

Twincat Library: Using TIOB

Application Note

Version	Date	Editor	Comment
002	27.11.12	mvx	Convert to new document style
003	2013-04-03	mvx	Update to lib 2.2.2

Document AN110_TwinCAT-Tiob_EP
Version 003
Source D:\svnroot\doc\ApplicationNotes\
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	2 PLC Code.....1
----------------------------	------------------

1 Target and Purpose

The Triamec TwinCat library comes with basic sample codes for NCI and CNC. This application note describes additional functions available in this library on using the TIOB board.

2 PLC Code

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

```
tiob      : TL_Tiob;
```

add the following code to the task MAIN_SLOW

```
(* The data received and sent depends on the parameter subAxis. *)
(* - subAxis=0  receive encoder position of axis 0 and Ain1 and Ain2. Send AO1 if publish=true *)
(* - subAxis=0  receive encoder position of axis 1 and Ain3 and Ain4. Send AO2 if publish=true *)
(* The parameter iAxis specifies the logical axis number with the same behavior as for drive axes. *)
tiob.iAxis := 5;
tiob.station := 30;
tiob.subAxis := 0;
(* an auxiliary register to be published (TIOB to master) *)
tiob.RegInAux := tiob.register.GeneralCommandsAnalogOutput1;
tiob.CallSlow(Trialink:=Trialink);
```

add the following declaration to the task MAIN_FAST

```
ToTiob      : TL_AxisFast;
TiobRoundtripTicks : REAL;
```

and the code

```
(* receive from TIOB: *)
(* tiob.encoder is encoder of the subAxis specified in slow task *)
(* tiob.AIn1or3 is Analog input 1 for subAxis0 and 3 for subAxis 1 *)
(* tiob.AIn2or4 is Analog input 2 for subAxis0 and 4 for subAxis 1 *)
(* send analog values as axis iAxis = unique axis number *)

tiob.CallFast(Trialink:=Trialink);

(* ToTiob.PosIn := 4.33; *) (* value to be sent *)
ToTiob.filter.freq_corner := 2000;
ToTiob.iAxis := tiob.iAxis;
ToTiob( Trialink := Trialink );

(* The rest is a way to calculate TIOB data roundtrip *)
TiobRoundtripTicks := (FromTiob.aux-ToTiob.PosIn)/0.001;
ToTiob.PosIn := ToTiob.PosIn + 0.001;
IF ToTiob.PosIn>=5 THEN
    ToTiob.PosIn := 0;
END_IF
(* Measured Roundtrip ticks are FastHandler.pll.BufferTicks + 1/f/Ts/6.3 of filter *)
(* BufferTick=3, 200Hz =1.6 x=2.3 gives 6.9 Ticks *)
(* BufferTick=3, 2000Hz=0.2 x=2.3 gives 5.5 Ticks *)
(* with BufferTicks=2.5 and Ts=0.5ms we do not get any Sync warnings *)
```