One Position Two Motors

Application Note 145

This document describes the setup of axes, where two motors are mechanically coupled to one position feedback.

Table of Contents

Document	AN145_OnePositionTwoMotors_EP
Version	003, 2024-05-15
Source	Q:\Doc\ApplicationNotes\
Destination	T:\Doc\ApplicationNotes
Owner	sm

1 Introduction

The mode offers the possibility to link two motors to one position controller. With this option, each motor is connected to its own output stage of a dual-axis *Drive (TSD)*. The coupling is made drive-internally, which allows precise control and setup.

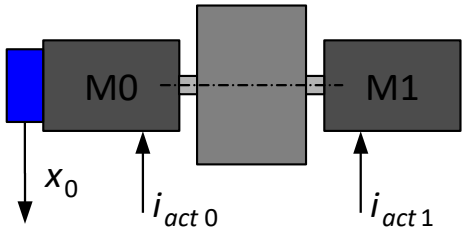

Figure 1: Rotary Example Setup

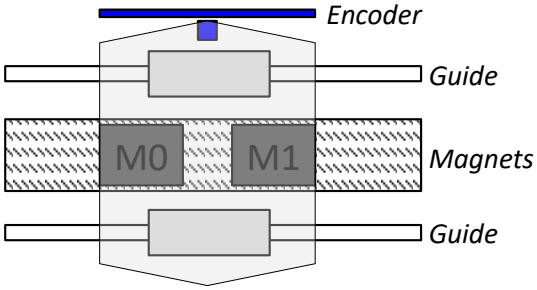

Figure 2: Linear Example Setup

1.1 Requirements and Restrictions

The following points are mandatory and some restrictions apply.

- The setting requires firmware 4.17 or higher and *TAM System Explorer* version 7.21 or higher. The current firmware and *TAM Software* can be downloaded from www.triamec.com.
- Only dual axis drives (*TSD*) support the *OnePositionTwoMotors* setting.
- Voltage feed forward is not supported.
- The software option *GY* is not required.

1.2 Implementation

The controller structure for this mode is visualized in Figure 3. It uses the standard position controller of Axis0. The coupling of both motor channels is realized by routing the setpoint current to the current controllers of each axis independently.

In between setpoint current and current controllers, a balance gain is configurable to consider the load distribution between the two motors.

