AN53: TMC4671 PI Tuning

Document Revision V1.2 • 2019-June-24

This appnote describes how to tune the TMC4671 using the USB-2-RTMI (RTMI) step by step. The interface converter is based on the FTDI FT4222H USB High-Speed to SPI bridge. It is USB bus powered and offers the same tiny 10pin connector with same pin assignment for the RTMI interface as can be found on the TMC4671 evaluation board. The TMCL-IDE provides the software tools to tune the different control loops. Thus the RTMI is an easy way to debug, monitor, and tune the motor drive in system.

Contents

TRINAMIC
MOTION CONTROL

1   Items used

- BLDC motor (e.g. QBL4208-61-04-013-1024-AT)
- TMC4671-EVAL-Kit
- USB-2-RTMI
- TMCL-IDE (3.0.24)
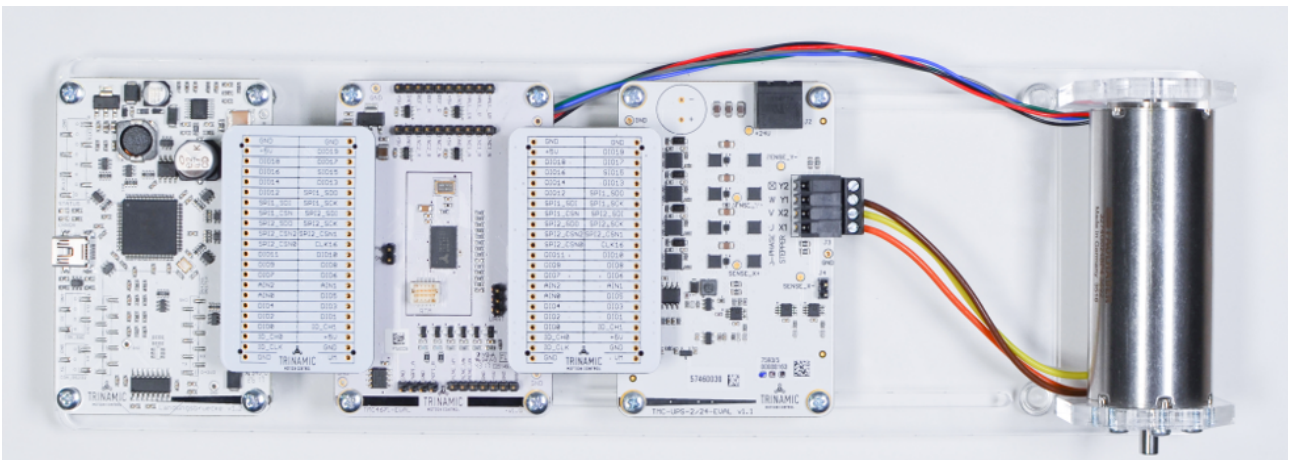- Power Supply (24V)
- Micro-USB Cable
- Mini-USB-Cable
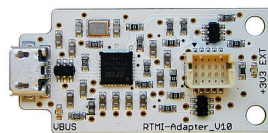

Figure 1: TMC4671-EVAL Kit and BLDC motor

Figure 2: USB-2-RTMI

2   USB-2-RTMI Driver Installation

Refer to USB-2-RTMI guide for driver installation.

3   Basic Configuration

Before using the Tuning tool, the TMC4671 must be configured (e.g. with the TMC4671 Wizard).
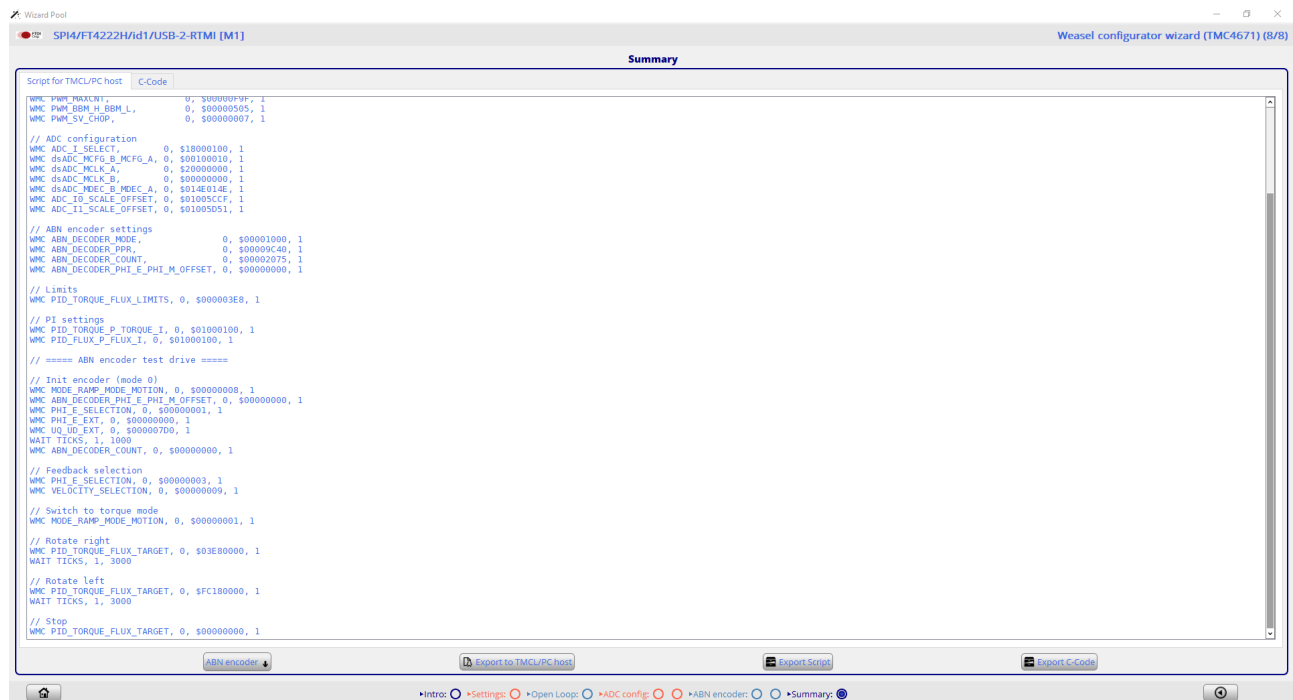

Figure 3: TMCL-IDE: TMC4671 Wizard - configuration code

After the basic configuration the parameters can be saved as C-Code or .tpc script

- In the summary choose ABN encoder
- Export the Script with *Export Script*

4   Tuning

4.1   Overview

The TMC4671 supports three main modes of operation, which requires PI Tuning:
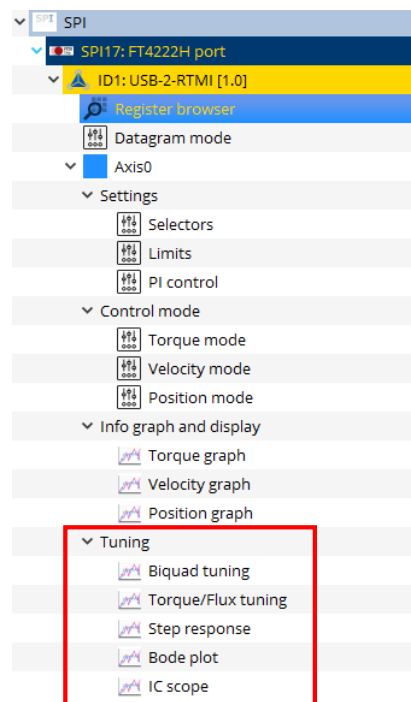
- current mode
- velocity mode
- position mode

Figure 5: PI tuning tools

Each mode can be tuned by the PI controller for every loop. Below picture gives an overview:
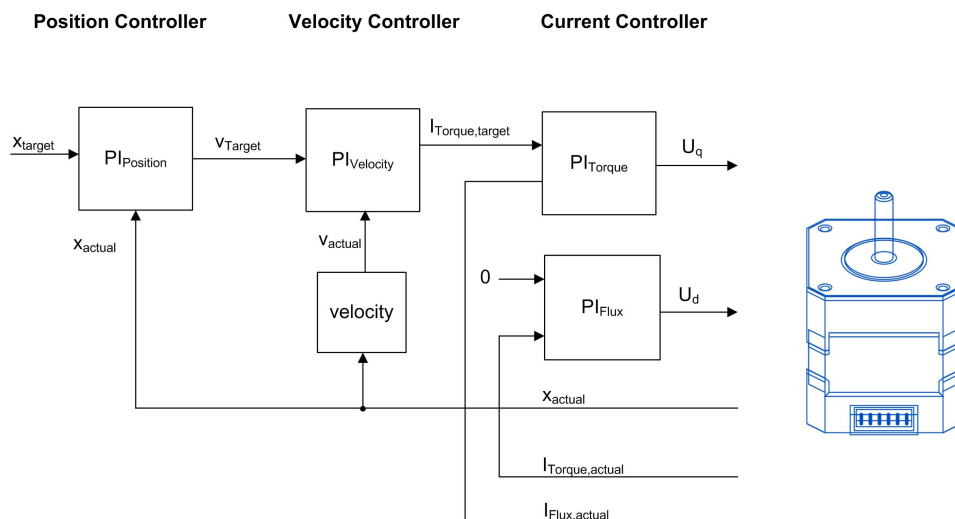

Figure 4: TMC4671 control loops

The loops are cascaded, thus the outer loops depends on the tuning of the inner loop. For example the current loop must be tuned before using the velocity loop. For the tuning of every PI controller tuning tools are provided. In the TMCL-IDE they can be accessed through the Tuning group
 The Tuning tools are:

• Biquad tuning: Alows to filter the target values of each loop

• Torque/Flux tuning: Identifies the PI parameters by open loop step response

- Step response: Closed Loop step responses for all control loops

- Bode plot: Bode plot of all loops

- IC scope: Monitoring, Read out of register values with PWM frequency

4.2 Limits

- Before using the Tuning tools it is recommended to set the output of the control voltage to maximum (set the PIDOU_UQ_UD_LIMITS = 32767). For stepper the default value is sufficient.

