EasyMotion Studio II

Quick Setup and Programing Guide

TECHNOSOFT

Technical Reference

1 About EasyMotion Studio II

EasyMotion Studio II is the software suite needed to perform the setup and motion programming of Technosoft intelligent drives and motors. The output of EasyMotion Studio II is a set of setup data and a motion program, which can be downloaded to the drive/motor EEPROM or saved on the PC for later use.

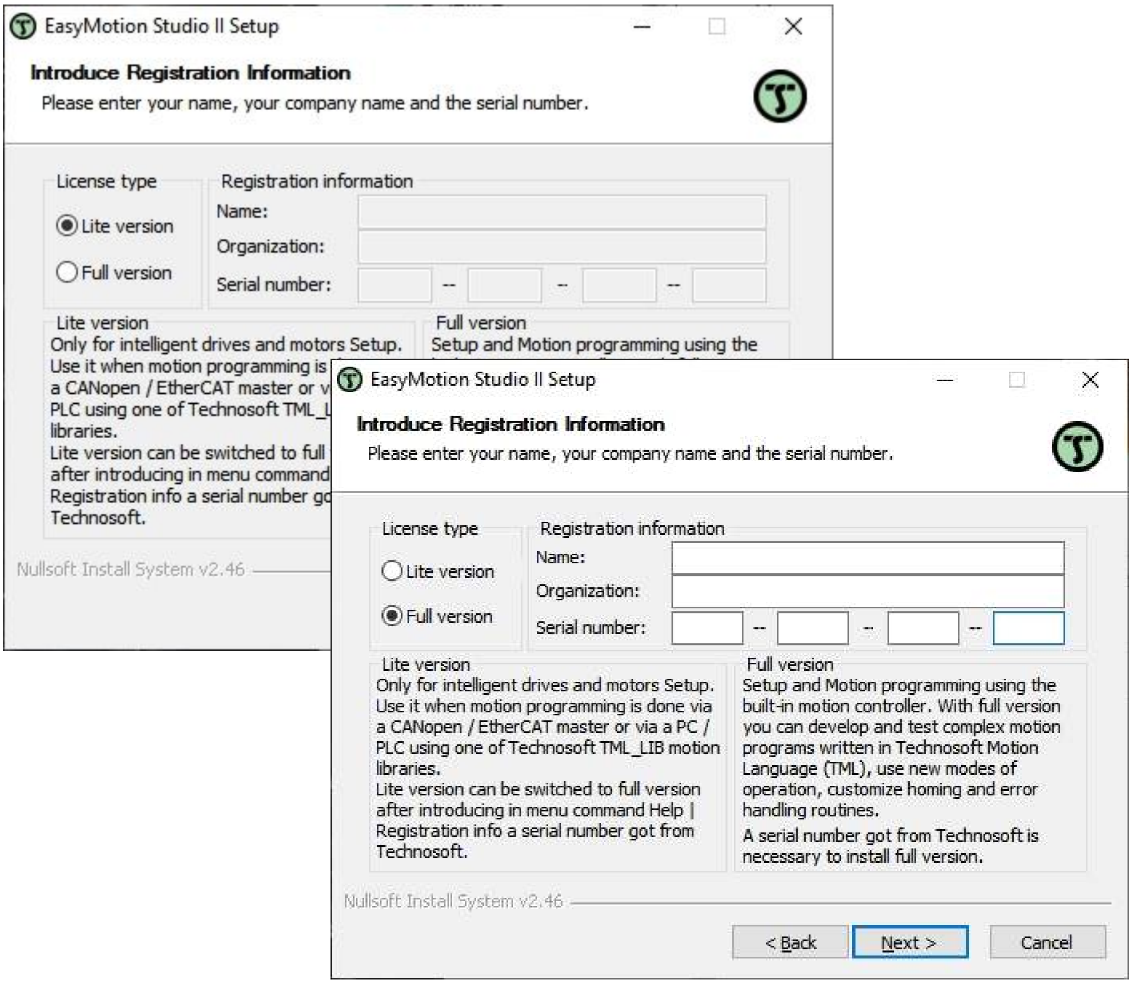
The setup contains all the information needed to configure and use a Technosoft drive/motor. This information is preserved in the drive/motor EEPROM as the *setup table*. The setup table is copied at power-on or reset into the RAM memory of the drive/motor and is used during runtime.

The motion component contains the implementation of the user/application requirements. This is described via a TML (Technosoft Motion Language) program, which is executed by the drives/motors built-in motion controller.

EasyMotion Studio II also includes a set of evaluation tools like Data Logger, Control Panels and the Command Interpreter which help to quickly test, measure and analyse the motion application.

2 Getting started

EasyMotion Studio II can be downloaded from Technosoft web page (link) and installed as **Lite version** (free of charge) or as **Full version** (it requires a serial number).



EasyMotion Studio II Setup

Introduce Registration Information

Please enter your name, your company name and the serial number.

License type
● Lite version
○ Full version

Registration information
Name:
Organization:
Serial number: -- -- --

Lite version
Only for intelligent drives and motors Setup.
Use it when motion programming is
a CANopen / EtherCAT master or v
PLC using one of Technosoft TML_L
libraries.
Lite version can be switched to full
after introducing in menu command
Registration info a serial number go
Technosoft.

Full version
Setup and Motion programming using the

Nullsoft Install System v2.46

EasyMotion Studio II Setup

Introduce Registration Information

Please enter your name, your company name and the serial number.

License type
○ Lite version
● Full version

Registration information
Name:
Organization:
Serial number: -- -- --

Lite version
Only for intelligent drives and motors Setup. Use it when motion programming is done via a CANopen / EtherCAT master or via a PC / PLC using one of Technosoft TML_LIB motion libraries.
Lite version can be switched to full version after introducing in menu command Help | Registration info a serial number got from Technosoft.

Full version
Setup and Motion programming using the built-in motion controller. With full version you can develop and test complex motion programs written in Technosoft Motion Language (TML), use new modes of operation, customize homing and error handling routines.
A serial number got from Technosoft is necessary to install full version.

Nullsoft Install System v2.46

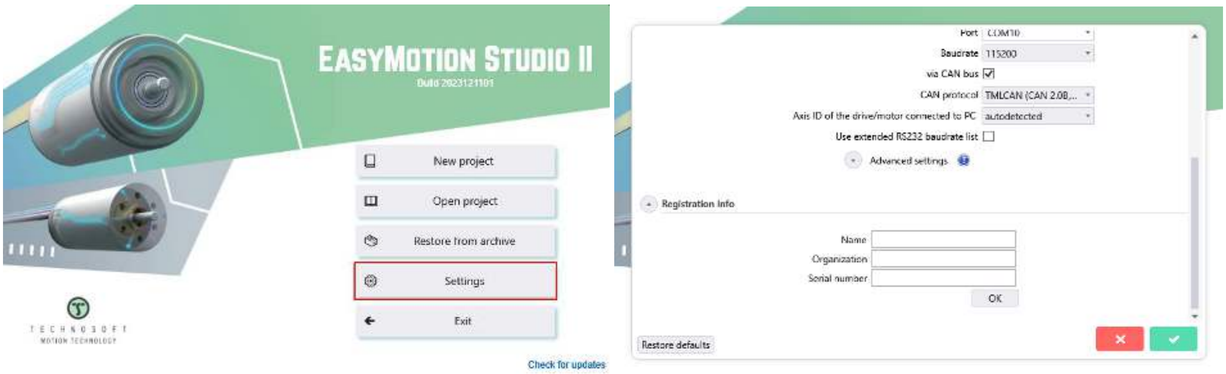
Figure 2.1 – EasyMotion Studio II – Version selection

Lite version allows performing the drive/motor commissioning and includes the needed tools (Data Logger, Scope, Control Panels etc.) to evaluate or even debug the application, if the case.

This version targets those projects where the Technosoft drives/motors are used as simple slaves in different networks (e.g. CANopen and EtherCAT).

Full version unlocks the TML program download operation, allowing the user to fully benefit of the drive built-in motion controller and implement (at the drive level) complex applications or share the intelligence between the master and the drive.

The *Full version* can be activated any time by entering the **Registration info** in the **Settings** dialogue that can be opened from the EasyMotion Studio II splash screen.



Port COM10
Baudrate 115200
via CAN bus ☑
CAN protocol TMLCAN (CAN 2.0B,...
Axis ID of the drive/motor connected to PC autodetected
Use extended RS232 baudrate list □
Name
Organization
Serial number

Figure 2.2 – Registration info in the Settings window

The **Registration info** window can also be accessed through the **Help** menu in the project window.

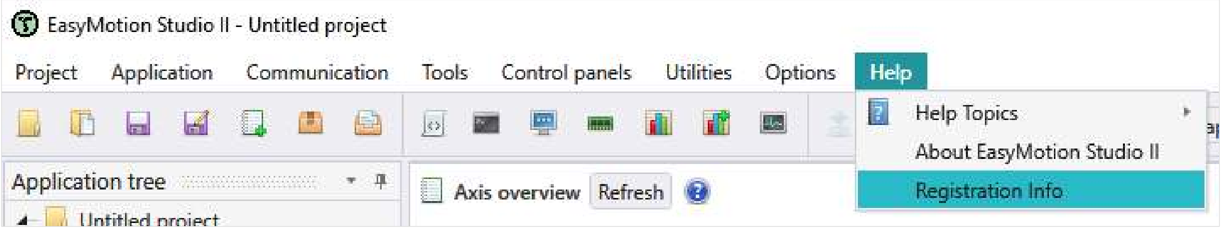


Figure 2.3 – Access the Registration info window via the Help menu

After EasyMotion Studio II is installed, there are 3 major steps in operating the Technosoft drives:

Step 1: **Hardware installation**
The hardware installation is described in details in each drive user manual.

Step 2: **Drive commissioning**
The drive commissioning will be detailed in this document.

Step 3: **Motion programming**

The programming can be done using one of the following options:

- EasyMotion Studio II using the TML;
- an EtherCAT master (depending on drive model);
- a CANopen master;
- a TML_LIB motion library for PCs (Windows or Linux)
- a TML_LIB motion library for PLCs
- a distributed control approach which combines the above options, like for example a host calling motion functions programmed on the drives in TML.

This document details the drive programming using the EasyMotion Studio II and the Technosoft Motion Language (TML).

The other programming option are detailed in the following user manuals:

- **CANopen Programming** manual - describes the CANopen drives programming;
- **CoE Programming User Manual** - describes the EtherCAT drives programming;
- **TML_LIB User manuals** - describes the project development using the PC / PLC libraries.

The above manuals can be downloaded from our website (link).

3 Create a new project

EasyMotion Studio II splash screen allows to create a new project, open a previously created one, restore a project archive or change the general settings (set the workspace location, change the software language, set the global communication parameters and insert/modify the registration info).

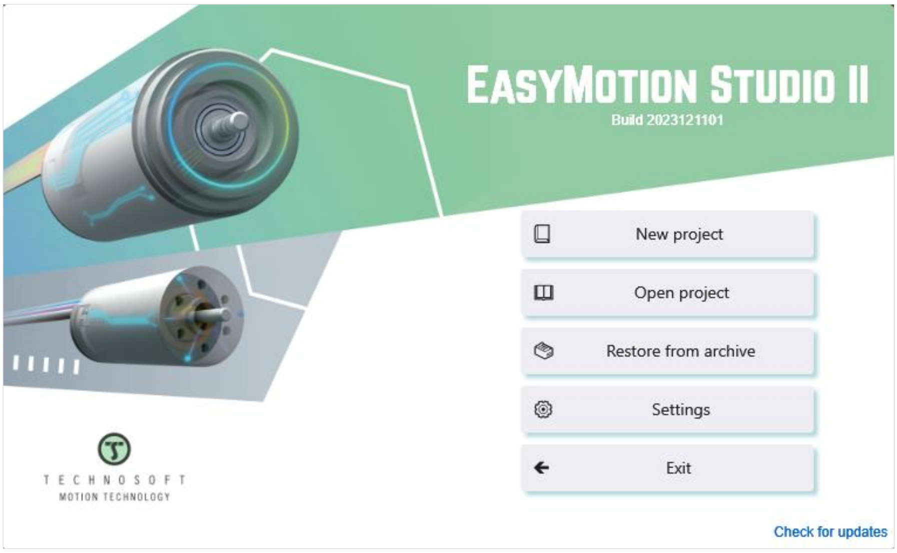
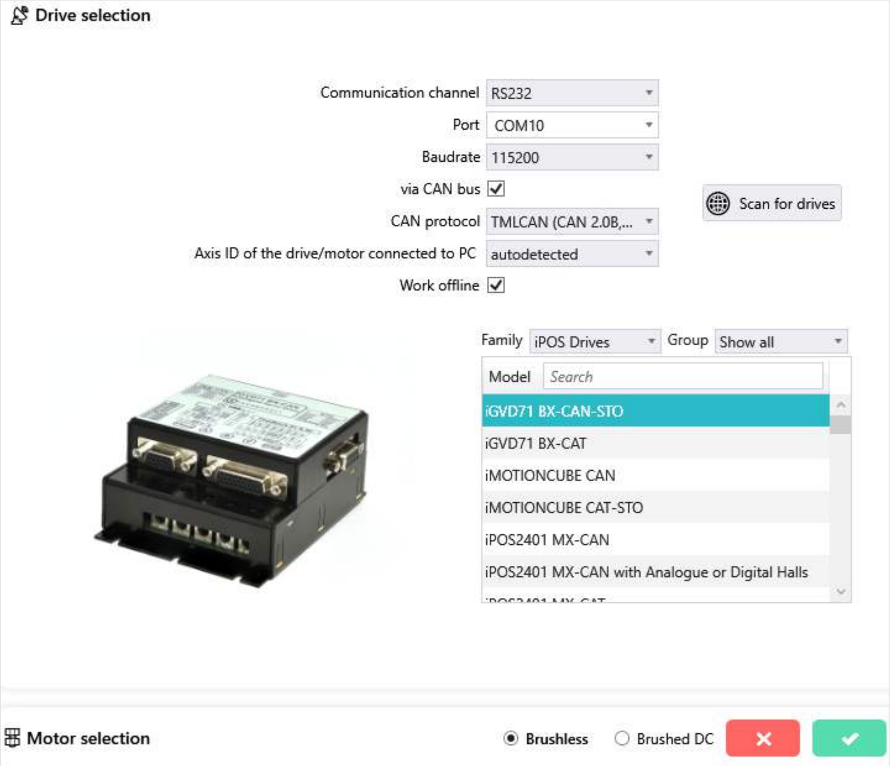


Figure 3.1 - EasyMotion Studio II splash screen

To create a new project, simply click on the **New project** button. This will open a wizard, allowing to select the drive model and the motor technology.

The drive selection can be done from a predefined list, if there's no drive connected or from a table with detected drives, that will be displayed after clicking the **Scan for drives** button.



Drive selection

Communication channel: RS232
Port: COM10
Baudrate: 115200
via CAN bus ☑
CAN protocol: TMLCAN (CAN 2.0B,...
Axis ID of the drive/motor connected to PC: autodetected
Work offline ☑

Family: iPOS Drives
Group: Show all
Model: Search

- iGVD71 BX-CAN-STO
- iGVD71 BX-CAT
- iMOTIONCUBE CAN
- iMOTIONCUBE CAT-STO
- iPOS2401 MX-CAN
- iPOS2401 MX-CAN with Analogue or Digital Halls

Motor selection: ● Brushless ○ Brushed DC

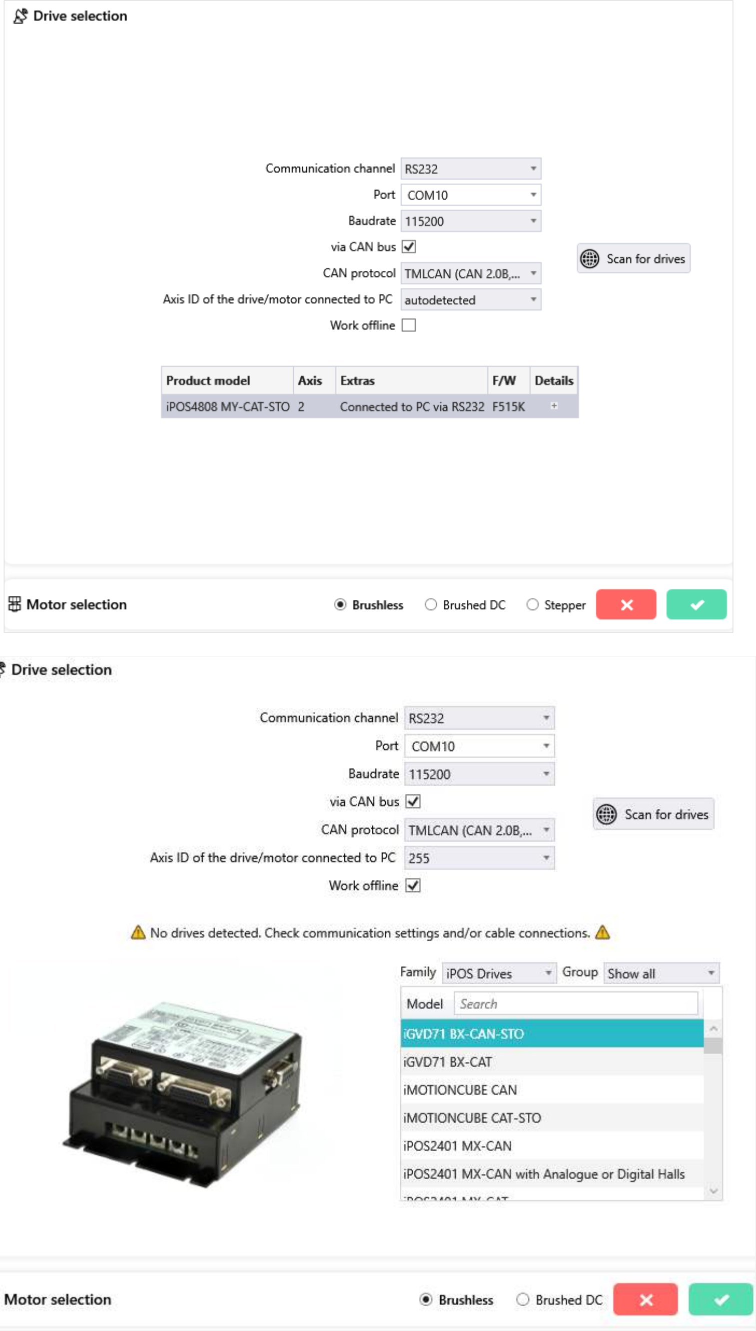
Figure 3.2 – Drive and Motor selection wizard

The **Scan for drives** option requires correct communication parameters. That's why a communication settings section is displayed in the **Drive selection** group box, allowing to set or edit the parameters.

The drives connection to the PC is possible through RS232, CAN, USB, Ethernet or Ethernet over EtherCAT (EoE), as follows:

1. Using an RS232 link, directly connected to the PC or via an USB to RS232 adapter;
2. Using the USB line;
3. Using the Technosoft Ethernet to RS232 or CAN adapter;
4. Using a CAN-bus link, directly connected to the PC through a PC-CAN interface;
5. Using the EoE protocol (available only in the EtherCAT drives);
6. Via another drive from the same CAN-bus network, which is connected directly to the PC.

When the scan for drives process ends, EasyMotion Studio II lists the detected drives or displays an error message, if no drive is found.



Drive selection

Communication channel: RS232
Port: COM10
Baudrate: 115200
via CAN bus ☑
CAN protocol: TMLCAN (CAN 2.0B,...
Axis ID of the drive/motor connected to PC: autodetected
Work offline □

Product model	Axis	Extras	F/W	Details
iPOS4808 MY-CAT-STO	2	Connected to PC via RS232	F515K	+

Motor selection ● Brushless ○ Brushed DC ○ Stepper

Drive selection

Communication channel: RS232
Port: COM10
Baudrate: 115200
via CAN bus ☑
CAN protocol: TMLCAN (CAN 2.0B,...
Axis ID of the drive/motor connected to PC: 255
Work offline ☑

⚠ No drives detected. Check communication settings and/or cable connections. ⚠

Family: iPOS Drives
Group: Show all
Model: Search

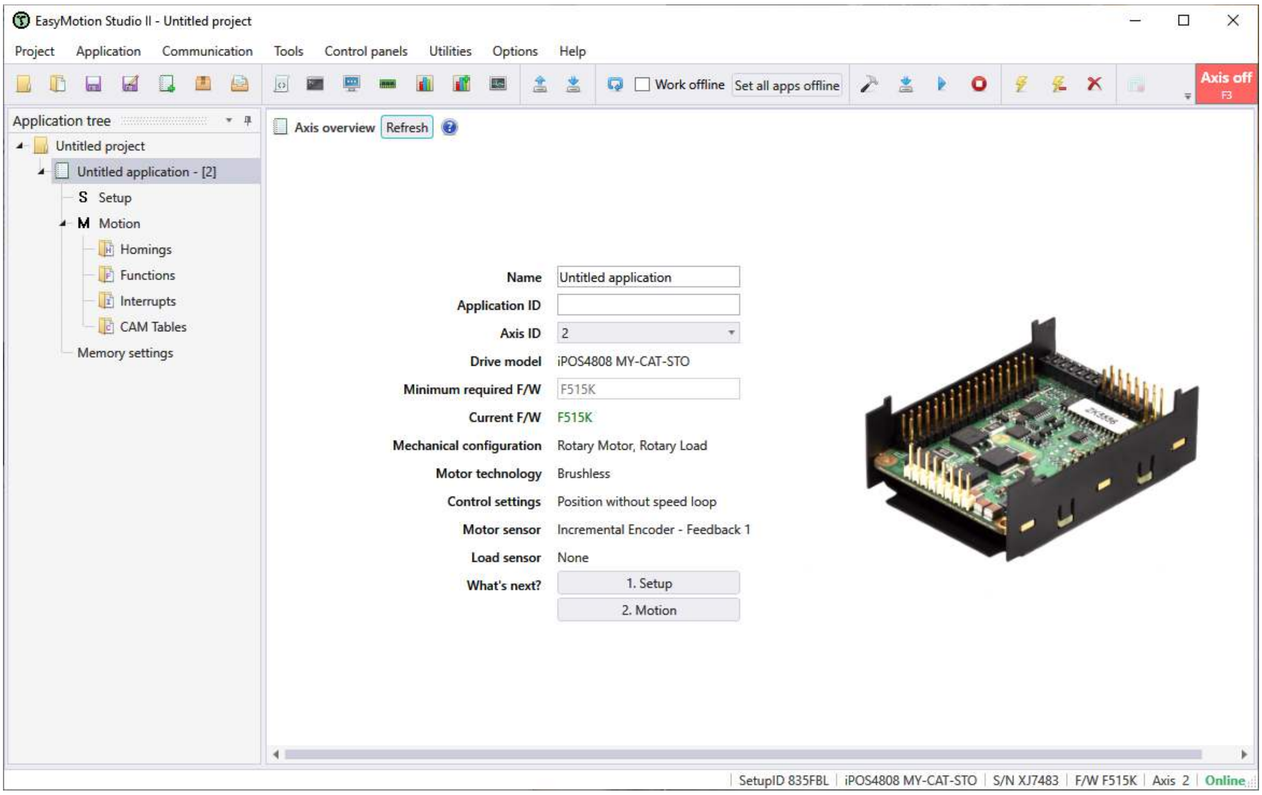
- iGVD71 BX-CAN-STO
- iGVD71 BX-CAT
- iMOTIONCUBE CAN
- iMOTIONCUBE CAT-STO
- iPOS2401 MX-CAN
- iPOS2401 MX-CAN with Analogue or Digital Halls

Motor selection ● Brushless ○ Brushed DC

Figure 3.2 – Scan for drives results

If the Scan for drives option returns an error and there’s one or more drives connected to the PC, check the cables and the power supply, make sure that the communication settings are correct and see if the operation system recognize the used adapter and has the proper drives for it.

