Triamec TwinCat

Quick Startup Guide

Version	Date	Editor	Comment
1.3	2010-05-05	mvx	Combine NC and CNC documentation
1.4	2011-04-01	mvx	Add TLC 100
1.5	2011-05-11	mvx	Add Hmi information and sample code
1.6	2011-10-31	mvx	New simulation mode and modulo axes
1.7	2011-11-21	mvx	Update for smart sync
1.8	2012-04-04	mvx	Update for sample code 1.7
2.0	2012-09-12	mvx	Update for sample code 2.0
021	2012-11-28	chm	Reformat
022	2013-02-08	mvx	Add TLO100, move error codes to AN103. Use with sample code 3.0.
023	2014-03-28	mvx	New object gCnc for Task exceed compensation
024	2016-02-02	chm	Section 3.1: USB observer loop-back configuration

Document SWTC_TwinCAT-UserGuide_EP
Version 024
Source Q:\doc\Software\SWTC\SWTC_TwinCAT-UserGuide\
Destination T:\doc\SW\
Owner mvx

Triamec Motion AG
Industriestrasse 49
6300 Zug / Switzerland

Phone +41 41 747 4040
Email info@triamec.com
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 Introduction

This is user guide for the Triamec interface to the Beckhoff Twincat NC system explains, how to use Beckhoff NCI or Beckhoff CNC in conjunction with Triamec drives. Setting up your Triamec drive for Beckhoff Twincat CNC requires the following steps:

- Install TwinCat NCI 2.11.1551 (`tcat_2110_1552.exe`) and
  CNC v2.11.1551 (`TwinCATCnc210_1019.exe`) or similar.
- Setup your drives using the Triamec TAM System Explorer as explained in the
  “Drive Setup User Guide” linked in the TAM SDK start menu.
- Assign the drive an address and store the data persistent as discussed here.
- Adjust the twincat sample configuration as discussed here.
- Use the sample PLC code to implement the drive in your control SW as discussed here.

2 Simulation Mode

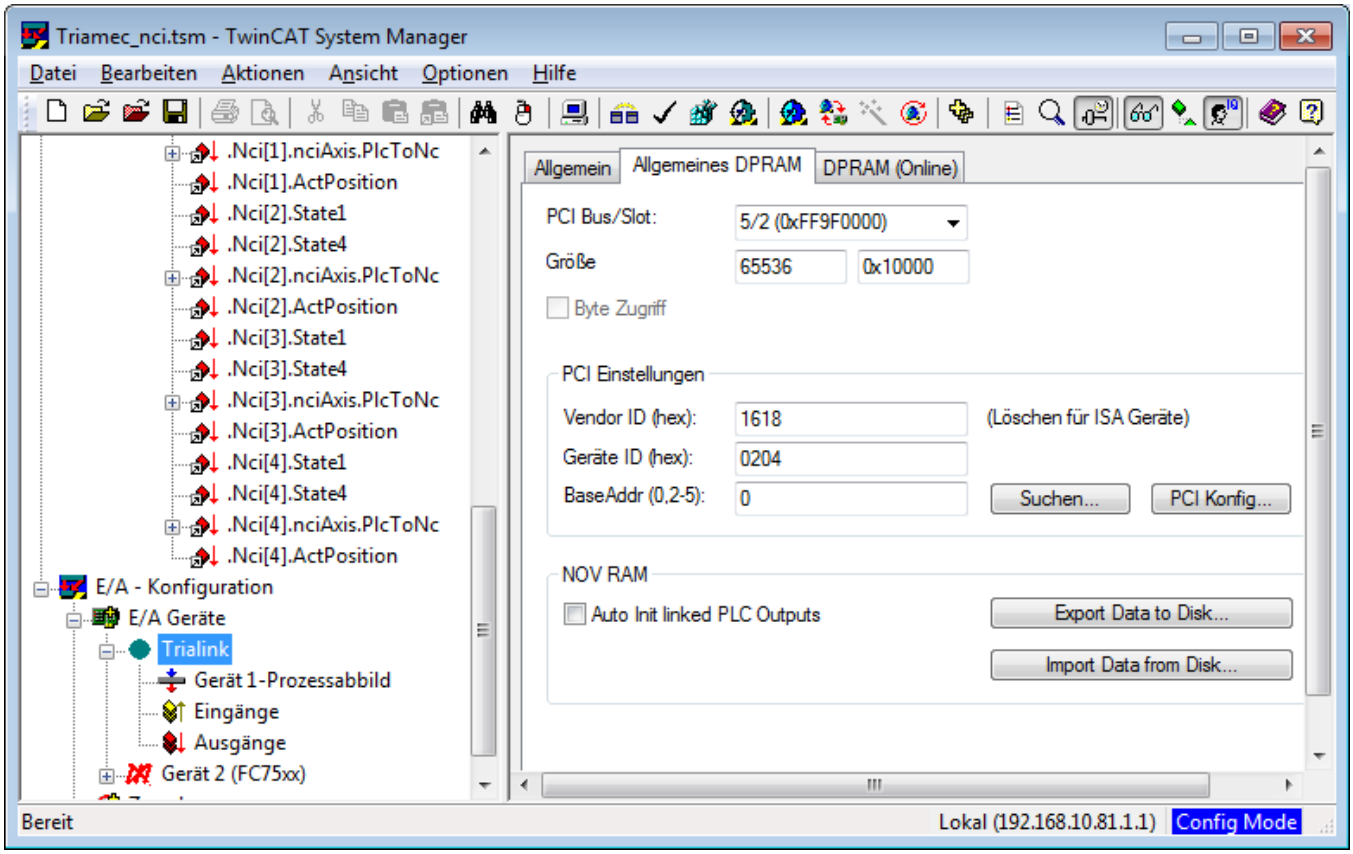
Two sample codes are available with increasing complexity

