

# SIMATIC

## S7-300/400 Tips

| Group | Topic                 |
|-------|-----------------------|
| 1     | Scaling Analog Values |

### Overview

Analog Inputs and Outputs in an S7 PLC are represented in the PLC as a 16-bit integer. Over the nominal span of the analog input or output, the value of this integer will range between -27648 and +27648. However, it is easier to use the analog values if they are scaled to the same units and ranges as the process being controlled. This applications tip describes methods for scaling analog values to and from engineering units.

### Program Description

The program for scaling analog values consists of 2 function (FC) blocks. These blocks are optimized for unipolar (0..27648) analog values, but will also work with bipolar analogs. Both blocks are shown in two different languages: LAD and STL.

FC100 converts an integer in a nominal 0..27648 range to a floating point number in the specified engineering units.

FC101 converts a floating point number in the specified engineering units to an integer in 0..27648 range.

**FC100 "Scale2Real" in LAD**

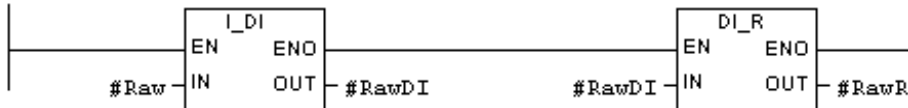
| Address | Decl. | Symbol | Data Type | Initial Value | Comment                                       |
|---------|-------|--------|-----------|---------------|---|
| 0.0     | in    | Raw    | INT       | 0             | value from Analog Input card, 0..27648 range  |
| 2.0     | in    | MinEU  | REAL      | 0.000000e+000 | Engineering Unit value corresponding to 0 raw |
| 6.0     | in    | MaxEU  | REAL      | 1.000000e+002 | EU value corresponding to 27648 raw           |
| 10.0    | out   | ValEU  | REAL      | 0.000000e+000 | Raw, scaled to engineering units              |
|         |       | in_out |           |               |   |
| 0.0     | temp  | RawDI  | DINT      |               | Raw, as 32-bit integer                        |
| 4.0     | temp  | RawR   | REAL      |               | raw, as floating point number                 |
| 8.0     | temp  | RngEU  | REAL      |               | EU range                                      |

**FC100 : Scale Analog Input to Engineering Units**

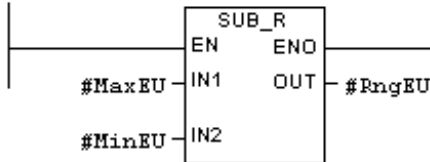
Converts integer Analog Input value from 0..27648 range to MinEU..MaxEU range. Conversion equation is:

$$ValEU = Raw * (MaxEU - MinEU) / 27648 + MinEU$$

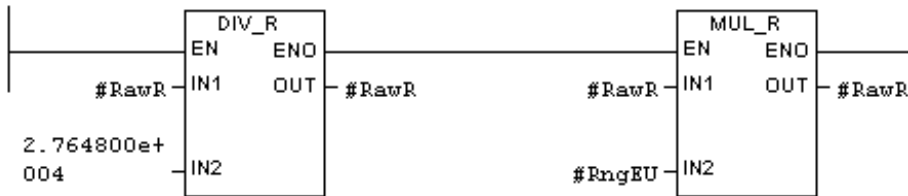
**Network 1 : Convert Raw to Real**



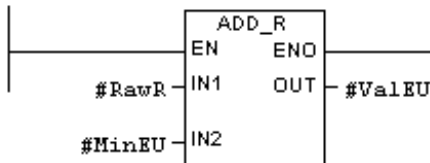
**Network 2 : Calculate EU span**



**Network 3 : Convert RawR to EU range**



**Network 4 : Add EU offset**



## FC100 "Scale2Real" in STL

| Address | Decl.  | Symbol | Data Type | Initial Value | Comment                                       |
|---------|--------|--------|-----------|---------------|---|
| 0.0     | in     | Raw    | INT       | 0             | value from Analog Input card, 0..27648 range  |
| 2.0     | in     | MinEU  | REAL      | 0.000000e+000 | Engineering Unit value corresponding to 0 raw |
| 6.0     | in     | MaxEU  | REAL      | 1.000000e+002 | EU value corresponding to 27648 raw           |
| 10.0    | out    | ValEU  | REAL      | 0.000000e+000 | Raw, scaled to engineering units              |
|         | in_out |        |           |               |   |
| 0.0     | temp   | RawR   | REAL      |               | Raw, as real                                  |

