# User Manual PNDS3

For Plug & Drive Studio 3 Version 3.1.8

User Manual Version: 1.3.1

# Contents


1 Document aim and conventions ..... 3

2 For your safety ..... 5
2.1 Warn and risk levels ..... 5

3 Before you start ..... 6
3.1 System and hardware requirements ..... 6
3.2 Intended use and audience ..... 6
3.3 Delivery scope and warranty ..... 7

4 Your product ..... 8

5 Installation and adapter ..... 10

6 User interface (UI) ..... 11
6.1 Header (1) ..... 12
6.2 Project bar (4) ..... 14
6.3 Work desk (3) ..... 17
6.4 Display wall (2) ..... 19

7 Project setup ..... 21

8 Special controls ..... 24
8.1 Complex controls ..... 24
8.2 Device communication ..... 26

9 Oscilloscope ..... 27

10 Imprint, versions ..... 28

# 1 Document aim and conventions

Beside technical data, this document explains product use and function. For possible combination with other Nanotec products, please ask your Nanotec sales partner. Before using the product, please note document font styles and conventions.

Underlined text marks a cross reference or hyperlink.

Example 1: Observe our safety notes.

Example 2: Download needed code templates from our website for EMEA / APAC or AMERICA.

**Gray bold italics** call out **menu paths, buttons, tab** and **file names**.

Example 1: Select **Home > Connect controller > CANopen**.

Example 2: In the **NanoJ** tab, select **NanoJ project** and open **Analog Input.cpp**.

*Plain italics* mark *Freehand entries* and *foreign-language* expressions. They also emphasize words of critical weight. Alternatively, bracketed exclaim marks(!) give critical weight.

Example 1: Enter *Plug & Drive Studio*. In addition to users (= *Nutzer; usuario; utente; utilisateur; utente* etc.), this document also addresses:

- Third-party users (= *Drittnutzer; tercero usuario; terceiro utente; tiers utilisateur; terzo utente* etc.).
- End users (= *Endnutzer; usuario final; utente final; utilisateur final; utente finale* etc.).

Example 2: Protect yourself, others and your equipment. Follow our *general* safety notes that are generally applicable to *all* Nanotec products. Also follow the *specific* safety notes that apply to *this* specific product.

`Courier` marks `code blocks` or `programming commands`.

Example 1: Via Bash, call `sudo make install` to copy shared objects; then call `ldconfig`.

Example 2: Use the following NanoLibAccessor function to change the logging level in NanoLib:

```
//
        ***** C++ variant *****
 void setLoggingLevel(LogLevel level);
```

### The verb *to co-click*

*Co-clicking* means a mouse click by secondary key to open context menus etc.

Example 1: Co-click the file, select **Rename**, and rename the file.

Example 2: Co-click the file to check and select **Properties**.

### Numerical values

Numbers appear in decimal. Hexadecimal notation ends in subscript *h.* Objects in the object dictionary notate in hexadecimal as `<Index>:<Subindex>`, non-notated subindices as $00_h$. Example: $1003_h$:$05_h$ is subindex 5 in object $1003_h$. And $6040_h$ is subindex 00 in object $6040_h$.

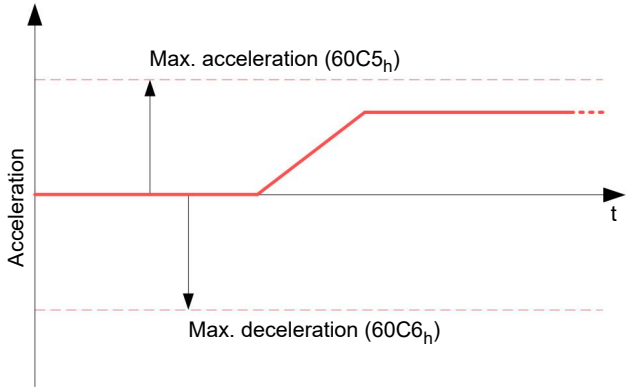
### Bits

Each object bit counts up from LSB (bit number 0), such as data type *UNSIGNED8*:

| | MSB | | | | | | | LSB | |
|---|---|---|---|---|---|---|---|---|---|
| Bit Nummer | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| Bits | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | ≙ $55_{hex}$ ≙ $85_{dec}$ |

### Count direction (arrows)

Illustrations always count arrow-wards; both example objects 60C5$_h$ and 60C6$_h$ are thus positive.



# 2 For your safety

Before product use, please ensure that all users read, understand and follow the instructions in this document fully.

## 2.1 Warn and risk levels

Please note: our hazard warnings, alert symbols and signal words mark different risk levels.

**CAUTION!**

**CAUTION warns of *possible physical danger!***

Minor / moderate injury possible.

► Instruction against **unhealthy** user errors.

**NOTICE**

**A NOTICE warns of wrong operation.**

Material or ecological damage possible (not strictly injury).

► Instruction against **destructive** user errors (= mere material risks).

**Note:** Explains or simplifies a process by additional information.

# 3 Before you start

Before product use, you need to prepare the PC and verify product intent / limits. Via online help, you can learn how to install and set up projects and how PNDS3 runs. Observe the safety notes in the manual (www.nanotec.com).

## 3.1 System and hardware requirements

Plug & Drive Studio 3 (PNDS3) needs 64-bit operating systems. Nanotec recommends controller firmware *FIR-v2213* or newer. PNDS3 offers a special control for firmware update.

**PNDS3**

v3.1.7

**64-bit OS requirements**

- Windows 10
- .NET Framework 4.8
- Display resolution 1920x1080

**Fieldbus adapters / cables**

- **CANopen:**
  - IXXAT USB-to-CAN V2
  - Nanotec ZK-USB-CAN-1
- **Modbus RTU:**
  - Nanotec ZK-USB-RS485-1 or equivalent USB-RS485 adapter
  - USB cable via virtual comport (VCP)
- **Modbus TCP:**
  - suitable ethernet cable
- **Ethernet (REST) , EtherCAT, Profinet:**
  - suitable ethernet cable
  - WinPcap 4.1.3, or Npcap installation, see Installation and adapter

## 3.2 Intended use and audience

**NOTICE**

**Damage: from unskilled staff!**

► Use the product only for the purpose described in this document.

► Restrict use to expert staff only.

► Follow valid OEM and system prescriptions for all equipment involved.

Plug & Drive Studio 3 (PNDS3) is a free software for easy Nanotec drive commissioning. The underlying operating system / hardware (PC) is **not** real-time capable. **Never** use PNDS3 for time-critical or synchronous multi-axis motion **nor** integrate it as a safety component in a product or system.

Add proper warnings and instructions for safe use / operation to each end user product with a Nanotec-produced component. Submit any Nanotec warning directly to end users. The product addresses skilled experts in industry use cases alone. Expert means:

- Training / experience in motor and controller handling
- Understanding this document plus Nanotec drive manuals
- Knowledge of valid regulations

## 3.3 Delivery scope and warranty

PNDS3 comes as a **.zip* folder from our download website for either EMEA / APAC or AMERICA. Duly store and unzip your download before setup. The product package contains:

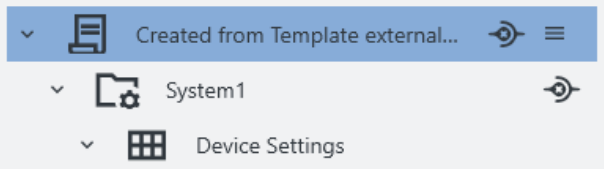
- Software as an executable file
- Project templates
- Current firmware release
- Online help file

For scope of warranty, please observe our terms and conditions for either EMEA / APAC or AMERICA. **Note:** Nanotec is not liable for wrong quality, handling, installation, operation, use, and maintenance of third-party equipment! Follow valid OEM instructions.