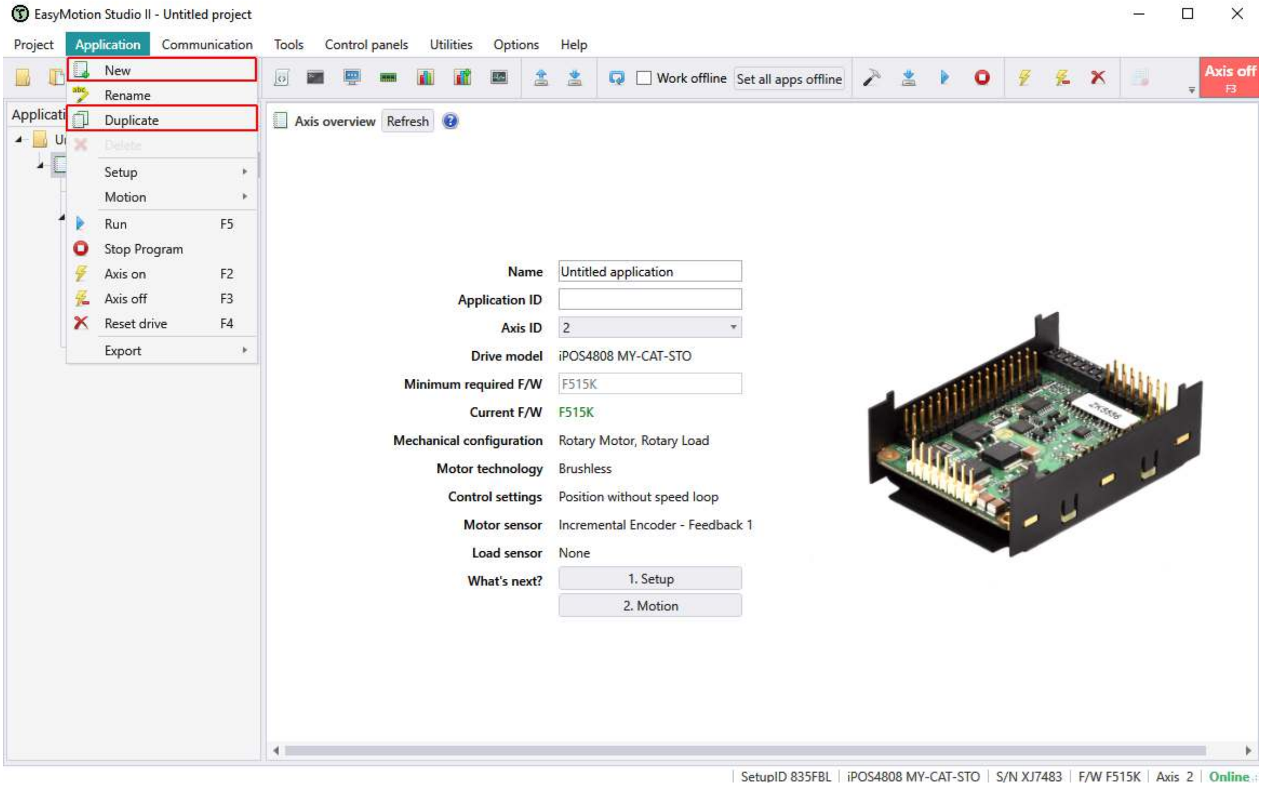
After selecting the drive and the motor technology, the new project can be started by clicking the ***green tick*** button in the **Motor selection** group box. This will end the wizard and open the project window.


EasyMotion Studio II - Untitled project
Project Application Communication Tools Control panels Utilities Options Help
Work offline
Set all apps offline
Axis off F3
Application tree
Untitled project
Untitled application - [2]
S Setup
M Motion
Homings
Functions
Interrupts
CAM Tables
Memory settings
Axis overview
Refresh
1. Setup
2. Motion
SetupID 835FBL | iPOS4808 MY-CAT-STO | S/N XJ7483 | F/W F515K | Axis 2 | Online

Name	Untitled application
Application ID	
Axis ID	2
Drive model	iPOS4808 MY-CAT-STO
Minimum required F/W	F515K
Current F/W	F515K
Mechanical configuration	Rotary Motor, Rotary Load
Motor technology	Brushless
Control settings	Position without speed loop
Motor sensor	Incremental Encoder - Feedback 1
Load sensor	None
What's next?	1. Setup 2. Motion

Figure 3.3 - Project window overview

When starting a new project, EasyMotion Studio II creates automatically a first application. Additional applications can be added later, using the **New** or **Duplicate** option in the **Application** menu.


EasyMotion Studio II - Untitled project
Project Application Communication Tools Control panels Utilities Options Help
New
Rename
Duplicate
Delete
Setup
Motion
Run F5
Stop Program
Axis on F2
Axis off F3
Reset drive F4
Export
Work offline
Set all apps offline
Axis off F3
Axis overview
Refresh
1. Setup
2. Motion
SetupID 835FBL | iPOS4808 MY-CAT-STO | S/N XJ7483 | F/W F515K | Axis 2 | Online

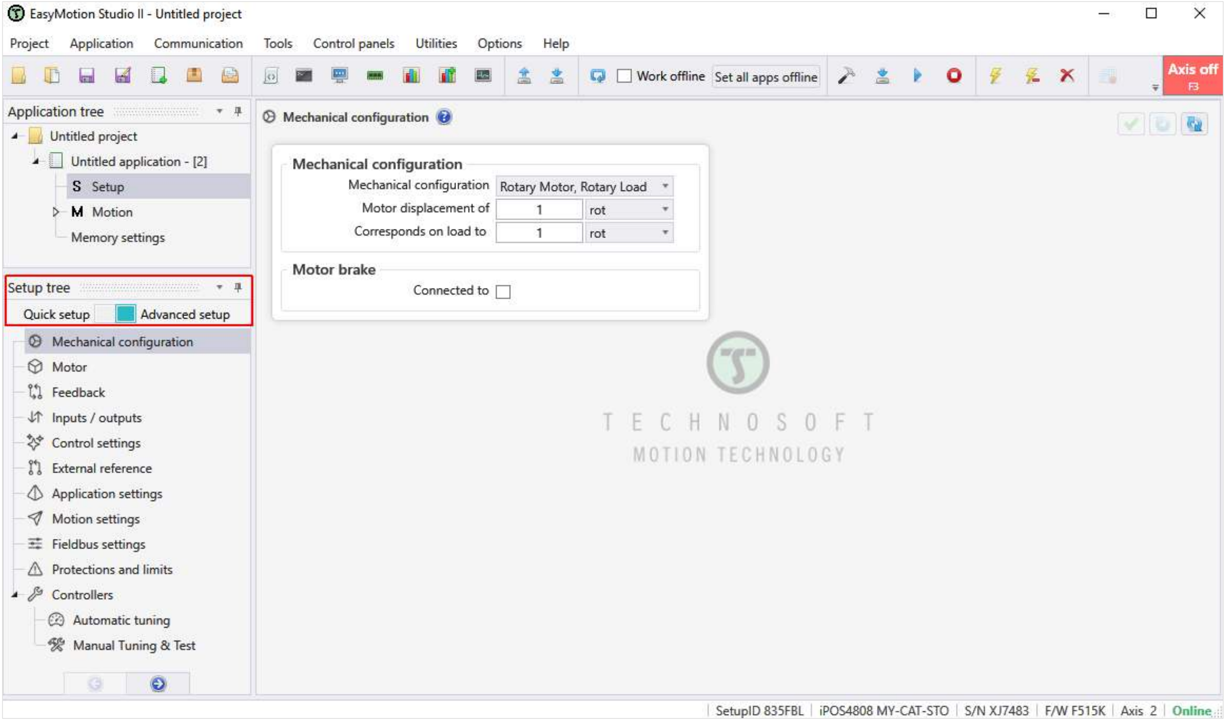
Name	Untitled application
Application ID	
Axis ID	2
Drive model	iPOS4808 MY-CAT-STO
Minimum required F/W	F515K
Current F/W	F515K
Mechanical configuration	Rotary Motor, Rotary Load
Motor technology	Brushless
Control settings	Position without speed loop
Motor sensor	Incremental Encoder - Feedback 1
Load sensor	None
What's next?	1. Setup 2. Motion

Figure 3.5 - Add a new application or duplicate the existing one

Each application includes 3 main branches: **Setup** (for drive/motor setup), **Motion** (for application development) and **Memory Settings** (for a memory usage overview and settings). The Setup and Motion branches, will be details in the next chapters.

4 Motor and drive setup

The **Setup** branch allows performing the drive commissioning using a **Quick setup** or an **Advanced setup** structure. The desired option can be selected through the radio button on top of the **Setup tree** section.



Mechanical configuration

Mechanical configuration: Rotary Motor, Rotary Load
Motor displacement of: 1 rot
Corresponds on load to: 1 rot

Motor brake

Connected to □

Figure 4.1 – Setup tree options

The **Quick setup** mode allows a fast drive commissioning, when all the connections were previously validated and the motor and feedback parameters are well known.

When a new project is started, it is recommended to use the **Advanced setup** mode, as it allows to perform different tests, to validate the motor, feedback and drive connections and functionality.

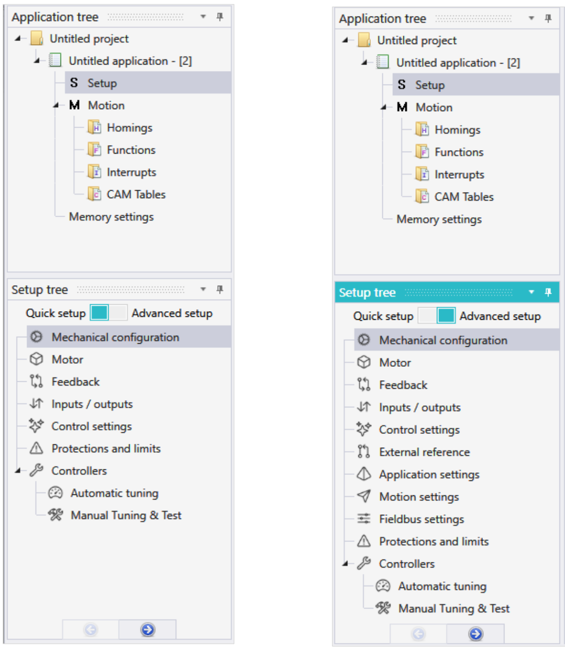


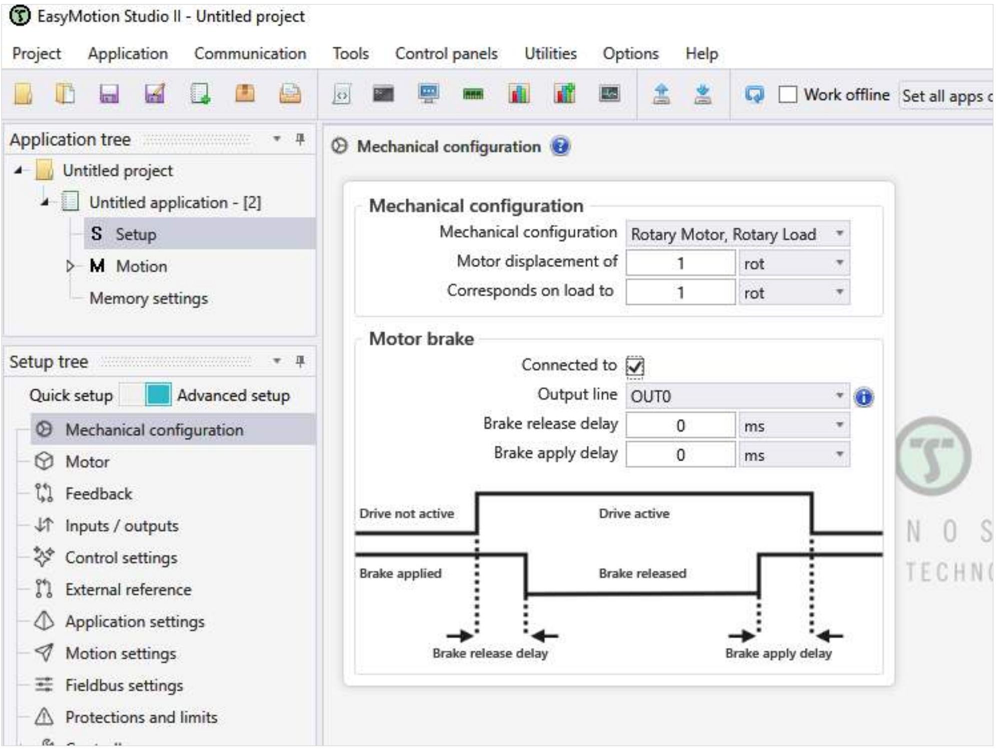
Figure 4.2 – Quick Setup vs. Advanced setup

Regardless of the selected setup structure (quick or advanced), the drive commissioning procedure assumes checking all the Setup tree branches (one by one, from top to the bottom), set the needed parameters and enable/disable the application related options.

4.1 Mechanical configuration

Available in both, *Quick setup* and *Advanced setup* modes, the **Mechanical configuration** section, allows defining the motor to load transmission type and ratio. The available options are:

- **Rotary Motor, Rotary Load** – defines those case where a rotary motor is directly connected to a rotary load or when the connection is done through a gearbox, a pulley or any other rotary to rotary transmission mechanism;
- **Rotary Motor, Linear Load** – it is used to define a linear actuator or any other system where the rotary motor movement is converted into a linear load movement, through a bet, a ball-screw, a rack and pinion system or through any other rotary to linear transmission mechanisms;
- **Linear Motor, Linear Load** – defines the cases where a voice coil or brushless linear motor is used.



Mechanical configuration

Mechanical configuration: Rotary Motor, Rotary Load

Motor displacement of: 1 rot

Corresponds on load to: 1 rot

Motor brake

Connected to: ☑

Output line: OUT0

Brake release delay: 0 ms

Brake apply delay: 0 ms

Figure 4.3 – Mechanical configuration

If the motor is equipped with an electro-mechanical brake, the drive can be configured to control it. The related settings are available in the **Motor brake** section from the **Mechanical configuration** branch, and became visible after the "**Connected to"** check box is selected.

The drive commands the brake through one of its digital outputs (usually "OUT0" or "OUT1"), specially designed for this purpose (it can output more current than a regular digital output), and sets its level automatically, function on the power stage status:

- when the drive power stage is disabled, the output is set to the inactive level, leading to the brake being applied, to prevent the load movement;
- when the drive power stage is enabled (there's power delivered to the motor), the output status is changed to the active level, allowing the brake release.

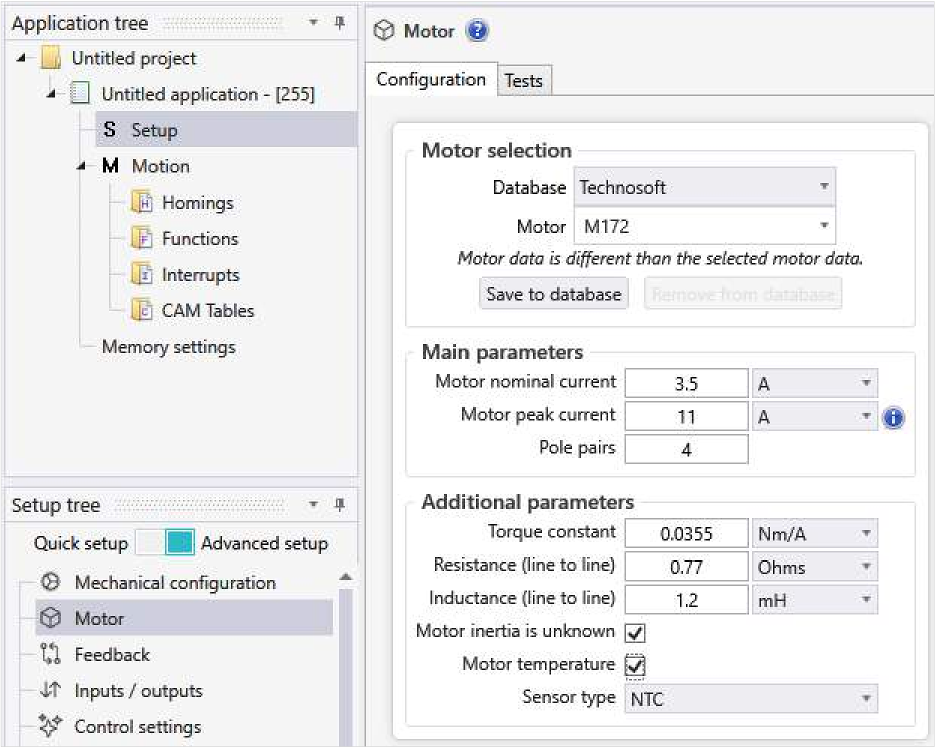
As many applications require a brake applying / releasing delay, the **Motor brake** section includes 2 fields that allows setting these parameters as follows:

- the **Brake release delay** represents the time that passes from the moment the drive power stage is enabled (the motor is energized) and the moment when the brake release command is given;
- the **Brake apply delay** represents the time that passes from the moment the brake is energised and the moment when the drive power stage is disabled (there's no power delivered to the motor).

The brake connections can often be made directly to the drive. The wiring diagram and other important details are presented in each drive user manual.

4.2 Motor setup

The motor parametrization page includes 2 tabs: **Configuration** and **Tests** (available only in the **Advanced setup** mode).



Application tree

- Untitled project
  - Untitled application - [255]
    - S Setup
    - M Motion
      - Homings
      - Functions
      - Interrupts
      - CAM Tables
    - Memory settings

Setup tree

Quick setup | Advanced setup

- Mechanical configuration
- Motor
- Feedback
- Inputs / outputs
- Control settings

Motor

Configuration | Tests

Motor selection

Database: Technosoft

Motor: M172

Motor data is different than the selected motor data.

Save to database | Remove from database

Main parameters

Motor nominal current	3.5	A
Motor peak current	11	A
Pole pairs	4	

Additional parameters

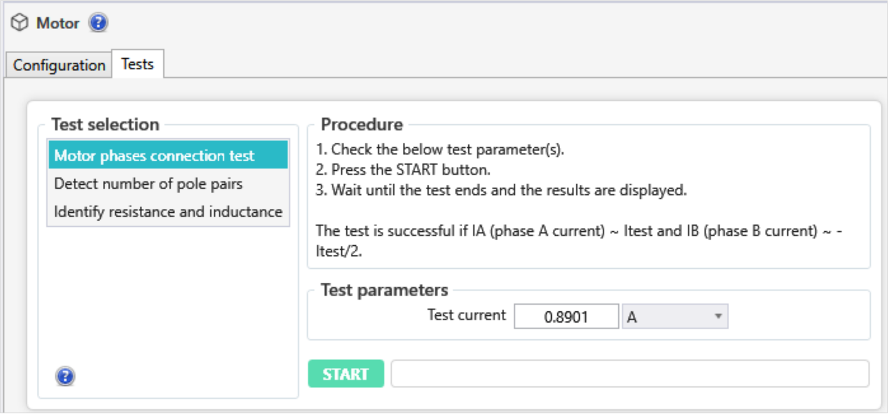
Torque constant	0.0355	Nm/A
Resistance (line to line)	0.77	Ohms
Inductance (line to line)	1.2	mH
Motor inertia is unknown	☑	
Motor temperature	☑	
Sensor type	NTC	

Figure 4.4 – Motor setup page

The **Configuration** tab allows parametrizing the motor and includes the following group boxes:

- **Motor selection** – allows to load the motor parameters from a predefined motors database or save the current parameters to the user motors database.
- **Main parameters** – allows setting the motor nominal and peak current values, and the number of motor pole pairs. All these values should be given in the motor data sheet.
- **Additional parameters** – allows setting the motor torque constant, the phase resistance, the phase inductance, the rotor inertia and the motor temperature sensor type (if present). These information should also be available in the motor data sheet. Nevertheless, they are not mandatory as there are some automated or manually tests that can be used to identify them.

