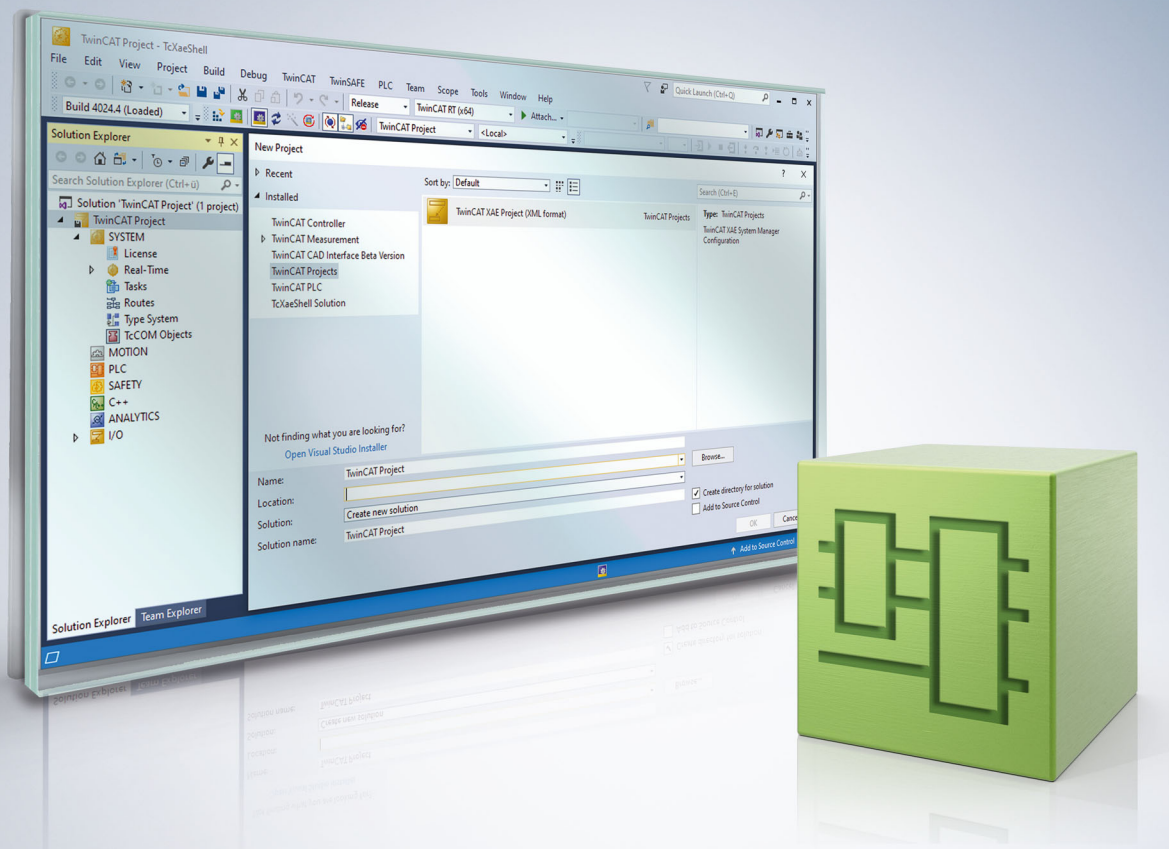
BECKHOFF New Automation Technology

Manual | EN

TE1000

TwinCAT 3 | PLC Library: Tc2_Math



2022-12-01 | Version: 1.2.1

Table of contents

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®, TwinCAT/BSD®, TC/BSD®, 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:
EP1590927, EP1789857, EP1456722, EP2137893, DE102015105702
with corresponding applications or registrations in various other countries.

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

Copyright

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!

⚠ DANGER

Serious risk of injury!

Failure to follow the safety instructions associated with this symbol directly endangers the life and health of persons.

⚠ WARNING

Risk of injury!

Failure to follow the safety instructions associated with this symbol endangers the life and health of persons.

⚠ CAUTION

Personal injuries!

Failure to follow the safety instructions associated with this symbol can lead to injuries to persons.

NOTE

Damage to the environment or devices

Failure to follow the instructions associated with this symbol can lead to damage to the environment or equipment.