# 4 Your product

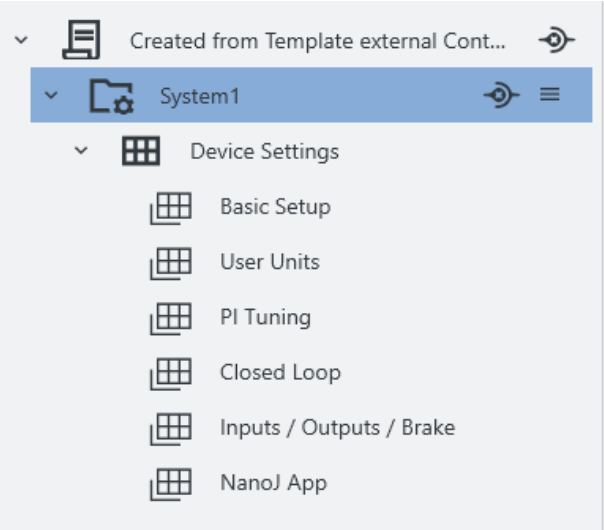
With PNDS3, you parametrize and commission Nanotec drives. Using templates for various Nanotec drives, you can add your own projects, systems and modules to the modular user interface. The software comes with a default folder structure (*Project, System, Module Group, Module,* etc.).

### Project




You manage all settings and device parameters in projects, save these as a file and im- / export them, say, as a template. Such a reusable **Project** can have multiple systems, say, the axes of a machine.

### System

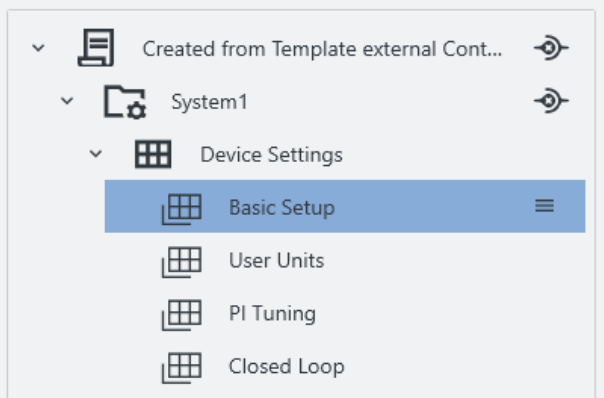



In a project (here: external controller), you create and store drive systems (here: X-axis). Each is im- / exportable as template.

You can extend such a reusable **System**, of at least motor and controller, by modules or module groups for encoder, gearbox, brake, settings, parameters, etc.

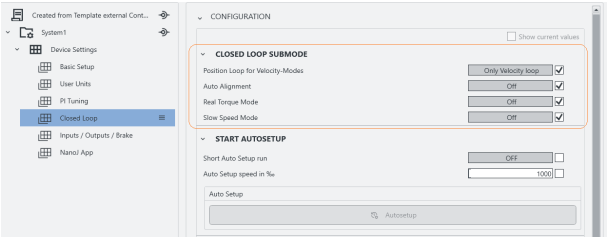
By parameters, sortable / poolable into several modules or module groups, you quickly control all system elements.

### Module (Group)




A module (group) contains parameters or controls (groups) and is im- / exportable, single or grouped, as template.

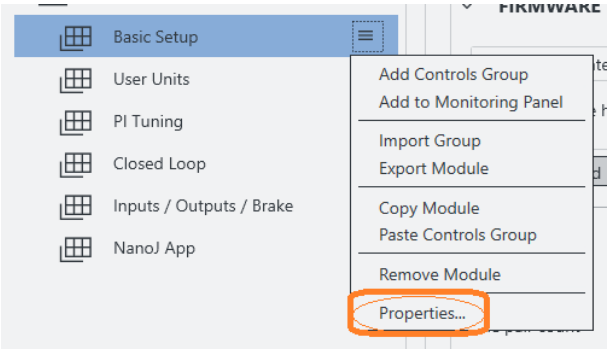
## Controls Group




A **Controls group** pools single device parameters (objects from the dictionary in the controller) and / or **Special controls**.

You im- / export such a control group together with set values, say, as template.

## Property editing




Simply co-click an element, select **Properties**, insert a visible name, version number, and description: This way you create your individual user interface.

# 5 Installation and adapter

Install the software, set up the adapter – and PNDS3 is ready to go. You find PNDS3 software online as a zip download.

1. Open the website **Nanotec > Products > Software > Plug & Drive Studio 3**.
2. Download and extract the product zip file.
3. Run the executable file **PNDS3.exe**.
4. Only with PNDS3 installed: Prepare your fieldbus adapter (see below).

### CANopen

1. Decide: **Ixxat USB-to-CAN**? Or **Nanotec ZK-USB-CAN-1**?
2. For **Ixxat USB-to-CAN:** Download the driver (www.ixxat.com/); install it by hand.
3. Connect the adapter to the computer. For **Nanotec ZK-USB-CAN-1:** Wait for self-installation.
4. Via correct cable (see product manual): Connect the installed adapter to the controller.

### USB: Nanotec Virtual COM-Port (VCP)

1. Connect the voltage supply to the controller and switch it on.
2. Via correct USB cable: Connect the PC to the controller (= "mass storage device").
3. In Explorer > Controller directory: Select `cfg.txt` (= `pd4ccfg.txt`for a PD4C).
4. Open the file via text editor (Notepad etc.).
5. Add the lines `2102|=0x100000` and `4015:01=0`. Save the file.
6. Restart the controller and check if its COM port appears in the device manager.

### Modbus RTU

1. For **Nanotec ZK-USB-RS485-1**: Connect the adapter to the computer and wait for self-installation.
2. For **other equivalent adapters**: Follow valid OEM instructions to install the driver.

### Modbus TCP

Configure the IP address of the drive and the ethernet adapter accordingly, as described in the drive manual.

### EtherCAT

Install WinPcap 4.1.3 or Npcap and make sure the corresponding driver is activated for the designated ethernet adapter.

### Profinet

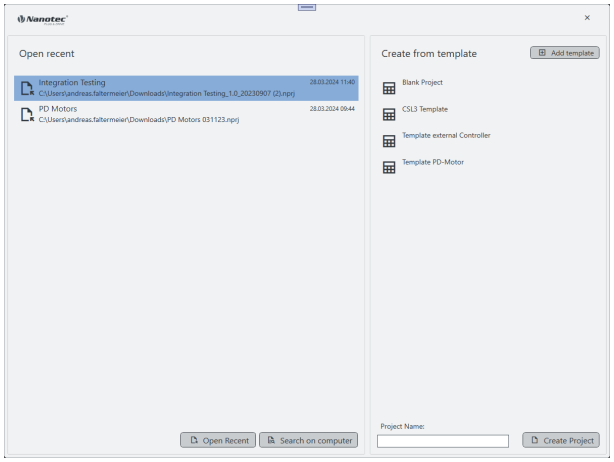
1. Install Win10cap or Npcap and make sure the corresponding driver is activated for the designated ethernet adapter.
2. Configure the IP address of the drive and the ethernet adapter accordingly, as described in the drive manual.

### Ethernet (REST)

Configure the IP address of the drive and the ethernet adapter accordingly, as described in the drive manual.

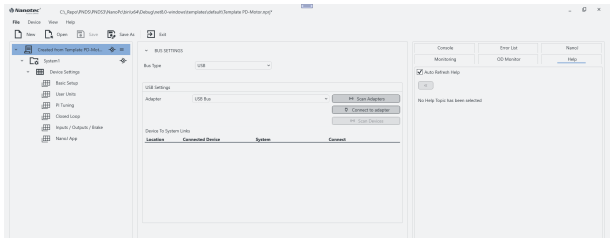
# 6 User interface (UI)

Thanks to flexible areas and windows, fitted into the main window or usable stand-alone, you can master a wide range of tasks. Before product use, please understand the UI structure.

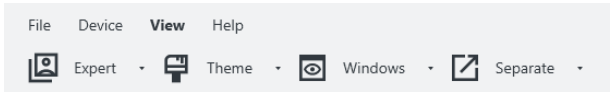



When PNDS3 starts for the first time, you are asked to create a new project: either a blank one or one based on a template.