The **Tests** tab allows testing the motor phases connection, detect the number of the motor pole pairs and identify the motor winding resistance and inductance.



Motor

Configuration | Tests

Test selection

- Motor phases connection test
- Detect number of pole pairs
- Identify resistance and inductance

Procedure

1. Check the below test parameter(s).
2. Press the START button.
3. Wait until the test ends and the results are displayed.

The test is successful if IA (phase A current) ~ Itest and IB (phase B current) ~ -Itest/2.

Test parameters

Test current: 0.8901 A

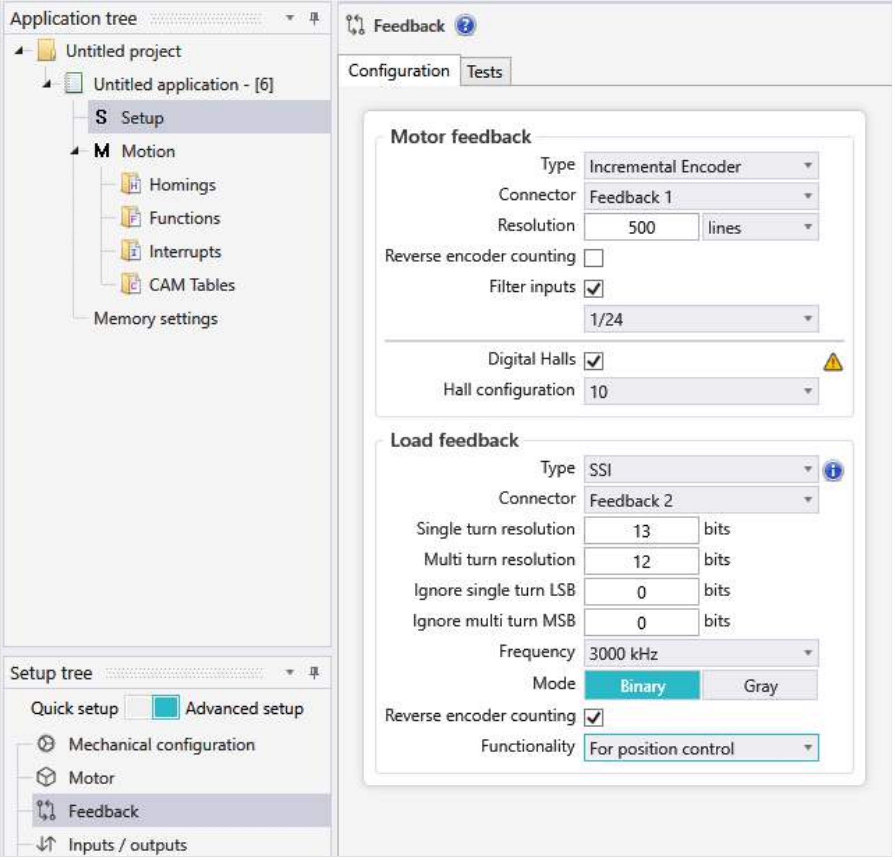
START

Figure 4.5 – Tests tab

A detailed description of the above mention tests can be found in the Help topics, that can be open from the **Help** menu in EasyMotion Studio II or by clicking the "**?**" sign in the tests dialogues.

4.3 Feedback setup

The motor and load feedback setup page can be accessed through the **Feedback** section and includes 2 tabs: **Configuration** and **Tests** (available only in the **Advanced setup** mode).



Motor feedback

Type: Incremental Encoder
Connector: Feedback 1
Resolution: 500 lines
Reverse encoder counting ☐
Filter inputs ☑ 1/24
Digital Halls ☑
Hall configuration: 10

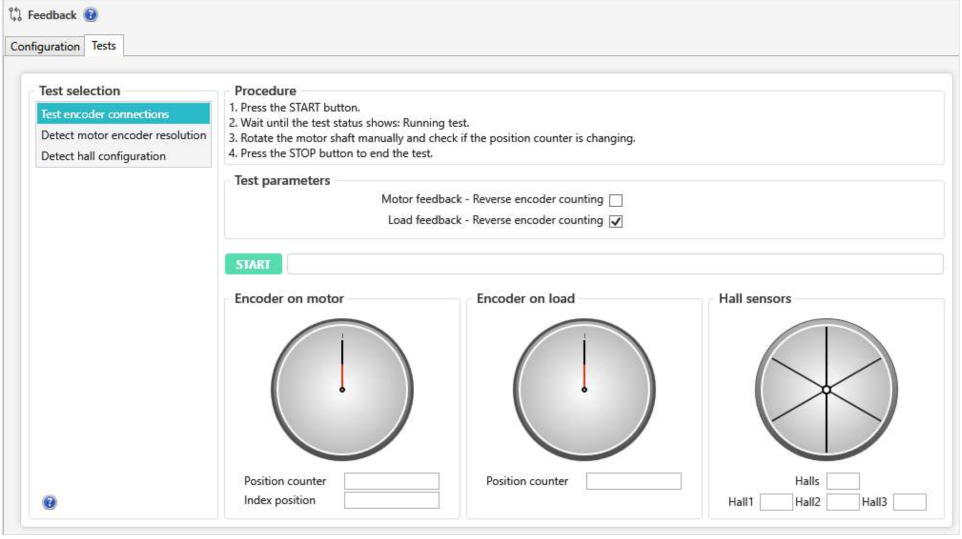
Load feedback

Type: SSI
Connector: Feedback 2
Single turn resolution: 13 bits
Multi turn resolution: 12 bits
Ignore single turn LSB: 0 bits
Ignore multi turn MSB: 0 bits
Frequency: 3000 kHz
Mode: Binary | Gray
Reverse encoder counting ☑
Functionality: For position control

Figure 4.6 – Feedback setup branch

The **Configuration** tab allows parametrizing the used feedback sensors, so its content changes function on the selected feedback type. The requested parameters should be available in the sensor data sheet.

The **Tests** tab allows testing the feedback sensors connection and detecting the resolution or other important sensor parameters.



Procedure

1. Press the START button.
2. Wait until the test status shows: Running test.
3. Rotate the motor shaft manually and check if the position counter is changing.
4. Press the STOP button to end the test.

Test parameters

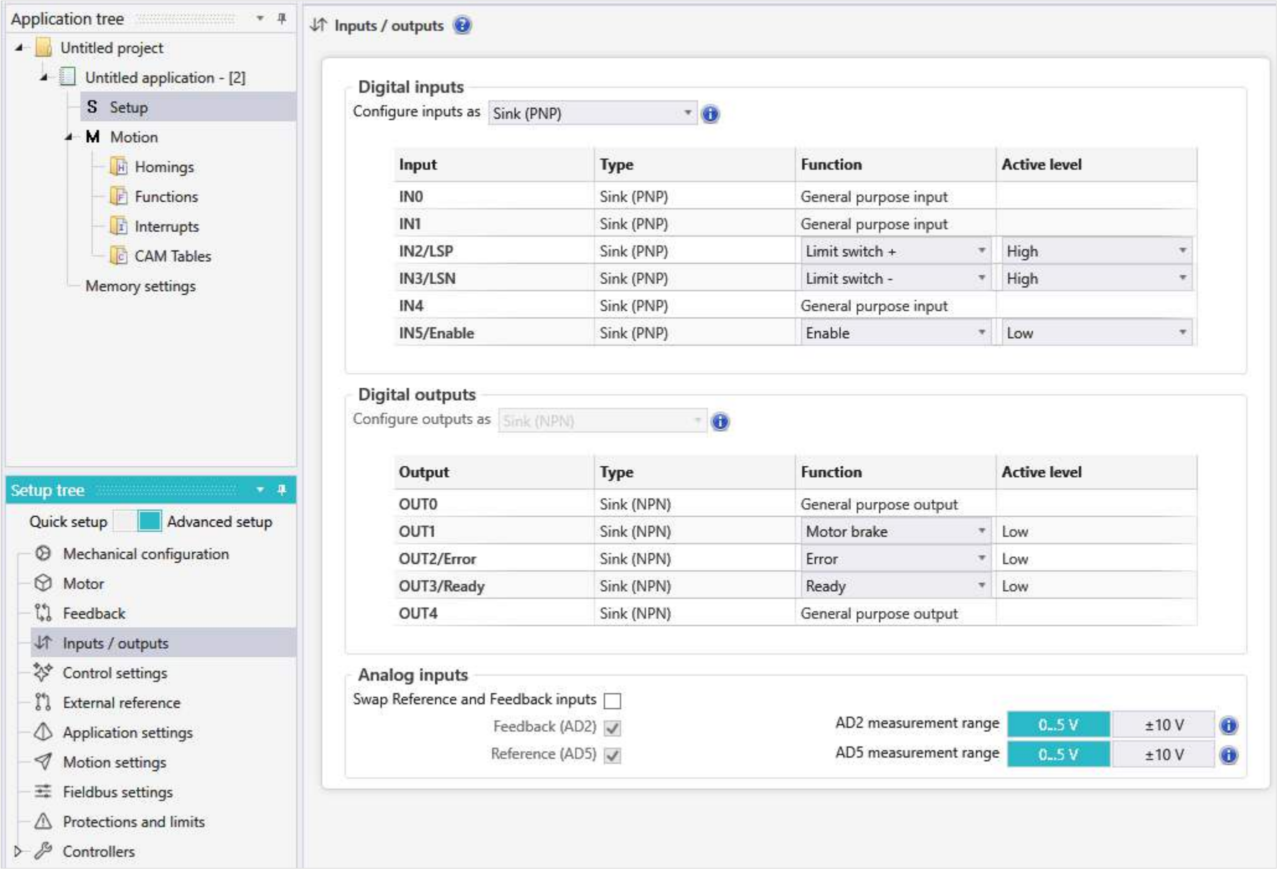
Motor feedback - Reverse encoder counting ☐
Load feedback - Reverse encoder counting ☑

Figure 4.7 – Tests tab

The parameters inside the **Configuration** tab as well as the **Tests** tab are described in details in the EasyMotion Studio II help topics, that can be open from the **Help** menu or by clicking the “**?**” sign in the **Feedback** page or in the tests dialogues.

4.4 Inputs / Outputs

The drive **Inputs** / **Outputs** parametrization (type, function and active level) can be easily done through the Inputs / outputs page.



Inputs / outputs

Digital inputs

Configure inputs as Sink (PNP)

Input	Type	Function	Active level
IN0	Sink (PNP)	General purpose input	
IN1	Sink (PNP)	General purpose input	
IN2/LSP	Sink (PNP)	Limit switch +	High
IN3/LSN	Sink (PNP)	Limit switch -	High
IN4	Sink (PNP)	General purpose input	
IN5/Enable	Sink (PNP)	Enable	Low

Digital outputs

Configure outputs as Sink (NPN)

Output	Type	Function	Active level
OUT0	Sink (NPN)	General purpose output	
OUT1	Sink (NPN)	Motor brake	Low
OUT2/Error	Sink (NPN)	Error	Low
OUT3/Ready	Sink (NPN)	Ready	Low
OUT4	Sink (NPN)	General purpose output	

Analog inputs

Swap Reference and Feedback inputs □

Feedback (AD2) ☑

Reference (AD5) ☑

AD2 measurement range 0...5 V ±10 V

AD5 measurement range 0...5 V ±10 V

Figure 4.8 – Inputs / Outputs setup page

By default all Technosoft drives and intelligent motors have general purpose inputs/outputs and some inputs/outputs with special functionality:

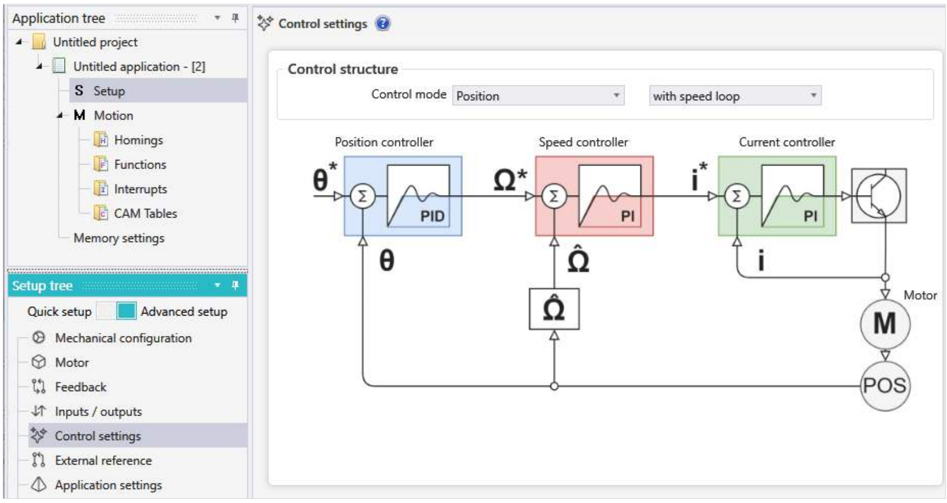
- **IN2 / LSN**
  - Type: digital input;
  - Default functionality: prevents negative load movements when active.
- **IN3 / LSP**
  - Type: digital input;
  - Default functionality: prevents positive load movements when active.
- **IN4 / ENABLE**
  - Type: digital input;
  - Default functionality: disables the drive power stage when inactive.
- **OUT2 / ERROR**
  - Type: digital output;
  - Default functionality: signals a fault on drive.
- **OUT3 / READY**
  - Type: digital output;
  - Default functionality: signals that the drive is operational.

The inputs/outputs special functionality can be disabled, by changing the related input/output **Function** to "***General purpose input/output"***.

The **Analog inputs** group box allows setting the drive to read analogue voltage signal (0...5 V or +/-10 V). This will be detailed in chapter *5.6 External reference*.

4.5 Control settings

This section allows choosing the control mode (**position**, **speed** or **torque**) and setting the control structure (positon control with or without the speed controller active),



Control settings

Control structure

Control mode: Position | with speed loop

Figure 4.9 – Control settings page

When the control mode is set to **position**, the drive will use the internal current and the position loops, allowing different position profiles (trapezoidal, S-Curve, PT, PVT, external etc.).

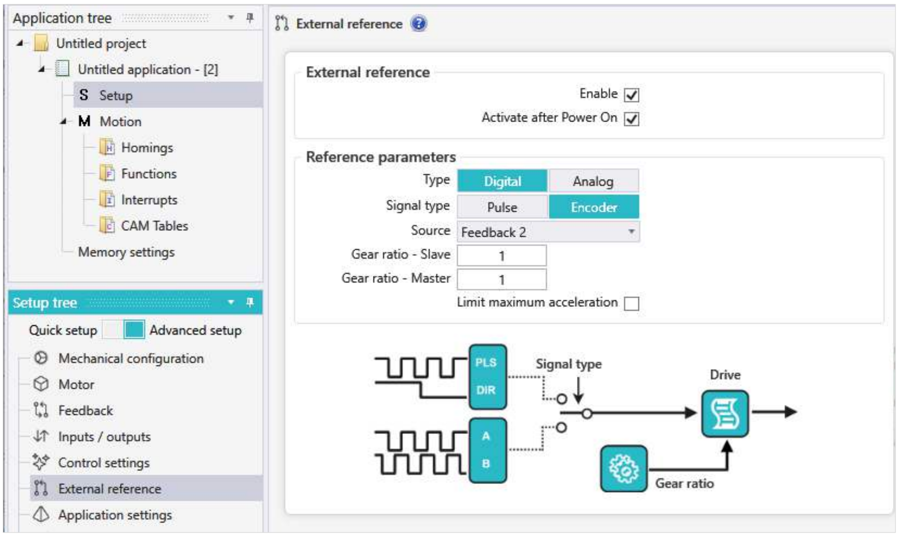
The **speed** control mode allows speed profiles and closes the current and speed loops.

Setting control mode to **torque**, will keep active only the current controller, allowing the motor torque control.

4.6 External reference

Available only in the Advanced setup mode, the **External reference** section allows setting the drive to read an external signal (analogue or digital) and convert it into a position, speed or torque reference.

When the **Digital** reference is selected, the drive can read **Pulse & Direction** signals or a quadrature **Incremental encoder** and compute the reference with a programmable gear ratio.



External reference

External reference

Enable ☑

Activate after Power On ☑

Reference parameters

Type: Digital | Analog (Digital selected)

Signal type: Pulse | Encoder (Encoder selected)

Source: Feedback 2

Gear ratio - Slave: 1

Gear ratio - Master: 1

Limit maximum acceleration □

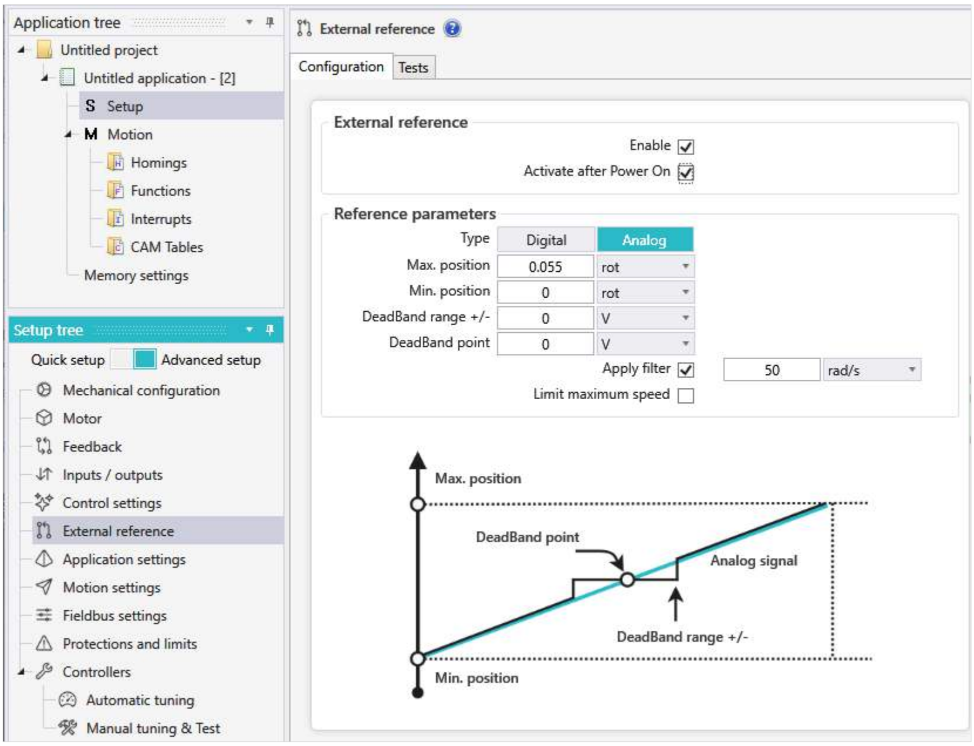
Figure 4.10 – Digital external reference settings

The **Automatically activated after Power On** option activates the external reference mode automatically, after the drive is powered ON, if the AUTORUN mode is enabled and there's no application program stored in the drive memory.

When the **Analog** reference is selected, the drive will read a 0 V .. 5 V or +/- 10 V voltage signal (check the drive user manual for details on the supported voltage range) and will convert it to a position, speed or torque reference, depending on the selected **Control mode**.

For this mode the **External reference** branch allow setting the **minimum** and the **maximum** command that corresponds to the minimum and maximum voltage reference values (there is a linear dependence between the 2 parameters).

Since the low voltage analogue signals can be easily affected by the electromagnetic noises and the analogue to digital conversion itself is subjected to some jitter, the drive has some means to compensate them: a **first order filter**, to make the reference smoother, a "**DeadBand Point**" and a "**DeadBand Range**" option, to create a voltage interval where the reference will be kept constant to prevent the motor movement / vibration when the analogue signal is stationary.



External reference

Enable ☑

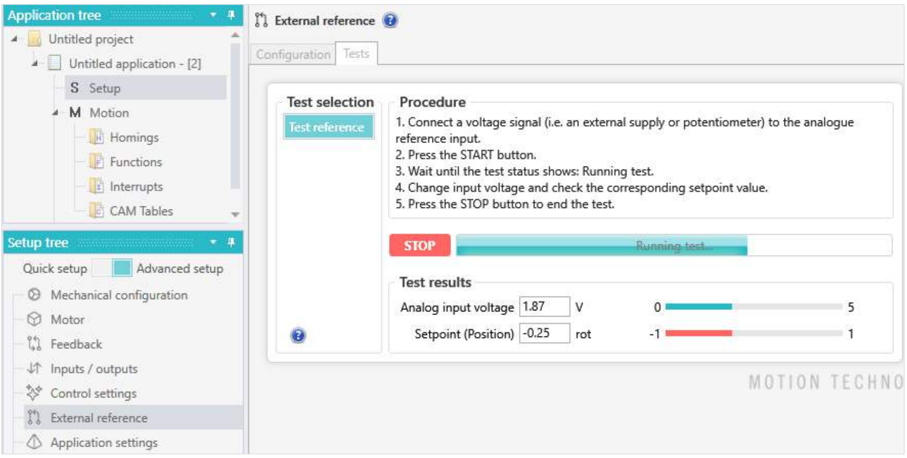
Activate after Power On ☑

Reference parameters

Type	Digital	Analog
Max. position	0.055	rot
Min. position	0	rot
DeadBand range +/-	0	V
DeadBand point	0	V
Apply filter ☑	50	rad/s
Limit maximum speed □		

Figure 4.11 – Analog external reference settings

The **Tests** tab allows checking the reference evolution function on the read analogue voltage signal.