- Triamec_nci strait forward use of the Beckhoff NCI module.
- Triamec_cnc using the Beckhoff CNC.

Open a Twincat sample configuration (`*.tsm`) in the folder “plc”. Open the DPRAM settings for the Tria-Link adapter board in “`I/O-Configuration/I/O Devices/Trialink/GenericDPRAM`”, see below. The vendor ID is 1618 and the device ID is

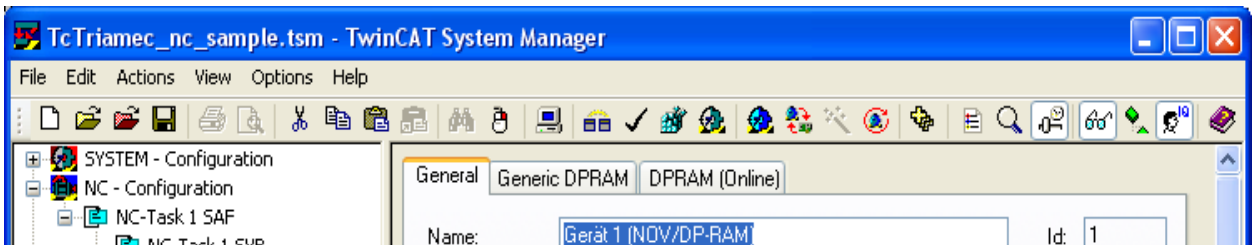
- 0200 for a TL100
- 0201 for a TLC201
- 0202 for a TLC 100
- 0203 for a TL300 PCI-Express
- 0204 for a TLO100 (with USB Observer)
- 0205 for a TLO300 (PCI-Express with USB Observer)
- 0206 for a TLOC100 (with USB Observer and additional controller)

Choose search and Twincat should find the Adapter.



Open the header “General=Allgemein” and see the device-ID “1”. This id identifies the Tria-Link board in the PLC code, see Trialink.Config.nDevId.

Activate the configuration.

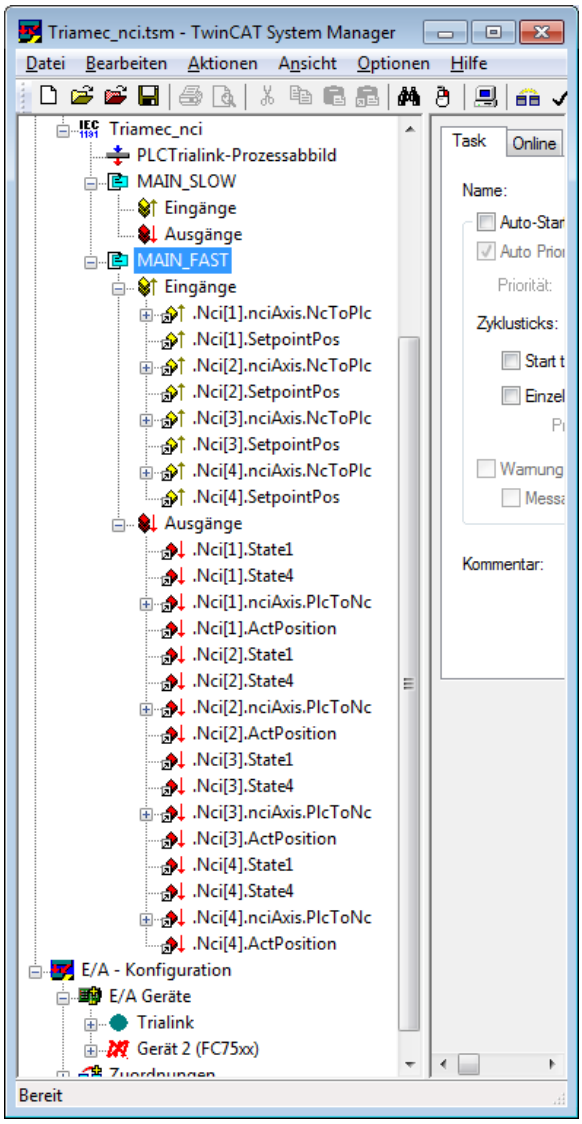


Open the PLC code (`*.pro`).