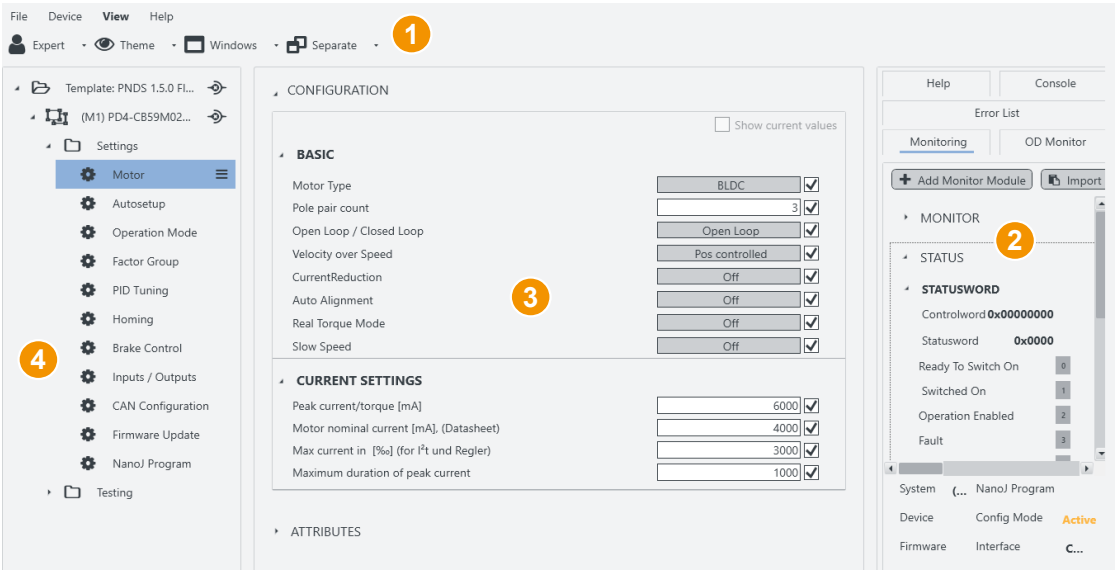
If you open a blank project, the project tree is empty at first.



If you create a new project or load a template, the interface fills up according to your needs. This way, you design your own UI.




Using the **View** options in the main menu you can further customize the UI by changing the theme, showing/hiding features or opening them in separate windows.

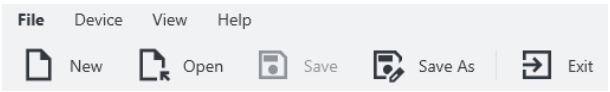



- Header for main menu (1).
- Display wall (2) for monitoring, object directory, help etc.
- Work desk (3) for user controls etc.
- Project (or side) bar (4) for systems etc.

## 6.1 Header (1)

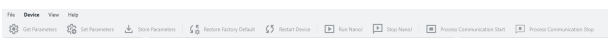
As a prominent layout bracket on top in the user interface, the UI header contains all basic functions and commands relevant to projects, devices and the UI view.

### File



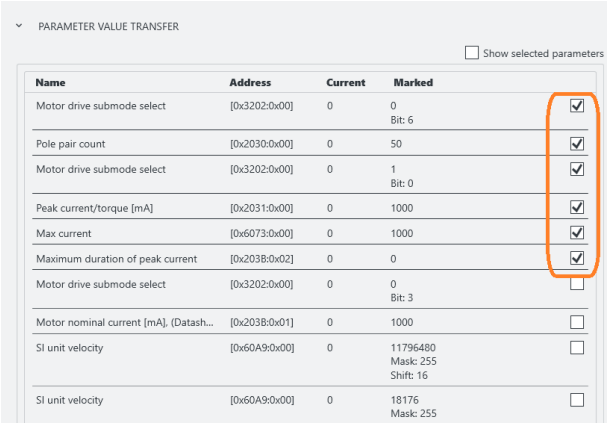

Leftmost above the header, you find the main menu for project files. You can load new – and save, reopen, edit existing projects.

### Device



Read, write, and save device parameters. Govern NanoJ programs and fieldbus network (with CANopen).

**Set Parameters:** Transmits *the selected* parameter values to system-connected controllers.




**Note:** You can select parameters to be set by ticking them. You can find a list of all used parameters under **Parameter Value Transfer** if you click the respective system.

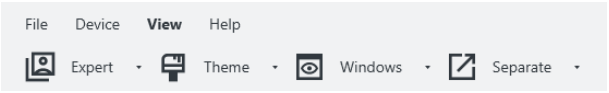
**Get Parameters:** *Reads* the values of system-connected controllers.

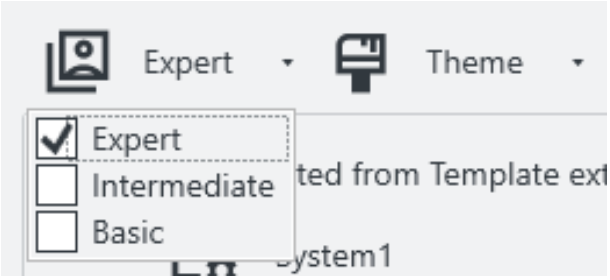
**Store Parameters:** *Stores* **Set**-transmitted values of system-connected controllers.

**Restore Factory Default:** The stored objects are discarded, with the exception of the category *Tuning* and the fieldbus-specific category. You can find further details in the chapter *Discarding the saved data* in the manual of your controller.

The change does not take effect until after the controller is restarted (**Restart Device**).

### View




Here you can customize the UI by changing the theme, showing/hiding features or opening them in separate windows.

You can also set the **User level**, to govern user rights for the following roles:

- **Expert**: Project owner with all rights. May create and edit projects, rights, visibilities, etc. Governs via **Properties**, for each single parameter up to a complete **Controls group**, *who* may see and edit exactly *what*.
- **Intermediate**: May change device parameters, but can't edit a project.
- **Basic**: Similar to **Intermediate**, but often gets fewer editing rights from **Expert**.

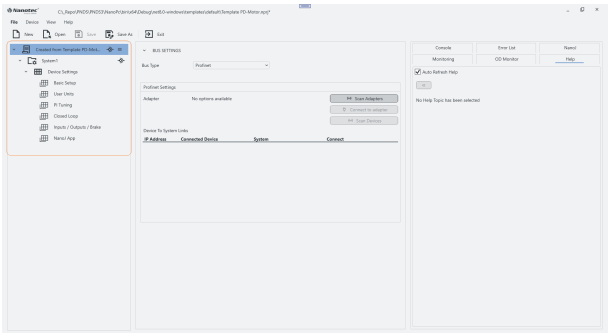
### Help


Open the online help or PNDS3 version info.

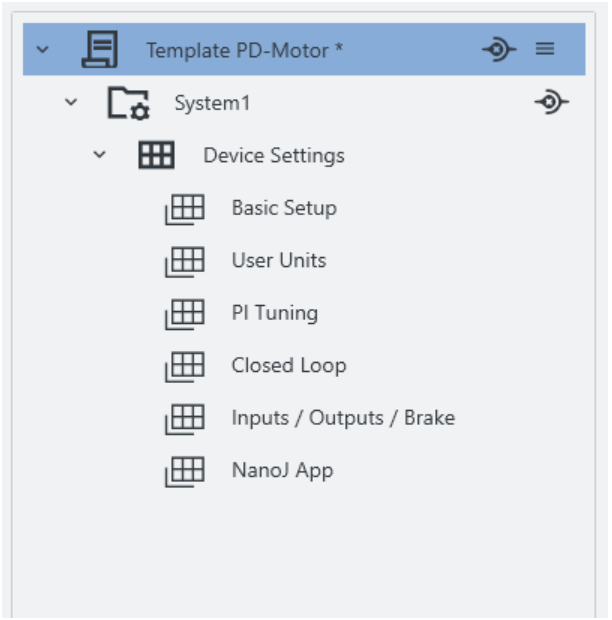
## 6.2 Project bar (4)

This side bar diplays your loaded project as a tree list by which you create the user interface. **Note:** Depending on assembly, you can check connections and attributes of all tree list items in the work desk (3).