Procedure

1. Connect a voltage signal (i.e. an external supply or potentiometer) to the analogue reference input.
2. Press the START button.
3. Wait until the test status shows: Running test.
4. Change input voltage and check the corresponding setpoint value.
5. Press the STOP button to end the test.

Test results

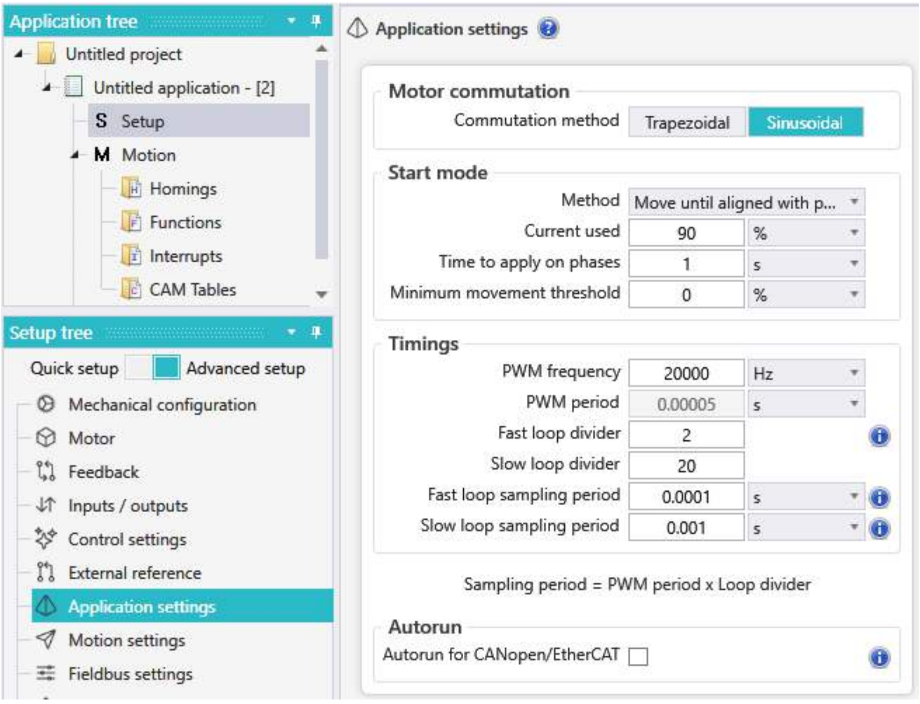
Analog input voltage	1.87	V	0	5
Setpoint (Position)	-0.25	rot	-1	1

Figure 4.12 – Analog external reference tests

During the test, the drive uses the parameters previously set in the **Reference parameters** group box from the **Configuration** tab.

4.7 Application settings

Available only in the *Advanced setup* mode, this section allows configuring the commutation method, stat mode, timings and the autorun mode for CANopen/EtherCAT.



Application tree

- Untitled project
  - Untitled application - [2]
    - S Setup
    - M Motion
      - Homings
      - Functions
      - Interrupts
      - CAM Tables

Setup tree

Quick setup ● Advanced setup

- Mechanical configuration
- Motor
- Feedback
- Inputs / outputs
- Control settings
- External reference
- Application settings
- Motion settings
- Fieldbus settings

Application settings

Motor commutation

Commutation method: Trapezoidal | [✓ Sinusoidal]

Start mode

Method	Move until aligned with p...	
Current used	90	%
Time to apply on phases	1	s
Minimum movement threshold	0	%

Timings

PWM frequency	20000	Hz
PWM period	0.00005	s
Fast loop divider	2	
Slow loop divider	20	
Fast loop sampling period	0.0001	s
Slow loop sampling period	0.001	s

Sampling period = PWM period x Loop divider

Autorun

Autorun for CANopen/EtherCAT □

Figure 4.13 – Application settings page

The **Commutation Method** section allows choosing the power stage commutation strategy:

- **Trapezoidal** – the motor is controlled as brushless DC using Hall sensors for commutation. In this mode, the motor currents are rectangular and the BEMF voltages are trapezoidal.
- **Sinusoidal** - the motor is treated like a PMSM (Permanent magnet synchronous motor) and is vector controlled using a FOC (field oriented control) algorithm. In this mode, the motor works with 3-phase sinusoidal voltages and currents.

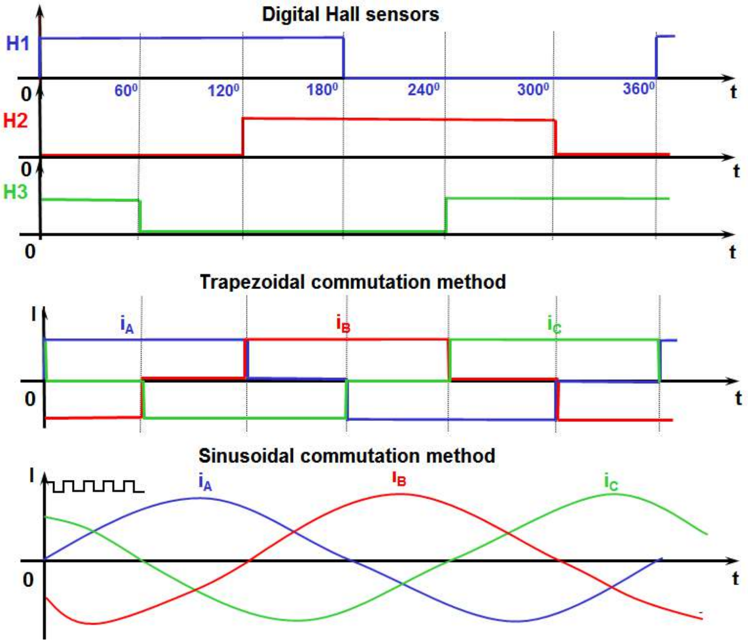


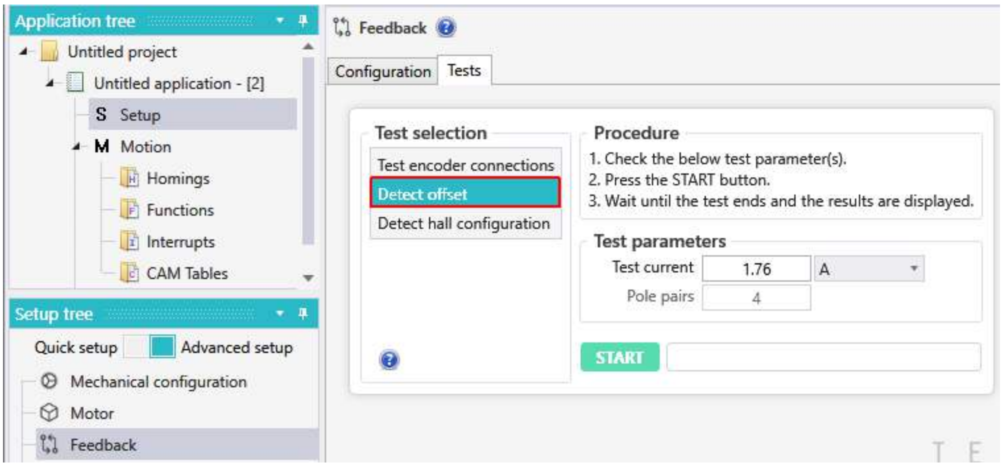
Figure 4.14 - Trapezoidal versus sinusoidal commutation method

<u>Remark</u> The sinusoidal mode offers a smoother operation, especially at low speeds, while the trapezoidal mode provides about 15% more torque to the motor in the exchange for noisier movement.

The field oriented control algorithm is possible only if the rotor position is precisely known. The drive checks it after Power On or Reset, through a start method. This method can be selected / configured though the **Start mode** group box. The following start methods are available:

- **Move till aligned with phase A**
  - the motor moves until it is aligned first with phase B, then with phase A.
  - this start method can be used for all applications with horizontal loads where there are no mechanical limits and a small motor jerk at the start is tolerated.
- **BLDC with Hall sensors** (available only if the motor is equipped with digital Hall sensors)
  - the motor starts in the trapezoidal mode and switches to the sinusoidal mode after the first Hall sensors transition is detected.
  - this start mode is recommended for applications with vertical loads, requiring a starting torque close the motor nominal value or where uncontrolled motor movement is not allowed even at start
- **PMSM with Hall sensors** (available only if the motor is equipped with digital Hall sensors)
  - very similar with the BLDC with HALL sensors except the motor starts directly in sinusoidal mode instead of trapezoidal and adjusts the rotor position after the first transition of the Hall sensors is detected.

  Motionless start (available only if there's an encoder feedback)
  - the drive determines the rotor position. The method involves only a very small movement around the initial rotor position;
  - this start mode is used when the Hall sensors are not available and the motor/load cannot move uncontrolled.
  - most complex method of all, requires tuning of the angle controller.
- **Direct using absolute position sensor** (available for motors with absolute encoders)
  - the motor absolute position is known, allows starting the motor directly;
  - an offset detection is required and can be done through the **Test** in the **Feedback** branch.



1. Check the below test parameter(s).
2. Press the START button.
3. Wait until the test ends and the results are displayed.

Test current 1.76 A

Pole pairs 4

Figure 4.15 – Absolute encoder offset detection

The **Timings** section allows changing the default PWM frequency and the control loops execution period.

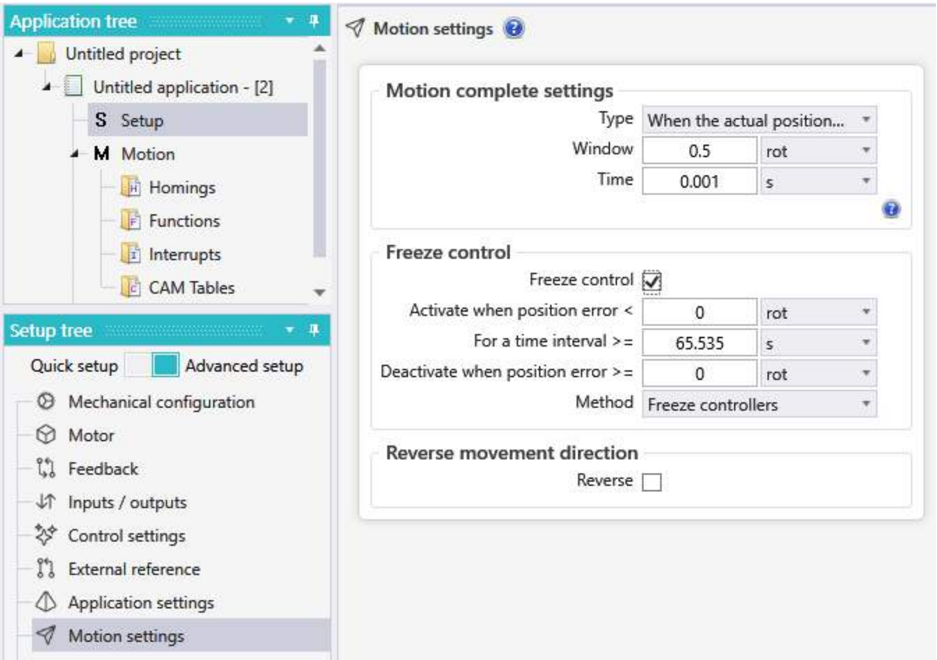
<u>*Remarks*</u>

1) *For a safe operation is recommended to keep the PWM frequency in the 20 kHz … 80 kHz interval;*
2) *The fast loop sampling period default value is 0.1 ms.*
3) *The slow loop sampling period default value is 1 ms.*

When the **Autorun for CANopen/EtherCAT** option is selected, the drive will execute the TML program at Power ON / Reset. This is valid only when a CAN drive is configured for CANopen operation or when an EtherCAT drive is used.

4.8 Motion settings

This section is present only in the *Advanced setup* mode and allows configuring the motion complete condition, activate and set the freeze control mechanism and reverse the movement direction.



Motion settings

Motion complete settings

Type: When the actual position...

Window: 0.5 rot

Time: 0.001 s

Freeze control

Freeze control ☑

Activate when position error <: 0 rot

For a time interval >=: 65.535 s

Deactivate when position error >=: 0 rot

Method: Freeze controllers

Reverse movement direction

Reverse □

Figure 4.16 – Motion setting branch

To qualify when a motion profile ends, a motion complete condition is used. The condition and the related parameters can be set through the **Motion complete settings** group box.

When the motion complete condition is set to wait for the actual positon to remain inside a settle band, for a time, the group box allows setting the settle band tolerance (position window) and the stabilize time, that should be according to the system possibilities/requirements.

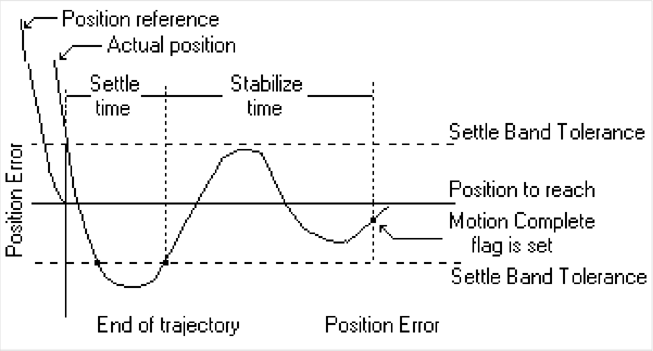


Figure 4.17 – Motion complete band details

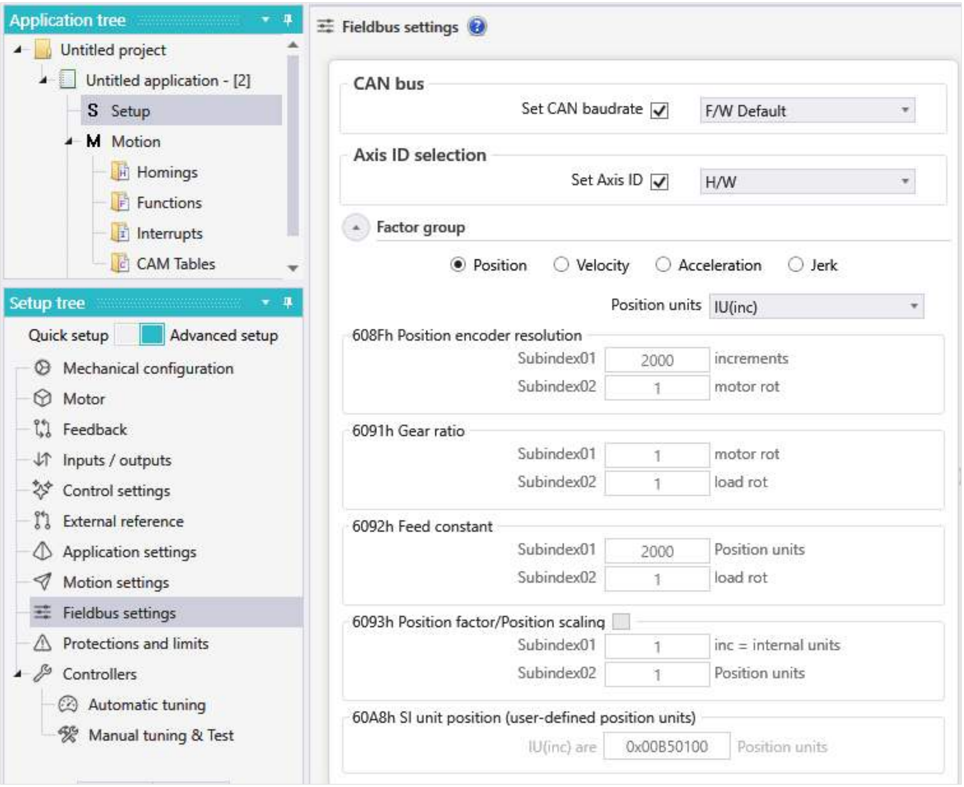
The motion complete on reference condition triggers, in position control, when the position reference arrives to the target position. In speed control, it triggers when speed reference is 0.

The **Freeze Control** group box allows setting the drive to freeze the controllers outputs, maintaining the last torque command, in an effort to avoid the motor oscillations. The algorithm requires to set a position error window and a time that are used to activate / deactivate the actual freezing.

When the motor rotation direction doesn't match the expected load movement direction, the motor connection should be modified (i.e. in case of the brushless motors, swap 2 phases). As in many cases, this operation is complicated, there's the **Reverse movement direction** option, that reverses the motor movement, without changing the motor wiring.

4.9 Fieldbus setting

Present only in the *Advanced setup* mode, this section allows setting the CAN baudrate, the drive Axis ID and the CANopen or EtherCAT factor group.



Fieldbus settings

CAN bus

Set CAN baudrate ☑ F/W Default

Axis ID selection

Set Axis ID ☑ H/W

Factor group

● Position ○ Velocity ○ Acceleration ○ Jerk

Position units IU(inc)

608Fh Position encoder resolution

Subindex01 2000 increments

Subindex02 1 motor rot

6091h Gear ratio

Subindex01 1 motor rot

Subindex02 1 load rot

6092h Feed constant

Subindex01 2000 Position units

Subindex02 1 load rot

6093h Position factor/Position scaling □

Subindex01 1 inc = internal units

Subindex02 1 Position units

60A8h SI unit position (user-defined position units)

IU(inc) are 0x00B50100 Position units

Figure 4.18 – Fieldbus settings branch

The **CAN bus** group box allow setting the CAN baudrate. The default value is **F/W default** (firmware default), which means 500 kbps. However, other speeds are also available: 125 kbps, 250 kbps, 500 kbps and 1 Mbps.

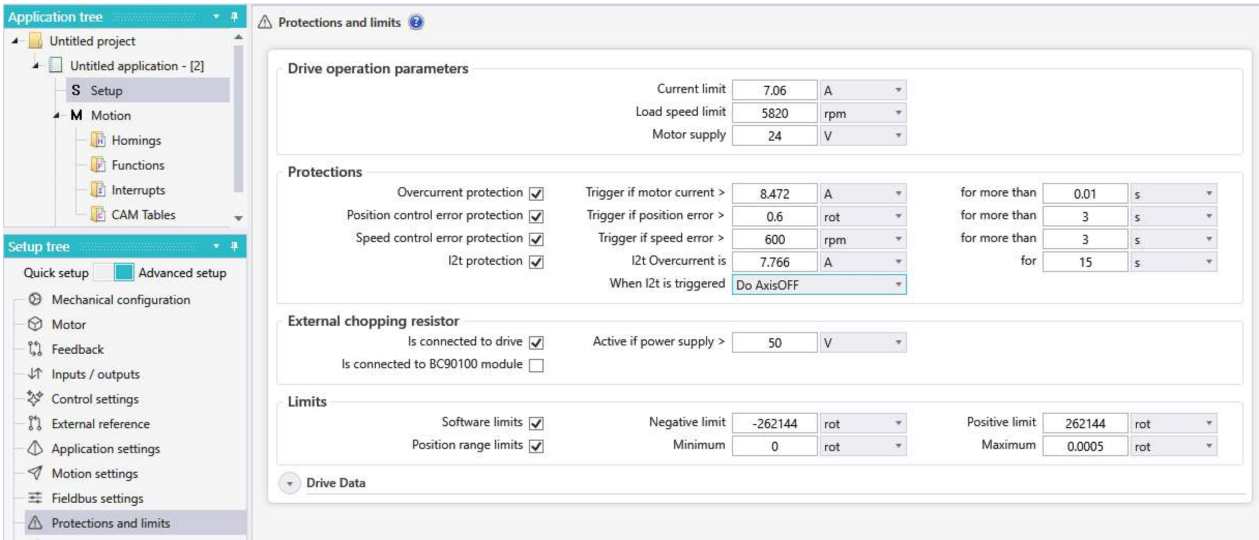
All Technosoft drives are identified with an Axis ID (a number between 1 and 127 in CANopen or between 1 and 255 in TMLCAN) that can be set by hardware (check the drive user manual) or by software, through the **Axis ID selection** option.

When multiple drives or intelligent motors should work in a CAN network, they must have the same CAN baudrate, use the same CAN protocol and have different Axis IDs.

For CANopen or EtherCAT applications, the **Factor group** section allows selecting the desired physical unit for position, speed, acceleration and jerk values. These settings are linked directly to the correspondent objects and complies with CiA 402 standard.

4.10 Protections and limits

This section allows setting the motor protections and the operational limits, preventing any system (motor, load, mechanics etc.) damage during the tests or in normal operation, when perturbations occurs.



Protections and limits

Drive operation parameters

Current limit	7.06	A
Load speed limit	5820	rpm
Motor supply	24	V

Protections

Overcurrent protection ☑	Trigger if motor current >	8.472	A	for more than	0.01	s
Position control error protection ☑	Trigger if position error >	0.6	rot	for more than	3	s
Speed control error protection ☑	Trigger if speed error >	600	rpm	for more than	3	s
I2t protection ☑	I2t Overcurrent is	7.766	A	for	15	s
	When I2t is triggered	Do AxisOFF				

External chopping resistor

Is connected to drive ☑	Active if power supply >	50	V
Is connected to BC90100 module □			

Limits

Software limits ☑	Negative limit	-262144	rot	Positive limit	262144	rot
Position range limits ☑	Minimum	0	rot	Maximum	0.0005	rot

Drive Data

Figure 4.19 – Protections and limits settings

I. Drive operation parameters

1) **Current limit** - represents the maximum current that the drive will apply to the motor. This value should be set a little bit higher than the motor nominal current, to allow dynamic movements, but smaller than the motor peak current because it may be applied to the motor for a long time.