Tip or pointer

This symbol indicates information that contributes to better understanding.

1.3 Notes on information security

The products of Beckhoff Automation GmbH & Co. KG (Beckhoff), insofar as they can be accessed online, are equipped with security functions that support the secure operation of plants, systems, machines and networks. Despite the security functions, the creation, implementation and constant updating of a holistic security concept for the operation are necessary to protect the respective plant, system, machine and networks against cyber threats. The products sold by Beckhoff are only part of the overall security concept. The customer is responsible for preventing unauthorized access by third parties to its equipment, systems, machines and networks. The latter should be connected to the corporate network or the Internet only if appropriate protective measures have been set up.

In addition, the recommendations from Beckhoff regarding appropriate protective measures should be observed. Further information regarding information security and industrial security can be found in our https://www.beckhoff.com/secguide.

Beckhoff products and solutions undergo continuous further development. This also applies to security functions. In light of this continuous further development, Beckhoff expressly recommends that the products are kept up to date at all times and that updates are installed for the products once they have been made available. Using outdated or unsupported product versions can increase the risk of cyber threats.

To stay informed about information security for Beckhoff products, subscribe to the RSS feed at https://www.beckhoff.com/secinfo.

2 Overview

The Tc2_Math library contains extended mathematical functions for TwinCAT PLC.

Functions

FLOOR [▶ 9]	The FLOOR function determines an integral value from a floating point number that is a fraction smaller than or equal that number.
FRAC [▶ 10]	The FRAC function determines the decimal component of a floating point number.
LMOD [▶ 10]	The LMOD function carries out a modulo division and returns the signed divide remainder.
LTRUNC [▶ 11]	The LTRUNC function determines the integral component of a floating point number.
MODABS [▶ 12]	The MODABS function carries out a modulo division and determines the unsigned modulo value within the modulo range.
MODTURNS [▶ 13]	The MODTURNS function carries out a modulo division and determines the signed integral component.
F_GetVersionTcMath [▶ 15]	Returns the version information of the library

3 Functions

3.1 FLOOR

The FLOOR function determines an 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 [▶ 11]

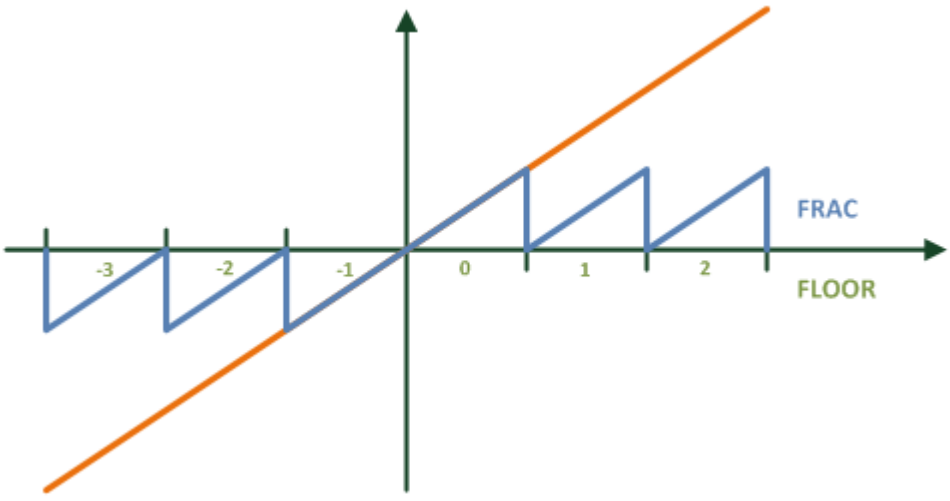
> Unlike FLOOR, the LTRUNC [▶ 11] function always determines the integral part of a number. For positive values, this number is smaller than or equal the input parameter, for negative values it is greater than or equal the input parameter.