- Make sure the library “Triamec_lib.lib” is in the search path.
- Choose **Project > CleanAll** before first time use of the PLC program and answer “Do not keep configuration unchanged”.
- Set TL_ConfigurationManager.axes[iAxis].simulate TRUE for all axes iAxis.
- **Project > Compile**, **Online > Login**, **Online > GenerateBootProject**.
- Re-read the PLC into the TwinCatSystemManager.
- The IO's of the axes are visible in the PLC part of the TwinCatSystemManager. *They might have to be moved in the TwinCatSystemManager to MAIN_FAST for proper task synchronization.*
- Make sure the ring is closed and, if drives are in the ring, the drives must have 24V supplied.
- Reactivate the configuration and start the PLC code.

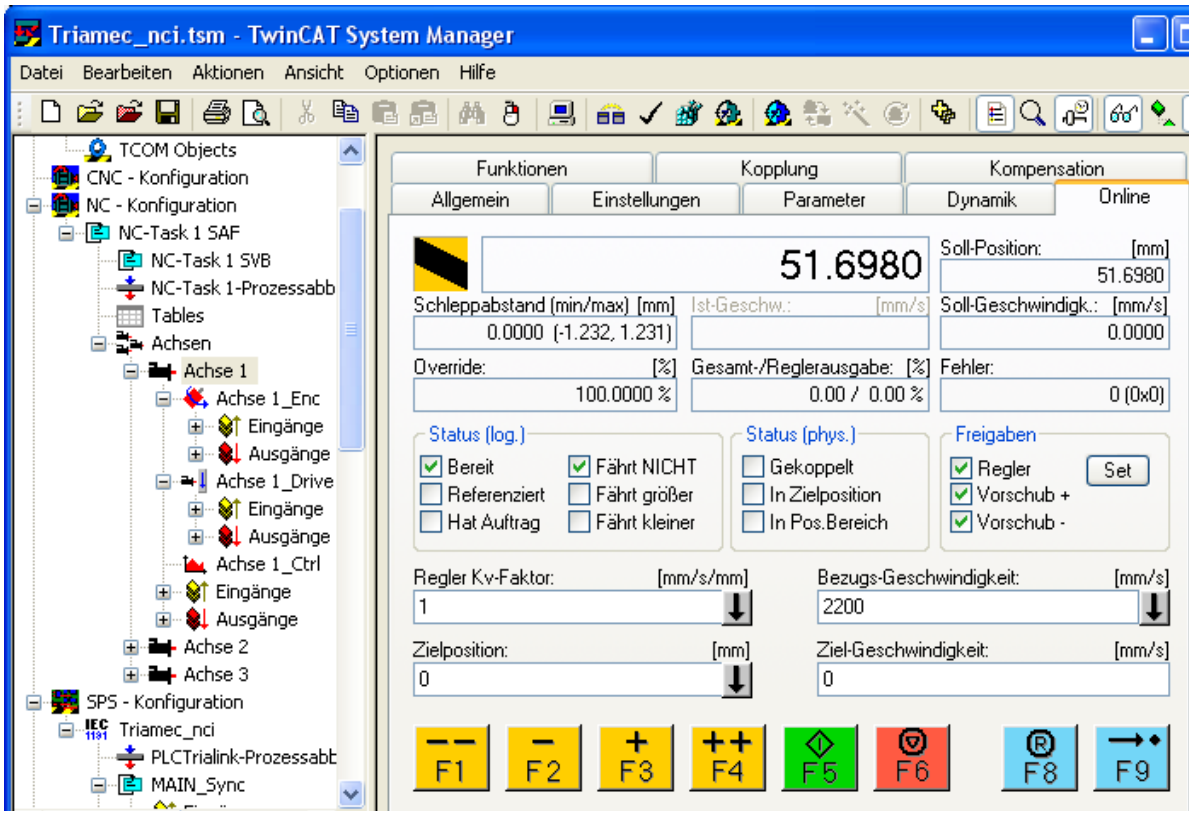
At this stage, you should be able to control the axes in simulated mode. The next chapters show how to use the TwinCat interface in the NCI and CNC case.

Later, we have to configure the real axes in the PLC code and switch each axis from simulation to real one by one.



2.1 Sample Code NCI

Set .AxisGroup.enable to TRUE. Then the NCI interface should look like



Press + and - to move the axis. The actual position should change accordingly. If the actual position is gray, the axis state is not valid. Check if enable is on and check the connections between TwinCat and PLC as shown in chapter “NCI configuration”.