2) **Load speed limit** - allows limiting the load speed, with the purpose of protecting the mechanical system and the operators in case of a bad command that may lead to a dangerous speed.

3) **Motor supply** – represents the voltage applied to the “Vmot” terminal.

II. Protections

1) **Over current** - it is defined through a current value and a time interval and it triggers if the motor current remains above the given current limit for the given time.

The current value should be set slightly higher than the **current limit** parameter but smaller than the motor peak current. The over current time must be set according with the motor data (smaller than the time the motor can support its peak current but long enough, to prevent false triggers).

2) **Control error** - monitors the position error or the speed error, function on used control mode, and triggers when the position / speed error remains above the position / speed error limit a time interval bigger than the position error time.

Note that if the absolute value of the position / speed error goes below the position / speed error limit, the position / speed error time counter is reset.

3) **I2t** thermal protection - prevents the motor damage due to long time operation over the nominal current. Its value is set by selecting a point on the motor I2t curve (should be available in the motor data sheet) and input the I2t over current value and the I2t time in the associated edit boxes. These points together with the nominal motor current “In” define the surface “SI2t” (see figure 5.20 - left).

When the current through the motor exceeds the nominal current (“In”), the drives starts to calculate, at each current loop (fast loop), the difference between the square of the instantaneous current and the square of the nominal current and adds them together. When the sum is equal or greater than the “SI2t” surface from the first picture, the drive I2t protection is triggered (see figure 5.20 - right).

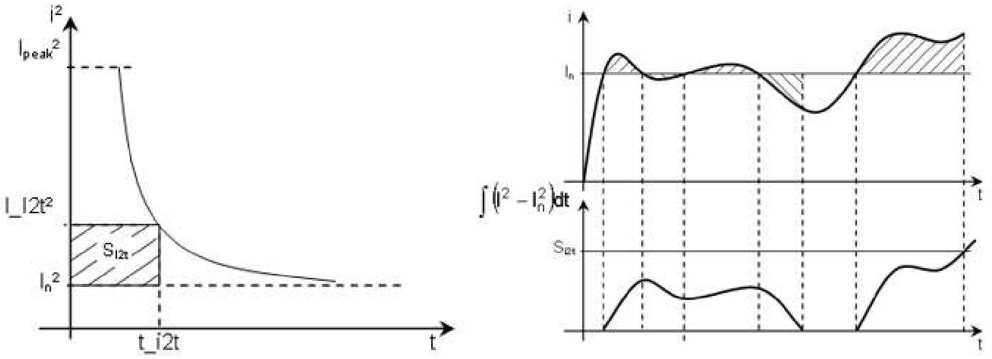


Figure 4.20 - I2t surface and triggering

Once triggered, the default behaviour of the I2t protection is to limit the motor current to 90% from the nominal current set in the Motor Setup dialogue. The current reverts back to the normal current limit after the internal protection value drops to 0. It can also be set to disable the drive power stage (“AXISOFF;”).

III. External chopping resistor

When the motor brakes or reverses, it transfers energy back to the power supply, causing the bus voltage to increase (if the supply can’t accept it). Depending on the amount of the energy and how fast it is transferred, the bus voltage can exceed the overvoltage protection and it can even damage the drive or the power supply. To prevent this unwanted situation, the over-voltage can be limited with a chopping resistor (see the drive user manual for the chopping resistor selection and connection).

Typically the activation level for the braking circuit must be set slightly above the drive supply voltage value and below its over-voltage protection value.

IV. Limits

1) **Software limit switches** - when the a limit is reached then the quick stop profile is automatically executed to stop the motor movement. No movements will be allowed in the respective direction.

2) **Position range limits** allows to:
- keep the position indicator between 2 limits (default operation mode)
- get to the target position using only negative or only positive movements
- get to the target using the shortest path

The position range limits behaviour can be set through the **bits 6** and **7** in the **POSOPTCODE** register:

Register name	Bit 7	Bit 6	Description
POSOPTCODE	0	0	Normal positioning (default option): if the position indicator reaches or exceeds one of the position range limits, it wrap automatically to the other end of the range.
	0	1	Positioning only in negative direction: if the target position is higher than the actual position, the axis moves over the min position limit to the target position.
	1	0	Positioning only in positive direction: if the target position is lower than the actual position, the axis moves over the max position limit to the target position.
	1	1	Positioning with the shortest way to the target position. NOTE: If the difference between actual value and target position in a 360° system is 180°, the axis moves in positive direction.

The figure below shows few movement examples (function on the settings of the bits 6 and 7 in the POSOPTCODE register), when the minimum position range limit is set to 0° and the maximum position range limit is set to 360°.

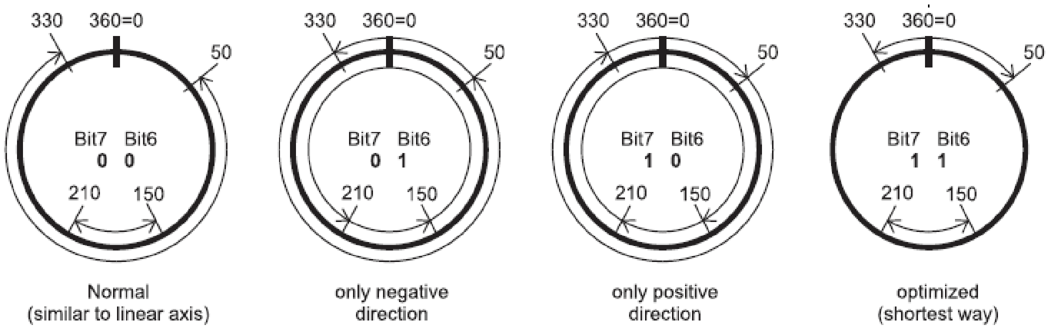
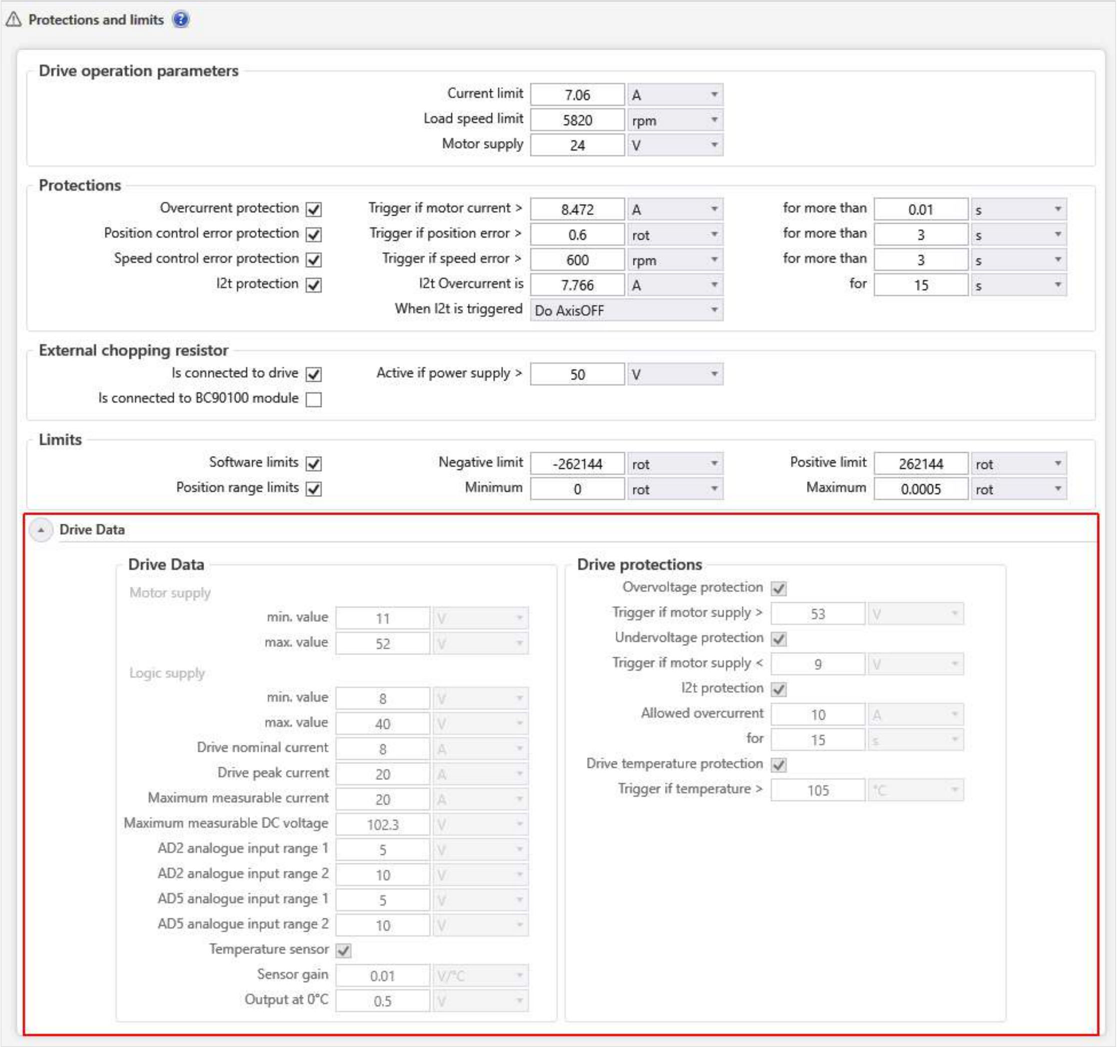


Figure 4.21 - Rotary axis position range limits examples

The **Limits** group box is only allowing to set the position range limits. Their behaviour (the **POSOPTCODE** register value) will be set in the TML program or, if there's a master, it can set the **POSOPTCODE** register accordingly.

V. Drive Data

The **Drive Data** page includes the drive nominal parameters and it protections. These are important as the motor protections should be correlated with the drive limits



Protections and limits

Drive operation parameters

Current limit	7.06	A
Load speed limit	5820	rpm
Motor supply	24	V

Protections

Overcurrent protection ☑	Trigger if motor current >	8.472	A	for more than	0.01	s
Position control error protection ☑	Trigger if position error >	0.6	rot	for more than	3	s
Speed control error protection ☑	Trigger if speed error >	600	rpm	for more than	3	s
I2t protection ☑	I2t Overcurrent is	7.766	A	for	15	s
	When I2t is triggered	Do AxisOFF				

External chopping resistor

Is connected to drive ☑	Active if power supply >	50	V
Is connected to BC90100 module □			

Limits

Software limits ☑	Negative limit	-262144	rot	Positive limit	262144	rot
Position range limits ☑	Minimum	0	rot	Maximum	0.0005	rot

Drive Data

Drive Data

Motor supply		
min. value	11	V
max. value	52	V
Logic supply		
min. value	8	V
max. value	40	V
Drive nominal current	8	A
Drive peak current	20	A
Maximum measurable current	20	A
Maximum measurable DC voltage	102.3	V
AD2 analogue input range 1	5	V
AD2 analogue input range 2	10	V
AD5 analogue input range 1	5	V
AD5 analogue input range 2	10	V
Temperature sensor ☑		
Sensor gain	0.01	V/°C
Output at 0°C	0.5	V

Drive protections

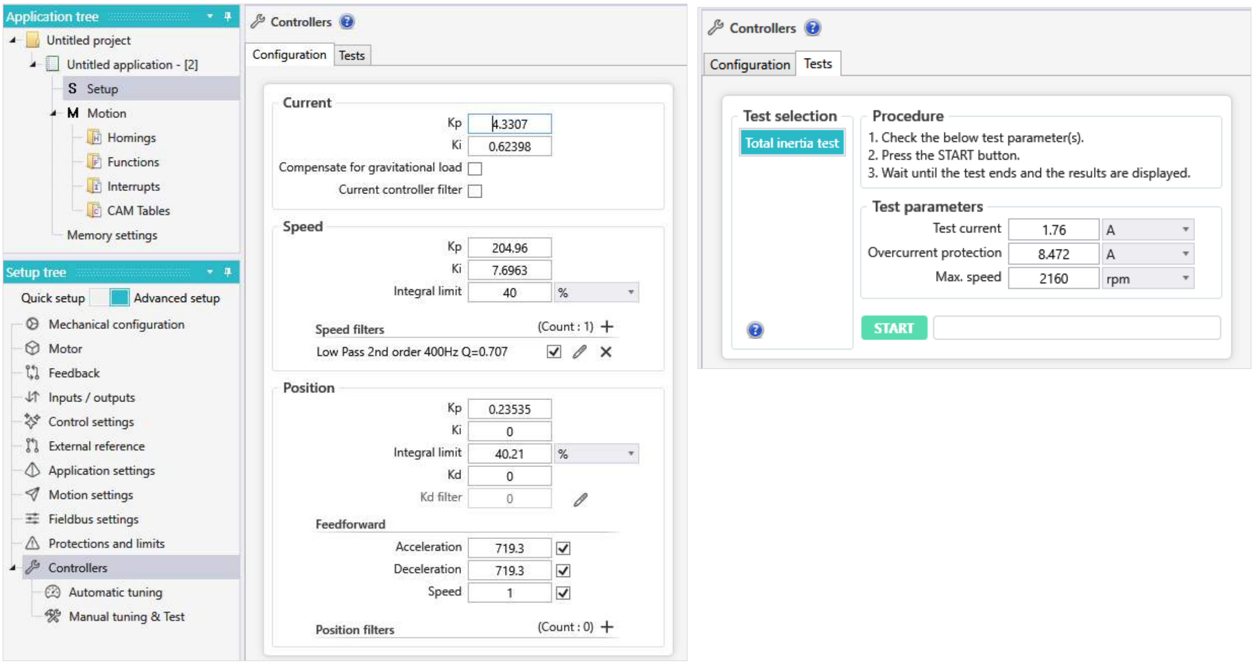
Overvoltage protection ☑		
Trigger if motor supply >	53	V
Undervoltage protection ☑		
Trigger if motor supply <	9	V
I2t protection ☑		
Allowed overcurrent	10	A
for	15	s
Drive temperature protection ☑		
Trigger if temperature >	105	°C

Figure 4.22 – Drive data page

*Remark By default, the **Drive Data** page starts minimized as it is read only and has only information purpose.*

4.11 Controllers

This section allows editing the controllers' parameters, detect the total system inertia (available only in *Advanced setup* mode) and perform the controller tuning (automatic or manually).



Current
Kp 4.3307
Ki 0.62398
Compensate for gravitational load □
Current controller filter □

Speed
Kp 204.96
Ki 7.6963
Integral limit 40 %
Speed filters (Count : 1) +
Low Pass 2nd order 400Hz Q=0.707 ☑

Position
Kp 0.23535
Ki 0
Integral limit 40.21 %
Kd 0
Kd filter 0
Feedforward
Acceleration 719.3 ☑
Deceleration 719.3 ☑
Speed 1 ☑
Position filters (Count : 0) +

Test selection
Total inertia test

Procedure
1. Check the below test parameter(s).
2. Press the START button.
3. Wait until the test ends and the results are displayed.

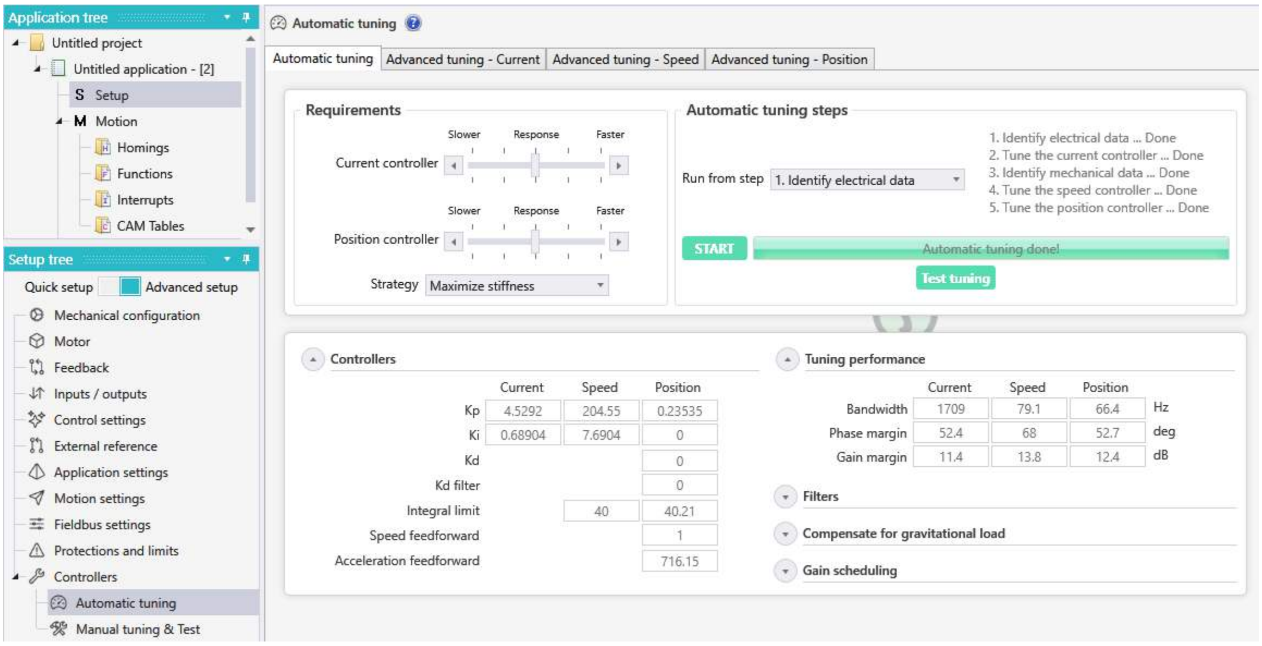
Test parameters
Test current 1.76 A
Overcurrent protection 8.472 A
Max. speed 2160 rpm

Figure 4.23 – Controllers configuration tab

The estimated inertia is used to compute the position / speed controllers parameters, when the selected tuning method is ”Set bandwidth” (see the Tune section in the Advanced tuning tabs for position and speed).

4.11.1 Automatic tuning

This option allows to easily find the controllers parameters, based on the required system response. The automatic tuning procedure is launched when the **START** button is pressed.



Requirements
Current controller
Position controller
Strategy Maximize stiffness

Automatic tuning steps
Run from step 1. Identify electrical data
1. Identify electrical data ... Done
2. Tune the current controller ... Done
3. Identify mechanical data ... Done
4. Tune the speed controller ... Done
5. Tune the position controller ... Done
Automatic tuning done!

Controllers

	Current	Speed	Position
Kp	4.5292	204.55	0.23535
Ki	0.68904	7.6904	0
Kd			0
Kd filter			0
Integral limit		40	40.21
Speed feedforward			1
Acceleration feedforward			716.15

Tuning performance

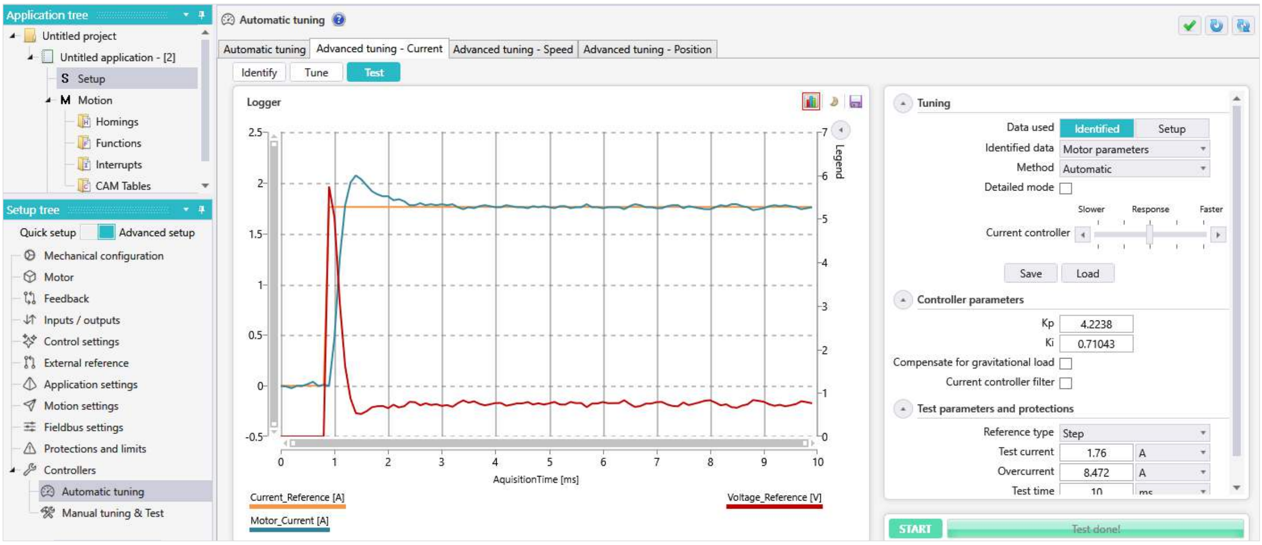
	Current	Speed	Position	
Bandwidth	1709	79.1	66.4	Hz
Phase margin	52.4	68	52.7	deg
Gain margin	11.4	13.8	12.4	dB

Figure 4.24 – Automatic tuning page

When the tuning procedure ends successfully, the “**Automatic tuning done!”** message will be displayed in the progress bar. The tuning parameters and the other detected or computed parameters will be displayed in the **Controllers** and **Tuning** performances group box.

If, for any reasons, the automatically tuning procedure fails, the "**Automatic tuning failed**" message will be displayed in the progress bar. Placing the cursor over the progress bar, will open a small information message, with more details on why the automatic tuning procedure failed.

The tuning parameters can be tested and adjusted through the **Advanced Tuning** tabs.