- Set on the PID_TORQUE_FLUX_LIMITS to limit the maximum current of the aplication.

- Set PID_POSITION_LIMIT_HIGH = 2 147 400 000 (don't exceed maximum register value)

- Set PID_POSITION_LIMIT_LOW = -2 147 400 000

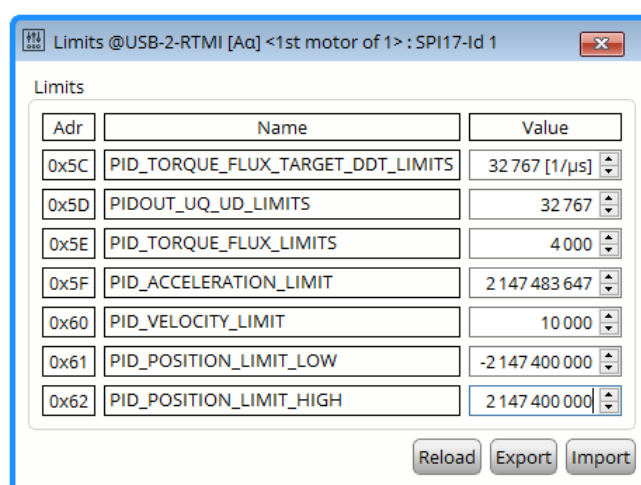| Adr	Name	Value
0x5C	PID_TORQUE_FLUX_TARGET_DDT_LIMITS	32 767 [1/µs]
0x5D	PIDOUT_UQ_UD_LIMITS	32 767
0x5E	PID_TORQUE_FLUX_LIMITS	4 000
0x5F	PID_ACCELERATION_LIMIT	2 147 483 647
0x60	PID_VELOCITY_LIMIT	10 000
0x61	PID_POSITION_LIMIT_LOW	-2 147 400 000
0x62	PID_POSITION_LIMIT_HIGH	2 147 400 000

Figure 6: TMCL-IDE: TMC4671 Limits

5 Tuning of the current loop

The current loop consists of two control loops: One for torque (current) and one for flux (current). Both loops can be tuned with the RTMI tools *Torque Flux/Tuning* tool and *Step response* tool. The Torque/Flux tool determines the PI parameters in an open loop mode. While the Step response tool is used to analyze the closed loop behaviour.

5.1 Torque/Flux Tuning Tool (Open Loop)

The Torque Flux/Tuning tool will automatically determine start values for the PI parameters by identifying the motor parameters. With this procedure, voltage steps are set and the current step response is evaluated. Only the flux current is stimulated for minimal/no motor movement in this tuning phase. It is recommended to use a PWM frequency of 25kHz for best identification results.

1. Connect the RTMI to the TMC4671-EVAL and the PC.

2. Open the Torque/Flux Tuning Tool.
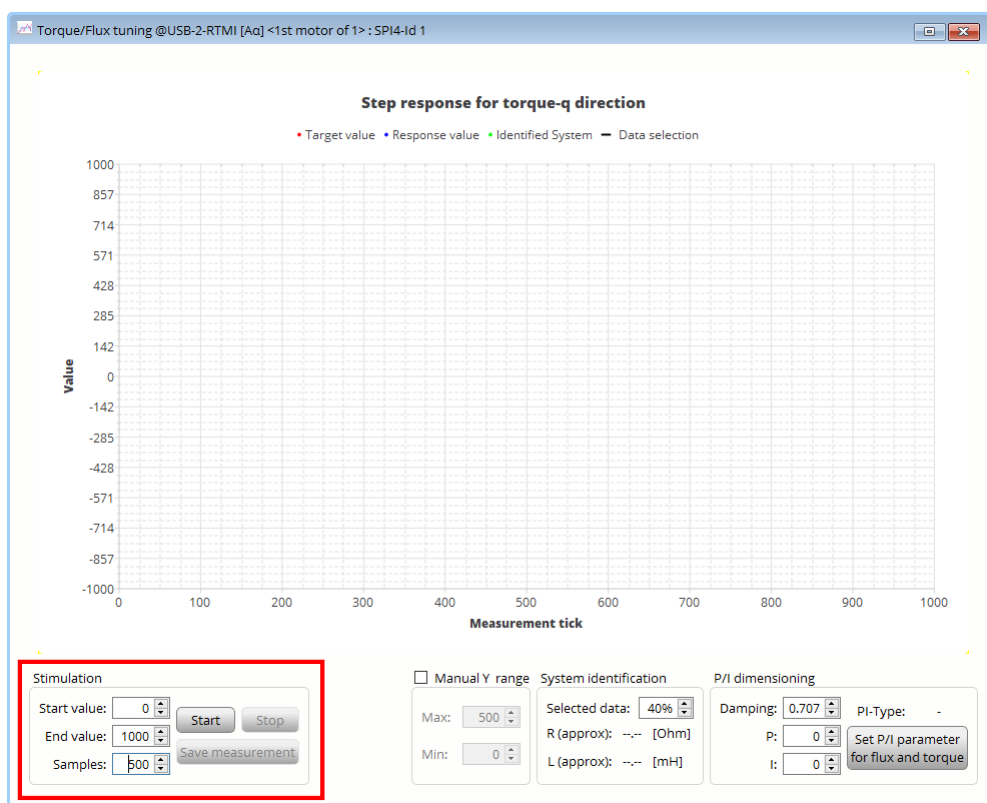
3. Start the identifcation by using the Start button.

Figure 7: Configuration and start of the step reponse

4. The response of the current on the voltage step is shown as well as the identified system
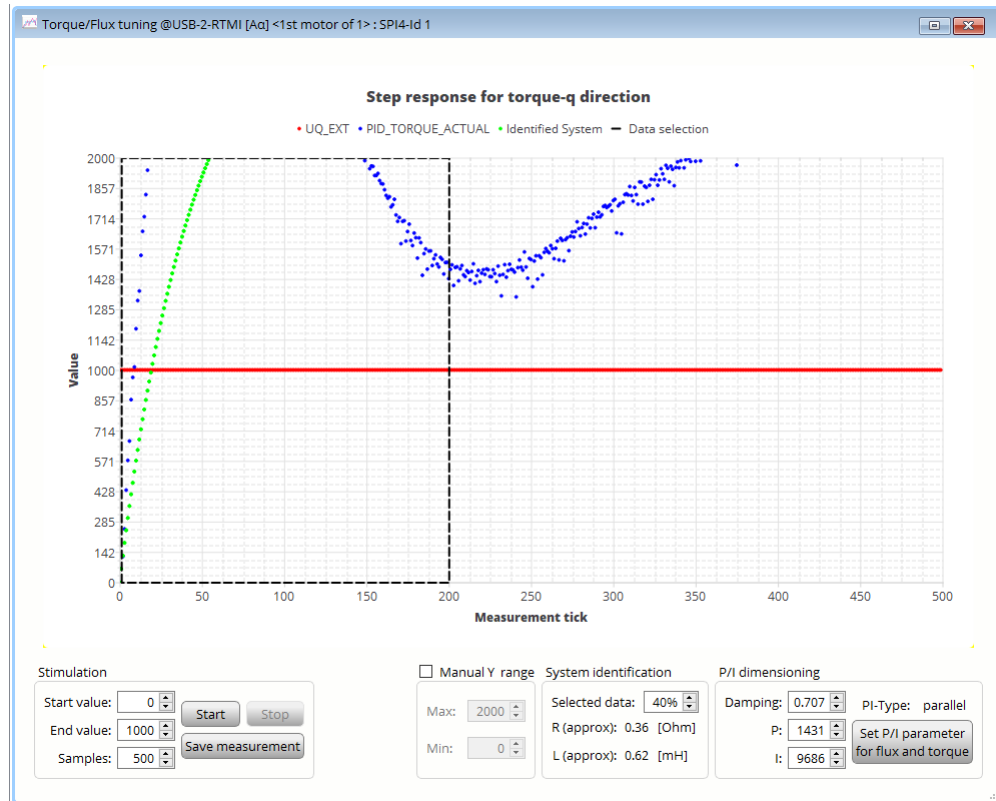

Figure 8: first step response

5. Check and adapt the Manual Y range for a better view

6. Re-Identify the system using the Start Button. After the motor has been aligned it will give better identification results.