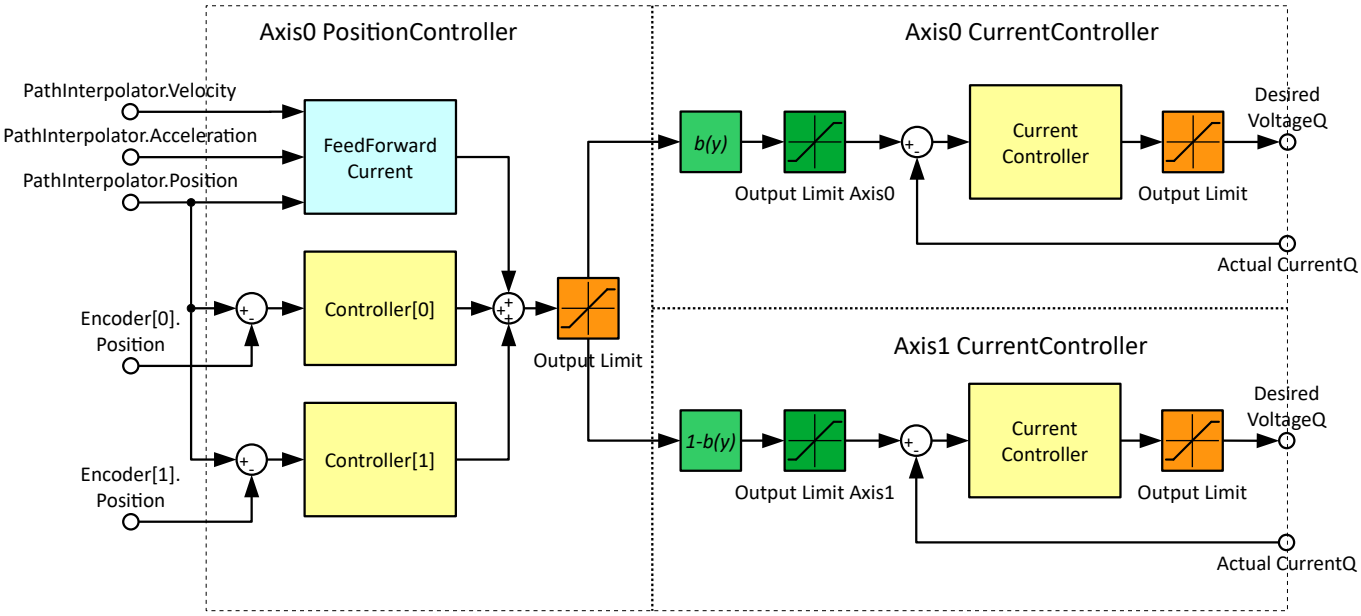

Figure 3: Block Diagram of the Controller Structure

This mode reuses parameters that were introduced for the *MIMO Gantry* mode. Therefore, some parameters for commissioning are found in the **Gantry** node of the **Register** tree (Figure 4).

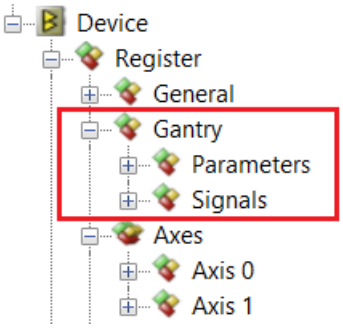

Figure 4: Gantry node in Register Tree

For further explanation of each register in the Gantry tree, refer to [2].

2 Commissioning

This chapter explains the required steps to commission an axis with one position reference and two motors.

2.1 TAM Parameter Preparation

First, setup General.Parameters and Axes[0].Parameters, as if it is a standard axis with one motor. Refer to [1], chapter 5.1-5.3, for the initial setup.

For Axes[1], setup the following register nodes, according to [1], chapter 5.1-5.3.

- Axes[1].Parameters.Motor
- Axes[1].Parameters.Commutation
- Axes[1].Parameters.CurrentController

The following parameters need special consideration:

General.Parameters.ControllerTopology

- Set this parameter to OnePositionTwoMotors.

Gantry.Parameters

- For initial commissioning, set BalanceOffset = 0.5. If the theoretically correct value is known, i.e. from CAD tools, set this value.
- As the dependency of another axis is not yet considered, set BalanceSlope = 0.
- With CurrentLimit0 and CurrentLimit1 the desired current applied to the motors can be restricted. Typically, these parameters are set about 10% below the peak current of the motor or the drive, whichever is smaller.

Axes[0].Parameters.PositionController.OutputLimit

- Set this value to Gantry.Parameters.CurrentLimit0 + Gantry.Parameters.CurrentLimit1.

Axes[0].Parameters.PositionController.Controllers[0|1].IntegratorOutputLimit

- Set this value to 40% - 80% of 2 * Axes[].Parameters.Motor.NominalCurrent.

2.2 Current Controllers

This section describes the tuning of the current controller.

- Make sure all parameters are set as described in chapter 2.1.
- Open the **Measure...** window in the **Frequency Response** module of **Axis 0**.
- Set the **Method** to **OpenLoop**, then follow the procedure for a standard axis, as described in [1].
- To determine the parameters Kr and Tn, execute a current controller tuning for **Axis 0**, as described in [1].
- If both motors are identical, the parameters Kr and Tn can be set identical in Axes[1].Parameters.CurrentController.
- Otherwise do the current controller tuning for axis 1 in the same way as described above for Axis 0. For the measurement, it's necessary to temporarily connect motor 1 to X40. Connect the motor 1 back to X41 after the tuning.

2.3 Motor Direction Check

The correct setting for the motor direction is crucial to not make the motors work against each other.

To find the correct directions for Axis 0 and Axis 1 experimentally, the direction analysis described in [1], chapter 5.5.1, can be used. The test must be set up and run on both axes individually. Adjust the Axes[].Parameters.Motor.InvertDirection for both axes, so that the MasterPosition of Axes[0] moves in positive direction for both axis tests. For the test with Axis 1, it's necessary to temporarily set the parameter General.Parameters.ControllerTopology to Standard. Set the parameter General.Parameters.ControllerTopology back to OnePositionTwoMotors after the test.