Data used: Identified
Identified data: Motor parameters
Method: Automatic
Detailed mode: □
Current controller
Kp: 4.2238
Ki: 0.71043
Compensate for gravitational load: □
Current controller filter: □
Reference type: Step
Test current: 1.76 A
Overcurrent: 8.472 A
Test time: 10 ms

Figure 4.25 – Advanced tuning tabs

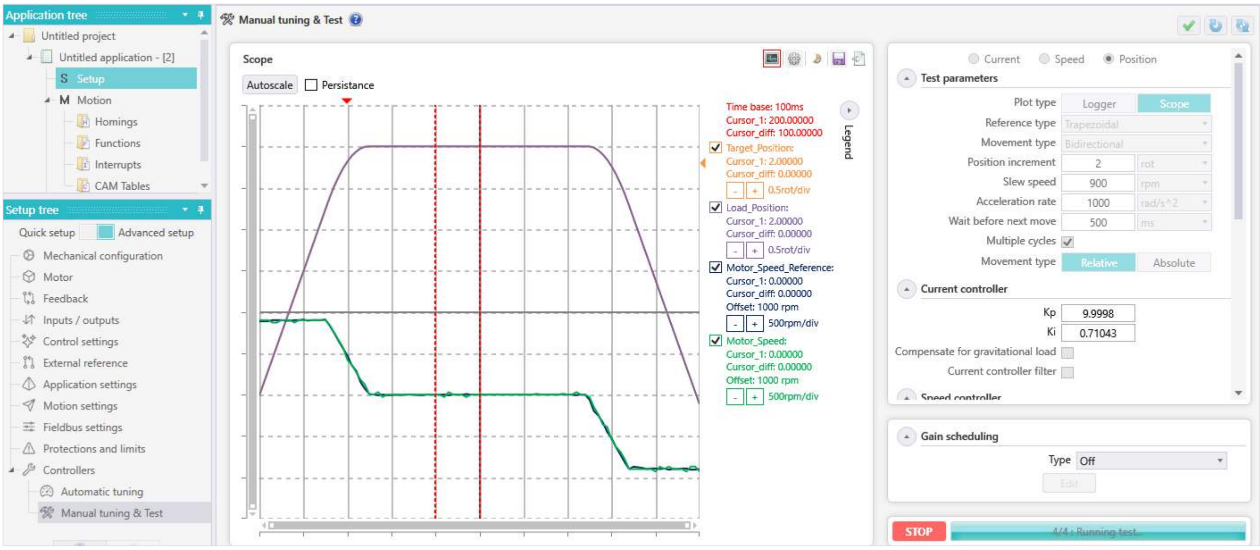
The **Advanced tuning** tabs includes 3 pages:

- **Identify** – displays the identified data, used to tune the respective controller and allows changing the identification method with the purpose of repeating the identification procedure.
- **Tune** – displays the computed controller parameters and the estimated controller answer. It is also used to start the respective controller tuning procedure again.
- **Test** – allows testing the respective controller response and change the tuning parameters.

For the best system performances, the tuning must be performed with the load connected to the motor.

4.11.2 Manual tuning & test

This option allows running a current, speed or positon profile and adjust the active controllers parameters, while checking the system behaviour, through the logger or scope tools.



Time base: 100ms
Cursor_1: 200.00000
Cursor_diff: 100.00000
☑ Target_Position:
Cursor_1: 2.00000
Cursor_diff: 0.00000
0.5rot/div
☑ Load_Position:
Cursor_1: 2.00000
Cursor_diff: 0.00000
0.5rot/div
☑ Motor_Speed_Reference:
Cursor_1: 0.00000
Cursor_diff: 0.00000
Offset: 1000 rpm
500rpm/div
☑ Motor_Speed:
Cursor_1: 0.00000
Cursor_diff: 0.00000
Offset: 1000 rpm
500rpm/div

○ Current ○ Speed ● Position
Plot type: Logger / Scope
Reference type: Trapezoidal
Movement type: Bidirectional
Position increment: 2 rot
Slew speed: 900 rpm
Acceleration rate: 1000 rad/s^2
Wait before next move: 500 ms
Multiple cycles: ☑
Movement type: Relative / Absolute
Kp: 9.9998
Ki: 0.71043
Compensate for gravitational load: □
Current controller filter: □
Type: Off

Figure 4.26 – Manual tuning page

The current and speed controllers are PI type ($u(t) = kp * \varepsilon(t) + ki * \int_0^t \varepsilon(t)dt$), while the position controller is PID type ($u(t) = kp * \varepsilon(t) + ki * \int_0^t \varepsilon(t)dt + kd * \frac{d\varepsilon(t)}{dt}$).

The controllers response characteristic (overshoot, rise time and steady-state error) are related to the Kp, Ki and Kd constants as follows:

- increasing **Kp** will result in increased overshoot but with reduced rise time and steady-state error;
- increasing **Ki** will result in increased overshoot and slightly decreased rise time but will drastically reduce the steady-state error while increasing the chances for oscillations (in an idealistic system where the output voltage will never saturate, the Ki can eliminate the position error);
- increasing **Kd** will result in increased rise time but with reduced overshoot (the steady-state error is not affected by Kd).

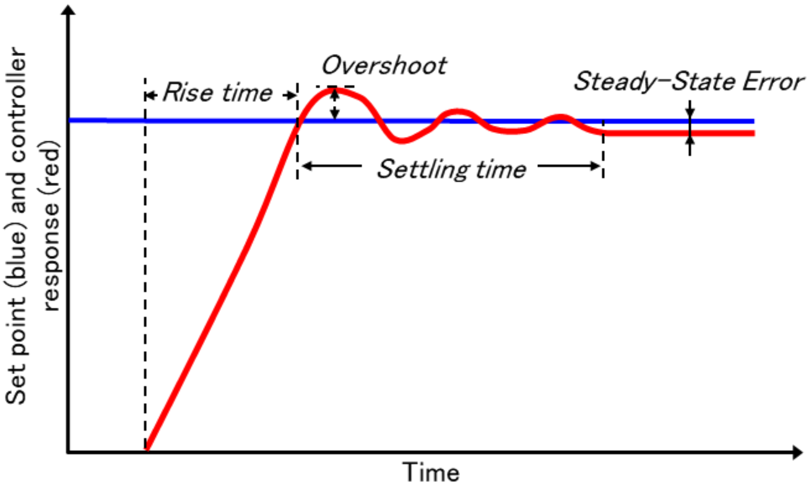


Figure 4.27 - Controllers response characteristics

The controllers tuning represents the last step in the setup process. Once it is ready, the setup can be downloaded to the drive, preparing it for the next step (TML application programming or master operation).

4.12 Download the Setup to the drive

The setup can be downloaded to the drive, using the **Write setup to drive** option in the **Application** menu or using the ribbon button, marked with a red square in the below image.

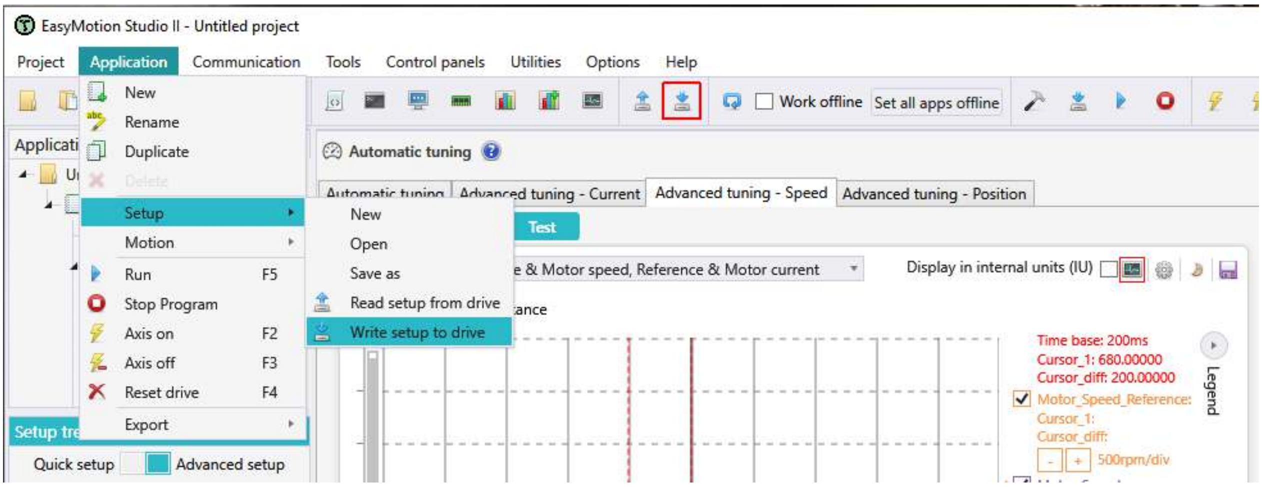


Figure 4.2 - Write the setup parameters to the drive memory

After the setup is written to the drive non-volatile memory (EEPROM), a reset is required to activate it.

5 Drive programming

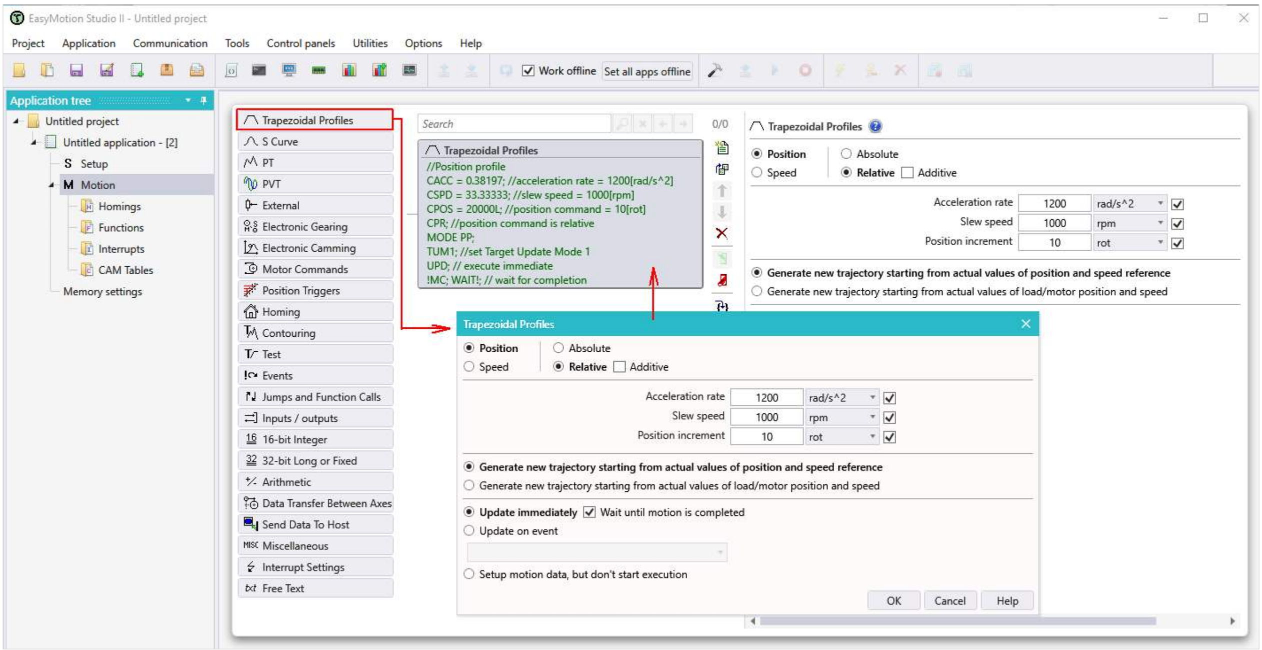
The Technosoft drives and intelligent motors can work as **standalone** devices or as **slaves** in a **CAN** (TMLCAN or CANopen) or **EtherCAT** network.

When the drive / intelligent motor is used as a standalone device or when the application functionality is **distributed** between the master and the Technosoft drives / intelligent motors, a **TML program** is required.

The TML program can be developed using the **modules** in the **Motion** branch and stored to the drive as a TML program or functions. Some predefined homing methods and interrupts are also available.

The modules in the Motion branch are described in the **Help** topics***:*** **Help | Help Topics | Application Programming | Motion Programming with built-in Motion Controller**.

As the drives / intelligent motors programming with EasyMotion Studio is graphical, the TML code is generated automaticity by the **wizard windows** behind the modules in the **Motion** branch.



Trapezoidal Profiles

```
//Position profile
CACC = 0.38197; //acceleration rate = 1200[rad/s^2]
CSPD = 33.33333; //slew speed = 1000[rpm]
CPOS = 20000L; //position command = 10[rot]
CPR; //position command is relative
MODE PP;
TUM1; //set Target Update Mode 1
UPD; // execute immediate
!MC; WAIT!; // wait for completion
```

Trapezoidal Profiles

● Position ○ Absolute
○ Speed ● Relative □ Additive

Acceleration rate 1200 rad/s^2 ☑
Slew speed 1000 rpm ☑
Position increment 10 rot ☑

● Generate new trajectory starting from actual values of position and speed reference
○ Generate new trajectory starting from actual values of load/motor position and speed

● Update immediately ☑ Wait until motion is completed
○ Update on event

○ Setup motion data, but don't start execution

Figure 5.1 – TML code example generated by the Trapezoidal profile wizard

More details on the TML instructions can be found in the **Help** topics***:*** **Help | Help Topics | Application Programming |Technosoft Motion Language.**

<u>TML program example</u>

Move the load relatively, for 1 rotation, using a trapezoidal position profile, wait 0.5 seconds and return, using an S-Curve profile .

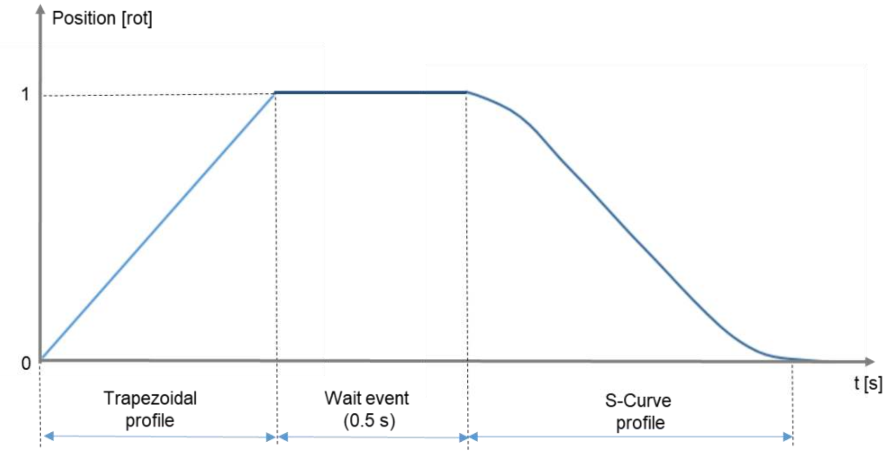
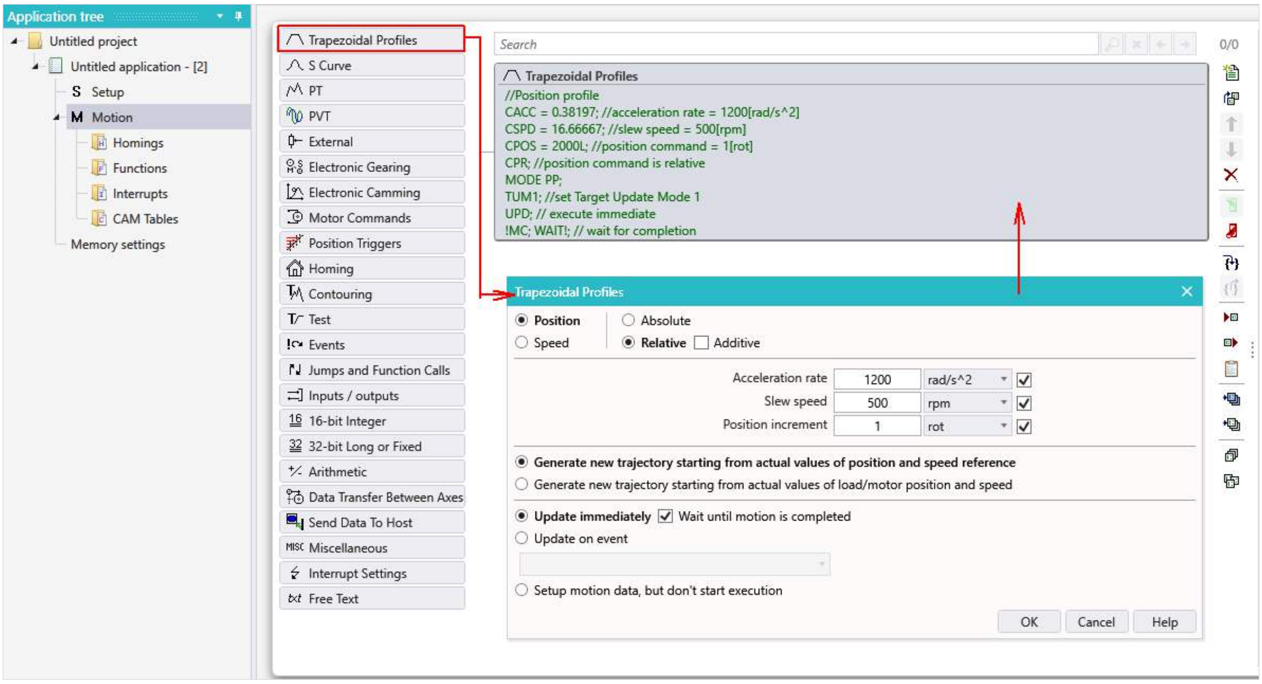


Figure 6.2 - Required movement profile

<u>Step 1</u>: Use the **Trapezoidal Profile** module, to insert the required position profile.



Trapezoidal Profiles

```
//Position profile
CACC = 0.38197; //acceleration rate = 1200[rad/s^2]
CSPD = 16.66667; //slew speed = 500[rpm]
CPOS = 2000L; //position command = 1[rot]
CPR; //position command is relative
MODE PP;
TUM1; //set Target Update Mode 1
UPD; // execute immediate
!MC; WAIT!; // wait for completion
```

Trapezoidal Profiles

● Position ○ Absolute

○ Speed ● Relative ☐ Additive

Acceleration rate 1200 rad/s^2 ☑

Slew speed 500 rpm ☑

Position increment 1 rot ☑

● Generate new trajectory starting from actual values of position and speed reference

○ Generate new trajectory starting from actual values of load/motor position and speed

● Update immediately ☑ Wait until motion is completed

○ Update on event

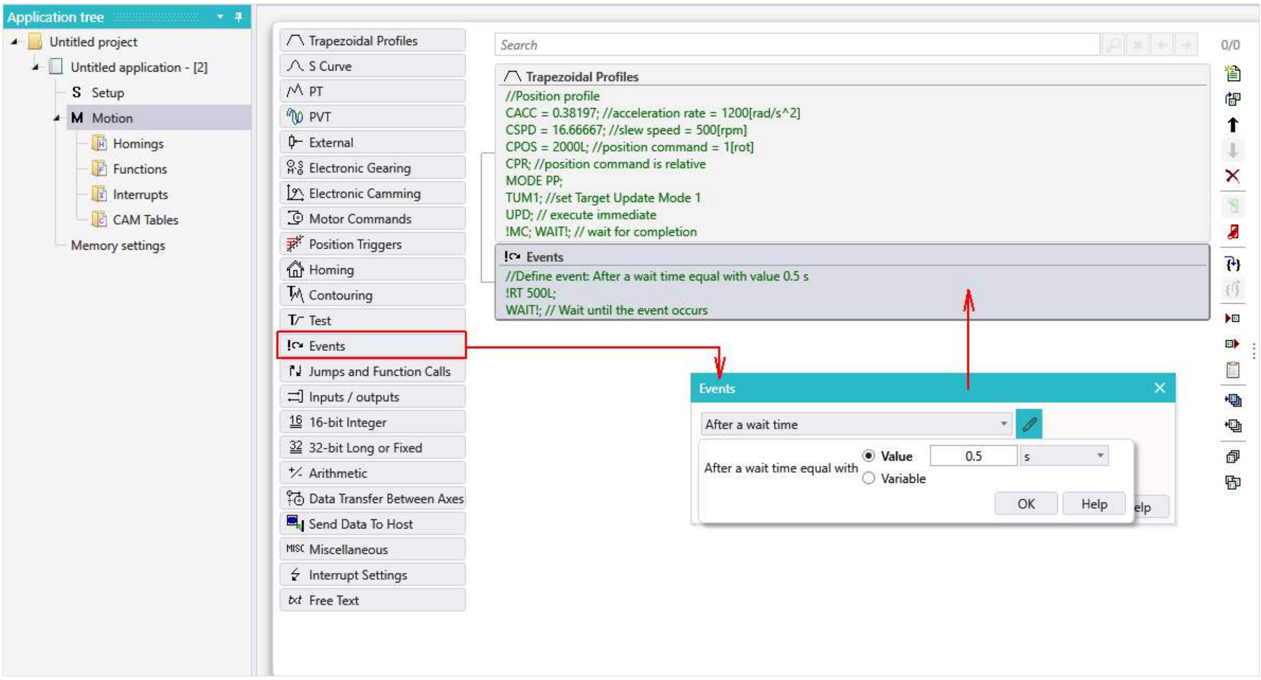
○ Setup motion data, but don't start execution

OK Cancel Help

Figure 6.3 - Insert a trapezoidal position profile

The **Trapezoidal Profile** wizard dialogue allows you to set the motion type (absolute or relative), the motion profile parameters (the acceleration, speed and position commands) and the trajectory generation strategy.

<u>Step 2</u>: Use the **Events** module to insert the 0.5 s wait time.