7. Adapt the window of the Selected Data if needed. In most cases the default settings will give good results.

   • The dynamic step response should be covered in the identification area (dashed frame).
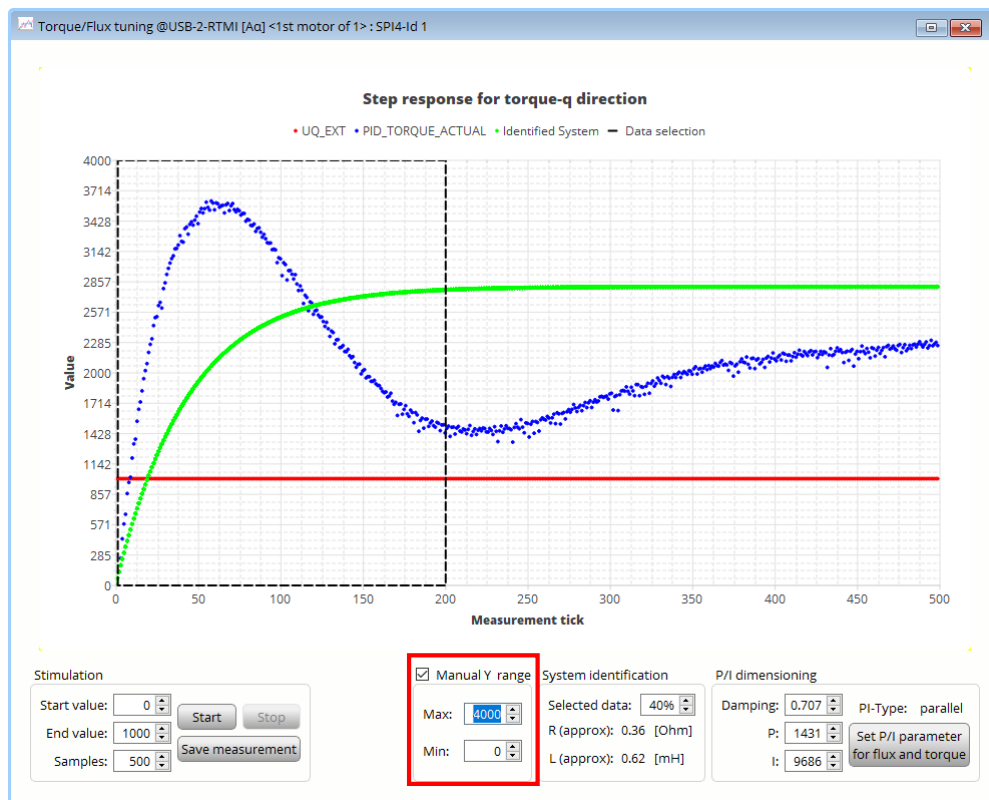
Figure 9: adapted Y range
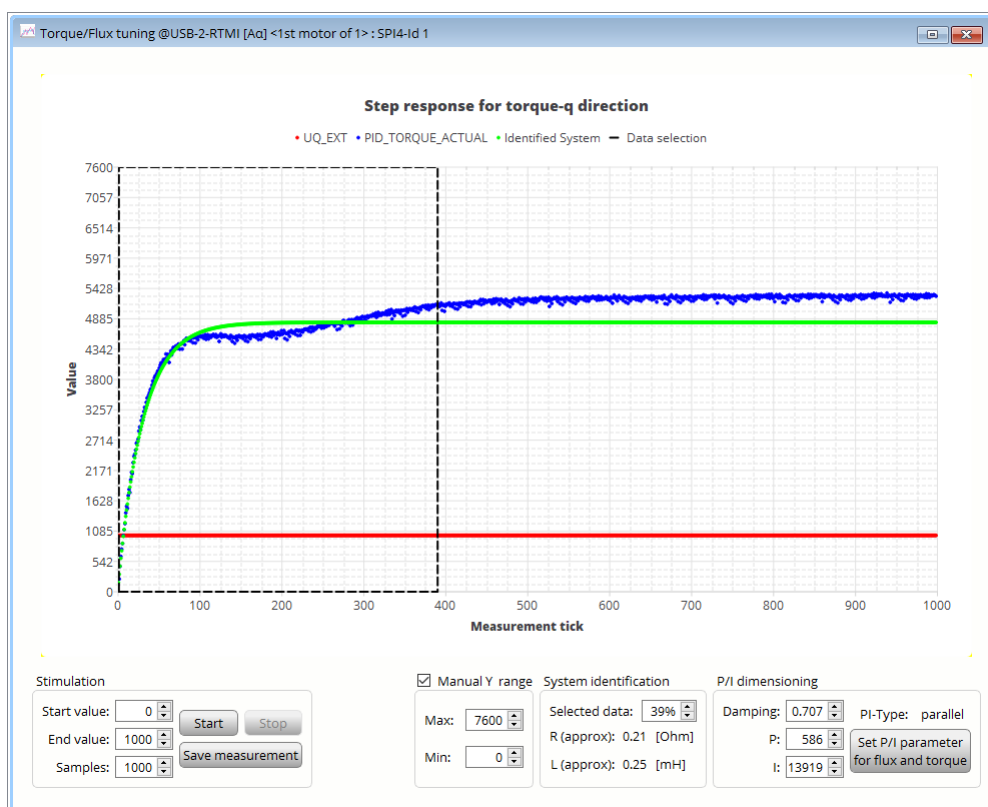
Figure 10: Step response: identification area good

- In figure 11 the step response is not fully covered. Identification area is too small.
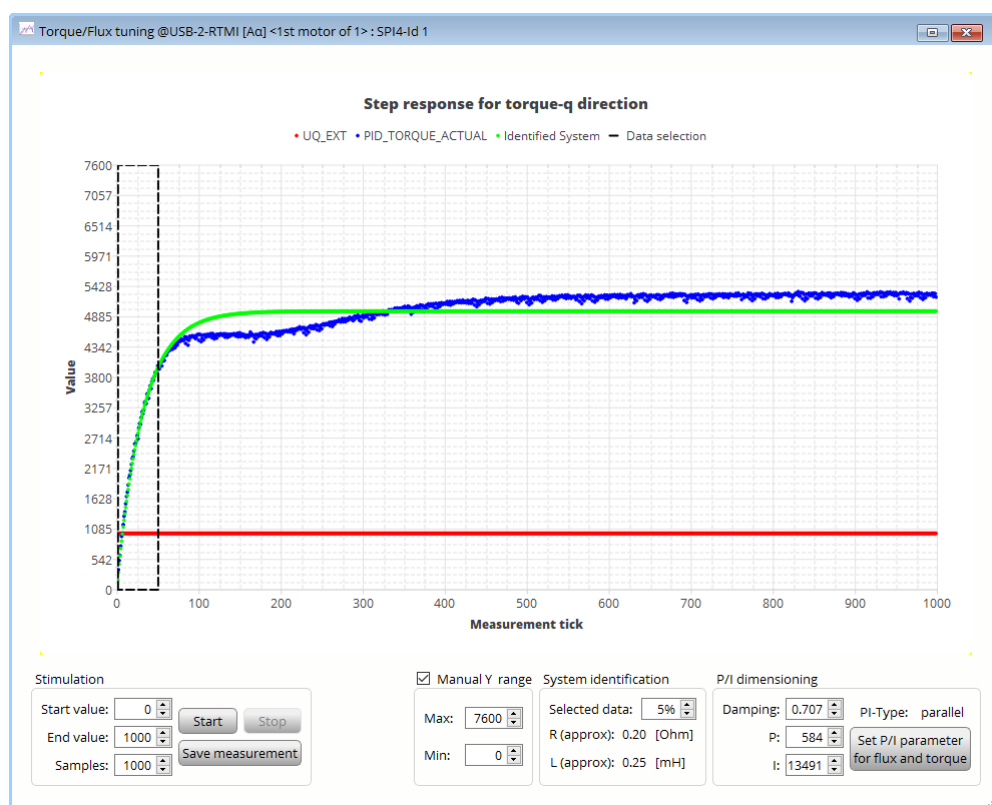
Figure 11: Step response: identification area too small