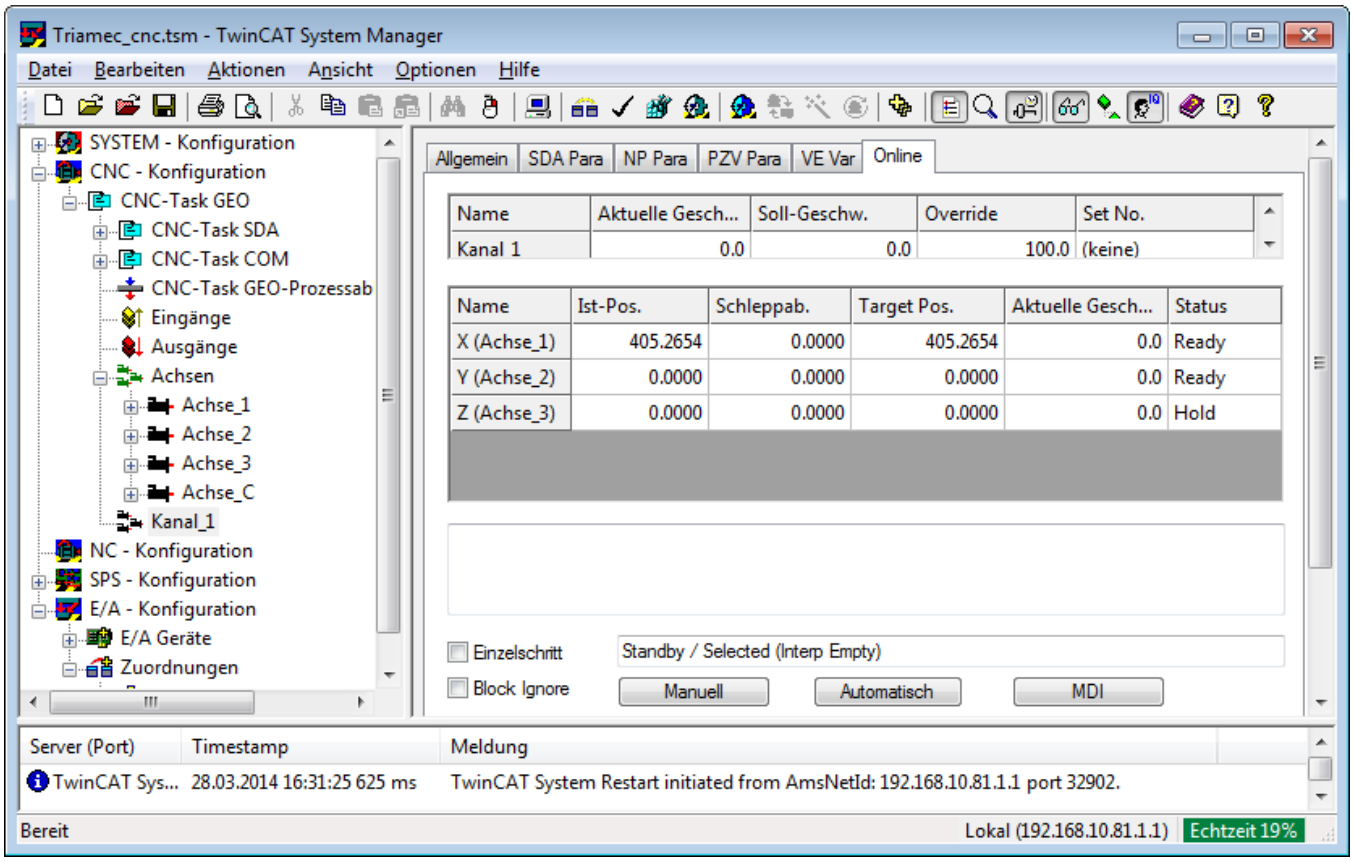
Set enable back to FALSE. If an axis is already configured as shown in the next chapters (but still simulated) the axis can be moved by hand and the ActualPosition should change correctly.

Set the dynamic parameters in NC/Achsen/Achse_n/Dynamik.

2.2 Sample Code CNC

Set AxisGroup.enable to TRUE and open the CNC control window: CNC-Task GEO/Kanal_1/Online.

Switch to Operation mode “Manual”. Select the first axis. Press + and - to move the axis. The actual position should change accordingly. The position “DistanceToGo” should reach zero at standstill.



Set enable back to FALSE. If an axis is already configured as shown in the next chapters (but still simulated) the axis can be moved by hand and the ActualPosition should change correctly in the CNC window.

Configure the CNC channel in CNC-Konfiguration/CNC-Task GEO/Kanal_1 and the CNC axes in CNC-Konfiguration/CNC-Task GEO/Achsen/Achse_n/Parameter as shown in the CNC-documentation.

3 Configuring the axes

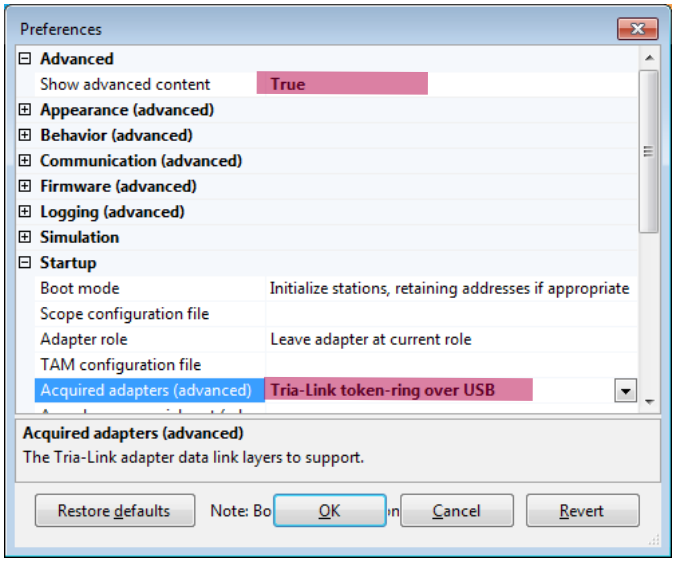
3.1 TAM System Explorer

Configure the drives with the TAM (Triamec) System Explorer to set drive parameters, install new firmware, and store parameters persistently. In a series environment with drives already setup, it is not necessary to instal the Explorer on the TwinCat PC. In a development environment, however, the Explorer should be installed. Before using the explorer for the *first time*, see ([1]).