FC100 : Scale Analog Input to Engineering Units

Converts integer Analog Input value from 0..27648 range to MinEU..MaxEU range.  
Conversion equation is:

$$\text{ValEU} = \text{Raw} * (\text{MaxEU} - \text{MinEU}) / 27648 + \text{MinEU}$$

Network 1 : Covert AI value to real number

```
L    #Raw
ITD                                //sign extend 16-bit Raw
DTR                                //covert Raw to Real
T    #RawR
```

Network 2 : Scale Analog Input value

```
L    #MaxEU                        //calculate EU span
L    #MinEU
-R
L    #RawR                          //re-scale Raw to EU span
*R
L    2.764800e+004
/R
L    #MinEU                          //add EU offset
+R
T    #ValEU
```

## FC101 "UnscaleReal" in LAD

| Address | Decl. | Symbol | Data Type | Initial Value | Comment                                       |
|---------|-------|--------|-----------|---------------|---|
| 0.0     | in    | ValEU  | REAL      | 0.000000e+000 | Engineering Unit value to convert to 0..27648 |
| 4.0     | in    | MinEU  | REAL      | 0.000000e+000 | EU value corresponding to 0 raw               |
| 8.0     | in    | MaxEU  | REAL      | 1.000000e+002 | EU value corresponding to 27648 raw           |
| 12.0    | out   | Raw    | INT       | 0             | Analog output value in 0..27648 range         |
|         |       | in_out |           |               |   |
| 0.0     | temp  | RngEU  | REAL      |               | EU span                                       |
| 4.0     | temp  | RawR   | REAL      |               | unscaled value as floating point              |
| 8.0     | temp  | RawDI  | DINT      |               | unscaled value as 32-bit integer              |

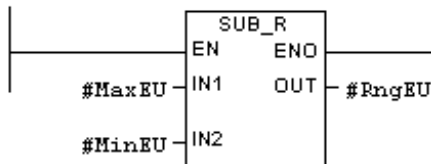
FC101 : Unscale from EU range to analog output range

Unscals from specified EU range to 0..27648 Analog output range. Unscale equation is:

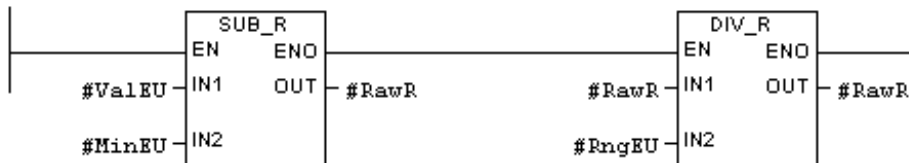
$$\text{Raw} = (\text{ValEU} - \text{MinEU}) * 27648 / (\text{MaxEU} - \text{MinEU})$$

If ValEU > MaxEU or ValEU < (2\*MinEU - MaxEU) then ValEU may not be convertible to a 16-bit integer. If this occurs, the output of this function is undefined. The output is also undefined if MaxEU = MinEU.

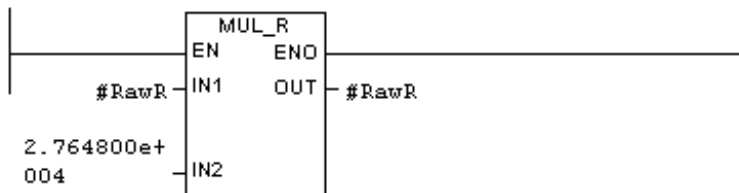
**Network 1** : Calculate EU span



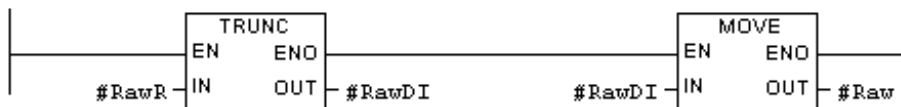
**Network 2** : Remove offset & convert ValEU to 0..1 span



**Network 3** : Convert output to 0..27648 range



**Network 4** : Convert to Integer



## FC101 "UnscaleReal" in STL

| Address | Decl.  | Symbol | Data Type | Initial Value | Comment                                       |
|---------|--------|--------|-----------|---------------|---|
| 0.0     | in     | ValeU  | REAL      | 0.000000e+000 | Engineering unit value to convert to 0..27648 |
| 4.0     | in     | MinEU  | REAL      | 0.000000e+000 | EU value corresponding to 0 raw               |
| 8.0     | in     | MaxEU  | REAL      | 1.000000e+002 | EU value corresponding to 27648 raw           |
| 12.0    | out    | Raw    | INT       | 0             | Analog output value in 0..27648 range         |
|         | in_out |        |           |               |   |
| 0.0     | temp   | RngEU  | REAL      |               |   |

FC101 : Unscale from EU range to analog output range

Unscals from specified EU range to 0..27648 Analog output range. Unscale equation is:

$$\text{Raw} = (\text{ValeU} - \text{MinEU}) * 27648 / (\text{MaxEU} - \text{MinEU})$$

If ValeU > MaxEU or ValeU < (2\*MinEU - MaxEU) then ValeU may not be convertible to a 16-bit integer. If this occurs, the output of this function is undefined. The output is also undefined if MaxEU = MinEU.

