## Table of contents

1 Foreword ................................................................................................................................. 5
   1.1 Notes on the documentation ............................................................................................... 5
   1.2 Safety instructions .............................................................................................................. 6

2 Overview .................................................................................................................................... 7

3 Functions ...................................................................................................................................... 8
   3.1 FLOOR .................................................................................................................................... 8
   3.2 FRAC ..................................................................................................................................... 9
   3.3 LMOD ................................................................................................................................... 9
   3.4 LTRUNC ................................................................................................................................. 10
   3.5 MODABS ............................................................................................................................... 11
   3.6 MODTURNS ......................................................................................................................... 12

4 [obsolete functions] .................................................................................................................. 14
   4.1 F_GetVersionTcMath ........................................................................................................... 14

5 Global constants ....................................................................................................................... 15
   5.1 Library version .................................................................................................................... 15
1 Foreword

1.1 Notes on the documentation

This description is only intended for the use of trained specialists in control and automation engineering who are familiar with applicable national standards.
It is essential that the documentation and the following notes and explanations are followed when installing and commissioning the components.
It is the duty of the technical personnel to use the documentation published at the respective time of each installation and commissioning.

The responsible staff must ensure that the application or use of the products described satisfy all the requirements for safety, including all the relevant laws, regulations, guidelines and standards.

Disclaimer

The documentation has been prepared with care. The products described are, however, constantly under development.
We reserve the right to revise and change the documentation at any time and without prior announcement.
No claims for the modification of products that have already been supplied may be made on the basis of the data, diagrams and descriptions in this documentation.

Trademarks

Beckhoff®, TwinCAT®, EtherCAT®, EtherCAT G®, EtherCAT G10®, EtherCAT P®, Safety over EtherCAT®, TwinSAFE®, XFC®, XTS® and XPlanar® are registered trademarks of and licensed by Beckhoff Automation GmbH.
Other designations used in this publication may be trademarks whose use by third parties for their own purposes could violate the rights of the owners.

Patent Pending

The EtherCAT Technology is covered, including but not limited to the following patent applications and patents:
with corresponding applications or registrations in various other countries.

EtherCAT® is a registered trademark and patented technology, licensed by Beckhoff Automation GmbH, Germany.

Copyright

© Beckhoff Automation GmbH & Co. KG, Germany.
The reproduction, distribution and utilization of this document as well as the communication of its contents to others without express authorization are prohibited.
Offenders will be held liable for the payment of damages. All rights reserved in the event of the grant of a patent, utility model or design.
1.2 Safety instructions

Safety regulations

Please note the following safety instructions and explanations!
Product-specific safety instructions can be found on following pages or in the areas mounting, wiring, commissioning etc.

Exclusion of liability

All the components are supplied in particular hardware and software configurations appropriate for the application. Modifications to hardware or software configurations other than those described in the documentation are not permitted, and nullify the liability of Beckhoff Automation GmbH & Co. KG.

Personnel qualification

This description is only intended for trained specialists in control, automation and drive engineering who are familiar with the applicable national standards.

Description of symbols

In this documentation the following symbols are used with an accompanying safety instruction or note. The safety instructions must be read carefully and followed without fail!

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DANGER</td>
<td>Serious risk of injury! Failure to follow the safety instructions associated with this symbol directly endangers the life and health of persons.</td>
</tr>
<tr>
<td>WARNING</td>
<td>Risk of injury! Failure to follow the safety instructions associated with this symbol endangers the life and health of persons.</td>
</tr>
<tr>
<td>CAUTION</td>
<td>Personal injuries! Failure to follow the safety instructions associated with this symbol can lead to injuries to persons.</td>
</tr>
<tr>
<td>NOTE</td>
<td>Damage to the environment or devices Failure to follow the instructions associated with this symbol can lead to damage to the environment or equipment.</td>
</tr>
</tbody>
</table>

Tip or pointer

This symbol indicates information that contributes to better understanding.
# Overview

The Tc2_Math library contains extended mathematical functions for TwinCAT PLC.

## Functions

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FLOOR [8]</td>
<td>The FLOOR function determines an integral value from a floating point number that is a fraction smaller than or equal that number.</td>
</tr>
<tr>
<td>FRAC [9]</td>
<td>The FRAC function determines the decimal component of a floating point number.</td>
</tr>
<tr>
<td>LMOD [9]</td>
<td>The LMOD function carries out a modulo division and returns the signed residual.</td>
</tr>
<tr>
<td>LTRUNC [10]</td>
<td>The LTRUNC function determines the integral component of a floating point number.</td>
</tr>
<tr>
<td>MODABS [11]</td>
<td>The MODABS function carries out a modulo division and determines the unsigned modulo value within the modulo range.</td>
</tr>
<tr>
<td>MODTURNS [12]</td>
<td>The MODTURNS function carries out a modulo division and determines the signed integral component.</td>
</tr>
</tbody>
</table>
3 Functions

3.1 FLOOR

The FLOOR function determines a integral value from a floating point number that is a fraction smaller than or equal that number. The resulting number is of type LREAL and is therefore not limited to the value range of integer variables.

Examples

FLOOR(2.8) = 2
FLOOR(-2.8) = -3

Similar functions: TRUNC, LTRUNC

Unlike FLOOR, the LTRUNC function always determines the integral part of a number. For positive values, this number is smaller than or equal the input parameter, for negative values it is greater than or equal the input parameter.

FUNCTION FLOOR : LREAL

VAR_INPUT
   lr_in : LREAL;
END_VAR

lr_in : Function parameters of type LREAL

Requirements

<table>
<thead>
<tr>
<th>Development environment</th>
<th>Target system type</th>
<th>PLC libraries to include</th>
</tr>
</thead>
<tbody>
<tr>
<td>TwinCAT v3.0.0</td>
<td>PC or CX (x86)</td>
<td>Tc2_Math</td>
</tr>
</tbody>
</table>
3.2 FRAC

The FRAC function determines the decimal component of a floating point number.