- In figure 12 the Identification area is too large
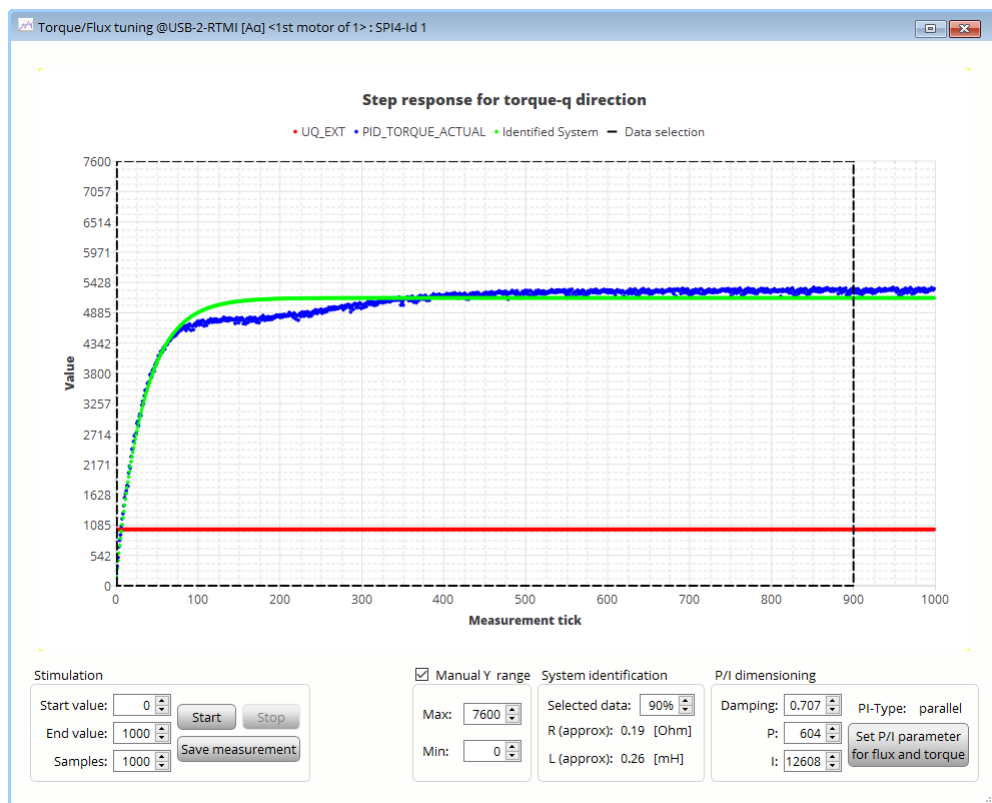
Figure 12: Step response: identification area too big
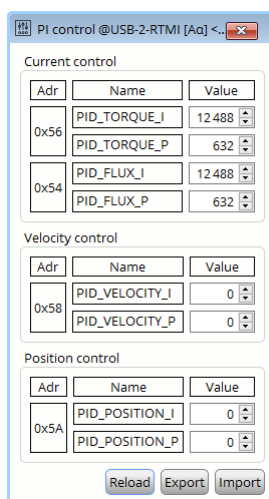
8. Write the identified PI values to the current controller

   • Use the following button to update the PI values to the current controller
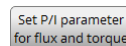
   • Note: For very high values identified I-parameter 0 is displayed. In this case the I parameter needs to be set manually (e.g. to 18000).

   • Confirm the new PI values with the PI control box. The Reload button may have to be used to update the values.

Figure 13: PI-Parameters

5.2   Step Response Toolbox (Closed Loop)

In previous step the PI parameters of the Torque/Flux current loop were determined in open loop mode. Now the Step response tool will be used to analyze the closed loop behaviour.

(a) Open the Step response toolbox
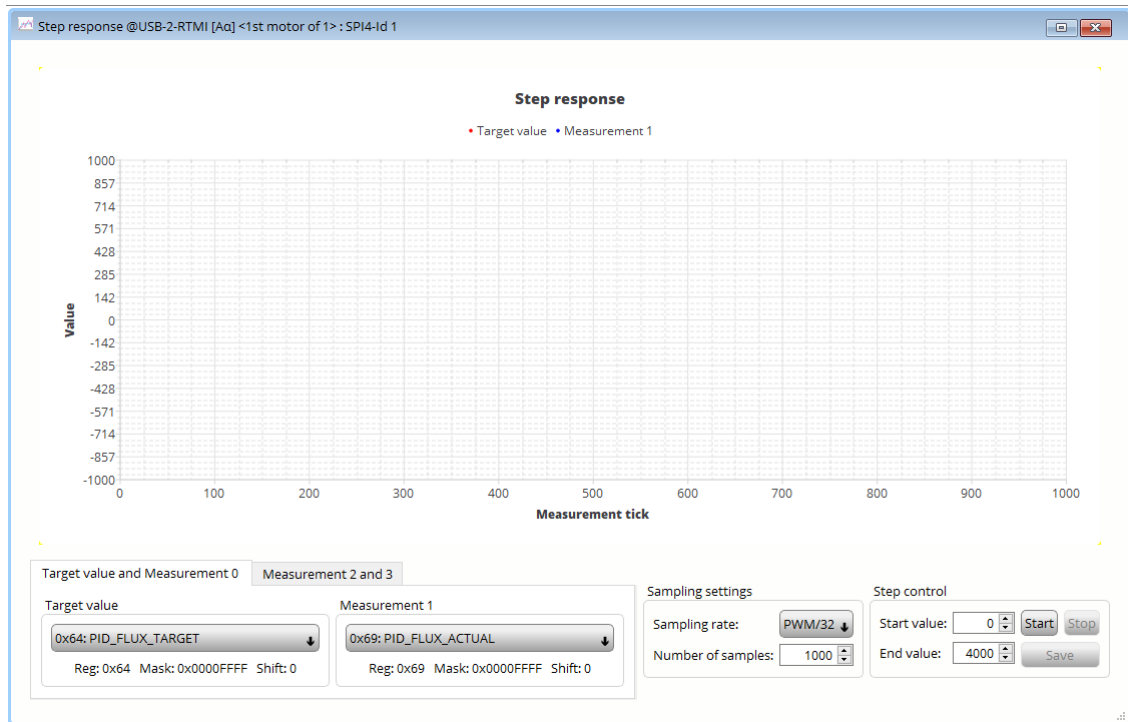
Figure 14: step response toolbox

(b) Configure the settings for flux current controller
  • Target value: 0x64 PID_FLUX_TARGET
  • Measurement 1: 0x69 PID_FLUX_ACTUAL
  • Sampling rate: PWM
  • Under Step Control
      Set Start value = 0
      Define end value for the flux target current (here 4000)
  • Start the step response with the start button

(c) The actual current and the target current is shown.
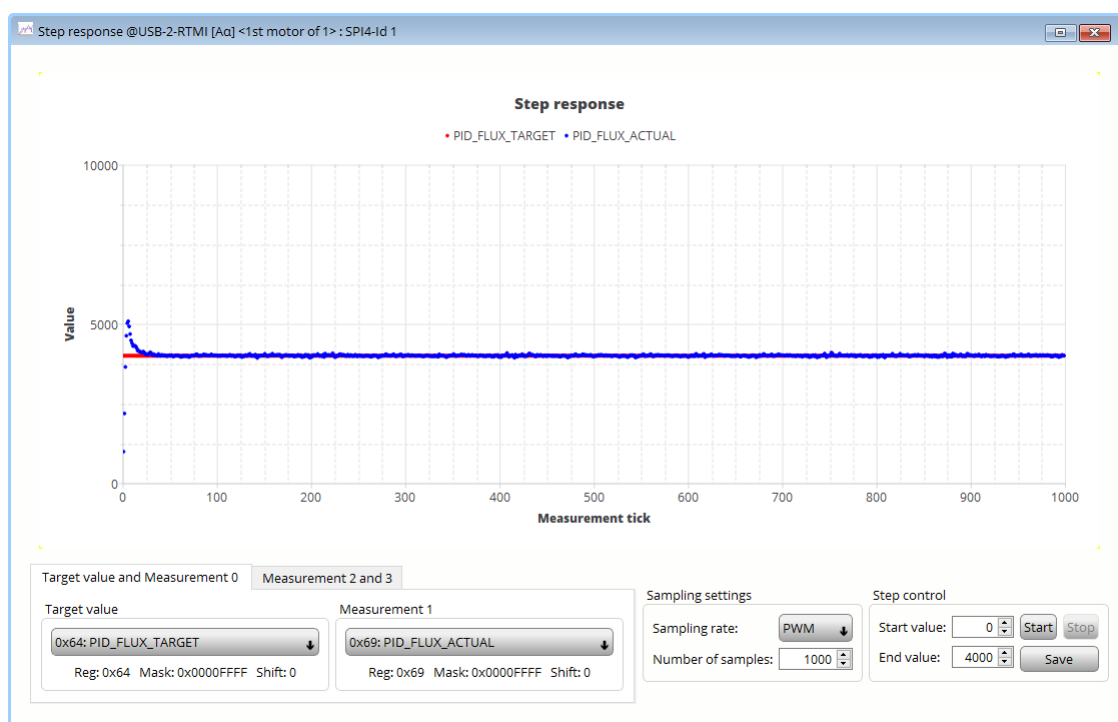
(d) Reduce the sample rate for better view
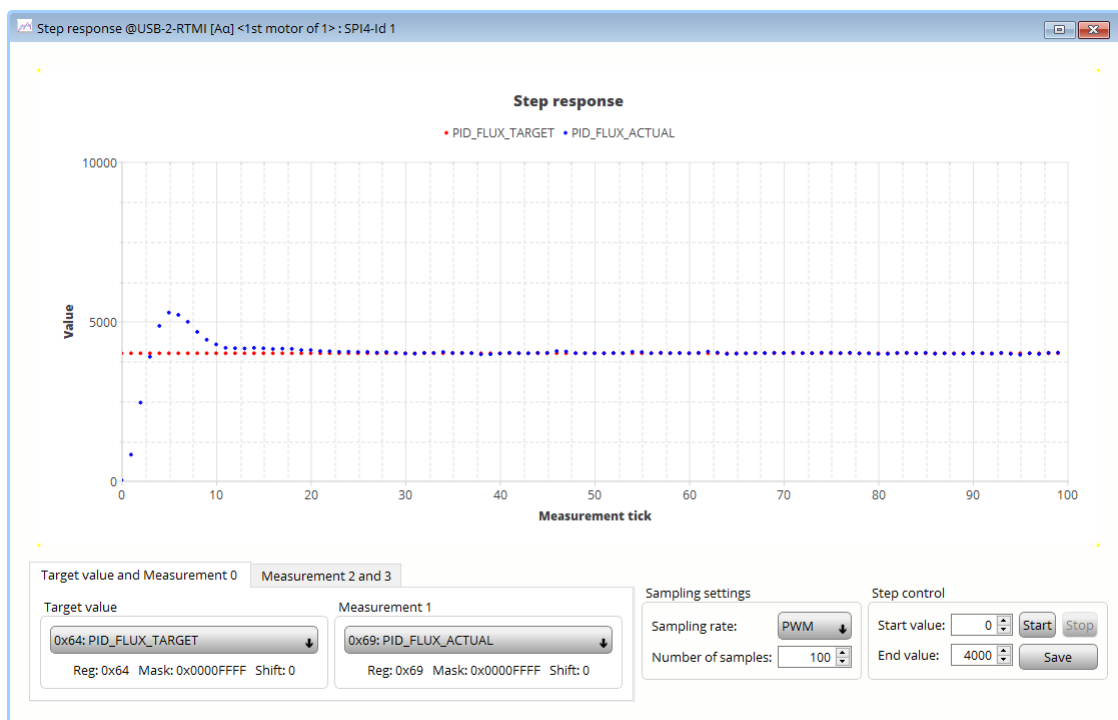
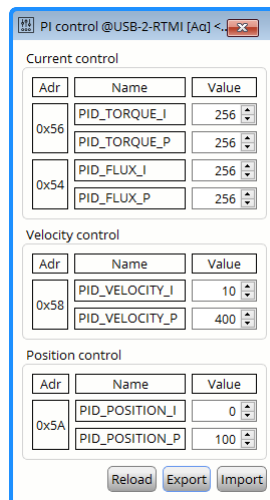(e) Optimise the behaviour by varying P and I