Be aware that the PCI Adapter cannot be accessed simultaneously by Beckhoff and Explorer. On a Beckhoff PC set Triamec.CommunicationEnable:=FALSE in the PLC code before starting the explorer.

The Explorer may also be used in Observer mode to observe any drive signals online, even while TwinCat is running. There are two methods. The preferred method is using a USB cable from an independent PC/Laptop to the PCI adapter in the control PC. The PCI adapter must be a TLO100, TLOC100 or TLO300 in this case. The observer mode is also possible with TL100 or similar ([2]).

If the USB cable is *looped back* into the host system, the TAM System Explorer needs to be configured to only work over USB. This is accomplished using the menu **File | Preferences**, temporarily enabling to show advanced content, and restricting the acquired adapters to USB.



3.2 Setting Up the Drives

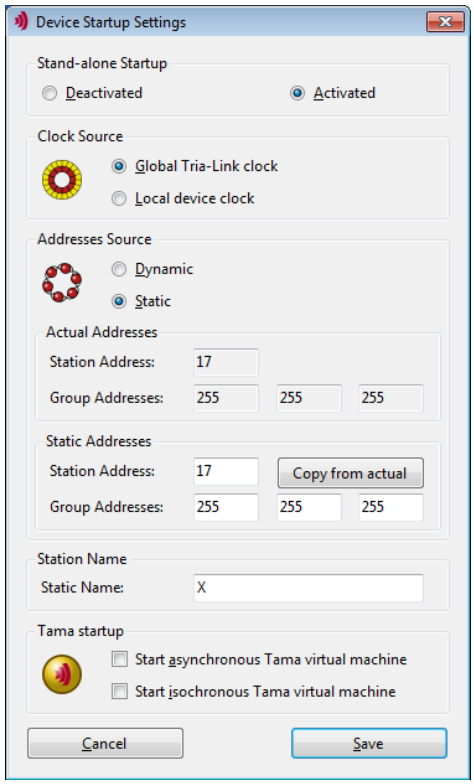
See the “Drive Setup Guide” ([3]) on the triamec CD on how to configure the drives and save the parameters of all drives in one configuration file `*.TAMcfg`.

1 After installation, use the Preferences menu of the TAM System Explorer and adjust the Startup parameter "Boot mode" to "Identify stations without assigning new addresses". If the explorer is used as Observer, set also "Adapter role" to "Observer".

2 In this case the second PC must also contain a PCI board: Include the PCI-Adapter of the observer PC into the Tria-Link ring. Start the Explorer on the observer PC. Start Beckhoff. Use the Explorer to adjust parameters and to log real-time data while Beckhoff is running.

3 SW_Drive-Setup-UserGuide_EPnnn.pdf

These settings are lost on power-down of a drive and must be loaded again from the configuration file after next power-up.



The settings are then permanently stored on a drive (Persistence). Use the TAM Explorer. Right-Click on the Drive/Device and choose “Device Startup Settings” as shown at the right side. Station addresses must be unique on the ring. As a convention we use static ***addresses*** starting at 17 for all drives (TS150, 151, 350, 351) and TIOB. These address settings will be entered as “***axis.Config.station***” parameter in the PLC code. Give the drive a meaningful static ***name***. Whatever settings are active before this step, will be made persistent.

Later changes of the configuration are only active until power-down and fall back to the persistent setting on the next power-up. Use this startup settings function again to update the persistent setting with a new configuration.

3.3 Setting Up the PLC

The PLC part of the axes is configured according to their logical axis number iAxis which is currently limited in the library to TL_CH_AX_MAX = 32.

- Configure the drive belonging to an axis with its station address in TL_ConfigurationManager.axes[iAxis].Config.Station
- Be aware that a drive may control two (DC-motor) axes. Therefore, two axes (iAxis) may be configured to the same station address. The second axis must then set TL_ConfigurationManager.axes[iAxis].Config.subAxis := TRUE;
- If the axis is in simulation mode (TL_ConfigurationManager.axes[iAxis].Config.Simulate := TRUE), the state information are set independent of the drive and the drive does not enable. However, if the axis is disabled, the actual position information is correctly propagated if possible.
- Start in simulation mode and then, if position information seem correct, turn simulation off. After changing this mode, toggle Triamec.CommunicationEnable or restart Beckhoff.

Now we are ready for position scaling, modulo, and referencing (see next chapters). Then download the PLC and start.