### Tree list

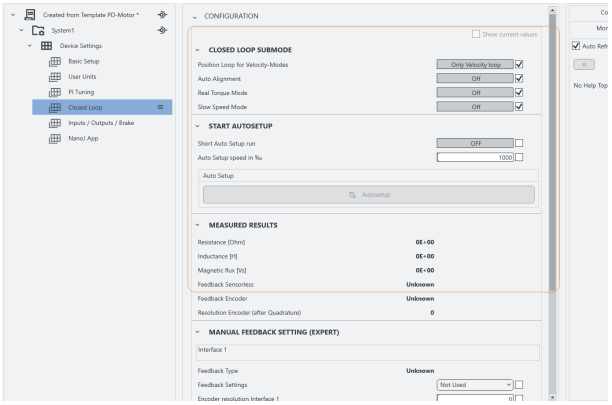


You find the project bar in the very left of the user screen.



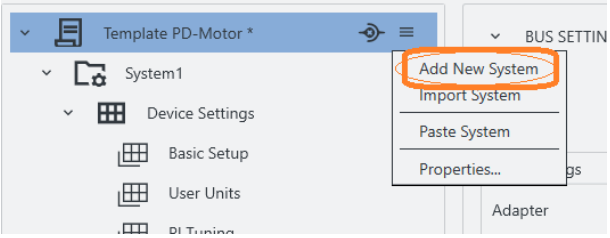

A project (here: for an *external controller*) tree-lists all systems and the items therein (see also Project setup). *One* project and *one* system are minimum; further items are optional and later on define the entire UI layout.

*System 1* contains the module groups *Quick Start* with modules for the basic settings and *Application Settings* with further controls and parameter groups.



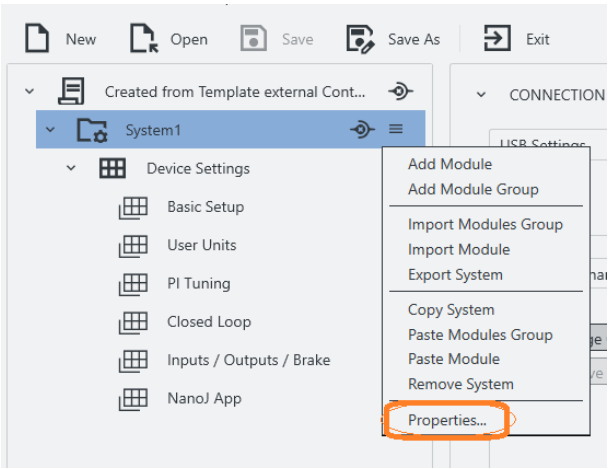
For each module, you may add one or more controls groups to the work desk (3) further to the right.

### Project > System

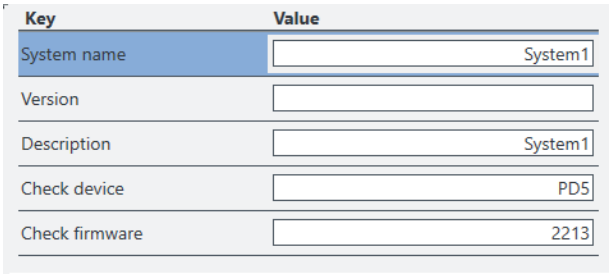



A system represents a motor with controller, that is, one per motor in a multi-axis application.

1. To set up a system: Co-click the project.
2. In the context menu: Either create a new system via **Add New System**.
3. Or fetch an existing one via **Import System**.



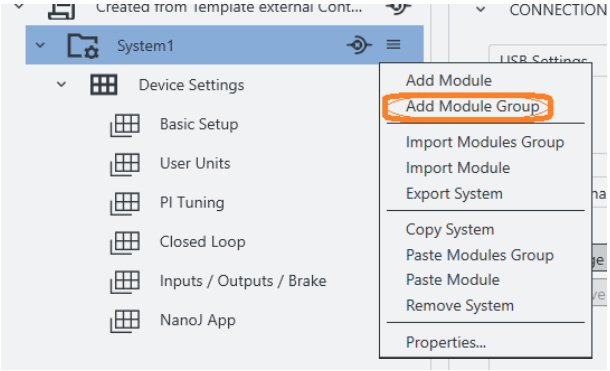

4. A new node (= blue) appears in the tree list.
5. To name it: Co-click the node, select **Properties**. **Note:** You can edit *any* object via **Properties**.



| Key | Value |
|---|---|
| System name | System1 |
| Version | |
| Description | System1 |
| Check device | PD5 |
| Check firmware | 2213 |

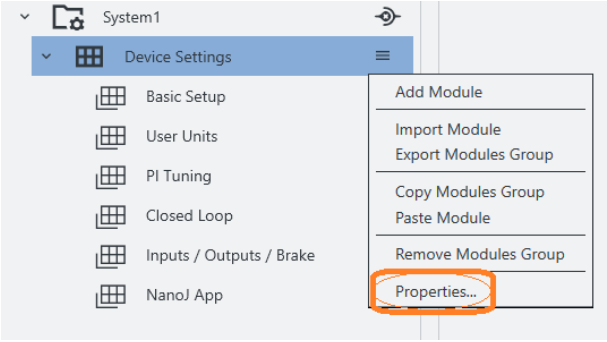
6. In the **Properties** window: Name the system as needed.
7. If needed: Versionize and describe the system. You can add a string for the device name and firmware version, which should be checked after connecting to a device.
8. After last entry: Set a tab stop (so that all is stored).
9. Assemble the system with module groups (see below).

### Project > System > Module group

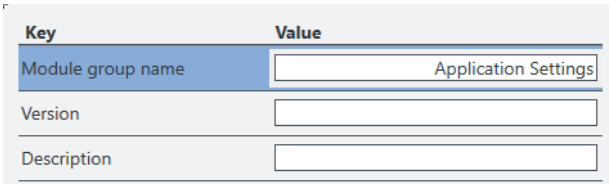



A module group bundles *several* motor functions (= modules). Depending on assembly, you can check its connections and attributes in the work desk (3).

1. To set up a module group: Co-click the system.
2. In the context menu: Either create a new module group via **Add module group**.
3. Or fetch an existing one via **Import module group**.

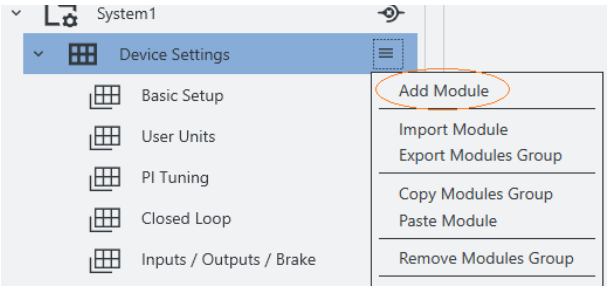



4. A new node appears in the tree list.
5. To name it: Co-click the node, select **Properties**.
   **Note:** You can edit *any* object via **Properties**.



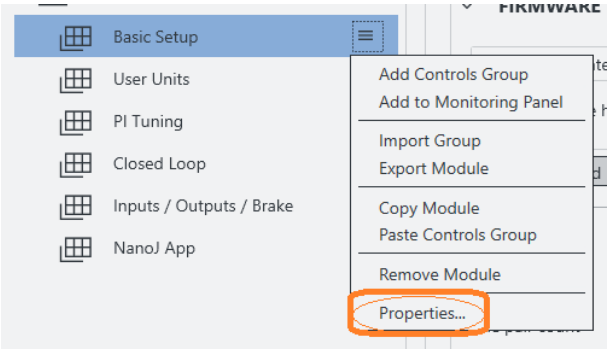

6. In the **Properties** window: Name the module group as needed (here: *Controller template*).
7. If needed: Versionize and describe the module group.
8. After last entry: Set a tab stop (so that all is stored).
9. Assemble the module group with modules (see below).