FUNCTION FLOOR : LREAL

Inputs

```
VAR_INPUT
    lr_in : LREAL;
END_VAR
```

Name	Type	Description
lr_in	LREAL	Function parameters of type LREAL



Requirements

Development environment	Target system type	PLC libraries to include
TwinCAT v3.0.0	PC or CX (x86)	Tc2_Math

3.2 FRAC

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

Examples

FRAC(2.8) = 0.8

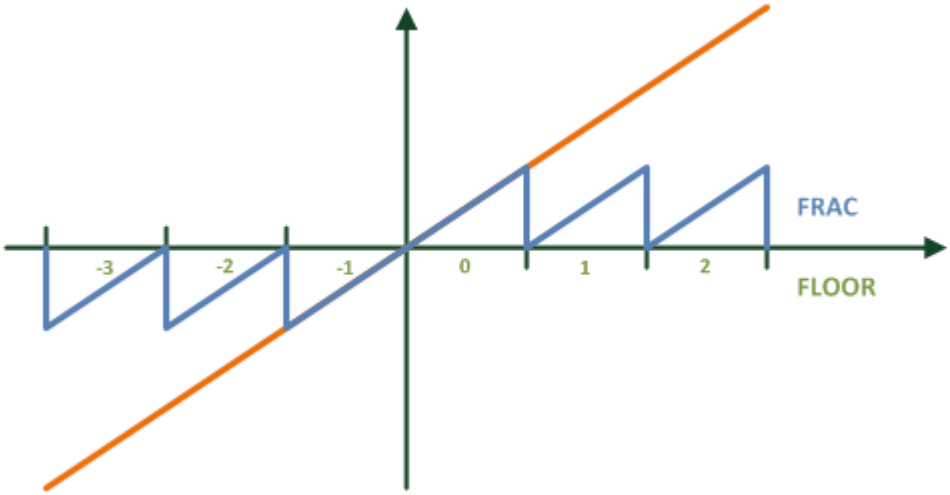
FRAC(-2.8) = -0.8

FUNCTION FRAC : LREAL

Inputs

```
VAR_INPUT
    lr_in : LREAL;
END_VAR
```

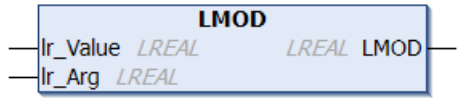
Name	Type	Description
lr_in	LREAL	Function parameters of type LREAL



Requirements

Development environment	Target system type	PLC libraries to include
TwinCAT v3.0.0	PC or CX (x86)	Tc2_Math

3.3 LMOD



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

Examples

LMOD(400.56, 360) = 40.56

LMOD(-400.56, 360) = -40.56

Similar functions: MOD, MODABS [▶ 12]

Unlike MOD, the LMOD function operates with floating point variables and also determines non-integer remainders.

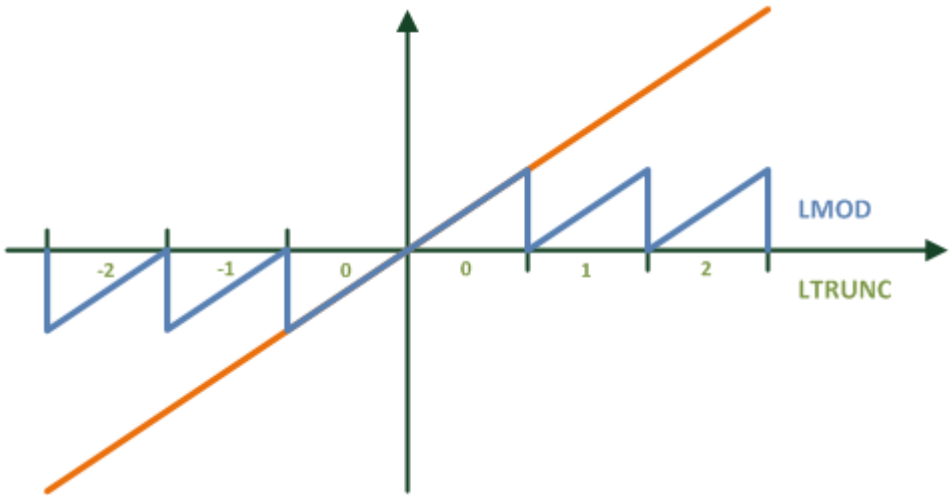
In the context of NC axes, modulo values are usually used unsigned. These can be calculated with the MODABS [▶ 12] function.