(f) Repeat from step 2. With PID_TORQUE_TARGET and PID_TORQUE_ACTUAL

5.3 Bode Plot

The Bode Plot tool is used to check the current loop dynamics.

(a) Open the Bode Plot tool

(b) Select 1: torque control loop

(c) Start the measurement with the Start button (leave other settings at default)

(d) Comparison example: Two below measurement show the bode plots with tuned and untuned/default PI parameter.

(e) Default PI parameter

Figure 15: PI parameter

Figure 16: Bode diagram with default PI

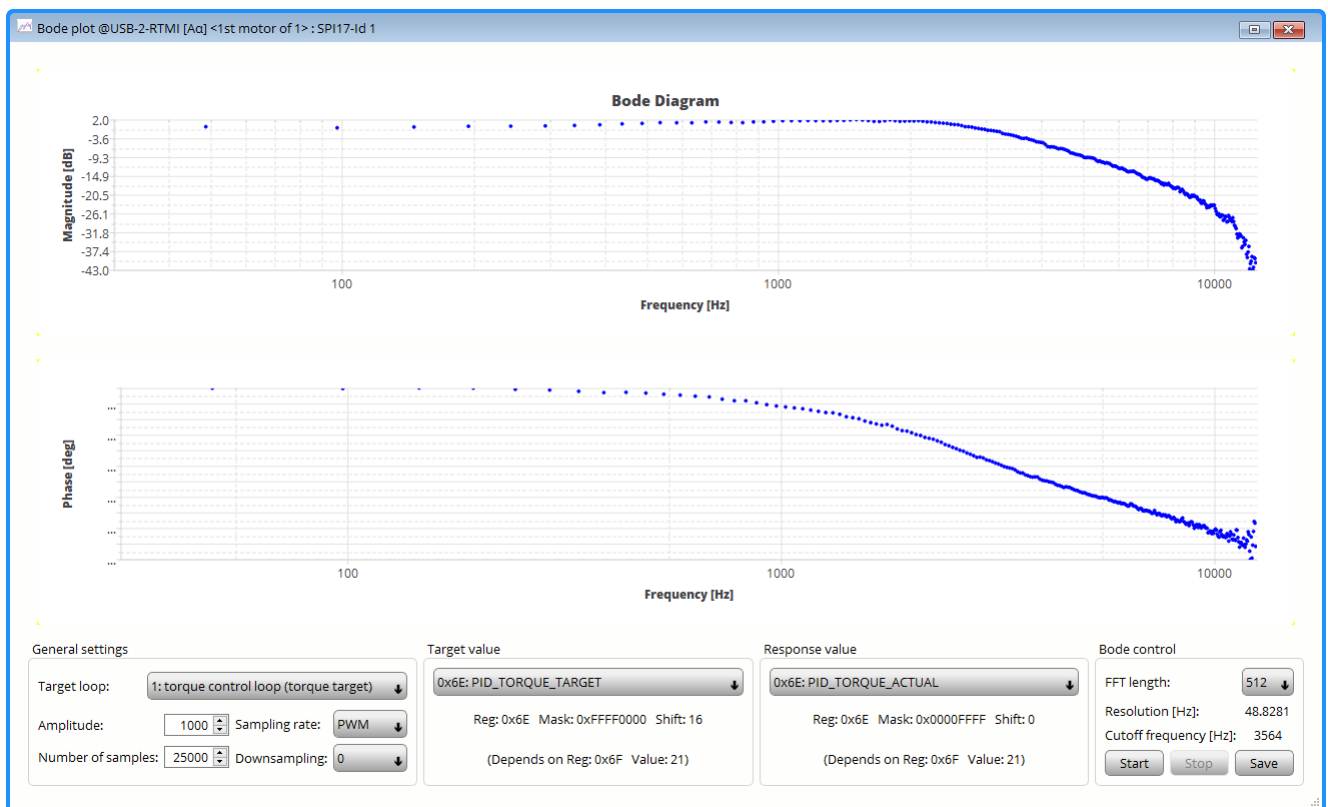(f) Tuned PI Parameter

Figure 17: PI parameter

Figure 18: Bode diagram with tuned PI

(g) The tuned PI control system has a higher cutoff frequency and thus higher dynamics.
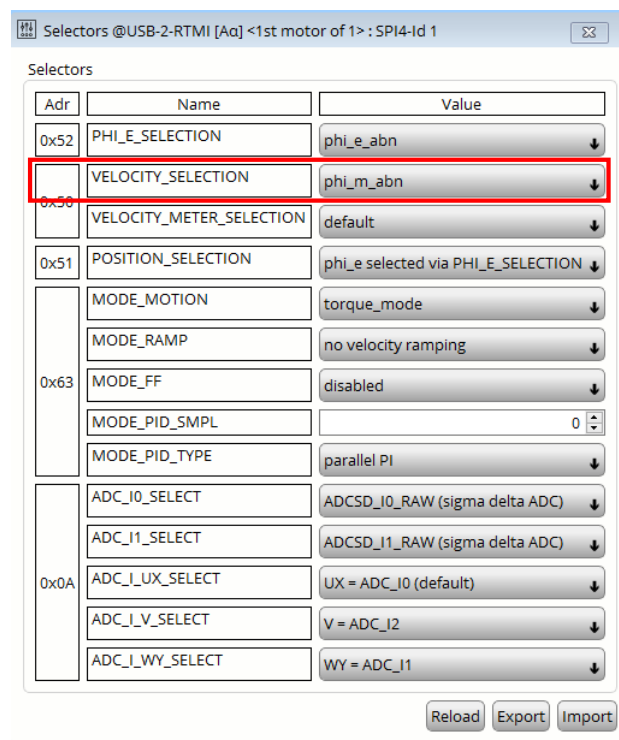
Figure 19: Selectors: Velocity Unit

6   Tuning of the velocity loop

In this section the velocity loop will be tuned. Precondition is the previous tuning of the current controller. To tune the velocity PI parameter the step response tool (closed loop) will be used.

(a) For ease of use the velocity unit will be altered. In the selectors toolbox VELOCITY_SELECTION (0x52) is changed from phi_e_selection to phi_m_abn.

> • phi_m: mechanical velocity is displayed and calculated in rpm
>
> • phi_e: electrical velocity is displayed and calculated in electrical periods

(b) Setting of the PI value for the velocity loop

Start with a low P value; set I = 0
   • 0x58: PID_VELOCITY_I = 0
   • 0x58: PID_VELOCITY_P = 100