### Project > System > Module group > Module

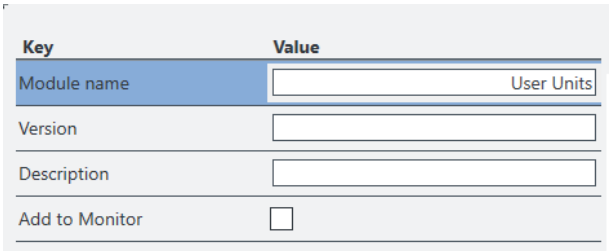



A module allows you to add a *single* motor function (= parameter set etc.). Depending on assembly, you can check its connections and attributes in the work desk (3).

1. To set up a module: Co-click the module group (here: *Controller template*).
2. In the context menu: Either create a new module via **Add Module**.
3. Or fetch an existing one via **Import Module**.

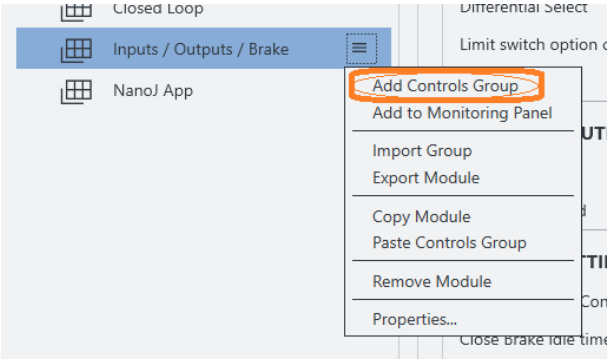



4. A new node appears in the tree list.
5. To name it: Co-click the node, select **Properties**.
   **Note:** You can edit *any* object via **Properties**.



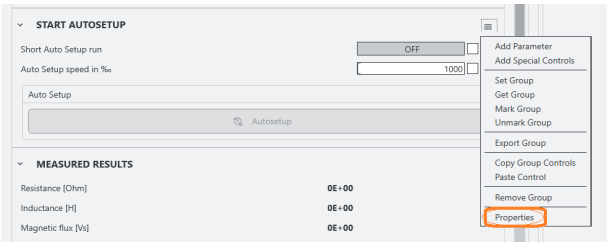

6. In the **Properties** window: Name the module as needed (here: *Communication settings*).
7. If needed: Versionize and describe the module, add it to a Monitor
8. After last entry: Set a tab stop (so that all is stored).

**Project > System > Module group > Module > Controls group**

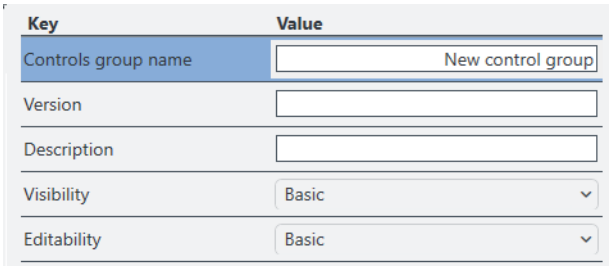



A controls group bundles individual operating elements or parameter sets.

1. To set up a controls group: Co-click the module.
2. In the context menu: Either create a new controls group via **Add Controls Group**.
3. Or fetch an existing one via **Import Group**.




4. In any case, the controls group appears in the work desk (3).
5. Right there: Co-click the group and its **Properties**. **Note:** You can edit *any* object via **Properties**.




6. In the **Properties** window: Name the controls group as needed.
7. If needed: Versionize and describe the group. **Note** the pull-downs for granted viewing and editing rights (here: both *Basic*).
8. After last entry: Set a tab stop (so that all is stored).

## 6.3 Work desk (3)

At the work desk, in the user screen's half-left, you edit the properties / contents / controls of your project and systems. Depending on assembly, different tabs are above the worktable:

An **Attributes** area accompanies all items (also module groups); **Bus settings**, by contrast, only the project itself. The **Connection settings** tab, finally, is for systems only; and **Configuration** is only for modules. Each tab opens different aspects:

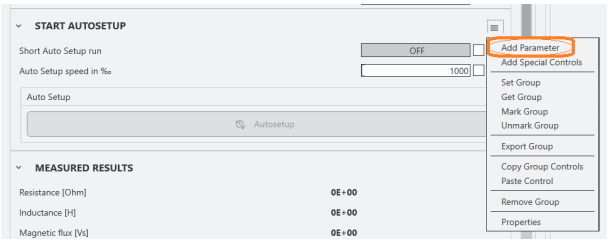
**Controls groups** Operator clusters

**Parameters** Operator values

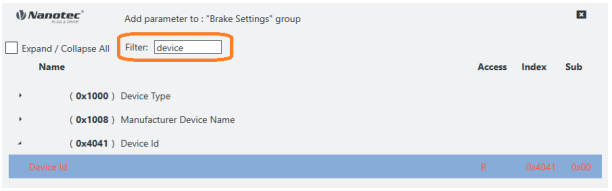
**Special controls** Feature operators

**Complex controls** Multi-level operators

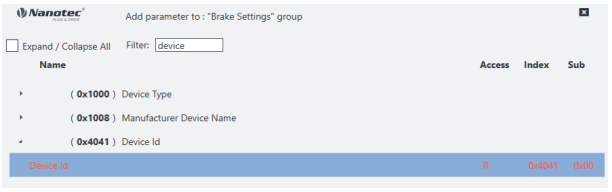
**Controls group > Parameter**



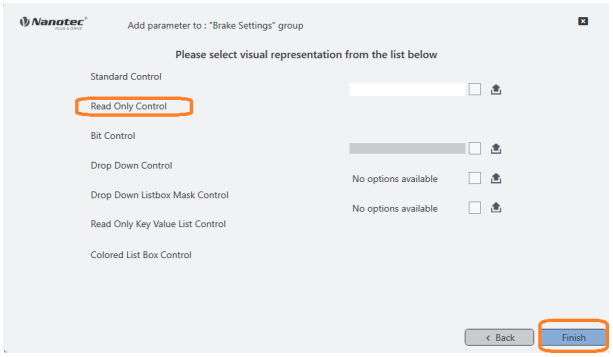

1. To add a parameter: Co-click the controls group and click **Add Parameter**.




2. In the pop-up: Enter *device* or *0x4041* to filter for the **Device Id** object.

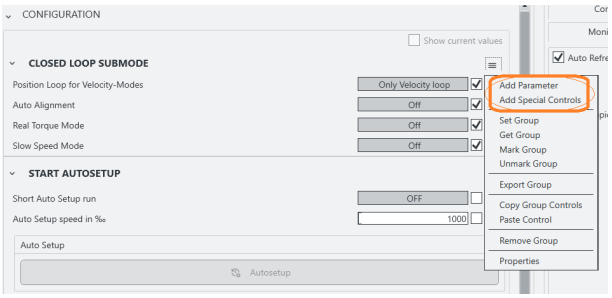



3. You may expand objects by mouse (or tick at **Expand all**).
4. Click **Device Id** and **Next** (if wrong: step **Back**).

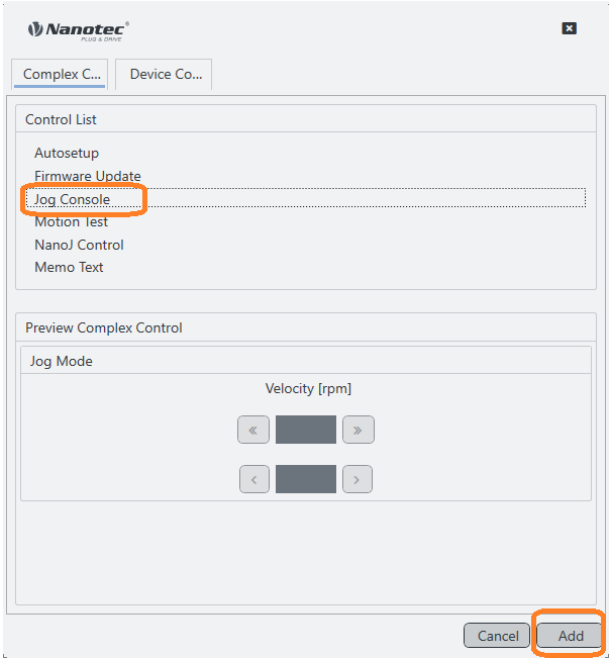



5. In the next pop-up: Select the visual reprentation and click **Finish**.

## Controls group > Special controls



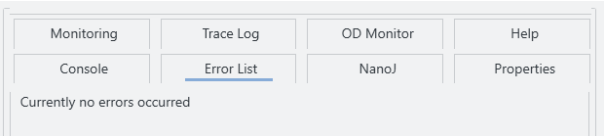

1. Co-klick a controls goup to open its context menu.
2. Select **Add Special Controls** to open the **Complex controls** list.