FUNCTION LMOD : LREAL

Inputs

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

Name	Type	Description
lr_Value	LREAL	Input value
lr_Arg	LREAL	Modulo range



Requirements

Development environment	Target system type	PLC libraries to include
TwinCAT v3.0.0	PC or CX (x86)	Tc2_Math

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 [▶ 9]

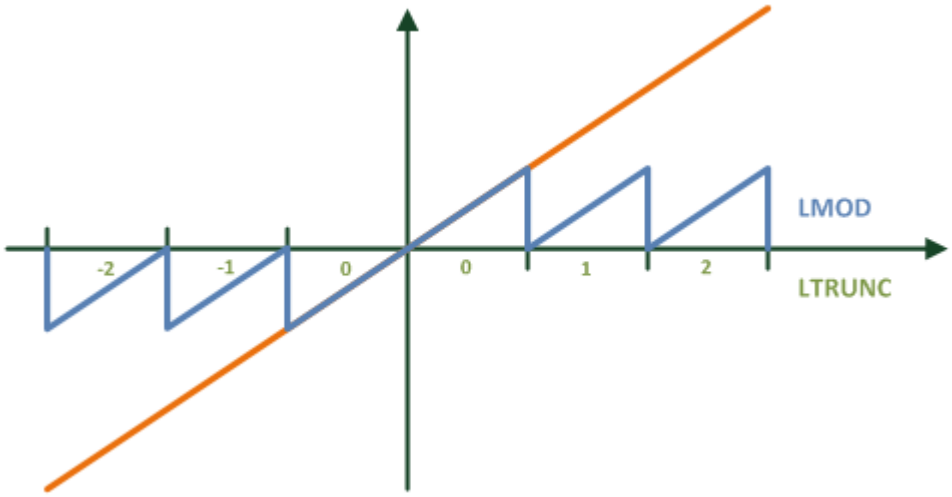
Unlike TRUNC, the result from LTRUNC is of type `LREAL` and is therefore not limited to the value range of integer variables.

FUNCTION LTRUNC : LREAL

Inputs

```
VAR_INPUT
    lr_in : LREAL;
END_VAR
```

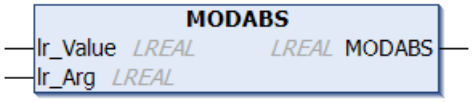
Name	Type	Description
lr_in	LREAL	Function parameters of type LREAL



Requirements

Development environment	Target system type	PLC libraries to include
TwinCAT v3.0.0	PC or CX (x86)	Tc2_Math

3.5 MODABS



The MODABS function performs 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 [▶ 10]

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

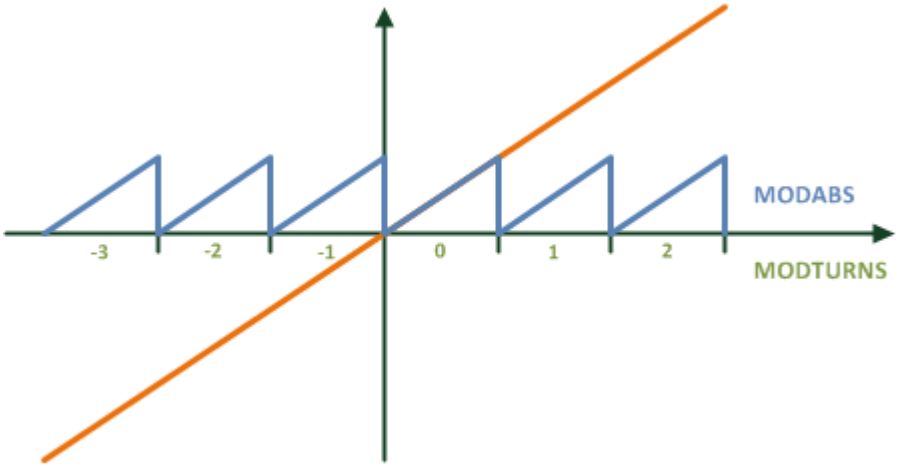
ModuloSetPosition := MODABS(NcToPlc.fPosSoll, 360);