Trapezoidal Profiles

```
//Position profile
CACC = 0.38197; //acceleration rate = 1200[rad/s^2]
CSPD = 16.66667; //slew speed = 500[rpm]
CPOS = 2000L; //position command = 1[rot]
CPR; //position command is relative
MODE PP;
TUM1; //set Target Update Mode 1
UPD; // execute immediate
!MC; WAIT!; // wait for completion
```

Events

```
//Define event: After a wait time equal with value 0.5 s
!RT 500L;
WAIT!; // Wait until the event occurs
```

Events

After a wait time

After a wait time equal with ● Value 0.5 s

○ Variable

OK Help

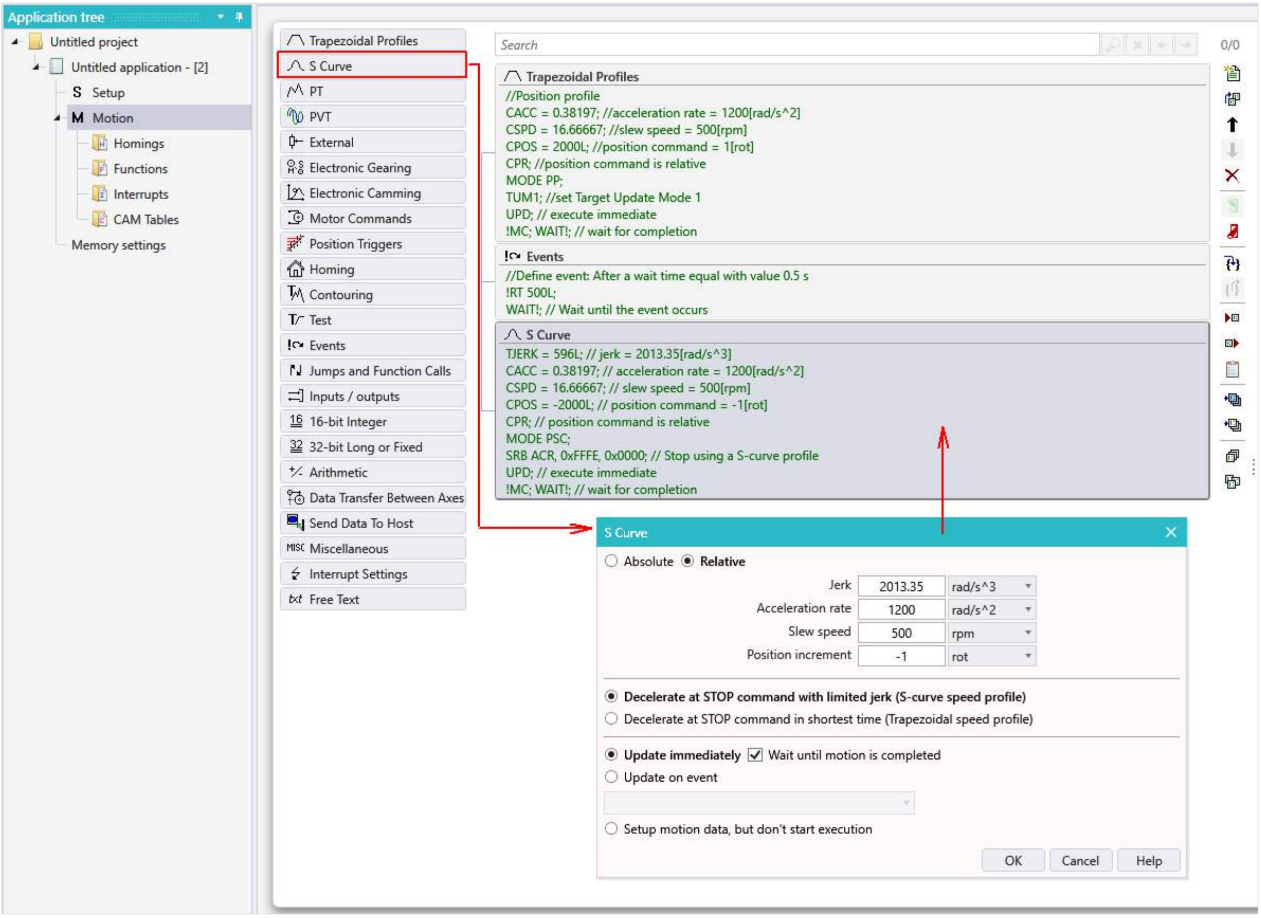
Figure 6.4 – Insert a time event

The **Events** dialogue, allows setting the drive to wait for a condition or even stop the motion when the respective events occurs.

The available events are:

- When actual motion is completed
- Function of motor or load position
- Function on motor or lad speed
- After a wait time
- Function of reference
- Function of input status
- Function of a 32-bit variable value

Step 3: Use the **S Curve** module, to insert the returning profile.



```
Trapezoidal Profiles
//Position profile
CACC = 0.38197; //acceleration rate = 1200[rad/s^2]
CSPD = 16.66667; //slew speed = 500[rpm]
CPOS = 2000L; //position command = 1[rot]
CPR; //position command is relative
MODE PP;
TUM1; //set Target Update Mode 1
UPD; // execute immediate
!MC; WAIT!; // wait for completion

Events
//Define event: After a wait time equal with value 0.5 s
!RT 500L;
WAIT!; // Wait until the event occurs

S Curve
TJERK = 596L; // jerk = 2013.35[rad/s^3]
CACC = 0.38197; // acceleration rate = 1200[rad/s^2]
CSPD = 16.66667; // slew speed = 500[rpm]
CPOS = -2000L; // position command = -1[rot]
CPR; // position command is relative
MODE PSC;
SRB ACR, 0xFFFE, 0x0000; // Stop using a S-curve profile
UPD; // execute immediate
!MC; WAIT!; // wait for completion
```

S Curve

○ Absolute ● Relative

Jerk	2013.35	rad/s^3
Acceleration rate	1200	rad/s^2
Slew speed	500	rpm
Position increment	-1	rot

● Decelerate at STOP command with limited jerk (S-curve speed profile)
○ Decelerate at STOP command in shortest time (Trapezoidal speed profile)

● Update immediately ☑ Wait until motion is completed
○ Update on event

○ Setup motion data, but don't start execution

Figure 6.5 – Insert an S Curve profile

Compared to a trapezoidal position profile, the S-Curve profile allows setting a “jerk” parameter too. This imposes how fast will the acceleration increase from 0 IU to the imposed value.

A detailed description for all options and parameters in the modules dialogues can be found in the help topics, that can be opened by clicking on the **Help** button in the respective module dialogue.

Step 4: Save the project and run the TML program

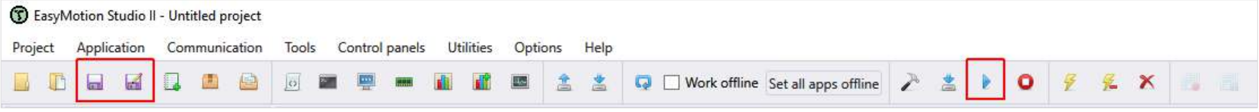


Figure 6.1 – Project “Save” and TML program “Run” buttons

The project can be saved using the **Save** option in the **Project** menu or using the pictogram in the ribbon.

The TML program can be started using the **Run** option in the **Application** menu or using the ribbon button.

After running the program, the motor will start performing the programmed motion profile (do 1 rotation in the positive direction, wait 0.5 s and them perform 1 rotation in the negative direction).

For more examples see the **Application Notes** on our website (link).

6 Motion evaluation and application debugging tools

EasyMotion Studio II includes a series of **control panels**, a **Logger** and a **Scope** that can be used to check the motor/load movement or to monitor, control and debug the TML application.

There are also some **tools** (Binary code viewer, Command Interpreter, Memory View and Output dialogue) that can be used for an advanced application/drive functionality check or debug.

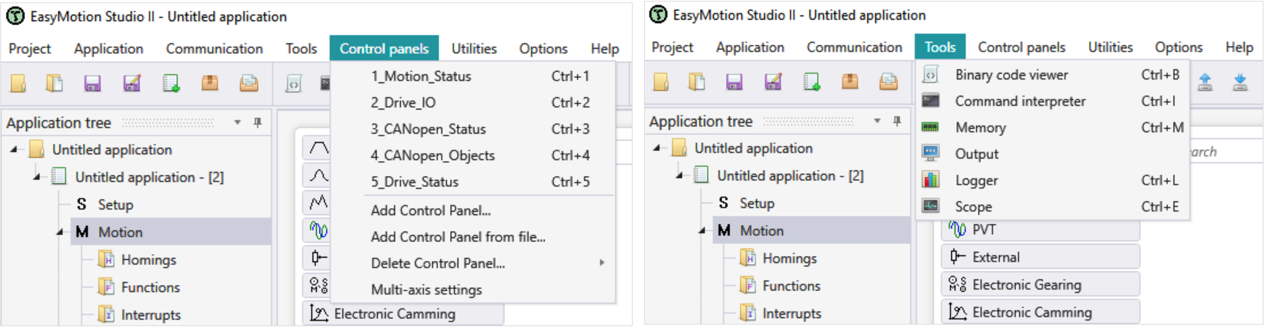


Figure 7.1 – Available control panels and tools

A detailed description of the evaluation tools is available in the EasyMotion Studio II **Help** topics (Help | Help topics | Evaluation Tools …).

6.1 Control panels usage

EasyMotion Studio II includes the following pre-defined control panels:

- **Motion Status** - displays details related to the load position and speed plus the motor current;
- **Drive IO** – displays the drive digital inputs & outputs status plus the analogue inputs level;
- **CANopen Status** – displays the CANopen machine status;
- **CANopen Objects** - display the value of most important CANopen objects;
- **Drive Status** - displays the bits values in the SRL, SRH, MER, DER and DER2 registers.

The default control panels content can be edited by right clicking on a control panel surface and choosing the **Customize** option.

It is also possible to create a new control panel from scratch or import/export an existent one, using the options in the **Control panels** menu.

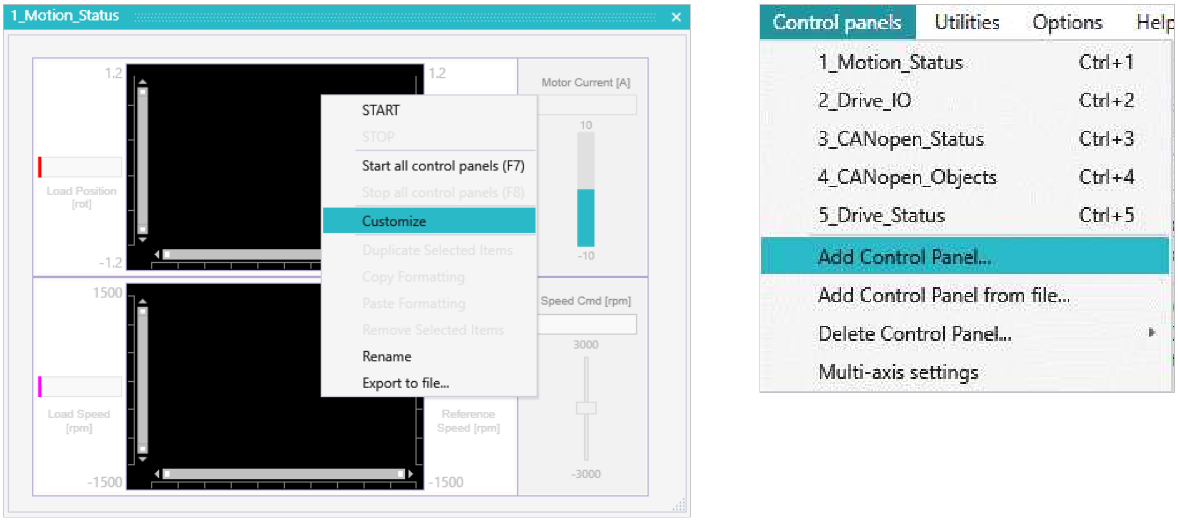


Figure 7.2 - Control panels options

Using the communication to get the information, the values displayed in control panels are not updated in real time. For real time and synchronized results, use the **Logger** or the **Scope** tools.

6.2 Logger usage

The Logger configuration and operation can be done through the buttons in the Logger ribbon or using the menu that opens when right clicking over the Logger surface.

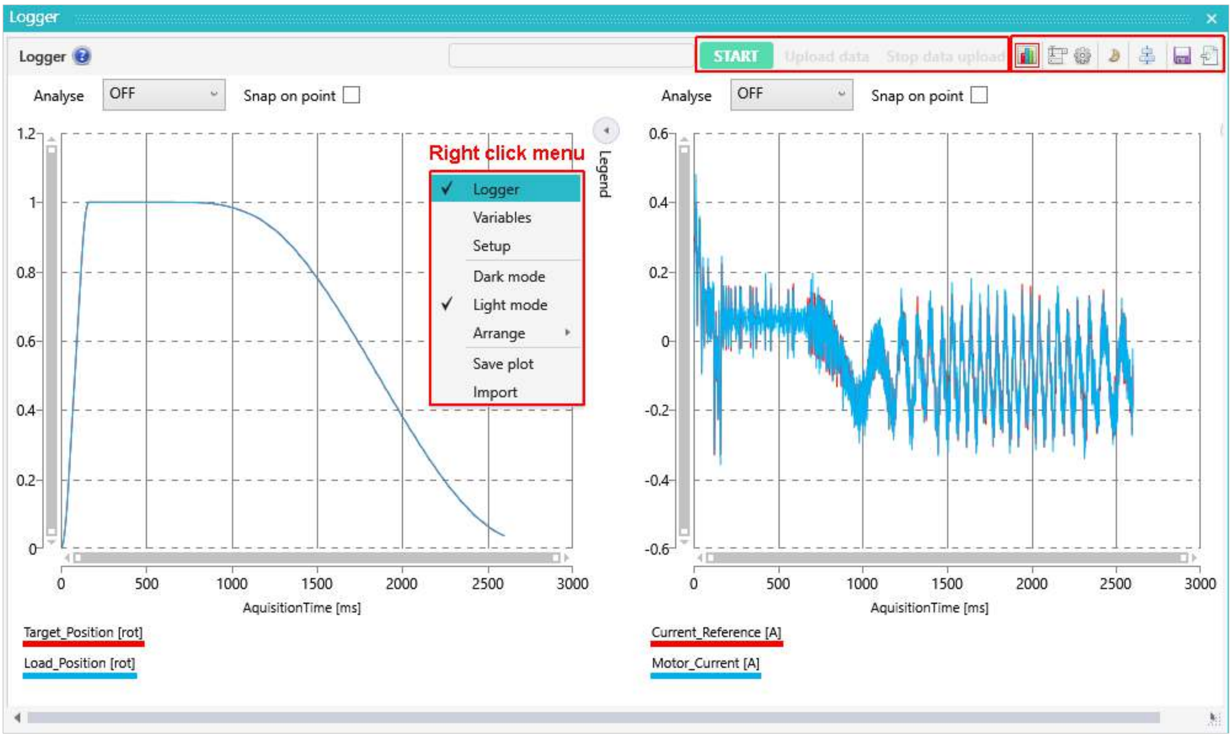
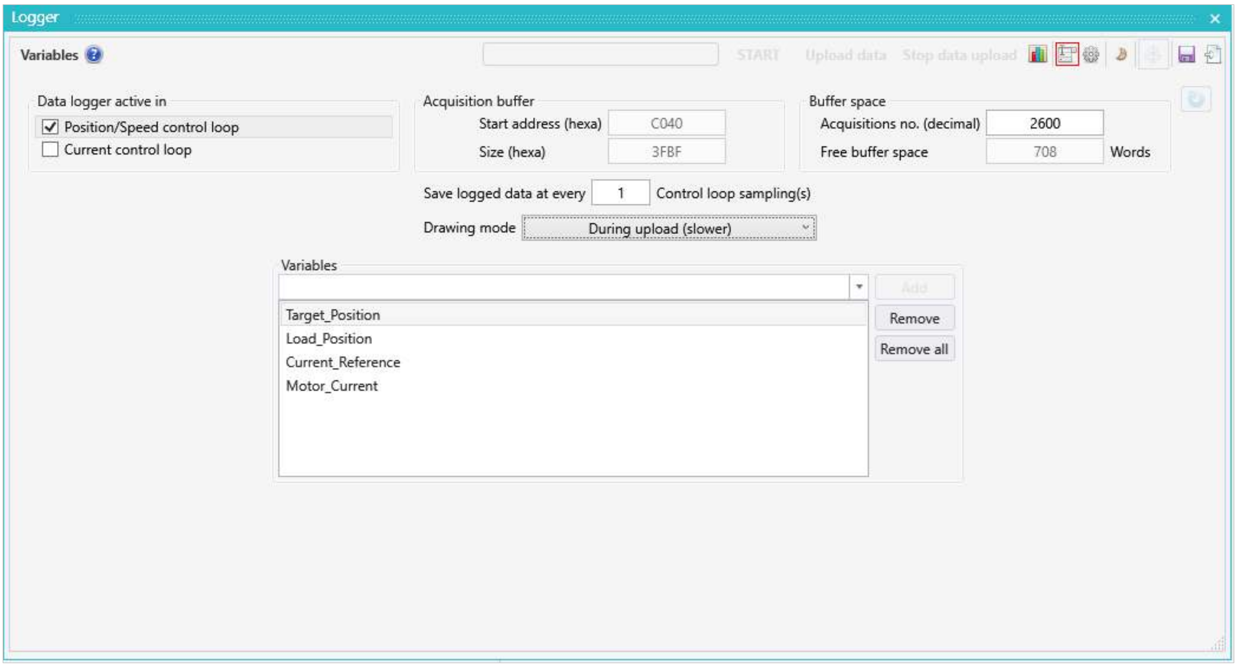


Figure 7.3 – Logger configuration and operation options

The first step in using the Logger consist in choosing the variables and setting the acquisition parameters.


Logger

Variables

START | Upload data | Stop data upload

Data logger active in
☑ Position/Speed control loop
□ Current control loop

Acquisition buffer
Start address (hexa) C040
Size (hexa) 3FBF

Buffer space
Acquisitions no. (decimal) 2600
Free buffer space 708 Words

Save logged data at every 1 Control loop sampling(s)

Drawing mode During upload (slower)

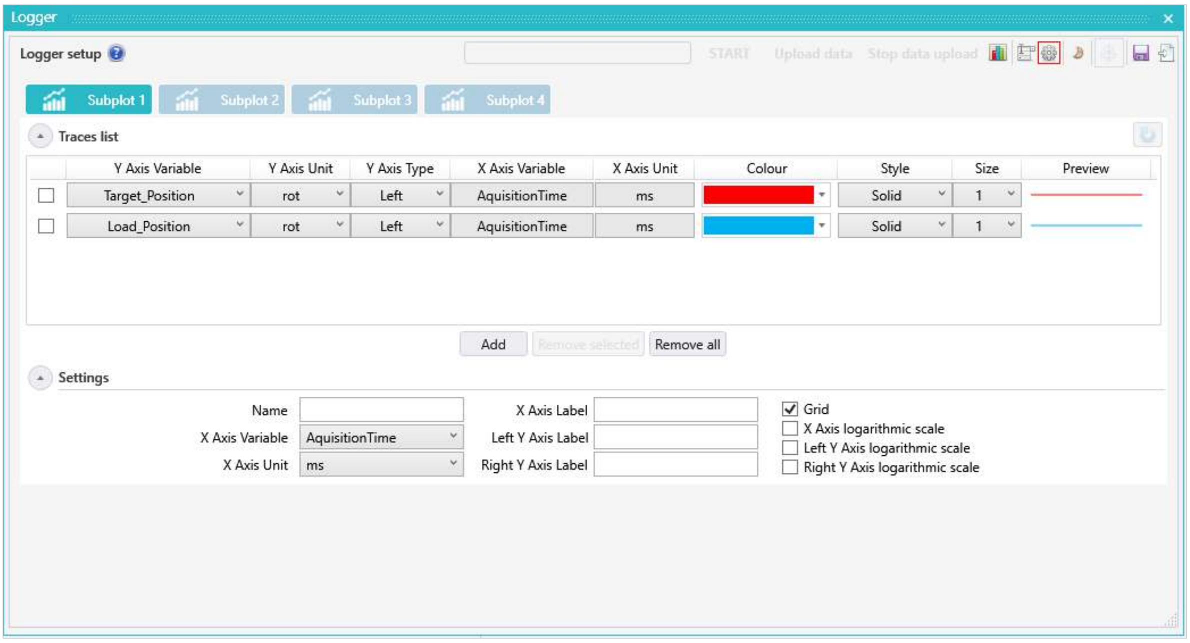
Variables
Target_Position
Load_Position
Current_Reference
Motor_Current

Add | Remove | Remove all

Figure 7.4 – Logger acquisition menu

Remark The Logger mechanism uses the RAM memory of the drive to save the points, so it's very accurate but can save and plot only a finite number of points due to the relatively small memory size.

The **Setup** option allows selecting the variables that will be plotted in each **Subplot** and parametrizing them.



	Y Axis Variable	Y Axis Unit	Y Axis Type	X Axis Variable	X Axis Unit	Colour	Style	Size	Preview
☐	Target_Position	rot	Left	AquisitionTime	ms	🟥	Solid	1	
☐	Load_Position	rot	Left	AquisitionTime	ms	🟦	Solid	1	

Name

X Axis Variable AquisitionTime

X Axis Unit ms

X Axis Label

Left Y Axis Label

Right Y Axis Label

☑ Grid

☐ X Axis logarithmic scale

☐ Left Y Axis logarithmic scale

☐ Right Y Axis logarithmic scale

Figure 7.5 – Logger setup menu

The data acquisition starts automatically when running the TML application but can also be triggered manually, using the **Start** button in the Logger ribbon.