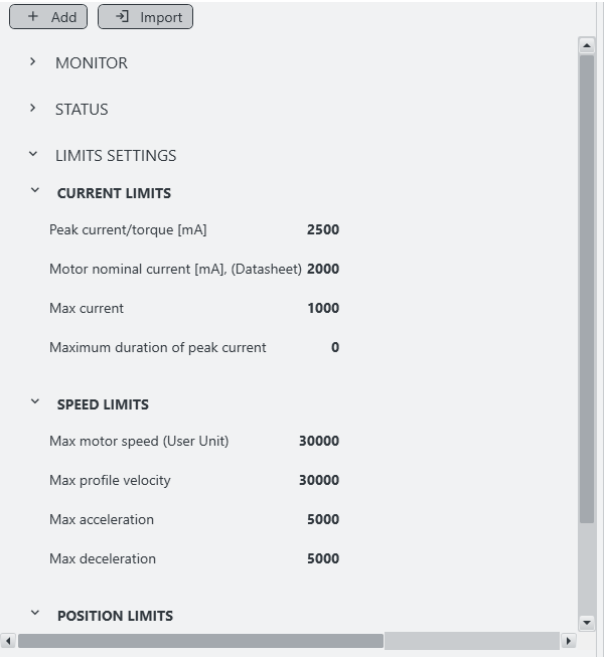

3. In the pop-up: Select the needed item.
4. To confirm: Click **Add**.

## 6.4 Display wall (2)

The display wall contains the monitors, error list, current OD values, properties, help and console.




Several tabs facilitate navigation in the display wall, in the user screen's upper right.




In the **Monitoring** tab, you combine either single or grouped monitors to track individual system behavior in real time.

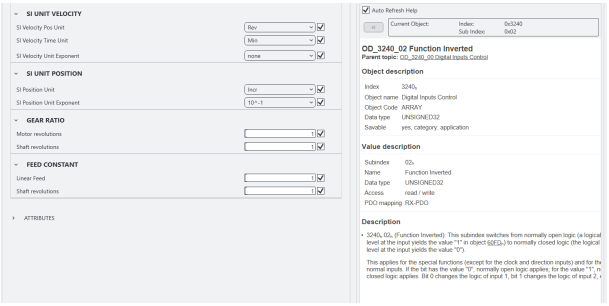
1. To set up a monitor: Co-click the tab **Monitoring**.
2. In the context menu: Either create a new monitor via **Add Monitor Module**.
3. Or fetch an existent one via **Import Module**.

The bottom part of the monitor display contains always the system monitor which shows the current device and connection status information.

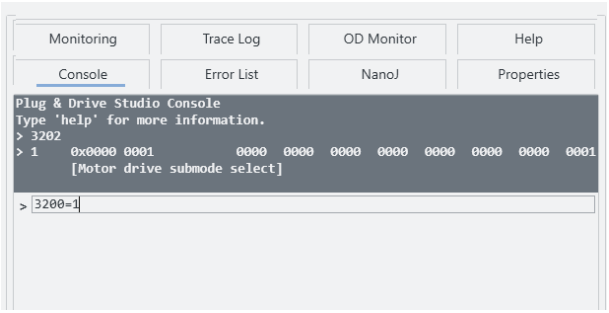

**OD Monitor:** Lists all objects from the controller's dictionary, together with their current values. For updates: Click **Read**.

To save the list as a text file on the hard drive: Click **Dump**. Keep the text file with current values ready in case of support enquiries.

Ypu can import an already saved dump file, compare it with the current OD and save it on another controller.



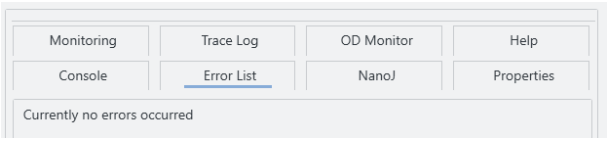

**Help:** Displays the description of the currently chosen element (OD object).




**Console:** Use this to quickly read/write.from/to the device's object dictionary.

Type <od index>:<od subindex> for read.

Type <od index>:<od subindex>=<value> for write.

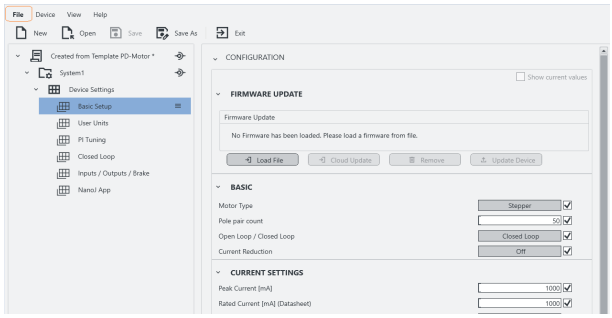



**Error List:** Here you can read the actual errors.

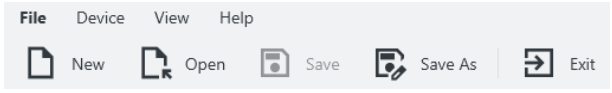
# 7 Project setup

In a project, you manage your devices, settings, connections, etc. **Note:** Ex works, in the software's templates folder, there is a sample project each for an external andan integrated controller. Nanotec recommends using these templates.

### Load / Create a project



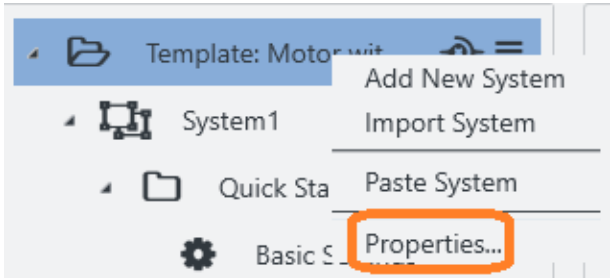
1. In the user screen: Visit the file menu (1)



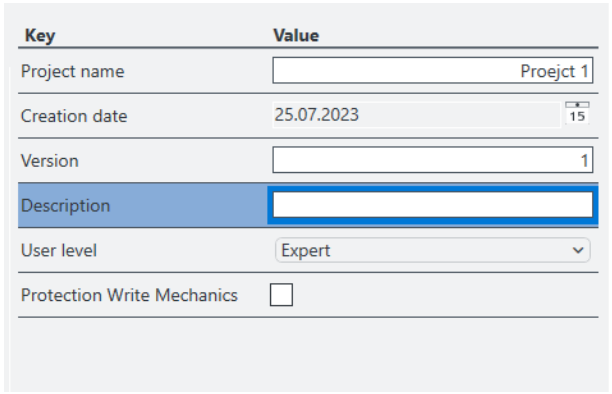

2. Preferably use **Project > Open** to select an existent sample project for template.
3. Or, for a new one instead: Select **Project > New**.