Figure 20: PI Values

(c) Open *Step Response* tool box

(d) Use the example configuration Step Response
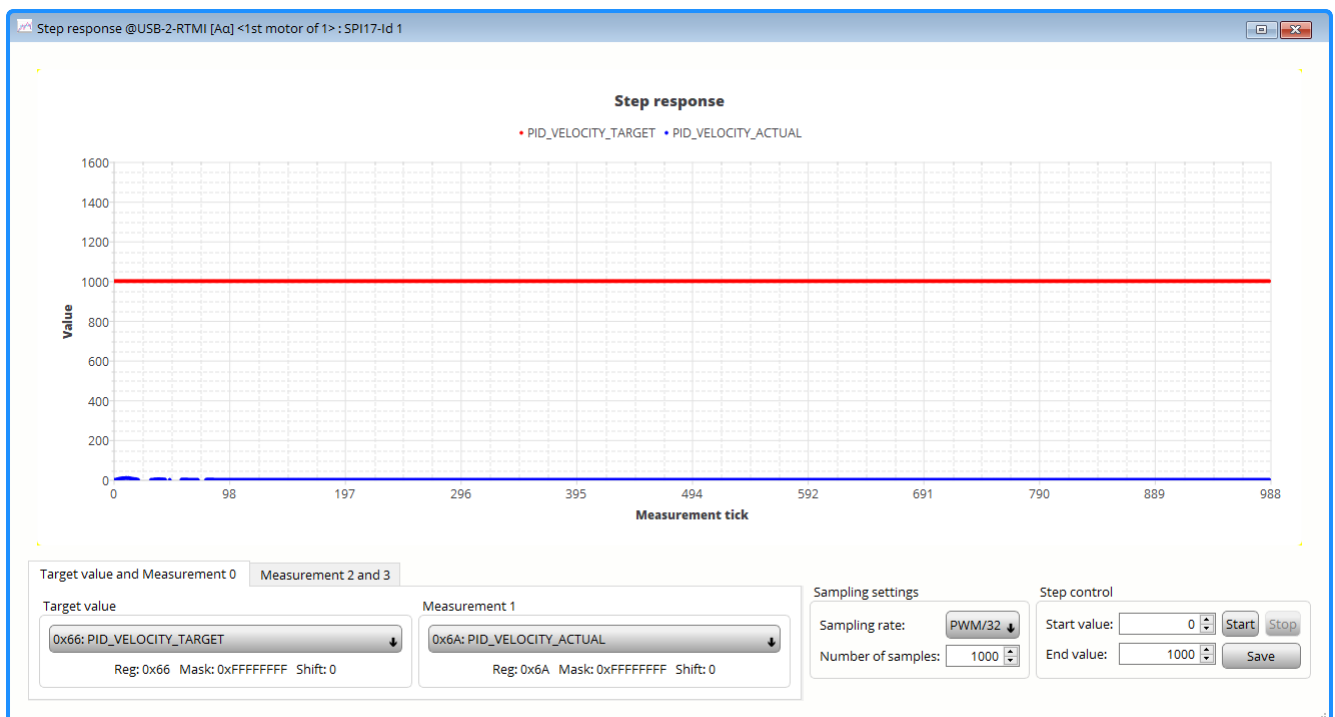
(e) Start the Step response with the Start Button

Figure 21: PID_VELOCITY_P = 100

(f) Gradually increase the PID_VELOCITY_P until the actual velocity (PID_VELOCITY_ACTUAL) reaches 50-75% of the target velocity (PID_VELOCITY_TARGET)

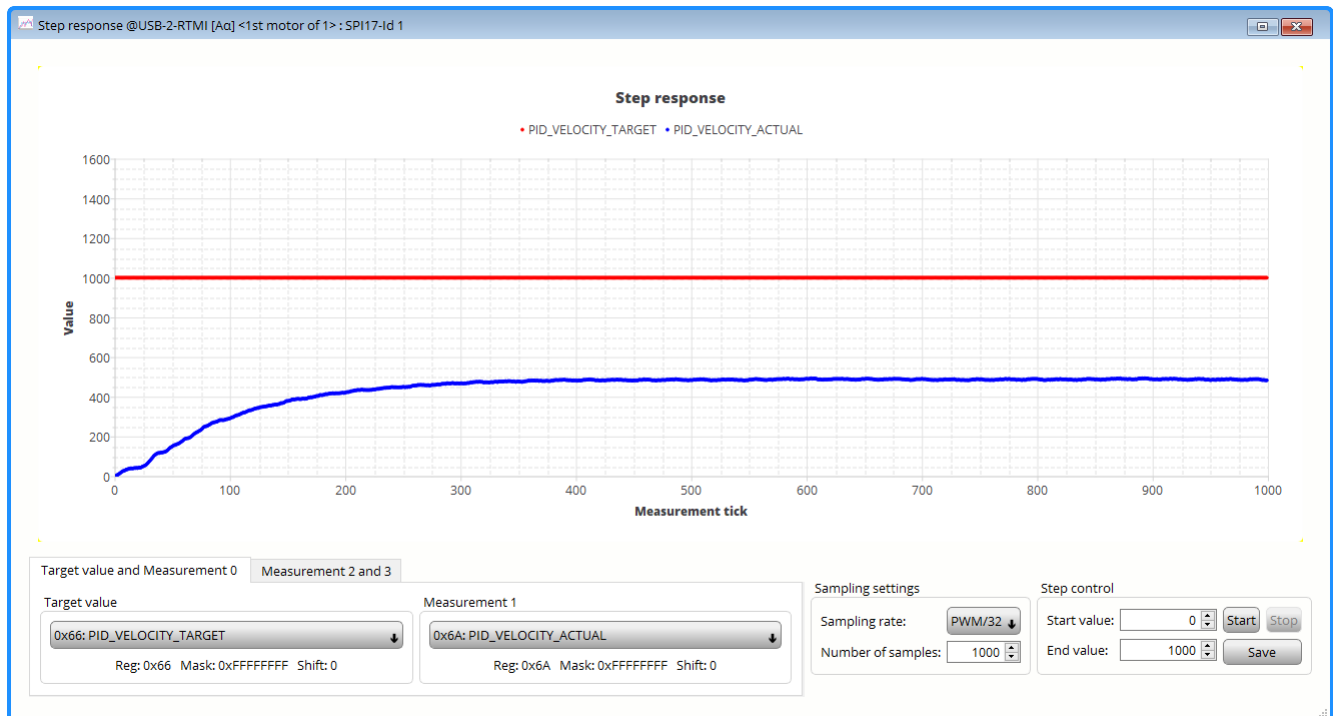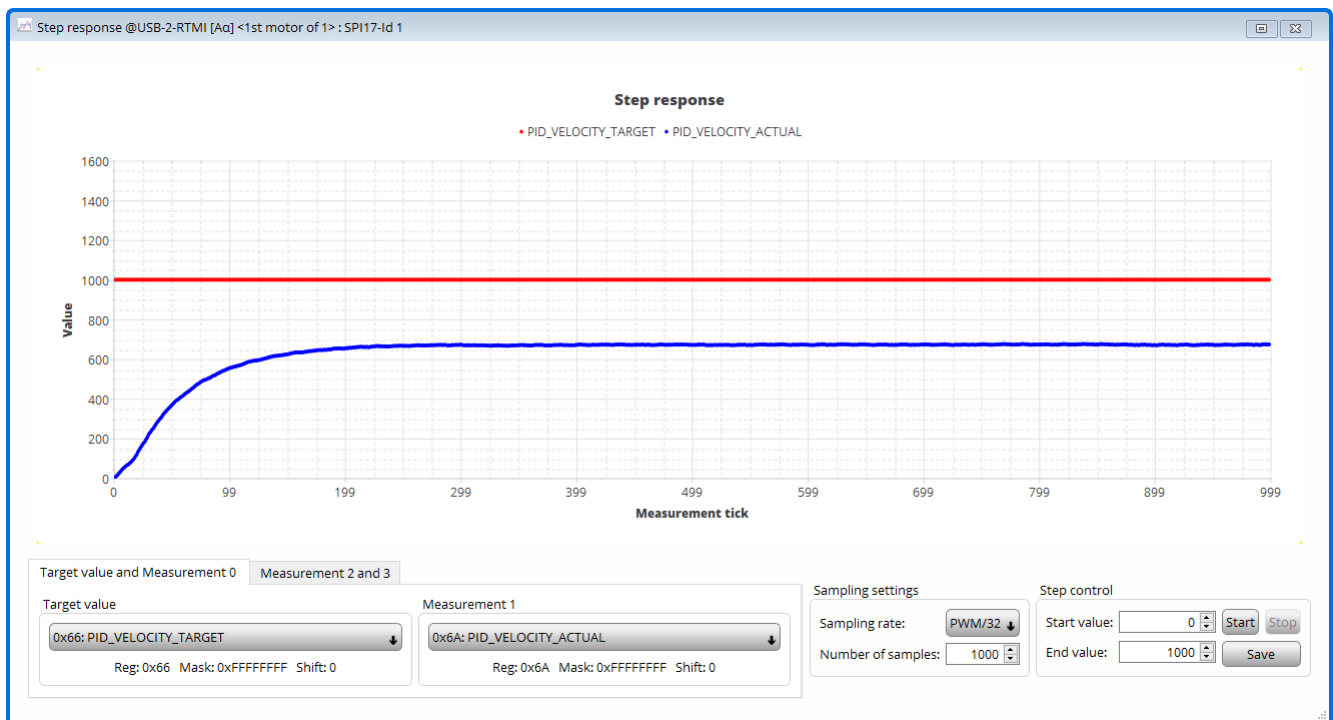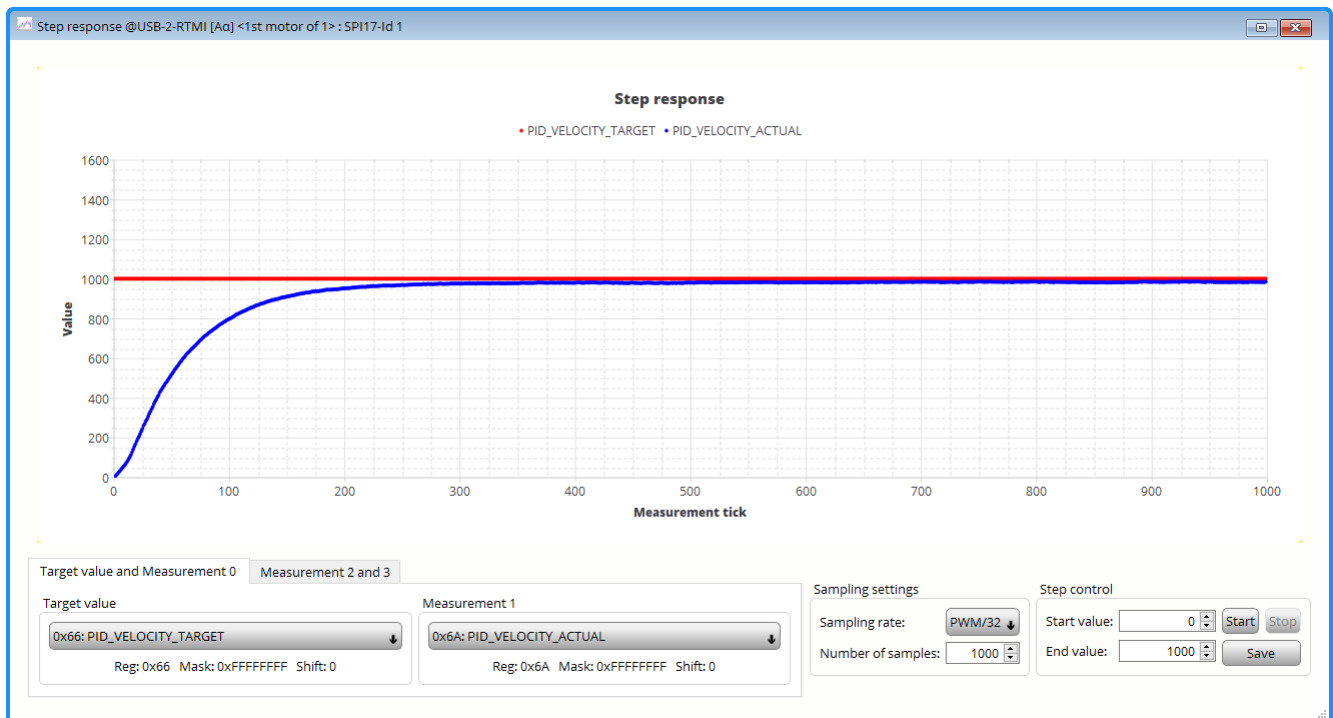*Figure 22: PID_VELOCITY_P = 300*