3.4 Position Scaling and Modulo

There are different position units in CNC/NCI (integer) and PLC (LREAL 64bit) and drive (40bit REAL). As a first use case we discuss the CNC in units of 0.1 µm, the PLC in mm and drive in m.

- Set the scale of the position data between CNC and PLC in TL_ConfigurationManager/gCnc[1].inc_per_unit:=10'000.
- Set the scale of the position data between PLC and Drives in TL_ConfigurationManager.axes[iAxis].Config.GearFactor := 1000;

If the axis is rotational with the CNC in units of 0.0001 °, PLC in °, and drive in rad.

- Set the scale of the position data between NCI/CNC and PLC in TL_ConfigurationManager/gCnc[1]:=10'000.
- Set the scale of the position data between PLC and Drives in TL_ConfigurationManager.axes[iAxis].Config.GearFactor := 180/PI;
- Set the modulo in TL_ConfigurationManager.axes[iAxis].Config.ModuloWrap:= 360;

For details on CNC axis, spindle, and channel settings see our Application notes and the Beckhoff documentation.

An other use case is NCI with finer resolution of 1nm and PLC in mm and Drive in m.

- Set the scale of the position data between NCI/CNC and PLC in TL_ConfigurationManager/gNci[1].inc_per_unit to 1E6.
- Set NC/Achsen/Achse_N/Achse_N_Enc/Parameter/Skalierungsfaktor to 1E-6.
- Set the Sercos-Axis Startup Value S-0-0077 to 1 and S-0-0078 to -9.
- Set the scale of the position data between PLC and Drives in TL_ConfigurationManager.axes[iAxis].Config.GearFactor := 1000;

3.5 Referencing (Homing)

The triamec homing function block TL_MC_Home implements several homing procedures. Our standard axis module ***TL_AxisSlow*** (called ***gAxis*** in the sample code) contains this function already and maps these parameters to gAxis[n].Config.Reference*. See TL_ConfigurationManager and ***AN108*** for details.

4 Introduction to the PLC Code

The PLC sample code is written in the language “structured text (ST)“ and implements two PLC Tasks

- MAIN_SLOW configuration, asynchronous state handling (Enable, Homing)
- MAIN_FAST synchronization between Trialink and TwinCat and coordinates

The sample code contains an array of axes ***gAxis[]*** of type ***TL_AxisSlow***. The logical axis numbers ***gAxis[].iAxis*** are unique among all axes belonging to one PCI-Adapter.

4.1 Path planner and Coordinates

While the axis *position controller* is always in the drive, the axis *path planner* is running either on the drive or in TwinCat. After enabling, the axis starts up in the drive path planner.

Drive pathplanner	The path planner runs on the drive. Moving to a new position or starting a velocity move is commanded asynchronously by PLCopen function blocks. These are used in the homing sequence.
TwinCat coupled pathplanner	The NC/CNC or a custom PLC code runs the path planner. An axis enters coupled mode using the input ***gAxis[].couple*** of TL_AxisSlow.

See *AN108* for more details.

The following blocks in the FOR-loop of MAIN_FAST are used for coordinate calculations and sending

position information to the PCI-Adapter in coupled mode.

- gNci[] gCnc[] gets NC/CNC path values.
- AxesPath[] takes the last positions, interpolates them to 50kHz and sends data to drives or TIOB.
- gNci[] gCnc[] sends actual position information back to NCI/CNC.

The actual position feedback is usually acquired with the rate of MAIN_SLOW and made available in AxesPath with a time correction. This is usually sufficiently fast, if all setpoint calculation is based on CNC data and no feedback from the drives must be used for setpoint calculation (actual position not in the control loop). If the actual position is required at higher speed, this drive needs to publish fast information to TwinCat, see application note AN105.

4.2 Accessing the adapter

For each PCI-Adapter, there is one global instance ***Trialink***. The call `Trialink.CallFast()` in MAIN_FAST synchronizes Beckhoff with the Tria-Link. The call `.Trialink.CallSlow` in MAIN_SLOW supplies the asynchronous communication to the Tria-Link.

If changing the task rate is necessary

- adjust the task cycle time in **Ressources > TaskConfiguration**.
- MAIN_FAST must have a high prio (0) and a rate not slower than 2ms (standard is 0.5ms).
- CNC-GEO or NCI-SAF must have the same cycle time as MAIN_FAST.
- MAIN_SLOW must be slower than MAIN_FAST (standard is 2ms).
- Go back to the TwinCat configuration, reload the PLC. Check the task priorities: **SystemConfiguration > EchtzeitEinstellungen > Prioritäten > Automatisch**.

The variable ***.CommunicationEnable*** boots the Trialink bus. It is TRUE by default. If rebooting is nessessary, toggle this variable FALSE and TRUE again. If the ring is successfully booted, the variable ***.CommunicationReady*** is TRUE. It may take up to 6s to boot the bus.

Some realtime issues on a heavily loaded system may be solved by synchronizing the CNC with the Triamec library. Replace the calls for the Trialink object by the gCnc object as shown in sample code TriamecCnc.

