TwinCAT Library: Streaming Mode

Application Node AN134

Version	Date	Editor	Comment
001	2018-10-17	DG	initial edit
002	2019-09-02	DG	direct feed is now also supported

Document	AN134_TwinCAT-StreamingMode_EP
Version	002
Source	Q:\doc\ApplicationNotes\
Destination	T:\Doc\ApplicationNotes
Owner	dg

Triamec Motion AG
Lindenstrasse 16
6340 Baar / 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 Target and Purpose

With TriamecLib 3.7.5.8 a new streaming mode is supported to feed the trajectory data to the drive. The new streaming mode has the following advantages:

- The task load of the fast task is reduced significantly.
- The new streaming mode is able to compensate TwinCAT task exceed failures.

This document describes the modification needed to integrate the new streaming mode.

2 Preconditions

The following conditions must be fulfilled to use the new steaming mode:

- Firmware TS >= 1050
- Firmware TSD >= 3.1.0
- TriamecLib >= 3.7.5.8

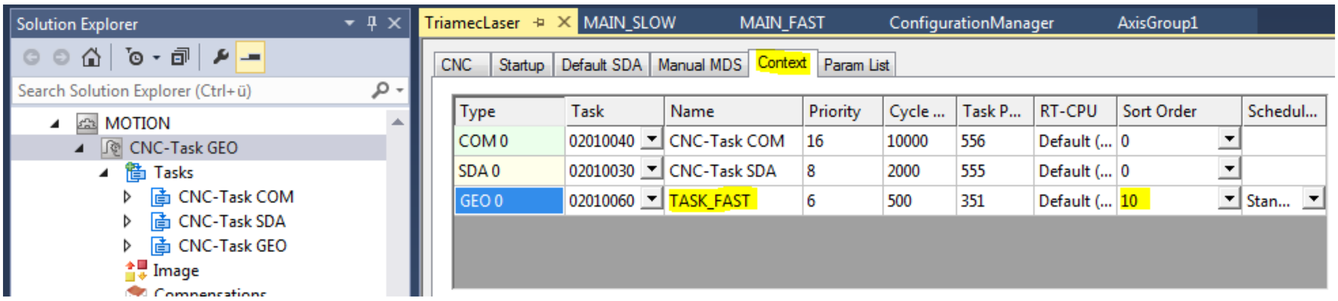
The low pass filter used to smooth the CNC data (Trialink.Config.FastFilterFrequency) is now applied to a signal with a sampling frequency defined by the TASK_FAST cycle ticks. With the previous method, the filter was running with 10 kHz. Consider, that this may reduce the efficiency of the filtering. This is especially the case if the FastFilterFrequency is close to the TASK_FAST cycle frequency.

3 Required Modifications

This section describes the required modifications to integrate the new streaming mode.

3.1 Task Settings

- The Triamec TASK_FAST and TASK_SLOW and the CNC-Task GEO must run on the same (isolated) core.
- The CNC-Task GEO has to be attached to the Triamec TASK_FAST. This can be achieved by adjusting the following settings:
    - Open the “Context” tab under “MOTION” - “CNC-Task GEO”.
    - For “GEO 0” select “TASK_FAST” from the pull down menu in the “Task” column.
    - With “Sort Order” the sequence of the calls can be configured. To call the “CNC-Task GEO” before the PLC sequence the value has to be set to 0 < “Sort Order” < 100.



Type	Task	Name	Priority	Cycle ...	Task P...	RT-CPU	Sort Order	Schedul...
COM 0	02010040	CNC-Task COM	16	10000	556	Default (...	0	
SDA 0	02010030	CNC-Task SDA	8	2000	555	Default (...	0	
GEO 0	02010060	TASK_FAST	6	500	351	Default (...	10	Stan...

3.2 Configuration

- BufferTicks: The value of the parameter “FastBufferTicks” is set to 6 by default. This buffer is used to compensate TwinCAT task exceeds and will therefore generate a delay between the trajectory calculation of the PLC and the execution of the trajectory on drive side. In normal cases this is allowable as all drives experience the same delay. For special applications, which are sensitive regarding delay between PLC and drive, the number of BufferTicks can be configured in the ConfigurationManager:

```
        Trialink.Config.FastBufferTicks = userBufferTicks;
```

  The value of userBufferTicks must fulfill the following condition: userBufferTicks <= min((10ms/cycleTime_fastTask)-1, 6)

- Maybe the Trialink.Config.FastFilterFrequency has to be adjusted if the frequency is close to the TASK_FAST cycle ticks.

3.3 Code

TL_Trialink Instance

Replace the function block TL_Trialink with TL_Trialink2 for all declarations of the TL_Trialink instance. E.g.:

```
VAR_GLOBAL
        ...
        ~~Trialink     : Triamec.TL_Trialink;~~
        Trialink     : Triamec.TL_Trialink2;
        …
END_VAR
```

TL_AxisSlow and TL_AxisFast Instance

TL_AxisSlow and TL_AxisFast are replaced by the TL_Axis2 function block. Therefore modify the existing TL_AxisSlow instance to a TL_Axis2 instance e.g.:

```
VAR_GLOBAL
        ...
        ~~gAxis   : ARRAY [1..N_AXIS] OF Triamec.TL_AxisSlow;~~
        gAxis   : ARRAY [1..N_AXIS] OF Triamec.TL_Axis2;
        …
END_VAR
```

and remove the TL_AxisFast instance:

```
        ~~axisFast: ARRAY[1..N_Axis] OF Triamec.TL_AxisFast;~~
```

Replace calls of the TL_AxisSlow object with a call of the CallSlow method e.g.

```
        ~~gAxis[iAxis](Trialink:=Trialink);~~
        gAxis[iAxis].CallSlow(Trialink:=Trialink);
```

Replace calls of the TL_AxisFast object with a call of the CallFast method e.g.

```
        ~~axisFast[iAxis](Trialink:=Trialink);~~
        gAxis[iAxis].CallFast(Trialink:=Trialink);
```

Replace all objects of TL_AxisSlow and TL_AxisFast with the new TL_Axis2 object and also consider the replacement of the following variables:

```
        ~~gCncAx[iAxis].PositionAct     :=axisFast[iAxis].actualPosition.Position;~~
        gCncAx[iAxis].PositionAct     :=gAxis[iAxis].act_pos;
```

```
        ~~axisFast[iAxis].PosIn := gCncAx[iAxis].PositionCmd;~~
        gAxis[iAxis].fastPositionCmd := gCncAx[iAxis].PositionCmd
```

Filter frequency:

```
        ~~axisFast[iAxis].filter.freq_corner    := 500;~~
```

has to be replaced by the following entry in the ConfigurationManager.TcPOU

```
        Trialink.Config.FastFilterFrequency   := 500;
```

Coupling

The couple state is handled internally by the TriamecLib. Therefore it is proposed to leave the couple-input TRUE . For example the following code should be changed:

```
        ~~gAxes[AxisId].couple   := gAxes[AxisId].enabled AND NOT gAxes[AxisId].referenceBusy;~~
        gAxes[AxisId].couple   := TRUE;
```

In case of a discrete motion command (e.g. TL_MC_MoveAbsolute) initialized by the application just the re-coupling of the axes has to be handled by the application by setting couple state FALSE and TRUE again.

4 Log File

In cases where the TwinCAT task exceeds cause a failure of the trajectory-generation a warning is added to the triamec-report-YYYY.txt file in the folder defined by the Trialink.Config.RootFolder variable in the ConfigurationManager, where YYYY is the current year.