Figure 23: PID_VELOCITY_P = 500

(g) Increase PID_VELOCITY_I until the actual velocity reaches the target velocity
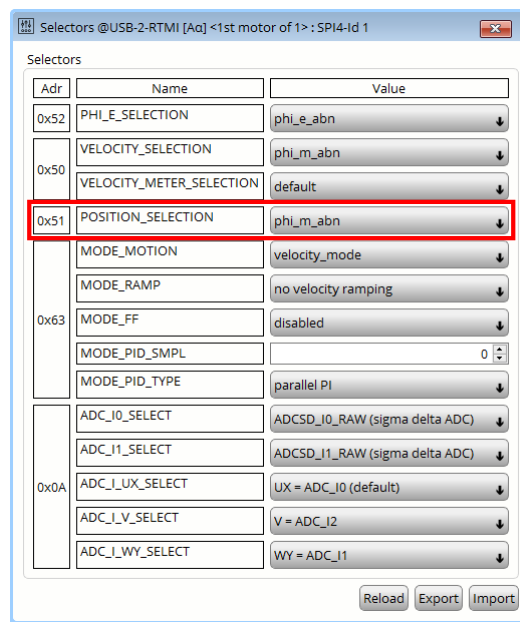

Figure 24: PID_VELOCITY_I = 10

Figure 25: Selectors: Position unit

7   Tuning of the position loop

In this section the position loop will be tuned. Before the position loop can be used, the current loop and the velocity loop needs to be tuned. For the tuning the step response tool is used. After that the motion controller of the Landungsbruecke evaluationboard is used to increase dynamics and accuracy.

7.1   Step Response Tool

(a) Set POSITION_SELECTION to phi_m_abn
     Thus one mechanical revolution equals 65535 counts

(b) Set starting value for PI Position controller
     • PID_POSITION_I = 0 (for most setups it is recommended to set to this to 0)
     • PID_POSITION_P = 10

Figure 26: PI parameter

(c) Open the step response tool

(d) Start the step response with Start button. The Motor will move back and forth the end value. In this case 1 revolution.
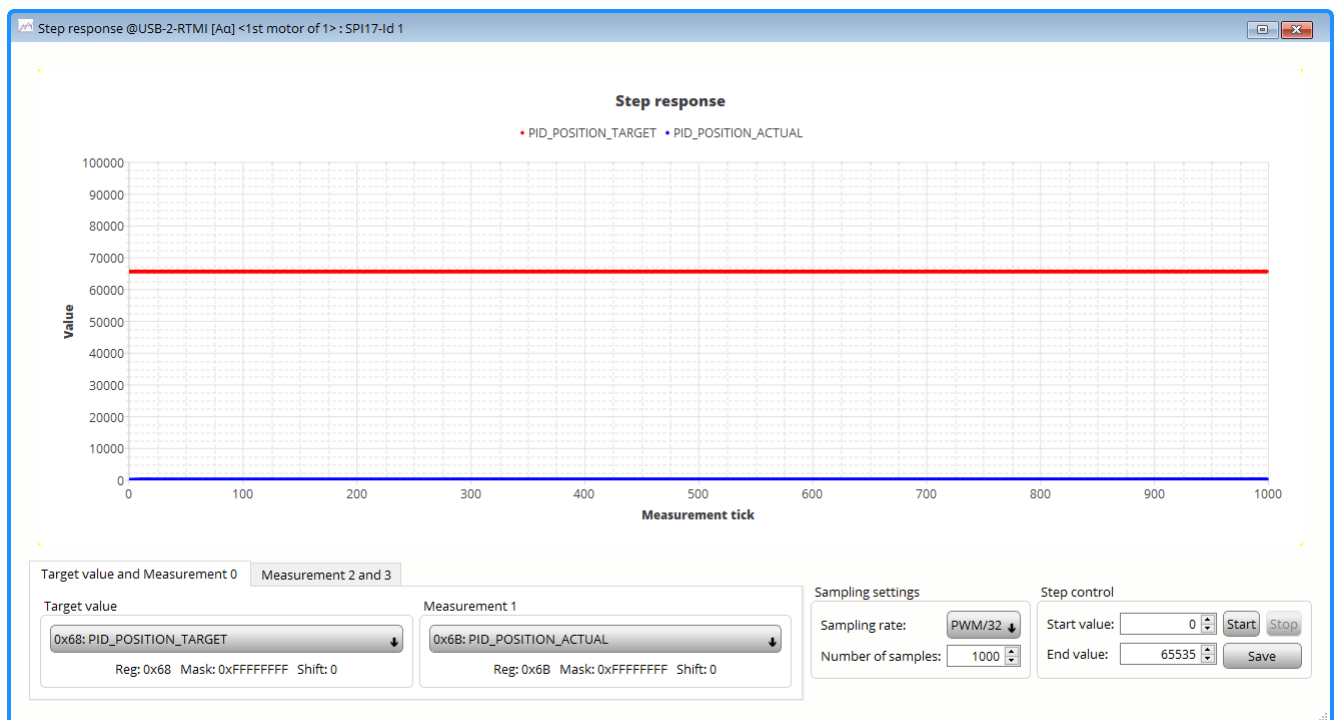
(e) No motor movement with PID_POSITION_P = 10

Figure 27: PID_POSITION_P = 10
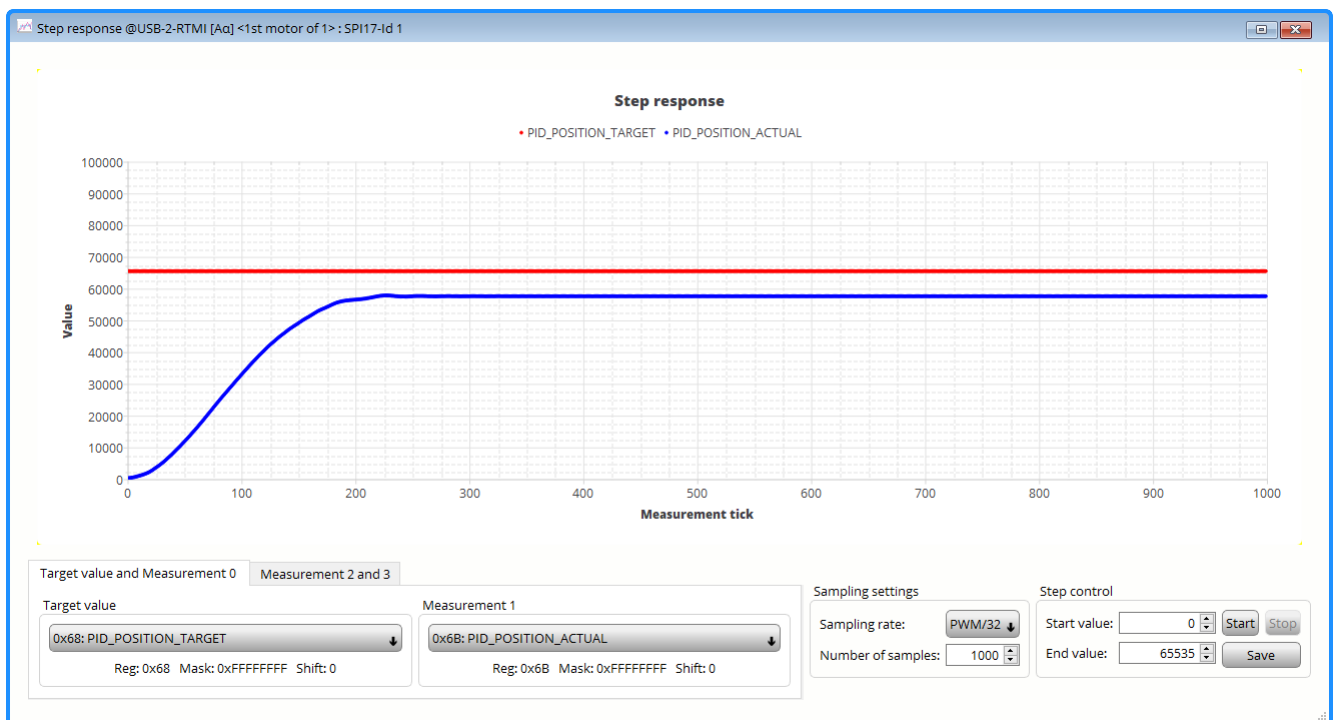
(f) Increase P parameter: PID_POSITION_P = 50