4.3 Configuration

Set the number of axes in the PLC program under **Global_Variables_Triamec > N_Axis**. These are all Triamec axes including handling axes, not only those that are to be accessed by CNC/NCI. The axes are configured in TL_ConfigurationManager. Make sure there is one configuration setting for each axis.

To add more axes, we use a dummy Sercos Master and add more Sercos Slaves. Later we will brake this connection and direct the data to our drives in the PLC.

- In I/O/Devices go to the device “Master FC75xx”.
- Master/AppendBox/”Drive Generic (Sercos)(for each additional axis)

Typical settings of the Sercos drives are

- SERCOS, Betriebsart Position 1 ohne Schleppabstand
- Startup S-0-0032 Hauptbetriebsart 11
- Eingänge S-0-0051 Lage-Istwert Geber 1
- Ausgänge S-0-0047 Lage-Sollwert
- Ausgänge S-0-0036 Geschwindigkeits-Sollwert

The further configuration is different for NCI and CNC:

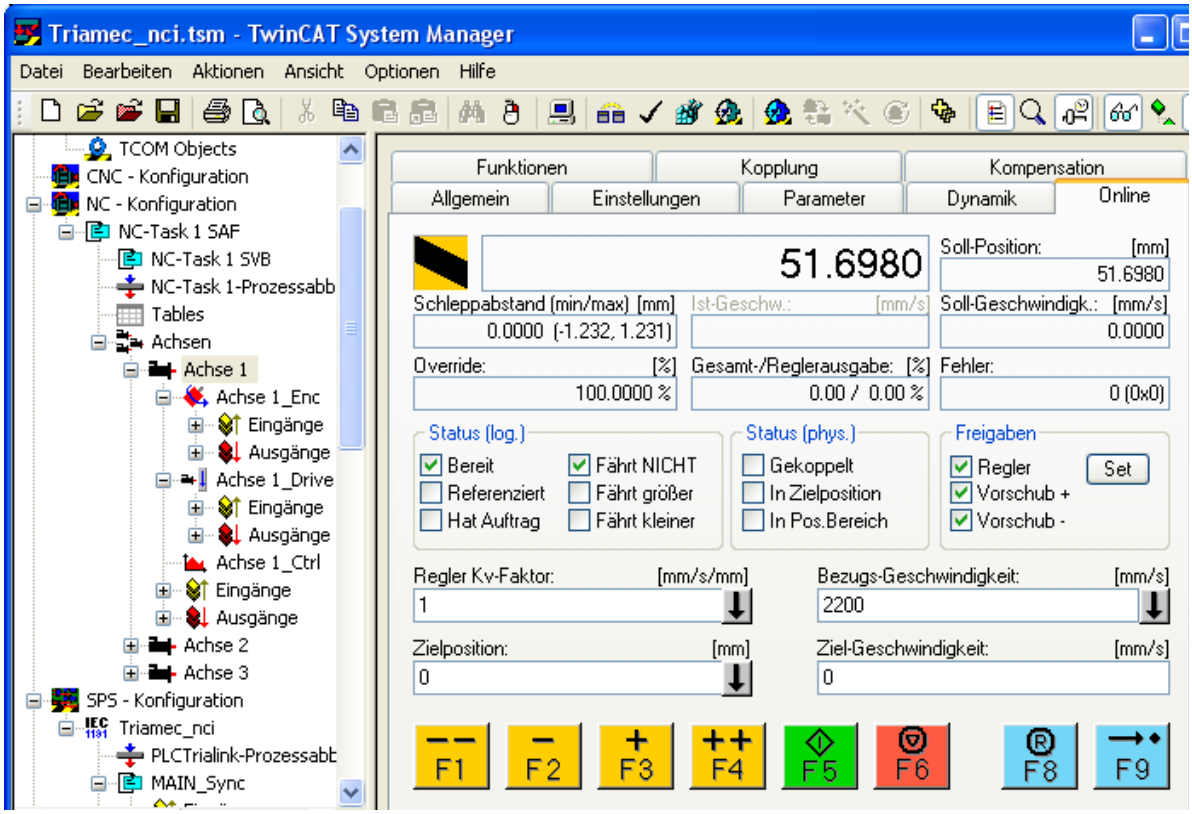
NCI-Configuration

Add a NC-Task to the configuration. Then add as many axes to the NCI as you need under **NC-Configuration > NC-Task > Achsen**. Choose Sercos under **Einstellungen > Achstyp** then connect these axes with the Sercos axes described above.

Now disconnect the automatic connections to these axes on the NCI side and connect the NCI with the PLC.

- .Nci[k].State1 Axis[k].Drive_In.nStatus1
- .Nci[k].State4 Axis[k].Drive_In.nStatus4
- .Nci[k].ActPosition Axis[k].Enc_in.nInData1 (ActPosition)
- .Nci[k].SetpointPos Axis[k].Drive_Out.nOutData1 (SetpointPos)

Finally disable the Sercos master by right-clicking on the icon but do not delete it. Activate the configuration then run the PLC.