Network 1 : Unscale ValeU

```

L   #MaxEU           //calculate EU span
L   #MinEU
-R
T   #RngEU           //store here for now
L   #ValeU           //remove EU offset from value
L   #MinEU
-R
L   #RngEU           //convert to 0..1 scale
/R
L   2.764800e+004    //convert to 0..27648 scale
*R
RND                    //convert to DINT
T   #Raw             //store result

```

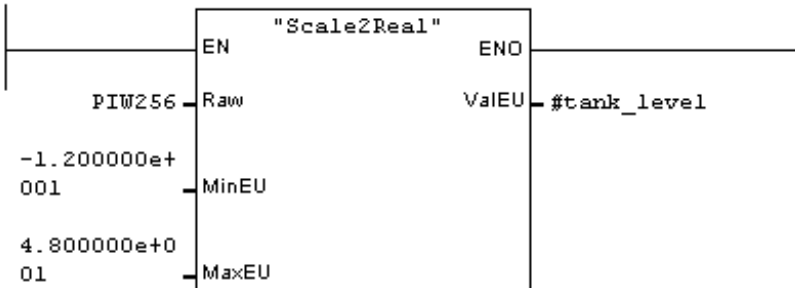
## Scaling Example

The following gives an example of calling each of the scaling blocks from within your program. The main thing you need to remember is that MinEU is the engineering unit value corresponding to a reading of 0 V or mA, and that MaxEU is the engineering unit value corresponding to the nominal maximum reading of the analog point.

**Network 1** : Convert Analog Input to Engineering units.

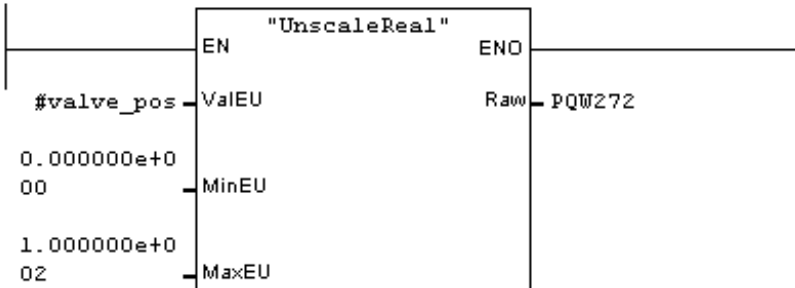
For this example, assume that there is a level transmitter at PIW256, generating a 4-20 mA signal, with 4 mA = 0 inches, and 20 mA = 48 inches.

MinEU must be set to the Engineering unit value corresponding to 0 mA, which is -12 inches. MaxEU is set to the EU value corresponding to 20 mA, or 48 inches. As long as the Analog Input signal is between 4 and 20 mA, tank\_level will be set to the current height in inches.



**Network 2** : Convert setting in Engineering Units to Analog Output

In this example, assume that PQW272 connects to a control valve that is fully closed at a 0 V output, and fully open at a 10 V output. valve\_pos is a number between 0 and 100, indicating the valve open percentage.



**General Notes**

The SIMATIC S7-300/400 Application Tips are provided to give users of the S7-300 and S7-400 some indication as to how, from the view of programming technique, certain tasks can be solved with this controller. These instructions do not purport to cover all details or variations in equipment, nor do they provide for every possible contingency. Use of the S7-300/400 Application Tips is free.

Siemens reserves the right to make changes in specifications shown herein or make improvements at any time without notice or obligation. It does not relieve the user of responsibility to use sound practices in application, installation, operation, and maintenance of the equipment purchased. Should a conflict arise between the general information contained in this publication, the contents of drawings or supplementary material, or both, the latter shall take precedence.

Siemens is not liable, for whatever legal reason, for damages or personal injury resulting from the use of the application tips.

All rights reserved. Any form of duplication or distribution, including excerpts, is only permitted with express authorization by SIEMENS.