4. If a pop-up wants to store the current project: Click **Yes**.
   - **No** will close the project unstored and without backup.
   - **Cancel** will just close the pop-up.
5. The newly loaded selection appears in the project bar (4)

### If needed: Name the project

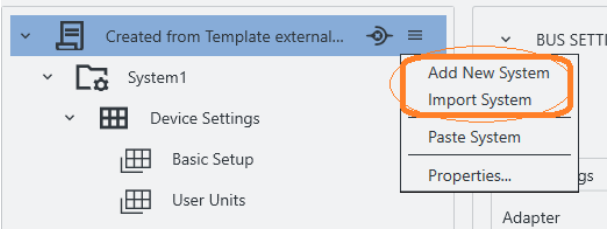



1. Go to the Project bar (4).
2. Co-click the current project and **Properties**.

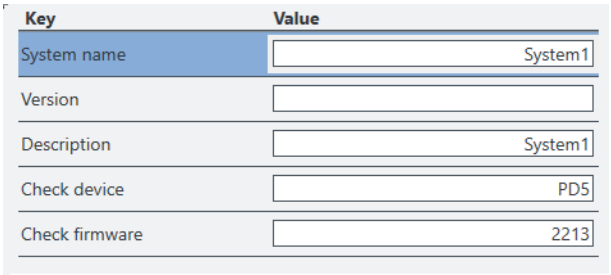



3. In the **Properties** window: Name, versionize, and describe the project. **Note** the pull-down for granted user rights (here: *Expert*).
4. After last entry: Set a tab stop (so that all is stored).

## Load / Create a system



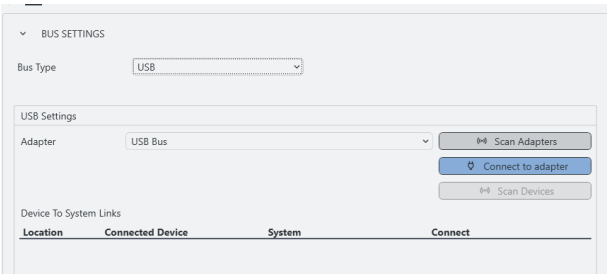

1. In the project: Preferably use **Import System** to select an existent sample system for template.
2. Or, for a new one instead: Select **Add new system**.




3. In the **Properties** window: Name, versionize, describe the system as needed.
4. After last entry: Set a tab stop (so that all is stored).
5. Repeat for each additional system.

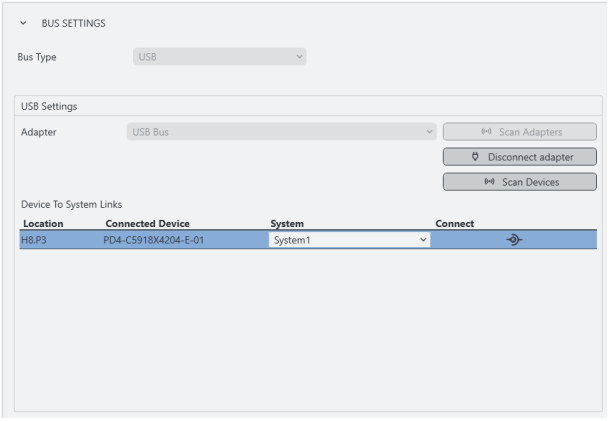
## Connect to adapter

1. In the project bar (4): Select your project.

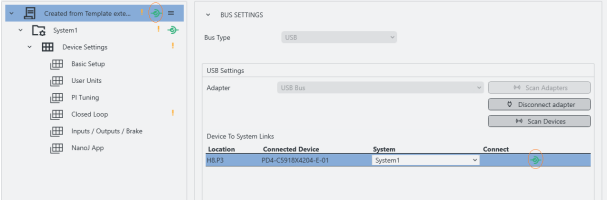



2. In the work desk (3): Open the **Bus settings** tab.
3. In the **Bus settings** tab: Select the **Bus type**.
4. Check setup by **Scan adapters**. If no result: Set up an adapter and check again.
5. Select the needed adapter.

## Connect to device



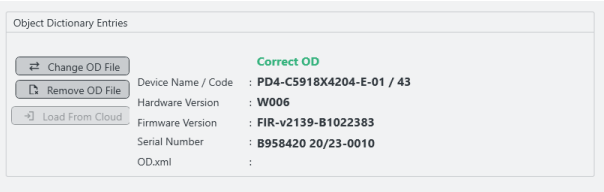

1. In the **Bus settings** tab: With the adapter linked, you can see all available devices.
2. Click **Scan devices**. Check **Connected device**.
3. By Pulldown: Select a **system** to link your device to.



4. You can link / unlink the device via **Connect** icon (here: green).

### Select the OD file

PNDS3 shows objects that match the controller firmware with correct OD file only (object dictionary). If the system is linked, a **Object Dictionary Entries** tab shows if the correct OD file is loaded. Otherwise, the generic file *Common OD* loads, by which you reach available objects of all Nanotec products.




1. Select the system.
2. Open the **Object Dictionary Entries** tab.

- *Common OD*: Reloadable via **Remove OD File**
- OD file of choice: Loadable via **Change OD File**
- Firmware-correct OD files for all Nanotec controllers: In the **Firmware** package on the PNDS3 website or via **Load From Cloud** (internet access necessary).

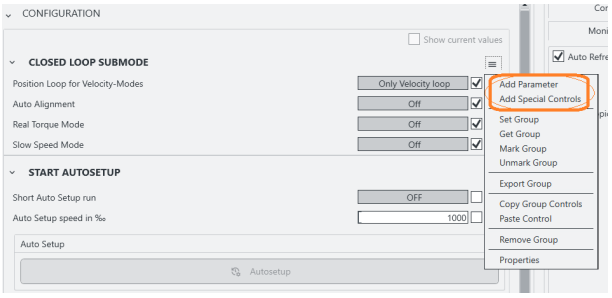
Wrong OD files report an error (= red).

# 8 Special controls

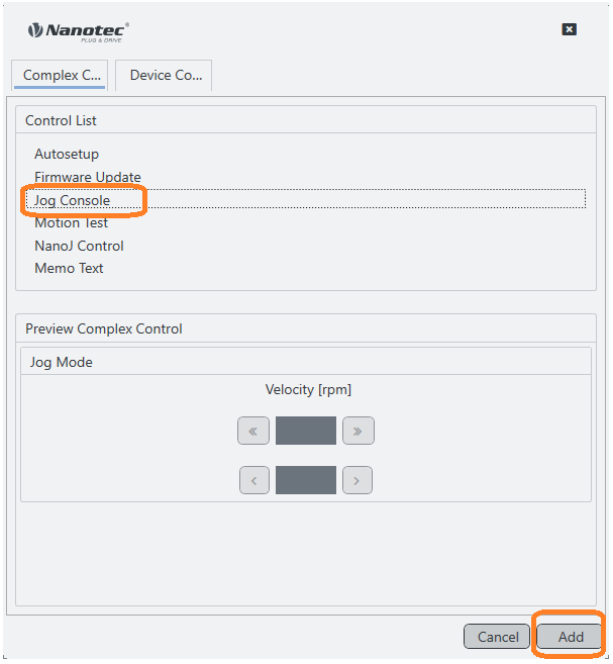
Via **Special controls**, you add **Complex controls** and **Device communication settings** to the user interface. Both help you to use advanced controller functions.

### Basic principle

Special controls define and monitor (as macro collections) the system behavior. Depending on assembly, you can check their connections and attributes in the work desk.




1. To set up complex controls or device communication: Co-click the controls group.
2. In the context menu: Select **Add Special Controls**.




3. **Complex controls**? **Device communication settings**? Open the tab of choice.
4. Select the needed item and **Add**.

→ The control / setting of choice appears in the work desk.

## 8.1 Complex controls

With the **Complex controls** macro collection, you create your own controller functions. Next to **Autosetup** and **Firmware update**, these include **Jog Console, Motion Test, NanoJ Control** and **Memo Text**.

### Autosetup

**Autosetup** detects the motor type and connected sensors (encoder / Hall sensors).

**CAUTION!**

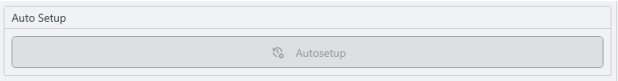
**Injury: from abrupt motor travel after auto-setup (= parameter loss)!**

- ► For motors with integrated controllers: Avert auto-setup (since it comes factory-run already).
- ► Otherwise: Restart the motor after auto-setup (homing alone won't suffice).
- ► Stay clear of moving motor parts.
- ► Touch the motor at standstill only.