2.4 Measure Phasing Offset

Given by the mechanical motor alignment, the electrical commutation angle of Axis 1 can be shifted in reference to Axis 0. If the magnetic poles are mechanically aligned by design, set both commutation angles to the same value. In most cases phasing is executed at angle zero:

- Axes[0].Parameters.Commutation.Angle = 0
- Axes[1].Parameters.Commutation.Angle = 0

If the magnetic poles are not mechanically aligned, subtract the phasing angle offset from the angle in Axis 1:

- Axes[0].Parameters.Commutation.Angle = 0
- Axes[1].Parameters.Commutation.Angle = 0 - *Offset*

If the offset is not known, measure it with the procedure described below.

Note For the following procedure it is mandatory to have a valid CurrentController configuration on both axes.

Set the following Commutation parameters for both axes.

- Axes[].Parameters.Commutation.PhasingMethod = RotorAlignment
- Axes[].Parameters.Commutation.EnablingMethod = ForcePhasing
- Axes[].Parameters.Commutation.Angle = 0

Setup the Scope with the following signals and a recording time of 30 seconds.

- Axes[0].Signals.PositionController.MasterPosition
- Axes[0].Signals.CurrentController.ActualCurrentD
- Axes[1].Signals.CurrentController.ActualCurrentD

Then follow the measurement procedure:

1. Set Gantry.Parameters.BalanceOffset = 1
2. Open the *Axis Module* for **Axis 0** and **Attach** the axis.
3. Start the *Scope*
4. **Enable** the axis and wait for the ActualCurrentD to drop back to 0.
5. **Disable** the axis
6. Set Gantry.Parameters.BalanceOffset = 0
7. **Enable** the axis and wait for the ActualCurrentD to drop back to 0.
8. Stop the *Scope* and **Disable** the axis.

Note Don’t forget to reset the Gantry.Parameters.BalanceOffset to the application value, once the offset is configured correctly!

The recorded Scope should contain both phasing procedures, similar to Figure 5.

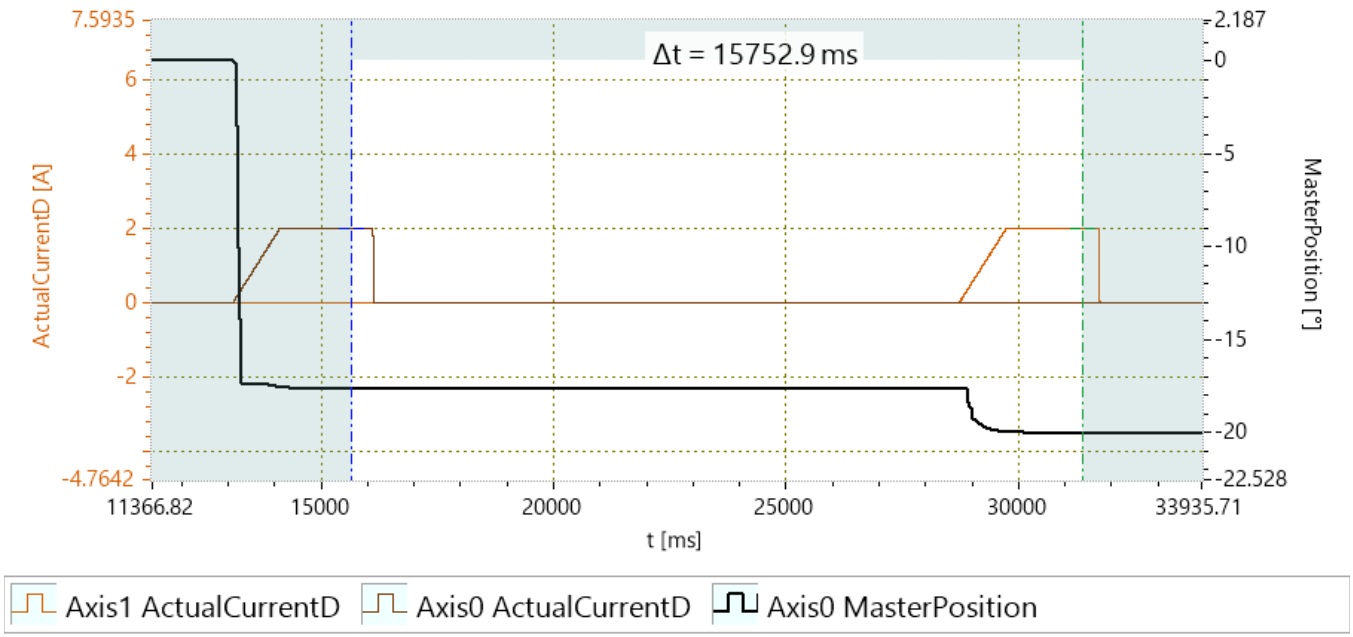

Figure 5: Phasing Offset Measurement

Now find the position difference of the two settled phasing states, using the cursors.