CNC-Configuration

Add a CNC-Task to the configuration. Then add as many axes to the CNC as you need under **CNC-Config-**

uration > CNC-Task > Achsen. Note that removing an axis is not possible, only disabling. Connect these axes with the Sercos axes.

Now disconnect any automatic connections to these axes on the CNC side and connect the CNC with the PLC.

- .gCnc[k].SercosStatus Axis[k].Eingänge.Statuswort
- .gCnc[k].SercosPhase Axis[k].Eingänge.SercosPhase
- .gCnc[k].SercosPosCmd Axis[k].Ausgänge.Lage-Sollwert
- .gCnc[k].SercosPosAct Axis[k].Eingänge.Lage-Istwert Geber 1

Finally disable the Sercos master by right-clicking on the icon but do not delete it.

4.4 Library modules

All axes are defined and controlled by axis modules ".gAxis[iAxis]" of type `TL_AxisSlow`, where iAxis is the logical axis number of an axis. Such a module contains the following inputs which are connected to internal PLCopen blocks

- enable connected to TL_MC_Power
- stop connected to TL_MC_Stop
- couple connected to TL_MC_MoveSynchronized (not PLCopen)
- referenceEnable connected to TL_MC_Home
  referenceStart This is a trigger input to set referenceEnable TRUE.
- reset connected to TL_MC_Reset

The code of this module is shown in Application Note AN108. The following internal PLC open blocks are described in Application note AN108.

- TL_MC_Power
- TL_MC_Stop
- TL_MC_MoveSynchronized

Optional function blocks for axis movements *TL_MC_MoveAbsolute* and *TL_MC_MoveVelocity* are also described in AN108. See application note AN109 on how to access drive registers with ***TL_MC_RegisterRead*** and ***TL_MC_RegisterWrite***.

A group of axes [iL..iH] is combined for enabling and referencing using the function block AxisGroup in the sample code.

4.5 TIOB

To receive data from and sending data to a *TIOB* drive see application note AN110.

4.6 Windows 7

To set the TwinCat bootmode, the TwinCat System Control must not be opened through the fast access window, because it will not open in admin mode this way. Instead open the System Control in the **start > Programs > TwinCat** list as Admin.

Be aware that the standard TwinCat EventConfigurator does not run under Windows7 as of version

2.11.1551. Ask Beckhoff for an upgrade if starting the configurator is not possible.

4.7 Beginners Hints on Using the PLC

With **Online > Create Bootproject** a boot-project is generated in the TwinCat bootPath `C:\TwinCat` and linked to the Twincat configuration. Now the application runs independent of the editor.

- With **Online > Login** the editor hooks up into the running code. After LOGIN the user might find it surprising, that, e.g., an initialization sequence is not gone through. Be aware, that the PLC code can already be running before LOGIN.
- A neat feature of the editor is on-the-fly-changing of code. We can edit code, compile and download to the TwinCat while code is running real-time. Since the behavior is strongly cyclic the code is just replaced between two calls of the task.

If more than one task is running (at for example different frequencies) only one can be logged at a time. The active task is defined using **PLC Control > Ressources > TaskConfiguration** then right click on, e.g., “main_slow” and set “debugging task”.

Be aware that the statement “IF a AND b THEN will always calculate both variables a and b before continuing. If you want to make sure b is not calculated if a=FALSE (for example if b contains an empty pointer, use two cascaded IF statements instead.

5 Error display using TwinCat Events

Errors, warnings and information are made available at the function block outputs and using the TwinCat event concept. The meaning of the output ***errorId*** of a function block is listed in AN103. An errorId not equal to zero corresponds to an error if ***error***=TRUE and a warning or information if ***error***=FALSE.

The error description file *triamec\plc\events\Event Configuration Triamec.ecpx* must be made available to TwinCat as described in *triamec\plc\events\readme.txt.*

6 Hmi

The sample code Triamec_cnc contains the task Standard/MAIN with the CNC and HMI code from Beckhoff. This must be the first task in the task configuration list (not the highest prio). Only code in the folder Triamec is from Triamec.

The HMI TcHmiProX is based on Beckhoff C# code with the following modifications

6.1 Messages

All TcEventClass objects are shown under “active”, but not all are logged by log4net. In the standard Hmi, only errors are logged. We added further events:

- TcEventClass.Warning: MainApp.log.Warn
- TcEventClass.ParameterError: MainApp.log.Error
- TcEventClass.Alarm: MainApp.log.Error
- TcEventClass.Instruction: MainApp.log.Info

- TcEventClass.StateInfo: MainApp.log.Info
- TcEventClass.Hint: MainApp.log.Info
- TcEventClass.Message: MainApp.log.Info
- TcEventClass.Maintenance: MainApp.log.Info

Other options of log4net which are not used: “debug” and “Fatal”

6.2 Status Display

Five status strings are shown in the HMI-CNC-window.

- .StateGeneral[1-4] 2x2 state displays in the header in the order 1rowLeft, 1rowRight, 2rowLeft, 2rowRight
- .StatusSpindle Next to spindle display

Detailed state information under **PlcStatus > Triamec > Positions** etc.