Figure 28: PID_POSITION_P = 50
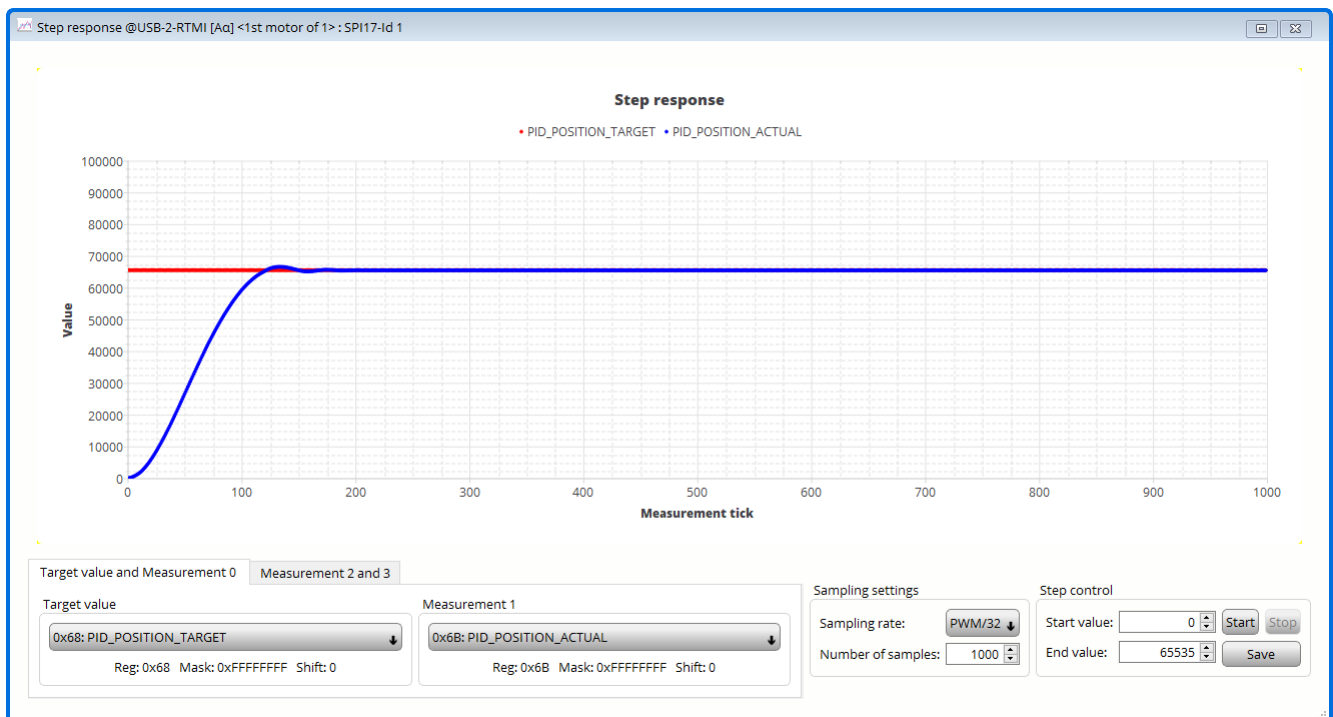
(g) Incrase P parameter: PID_POSITION_P = 100

Figure 29: PID_POSITION_P = 100

(h) PI configuration after tuning.

Figure 30: PI parameter

7.2   Motion Controller

The Landungsbrücke features trapezoidal ramps which can be used with the TMC4671-EVAL. The TMC4671 does not integrates a motion controller.

(a) Connect the Landungsbruecke to the PC with a Mini USB cable

(b) In the Landungsbruecke dialog (not available USB-2-RTMI) open Position Mode toolbox

(c) Enable Velocity Ramp and limit acceleration

(d) The motor is turned 10 revolution with the position mode toolbox
   • *Clear* the position
   • Set Target pos. = 655350
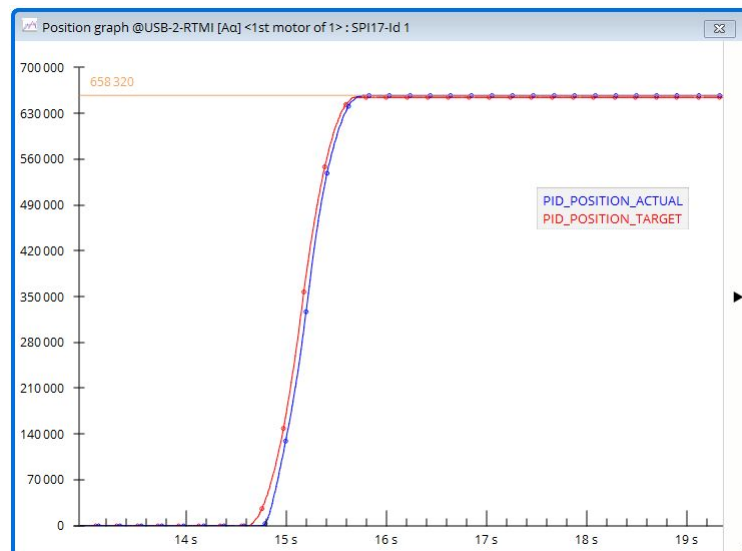   • Start the movement with the Absolute button

Figure 31: Movement with ramp
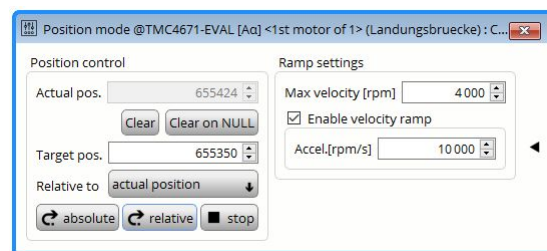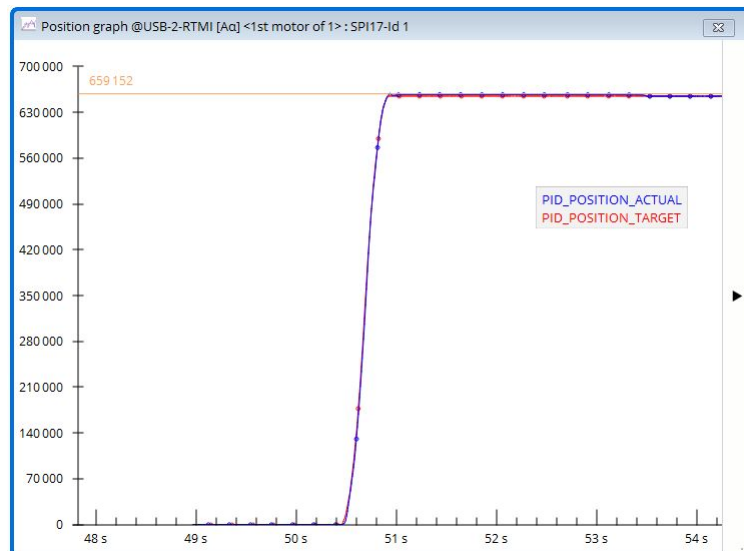
(f) Increase the acceleration

Figure 32: Movement with ramp

(g) Repeat the motion of 10 revolution with MC and increased accelerations settings, the PI values can be adapted as well:

(h) PI configuration after adding motion controller:

Figure 33: PI parameter

(i) Save PI configuration with the Export option

8 Summary

The RTMI interface of the TMC4671 provides a powerful option to tune the IC. Fast realtime access to the IC, allows tuning of the control loops and monitoring. The TMCL-IDE provides ready to use software tools for in system developement. For more details on the TMC4671 and how to use, please look into the TMC4671 datasheet.

9   Revision History

Version	Date	Author	Description
V1.0	25.04.2019	JPX	Inital version
v1.1	03.05.2019	ED, JPX	refinement
v1.2	24.06.2019	JPX	changed header, changed PID_POSITION_LIMIT_X values

Table 1: Document Revision