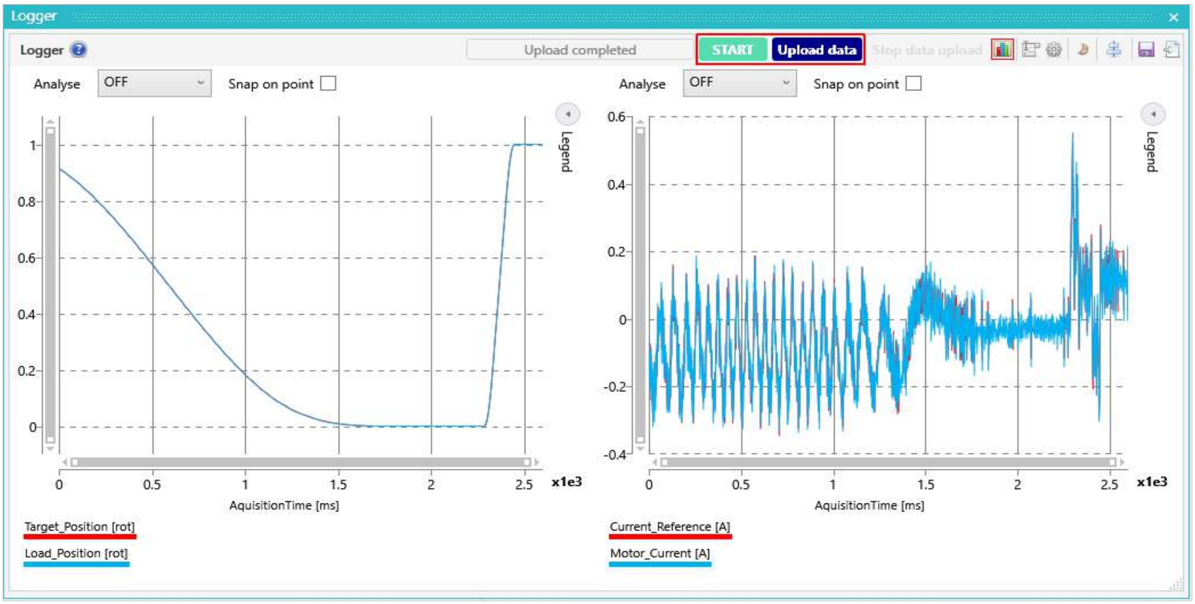


Figure 7.6 – Logger operation options

The **Upload data** button in the Logger ribbon will start the plotting process, while the **Stop data upload** option stops the uploading process, keeping on the Logger just the portion of the recording that was already plotted.

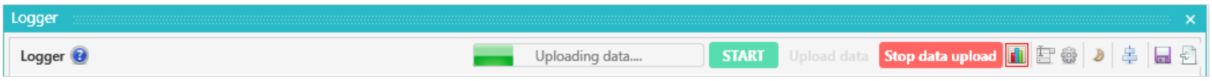


Figure 7.7 – Logger uploading cancelation

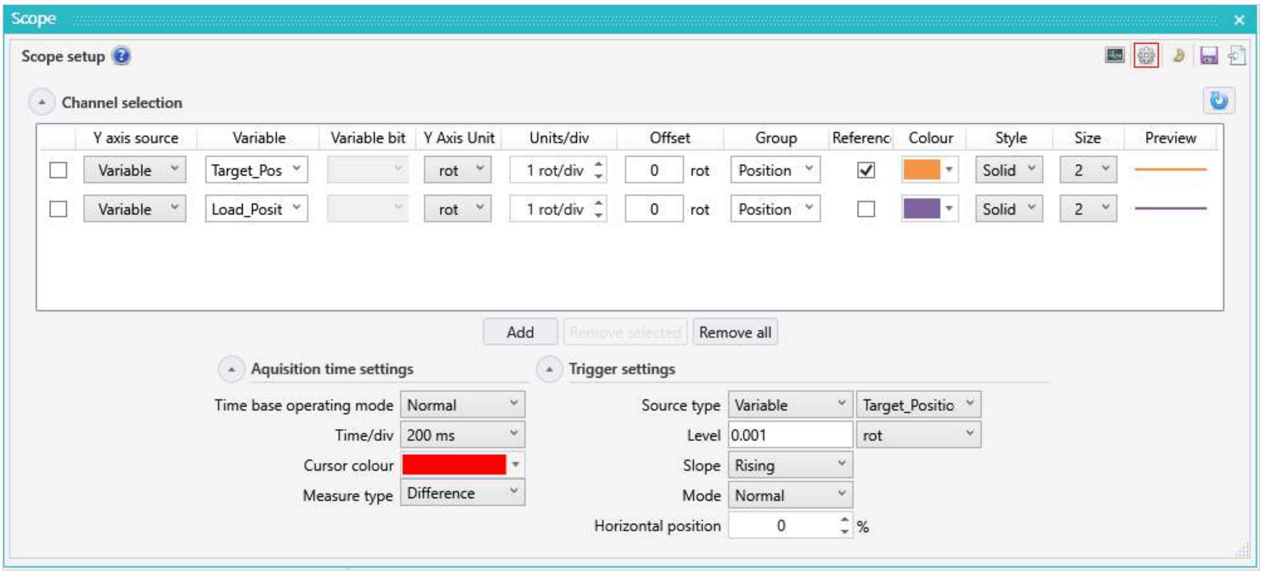
The plots can be saved, using the **Save** button in the Logger ribbon, as a Logger files (*.lgs), a text file ("*.txt"), an image ("*.png") or as an Excel files ("*.csv").

The Logger options and its functionality it is also detailed in the related help topics that can be open from the main **Help** menu or by clicking the "**?**" sign in the Logger interface.

6.3 Scope usage

Working as a 4-channels oscilloscope, the **Scope** tool can be set to monitor the real time evolution of different variables, parameters, registers or inputs/outputs status.

The Scope **setup page** can be accessed through the button in the Scope ribbon or using the right click menu and allows configuring the 4 available channels and setting the acquisition and triggering parameters.



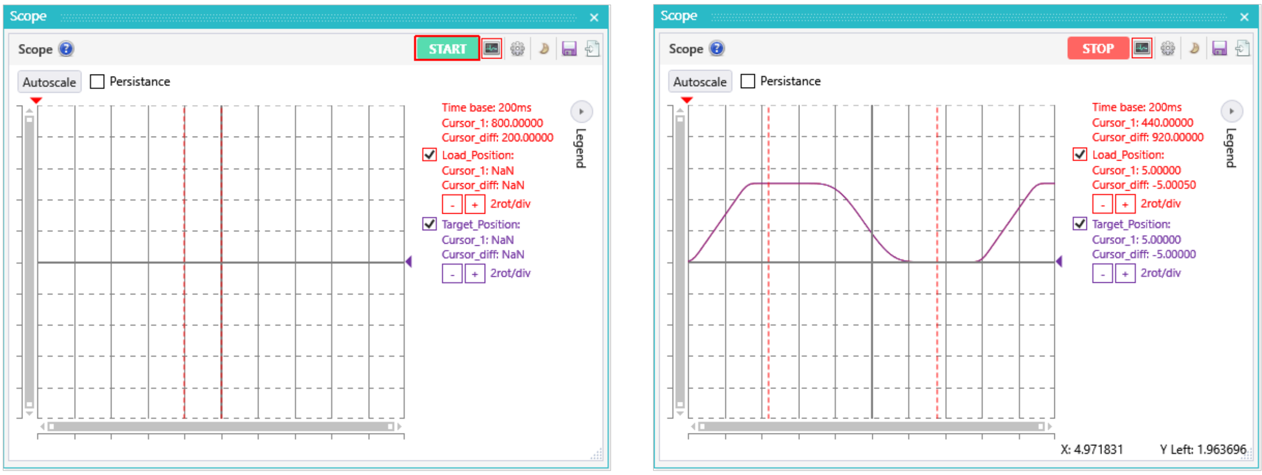
	Y axis source	Variable	Variable bit	Y Axis Unit	Units/div	Offset		Group	Referenc	Colour	Style	Size	Preview
□	Variable	Target_Pos		rot	1 rot/div	0	rot	Position	☑	🟧	Solid	2	
□	Variable	Load_Posit		rot	1 rot/div	0	rot	Position	□	🟪	Solid	2	

Time base operating mode: Normal
Time/div: 200 ms
Cursor colour: 🟥
Measure type: Difference

Source type: Variable, Target_Positio
Level: 0.001 rot
Slope: Rising
Mode: Normal
Horizontal position: 0 %

Figure 7.8 - Scope Setup window

The data acquisition starts when pressing the START button, in the Scope ribbon, and the information is updated continuously until the acquisition is stopped, using the STOP button in the Scope ribbon.



Time base: 200ms
Cursor_1: 800.00000
Cursor_diff: 200.00000
☑ Load_Position:
Cursor_1: NaN
Cursor_diff: NaN
2rot/div
☑ Target_Position:
Cursor_1: NaN
Cursor_diff: NaN
2rot/div

Time base: 200ms
Cursor_1: 440.00000
Cursor_diff: 920.00000
☑ Load_Position:
Cursor_1: 5.00000
Cursor_diff: -5.00050
2rot/div
☑ Target_Position:
Cursor_1: 5.00000
Cursor_diff: -5.00000
2rot/div

X: 4.971831 Y Left: 1.963696

Figure 7.9 - Motion evaluation using the Scope

The plot scale can be changed on the fly using the “-“ and “+” buttons in the Scope legend or the “Autoscale” button.

For simple measurements (value or difference), 1 or 2 cursors are available. They can be activated and set though the **Acquisition time settings** group box in the **Scope setup** page.

The graphs can be saved, using the **Save** button in the Scope ribbon or the **Save plot** option in the right click menu, as a Scope file (*.osc2), a text file (“*.txt”), an image (“*.png”) or as an Excel files (“*.csv”).

All the Scope options and functions are detailed in the help topics that can be opened from the main **Help** menu or by clicking the “**?**” sign in the Scope interface.

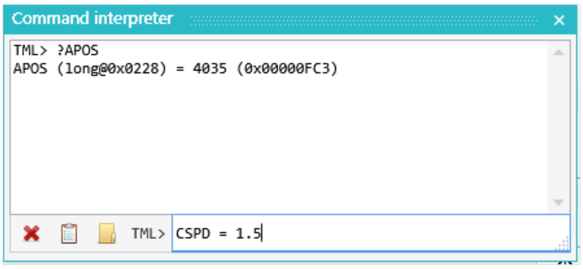
sad

6.4 Advanced tools for application and drive functionality check or debug

Command interpreter

The Command Interpreter tool allows sending on-line TML commands to the online drive(s) or intelligent motor(s).

This tool can be used to set or get TML data: registers, parameters and variables or to display memory locations.



```
TML> ?APOS
APOS (long@0x0228) = 4035 (0x00000FC3)
```

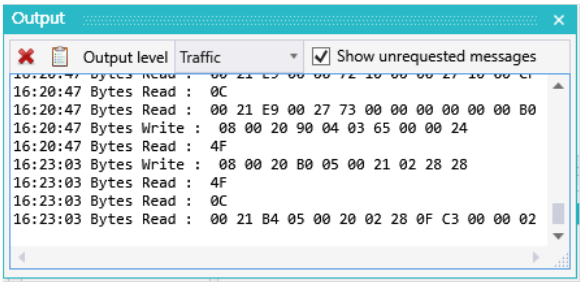
```
TML> CSPD = 1.5
```

Figure 7.10 – Command interpreter overview

Output view

The Output widow allows checking the traffic on the RS232 line and see the following messages:

- error messages
- warning messages
- RS232 traffic
- unrequested messages



Output level: Traffic

☑ Show unrequested messages

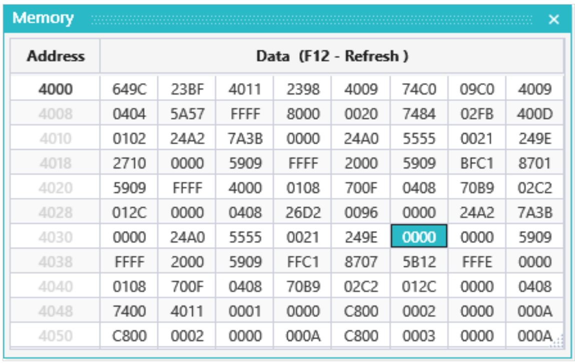
```
16:20:47 Bytes Read :  0C
16:20:47 Bytes Read :  00 21 E9 00 27 73 00 00 00 00 00 00 B0
16:20:47 Bytes Write :  08 00 20 90 04 03 65 00 00 24
16:20:47 Bytes Read :  4F
16:23:03 Bytes Write :  08 00 20 B0 05 00 21 02 28 28
16:23:03 Bytes Read :  4F
16:23:03 Bytes Read :  0C
16:23:03 Bytes Read :  00 21 B4 05 00 20 02 28 0F C3 00 00 02
```

Figure 7.11 – Output view overview

Memory

The Memory window allows viewing or even modifying the contents of the Technosoft drive's and intelligent motor's EEPROM memory.

Remark As this feature is a very low level function, it is NOT recommended to modify the memory contents without a deep knowledge of the use made by the Technosoft drive or intelligent motor of each memory location you intend to modify

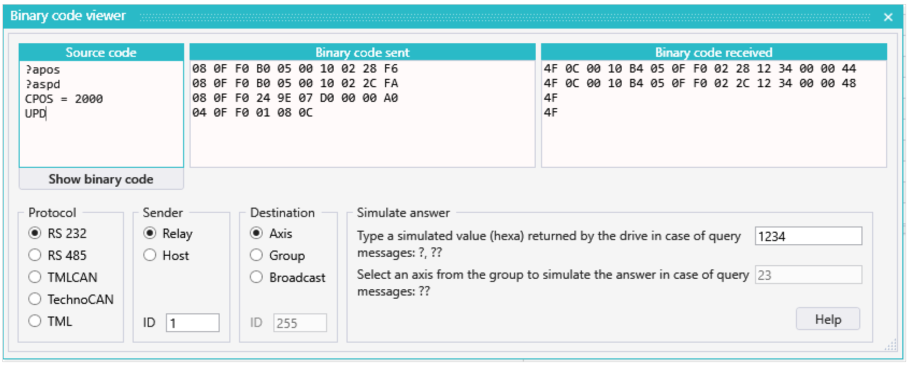


Address	Data (F12 - Refresh)							
4000	649C	23BF	4011	2398	4009	74C0	09C0	4009
4008	0404	5A57	FFFF	8000	0020	7484	02FB	400D
4010	0102	24A2	7A3B	0000	24A0	5555	0021	249E
4018	2710	0000	5909	FFFF	2000	5909	BFC1	8701
4020	5909	FFFF	4000	0108	700F	0408	70B9	02C2
4028	012C	0000	0408	26D2	0096	0000	24A2	7A3B
4030	0000	24A0	5555	0021	249E	0000	0000	5909
4038	FFFF	2000	5909	FFC1	8707	5B12	FFFE	0000
4040	0108	700F	0408	70B9	02C2	012C	0000	0408
4048	7400	4011	0001	0000	C800	0002	0000	000A
4050	C800	0002	0000	000A	C800	0003	0000	000A

Figure 7.12 – Memory dialogue overview

Binary code viewer

The Binary Code Viewer tool offers a quick way to program the host for exchanging messages with a Technosoft drive or intelligent motor. This tool shows how to encapsulate a TML command for all the communication types and the supported protocols. It will display the contents of the messages that needs to be sent and the expected answer, if it is the case.



Source code

```
?apos
?aspd
CPOS = 2000
UPD
```

Binary code sent

```
08 0F F0 B0 05 00 10 02 28 F6
08 0F F0 B0 05 00 10 02 2C FA
08 0F F0 24 9E 07 D0 00 00 A0
04 0F F0 01 08 0C
```

Binary code received

```
4F 0C 00 10 B4 05 0F F0 02 28 12 34 00 00 44
4F 0C 00 10 B4 05 0F F0 02 2C 12 34 00 00 48
4F
4F
```

Protocol: ● RS 232 ○ RS 485 ○ TMLCAN ○ TechnoCAN ○ TML

Sender: ● Relay ○ Host — ID 1

Destination: ● Axis ○ Group ○ Broadcast — ID 255

Simulate answer

Type a simulated value (hexa) returned by the drive in case of query messages: ?, ?? 1234

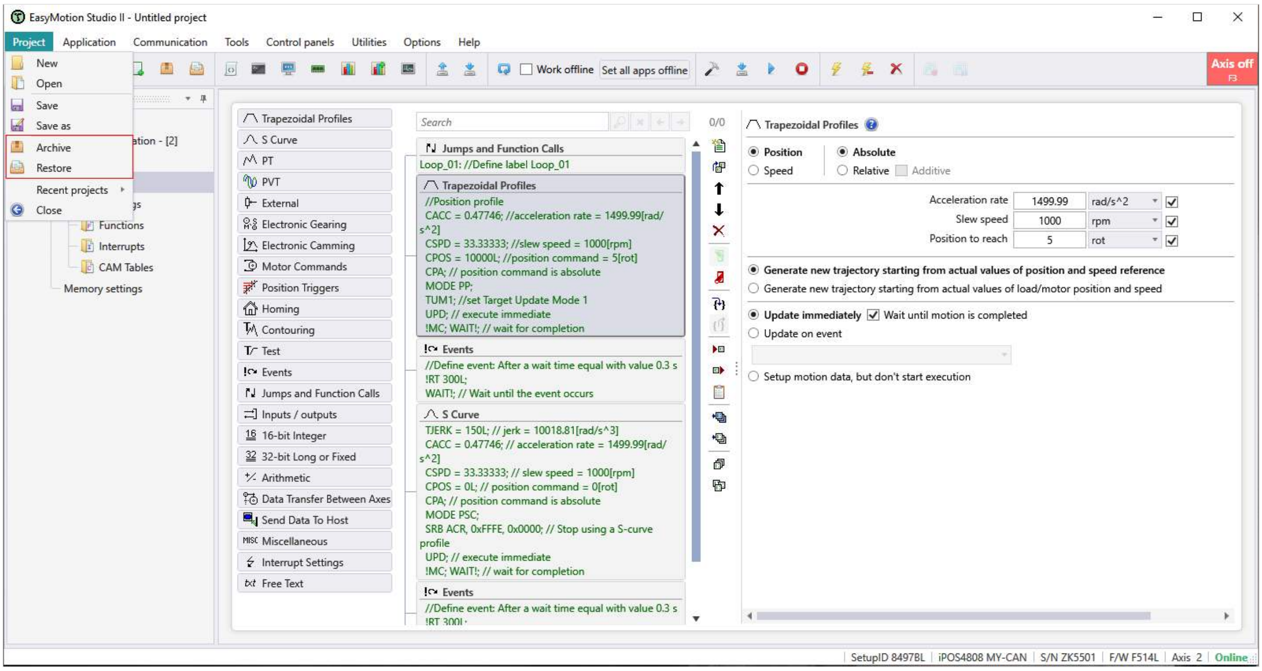
Select an axis from the group to simulate the answer in case of query messages: ?? 23

Figure 7.13 – Binary code viewer overview

The advanced tools are detailed in the EasyMotion Studio II help topics (**Help** | **Help topics** | **Evaluation tools** …).

7 Miscellaneous – Project archiving and restoring

For project storring or transfer purpose EasyMotion Studio II includes an **Archive** and a **Restore** option.



Jumps and Function Calls

```
Loop_01: //Define label Loop_01
```

Trapezoidal Profiles

```
 //Position profile
 CACC = 0.47746; //acceleration rate = 1499.99[rad/s^2]
 CSPD = 33.33333; //slew speed = 1000[rpm]
 CPOS = 10000L; //position command = 5[rot]
 CPA; // position command is absolute
 MODE PP;
 TUM1; //set Target Update Mode 1
 UPD; // execute immediate
 !MC; WAIT!; // wait for completion
```

Events

```
 //Define event: After a wait time equal with value 0.3 s
 !RT 300L;
 WAIT!; // Wait until the event occurs
```

S Curve

```
 TJERK = 150L; // jerk = 10018.81[rad/s^3]
 CACC = 0.47746; // acceleration rate = 1499.99[rad/s^2]
 CSPD = 33.33333; // slew speed = 1000[rpm]
 CPOS = 0L; // position command = 0[rot]
 CPA; // position command is absolute
 MODE PSC;
 SRB ACR, 0xFFFE, 0x0000; // Stop using a S-curve profile
 UPD; // execute immediate
 !MC; WAIT!; // wait for completion
```

Events

```
 //Define event: After a wait time equal with value 0.3 s
 !RT 300L;
```

Trapezoidal Profiles

● Position ○ Speed

● Absolute ○ Relative □ Additive

Acceleration rate	1499.99	rad/s^2	☑
Slew speed	1000	rpm	☑
Position to reach	5	rot	☑

● Generate new trajectory starting from actual values of position and speed reference
○ Generate new trajectory starting from actual values of load/motor position and speed

● Update immediately ☑ Wait until motion is completed
○ Update on event

○ Setup motion data, but don't start execution

SetupID 8497BL | iPOS4808 MY-CAN | S/N ZK5501 | F/W F514L | Axis 2 | Online

Figure 7.1 – "Archive" and "Restore" options

A project archive (file extension "*.e2pf") can be created by choosing **Project** | **Archive**. The project archive includes all the project information, the Logger settings, the control panels and other custom files too. This can be used to create a back-up or to transfer the project to other PCs.

The **Restore** option extracts the "*.e2pf" archive content and restores it as a full EasyMotion Studio II project. The restored project is identically with the one from where the archive was created.

TECHNOSOFT

MOTION TECHNOLOGY