**Examples**

\[
\text{FRAC}(2.8) = 0.8 \\
\text{FRAC}(-2.8) = -0.8
\]

**FUNCTION FRAC : LREAL**

\[
\text{VAR_INPUT} \\
\hspace{1cm} \text{lr_in : LREAL; } \\
\text{END_VAR}
\]

\[
\text{lr_in} : \text{Function parameters of type LREAL}
\]

**Requirements**

<table>
<thead>
<tr>
<th>Development environment</th>
<th>Target system type</th>
<th>PLC libraries to include</th>
</tr>
</thead>
<tbody>
<tr>
<td>TwinCAT v3.0.0</td>
<td>PC or CX (x86)</td>
<td>Tc2_Math</td>
</tr>
</tbody>
</table>

3.3 LMOD

The LMOD function carries out a modulo division and returns the signed residual.

**Examples**

\[
\text{LMOD}(400.56, 360) = 40.56 \\
\text{LMOD}(-400.56, 360) = -40.56
\]

**Similar functions:** MOD, MODABS [11]
Unlike MOD, the LMOD function operates with floating point variables and also determines non-integer residuals.

In the context of NC axes, modulo values are usually used unsigned. These can be calculated with the MODABS \[\text{MODABS}\] function.

FUNCTION LMOD : LREAL

VAR_INPUT
  lr_Value : LREAL;
  lr_Arg   : LREAL;
END_VAR

lr_Value : Input value
lr_Arg   : Modulo range

![Graph of LMOD and LTRUNC functions]

Requirements

<table>
<thead>
<tr>
<th>Development environment</th>
<th>Target system type</th>
<th>PLC libraries to include</th>
</tr>
</thead>
<tbody>
<tr>
<td>TwinCAT v3.0.0</td>
<td>PC or CX (x86)</td>
<td>Tc2_Math</td>
</tr>
</tbody>
</table>

3.4 LTRUNC

The LTRUNC function determines the integral component of a floating point number.

Examples

LTRUNC(2.8) = 2
LTRUNC(-2.8) = -2

Similar functions: TRUNC, FLOOR \[\text{FLOOR}\]

Unlike TRUNC, the result from LTRUNC is of type LREAL and is therefore not limited to the value range of integer variables.
FUNCTION LTRUNC : LREAL

VAR_INPUT
  lr_in : LREAL;
END_VAR

lr_in : Function parameters of type LREAL

Requirements

<table>
<thead>
<tr>
<th>Development environment</th>
<th>Target system type</th>
<th>PLC libraries to include</th>
</tr>
</thead>
<tbody>
<tr>
<td>TwinCAT v3.0.0</td>
<td>PC or CX (x86)</td>
<td>Tc2_Math</td>
</tr>
</tbody>
</table>

3.5 MODABS

The MODABS function carries out a modulo division and determines the unsigned modulo value within the modulo range.

Examples

MODABS(400.56, 360) = 40.56
MODABS(-400.56, 360) = 319.44

Similar functions: MOD, LMOD

The MODABS function can be used to calculate the modulo set position of an NC axis from its absolute set position.

ModuloSetPosition := MODABS( NcToPlc.fPosSoll, 360 );

FUNCTION MODABS : LREAL

VAR_INPUT
  lr_val : LREAL;
  lr_mod : LREAL;
END_VAR

lr_val : Input value
**MODTURNS**

The MODTURNS function carries out a modulo division and determines the signed integral component (modulo periods, modulo rotations).

**Examples**

MODTURNS (800.56, 360) = 2

MODTURNS (-400.56, 360) = -2

The MODTURNS function can be used to calculate the number of modulo rotations of an NC axis from its absolute set position.

ModuloSetTurns := MODTURNS ( NcToPlc.fPosSoll, 360 );

**FUNCTION MODTURNS : LREAL**

```plaintext
VAR_INPUT
  lr_Value : LREAL;
  lr_Arg   : LREAL;
END_VAR

lr_Value : Input value

lr_Arg : Modulo range
```
Requirements

<table>
<thead>
<tr>
<th>Development environment</th>
<th>Target system type</th>
<th>PLC libraries to include</th>
</tr>
</thead>
<tbody>
<tr>
<td>TwinCAT v3.0.0</td>
<td>PC or CX (x86)</td>
<td>Tc2_Math</td>
</tr>
</tbody>
</table>
4 [obsolete functions]

4.1 F_GetVersionTcMath

The function returns library version info.

FUNCTION F_GetVersionTcMath : UINT
VAR_INPUT
  nVersionElement : INT;
END_VAR

nVersionElement : Version parameter:
• 1 : major number;
• 2 : minor number;
• 3 : revision number;

Requirements

<table>
<thead>
<tr>
<th>Development environment</th>
<th>Target system type</th>
<th>PLC libraries to include</th>
</tr>
</thead>
<tbody>
<tr>
<td>TwinCAT v3.0.0</td>
<td>PC or CX (x86)</td>
<td>Tc2_Math</td>
</tr>
</tbody>
</table>
5 Global constants

5.1 Library version

All libraries have a specific version. This version is shown in the PLC library repository too. A global constant contains the library version information:

Global_Version

VAR_GLOBAL CONSTANT
  stLibVersion_Tc2_Math : ST_LibVersion;
END_VAR


To compare the existing version to a required version the function F_CmpLibVersion (defined in Tc2_System) is offered.

All other possibilities known from TwinCAT2 libraries to query a library version are obsolete!