FUNCTION MODABS : LREAL

Inputs

```
VAR_INPUT
    lr_val : LREAL;
    lr_mod : LREAL;
END_VAR
```

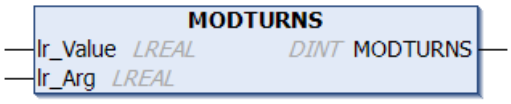
Name	Type	Description
lr_val	LREAL	Input value
lr_mod	LREAL	Modulo range



Requirements

Development environment	Target system type	PLC libraries to include
TwinCAT v3.0.0	PC or CX (x86)	Tc2_Math

3.6 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.

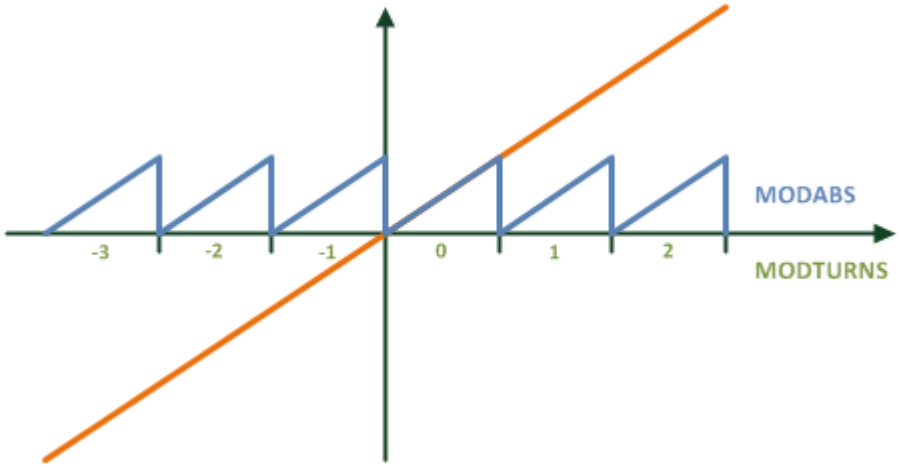
ModuloSetTurns := MODTURNS (NcToPlc.fPosSoll, 360);

FUNCTION MODTURNS : LREAL

Inputs

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

Name	Type	Description
lr_Value	LREAL	Input value
lr_Arg	LREAL	Modulo range

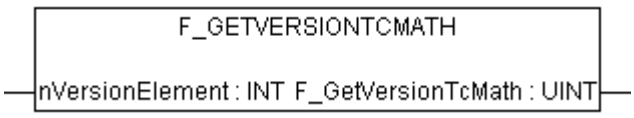


Requirements

Development environment	Target system type	PLC libraries to include
TwinCAT v3.0.0	PC or CX (x86)	Tc2_Math

4 [obsolete functions]

4.1 F_GetVersionTcMath



This function can be used to read PLC library version information.

FUNCTION F_GetVersionTcMath : UINT

Inputs

```
VAR_INPUT
    nVersionElement : INT;
END_VAR
```

Name	Type	Description
nVersionElement	INT	Version element to be read. Possible parameters: • 1: major number; • 2: minor number; • 3: revision number

Requirements

Development environment	Target system type	PLC libraries to include
TwinCAT v3.0.0	PC or CX (x86)	Tc2_Math

5 Global constants

5.1 Library version

All libraries have a certain version. The version is indicated in the PLC library repository, for example. A global constant contains the information about the library version:

Global_Version

```
VAR_GLOBAL CONSTANT
    stLibVersion_Tc2_Math : ST_LibVersion;
END_VAR
```

Name	Type	Description
stLibVersion_Tc2_Math	ST_LibVersion	Version number of the Tc2_Math library (type: ST_LibVersion)

To see if you have the version you need, use the function F_CmpLibVersion (defined in Tc2_System).

All other options for comparing library versions, which you may know from TwinCAT 2, are outdated!

More Information:
www.beckhoff.com/te1000

Beckhoff Automation GmbH & Co. KG
Hülshorstweg 20
33415 Verl
Germany
Phone: +49 5246 9630
info@beckhoff.com
www.beckhoff.com