1. Place the cursors into the constant current region of the phasing, where the position is settled.
2. Activate the *Δy* option from the Scope Analysis drop down menu.
3. Calculate the *Offset*, using the below formulas, dependent on the motor type.
4. Subtract the resulting *Offset* from Axes[1].Parameters.Commutation.Angle.

Linear motor configured in mm:

$$\text{Offset} = \frac{\Delta y\,[\text{mm}]}{\text{PolePairPitch}\,[\text{mm}]} \cdot 2 \cdot \pi$$

Rotational motor configured in radian:

$$\text{Offset} = \Delta y\,[\text{rad}] \cdot \text{PolePairs}$$

Rotational motor configured in degrees:

$$\text{Offset} = \Delta y\,[^\circ] \cdot \text{PolePairs} \cdot \frac{\pi}{180}$$

Note	Dependent on the axis units, the value must be converted. The resulting *Offset* must be in radians.
Note	The sign of *Offset* depends on the electrical motor direction. If Axes[].Parameters.Motor.InvertDirection is changed, the sign of *Offset* must be changed as well.

2.5 Position Controller

For the tuning of the position controller the following sequence is recommended:

- Open the **Measure...** window in the **Frequency Response** module of **Axis 0**.
- Set the **Method** to **OnePositionTwoMotors**.
- Set all parameters, execute the measurement and save it.
- For the tuning of the position controller, click on the **Tune...** button in the **Frequency Response** Module for **Axis 0** and load the corresponding frequency response measurement. Then setup the controller as described in [1].
- Also refer to [1] on how to verify the controller tuning in time domain.

2.6 Acceleration Feed Forward

The acceleration feed forward of the axis can also be determined or verified based on the frequency response measurement and needs to be set in register Axes[0].Parameters.PositionController.FeedForwardAcceleration (see also [1]). The register is accessible in the **Frequency Response** module directly. Verify the acceleration feed forward as described in [1] chapter 5.5.4. Here the current Ganry.Signals.ActualCurrentQLin must be compared with the feed forward current.

2.7 Completion of Commissioning

Phasing

For first commissioning, the PhasingMethod was set to RotorAlignment. This causes a movement when the phasing is executed. With the following adjustments in node Parameters.Commutation of Axes[0] and Axes[1], this movement can be avoided:

- Set PhasingMethod to AngleSearch
- Set RampRiseTime to about 0.1s and RampConstTime to 0.5s.

If the phasing with this new setup works as expected, the EnablingMethod can be set to Automatic. With this, the phasing will only be executed after a restart of the servo drive. See also [1] for further information on phasing validation.

Balance Gain

The balance gain defines the load distribution to the two motors. For a symmetric setup the balance gain is 0.5. With *OnePositionTwoMotors,* this parameter may only change with special axis architecture or when combining different motor types.
For axes with gantry topology this gain is set according to the location of the center of mass. In this case, follow the recommendations in [2] in addition to this document.

Save the Configuration

When the configuration of the axis is done and tested, it is recommended to persist the configuration on the servo drive and to save it on the PC. The basic commissioning is done and the axis should be ready for application specific tuning and integration procedures.

References

[1] “Servo Drive Setup Guide”, ServoDrive-SetupGuide_EP020.pdf, Triamec Motion AG, 2025.
[2] “MIMO Gantry Commissioning”, AN139_MIMOGantryCommissioning_UserGuide_EP006.pdf, Tramec Motion AG, 2023

Revision History

Version	Date	Editor	Comment
001	2023-03-21	sm	Initial edit
002	2024-04-05	ab	Review, additions in 2.2, 2.3 and 2.6, correction in 2.4
003	2024-05-15	yz	Correction in Current Controller setup, Replaced Bode with Frequency Response

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.