**NOTICE**

**Motor malfunction: from auto-setup user error!**

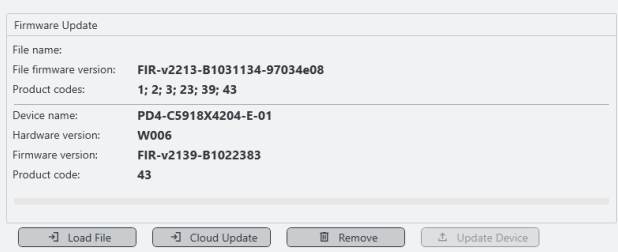
- ► Close possible NanoJ programs (object $2300_h$:$00_h$ Bit 0 = "0"; cf. 2300h NanoJ Control).
- ► Keep the motor load-free, and freely rotable in any direction.
- ► **Don't** touch the motor.




As long as the motor on the controller or the feedback sensors (encoder / Hall) remain the same: Run **Autosetup** only once, on initial commissioning.

### Firmware update

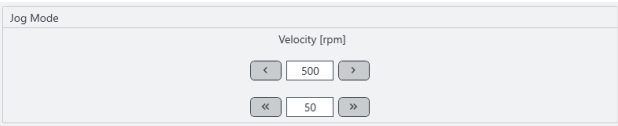
Nanotec recommends controller firmware *FIR-v2213* or newer. Please find the current version in the **Firmware** folder on the PNDS3-website or via **Load From Cloud** (see Select the OD file internet access necessary).




1. Open or add the**Firmware update** control.
2. Click **Load from file**.
3. Select a firmware file and click **Open**.
4. PNDS3 checks via product code if the chosen file fits to the product.
5. Click **Update device**.
6. Firmware updates itself.

**Note:** The chosen firmware file will be stored as part of the project the next time the latter is stored. If you don't want this to happen, click **Remove from project** before.

### Jog Console

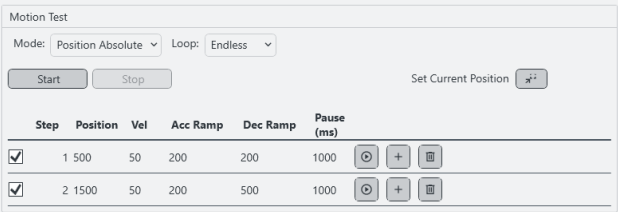



Via **Jog Console**, you test the motor in velocity mode. You can select two target speeds. The motor runs as long as you use the mouse to press the button for left / right rotation.

**NOTICE**

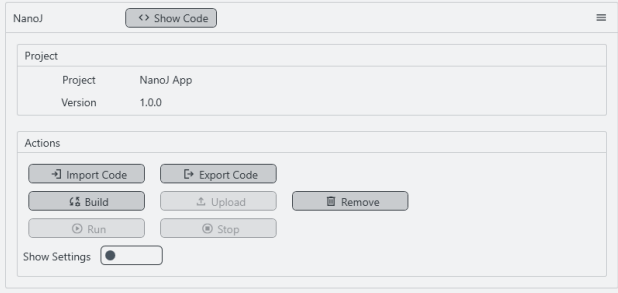
The console may be deactivated, in case the controller is cu

### Motion Test




In **Motion Test**, you test the motor in position / velocity / torque mode. Your options include target values, acceleration / deceleration ramps, repetition cycles, test run duration etc.

### NanoJ Control




In **NanoJ control**, you create a NanoJ new project (= **New**) or **Import** an existing one. The button **Build** compiles the project.

**Note:** The next time you store the project, the selected NanoJ file merges into the project. If you don't wish this to happen, click **Remove** before.

### Memo Text

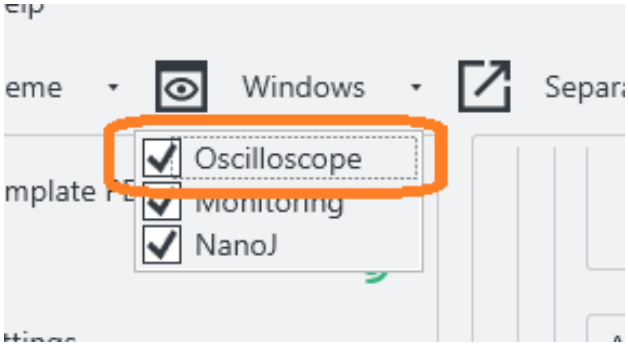
Adds a freely editable text box.

## 8.2 Device communication

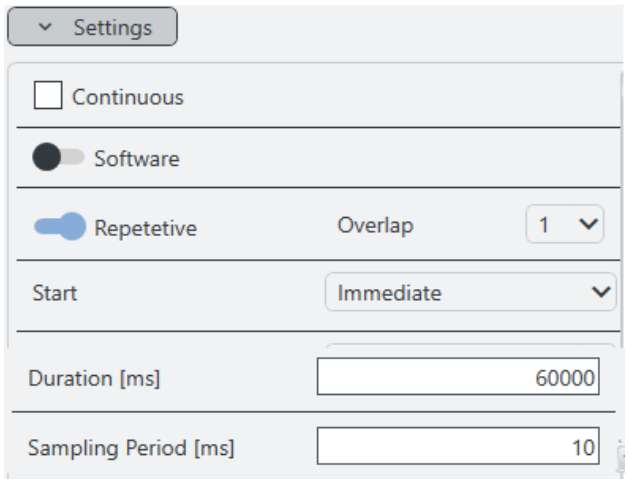
With these controls, you parametrize the device communication. **Note:** Coding switches for setting the communication parameters overwrite the software settings on some devices. For details: Follow valid OEM instructions.

# 9 Oscilloscope

Via **Oscilloscope**, you monitor and control in real time the current value of device parameters from the object dictionary, say, for recording.



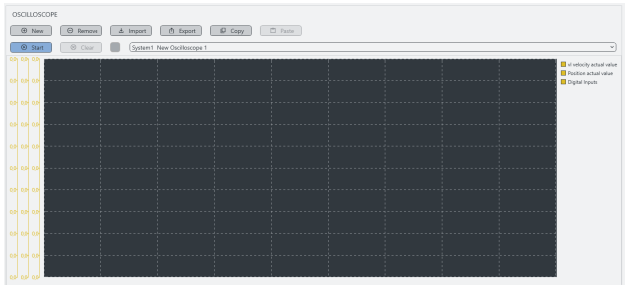

To open the oscillosope, go to the **Main Menu > Windows** and select it.




Under **Settings** you can configure the following:

- **Continuous:** If chosen, the oscilloscope starts imeediately and runs continuously, until the buffer is full (if **On-Device**) or after the set **Duration** expires (if **Software** is selected). If not selected you can further define the conditions for start/stop.
- **Start: Immediate**, **Condition** (**On-Device**, as soon as a parameter of choice changes), or **Motion test** (**Software**, Motion test triggers the scope).
- **Stop: Duration** (of recording) or manual.

In the right bottom corner you can add channels by selecting from the object dictionary or remove them.



For recording, you open a new (or import an existing) oscilloscope in the tab above the settings and click **Start**.

# 10 Imprint, versions



**Nanotec Electronic GmbH & Co. KG** │ Kapellenstraße 6 │ 85622 Feldkirchen │ Germany

Tel. +49 (0)89 900 686-0 │ Fax +49 (0)89 900 686-50 │ info@nanotec.de │ www.nanotec.com

| Document | Changes | PNDS3 |
|---|---|---|
| 1.0.0 (06/2022) | Edition | V1.3.0 |
| 1.0.1 (11/2022) | New software version V1.4.0, new firmware FIR-v2213 | V1.4.0 |
| 1.1.0 (11/2023) | New software version V1.5.2 | V1.5.2 |
| 1.2.0 (04/2024) | New software version V1.6.0 | V1.6.0 |
| 1.3.0 (10/2024) | New software version V3.1.7 (1.7.0 according to the old versioning).<br>Modbus TCP support added.<br>Firmware and OD file can now be directly downloaded from cloud, see Select the OD file. | V3.1.7 |
| 1.3.1 (04/2025) | New software version V3.1.8 | V3.1.8 |