



Twincat Library: Controlling Brakes

Application Note

Version	Date	Editor	Comment
002	2012-11-27	mvx	Convert to new document style
003	2013-05-01	mvx	Add detail on brake timing

Document AN130_TwinCAT-BrakeControl_EP
Version 003
Source Q:\doc\ApplicationNotes\AN130_TwinCAT-BrakeControl\
Destination T:\doc\ApplicationNotes
Owner mvx

Copyright © 2012	Triamec Motion AG	Phone	+41 41 747 4040
Triamec Motion AG	Industriestrasse 49	Email	info@triamec.com
All rights reserved.	6300 Zug / Switzerland	Web	www.triamec.com

Disclaimer

This document is delivered subject to the following conditions and restrictions:

- This document contains proprietary information belonging to Triamec Motion AG. Such information is supplied solely for the purpose of assisting users of Triamec products.
- The text and graphics included in this manual are for the purpose of illustration and reference only. The specifications on which they are based are subject to change without notice.
- Information in this document is subject to change without notice.

Table of Contents

1 Target and Purpose.....1	3 Manual Control.....3
2 Automatic Control.....2	3.1 Configuring the HMI.....3
	3.2 PLC Code.....3

1 Target and Purpose

The Triamec TwinCat library comes with basic sample codes for NCI and CNC. This application note describes the implementation of an axis brake and additional functions available in this library.

Many mechanical axes are equipped with mechanical brakes to prevent motion when the drive is disabled. The Triamec firmware contains all the functions required to automatically control a brake that is wired with the IO-connector. The settings are specified with the TAM System Explorer and made persistent in a TwinCat environment.

This application note describes the brake behavior in the TwinCAT context. If an application requires additional control of the brakes, it may use the function-blocks described in the second part to control the brakes through the manual control section of the Beckhoff HMI.

2 Automatic Control

If the axis brake is configured correctly in the TAM System Explorer and made persistent, the brake action takes place automatically as soon as the axis is being disabled and enabled.

There are timing issues, however, where the mechanical brake requires some time to enter the hold state. Disabling at the same time of braking might cause a certain displacement of the axis in this case. Then it is desirable for braking some time ahead of disabling an axis. The standard enabling block TL_MC_Power (which is part of TL_AxisSlow) contains this functionality with the following parameter:

gAxis[i].MC_axis_Power.BrakeTimeBeforeDisable	Default: 0.01 seconds
---	-----------------------

3 Manual Control

3.1 Configuring the HMI

The following definition should exist in Manual.dat in

triamec\TcHmiPro\TcApplication\bin\Debug\System

```
Gruppe7|x|True|False|True|6|0||
801|.ManualFunctionsF20ff|.ManualFunctionsF20n|.ManualFunctionsF2Status|.ManualFunctionsF2Locked|.ManualFunctionsF2Info|8
8|Triamec Communication|Stop|Restart|8
7|Triamec Brake 3|Hold|Release|7
6|x|x|x|6
5|x|x|x|5
4|x|x|x|4
3|x|x|x|3
2|x|x|x|2
1|x|x|x|1
```

The corresponding language texts are controlled under HMI/Options/Strings

3.2 PLC Code

The following declaration is made in the global definition "Global_Manual"

```
ManualFunctionsF20ff :ARRAY[1..20] OF BOOL;
ManualFunctionsF20n :ARRAY[1..20] OF BOOL;
ManualFunctionsF2Status :ARRAY[1..20] OF BYTE;
```

add the following code to the function block Triamec

```
(* ----- declaration ----- *)
VAR
```

```
brakesHold : ARRAY [1..N_AXIS] OF TL_MC_Brake;
brakesRelease : ARRAY [1..N_AXIS] OF TL_MC_Brake;
END_VAR
VAR CONSTANT
    cManual2DrivesBrake          : INT := 6;
END_VAR
(* ----- Code ----- *)

(* manual brake control by HMI. Automatic control is specified in Triamec-Configuration-file *)
ManualFunctionsF2Status[cManual2DrivesBrake] := 0;
FOR k:=1 TO N_AXIS DO
    brakesHold[k].Execute          := ManualFunctionsF2Off[cManual2DrivesBrake];
    brakesHold[k]( BrakeRelease := FALSE, axis:=gAxis[k].MC_axis , Trialink:=Trialink );
    brakesRelease[k].Execute := ManualFunctionsF2On[cManual2DrivesBrake];
    brakesRelease[k](BrakeRelease := TRUE, axis:=gAxis[k].MC_axis, Trialink:=Trialink );
END_FOR
```