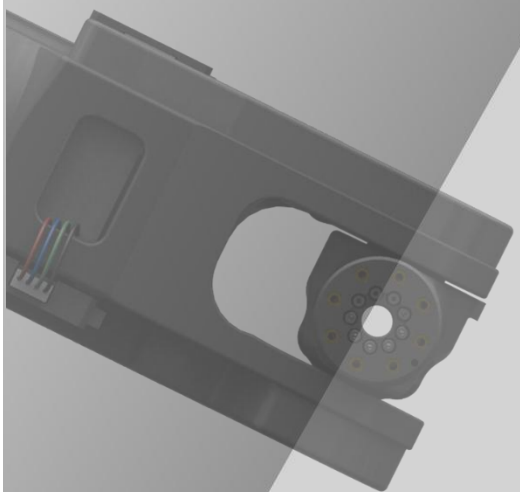
Kawasaki Robotics

Astorino

SimBox Operation Manual



Preface

This manual describes the handling of the 6-axis robot “astorino" simulator.

The ASTORINO is a learning robot specially developed for educational institutions. Pupils and students can use the ASTORINO to learn robot-assisted automation of industrial processes in practice.

1. The "astorino" software included with the ASTORINO is licensed for use with this robot only and may not be used, copied or distributed in any other environment.
2. Kawasaki shall not be liable for any accidents, damages, and/or problems caused by improper use of the ASTORINO robot.
3. Kawasaki reserves the right to change, revise, or update this manual without prior notice.
4. This manual may not be reprinted or copied in whole or in part without prior written permission from Kawasaki.
5. Keep this manual in a safe place and within easy reach so that it can be used at any time. If the manual is lost or seriously damaged, contact Kawasaki.

Symbols

Items that require special attention in this manual are marked with the following symbols.

Ensure proper operation of the robot and prevent injury or property damage by following the safety instructions in the boxes with these symbols.

⚠ WARNING

Failure to observe the specified contents could possibly result in injury or, in the worst case, death.

[**ATTENTION**]

Identifies precautions regarding robot specifications, handling, teaching, operation, and maintenance.

⚠ WARNING

1. **The accuracy and effectiveness of the diagrams, procedures and explanations in this manual cannot be confirmed with absolute certainty. Should any unexplained problems occur, contact Kawasaki Robotics GmbH at the above address.**
2. **To ensure that all work is performed safely, read and understand this manual. In addition, refer to all applicable laws, regulations, and related materials, as well as the safety statements described in each chapter. Prepare appropriate safety measures and procedures for actual work.**

Paraphrases

The following formatting rules are used in this manual:

- For a particular keystroke, the respective key is enclosed in angle brackets, e.g. <F1> or <Enter>.
- For the button of a dialog box or the toolbar, the button name is enclosed in square brackets, e.g. [Ok] or [Reset].
- Selectable fields are marked with a square box ☐.
  If selected a check mark is shown inside the symbol ☑.

List of contents

1 Nomenclature in this manual

The author of the manual tries to use generally valid terminology while achieving the greatest possible logical sense. Unfortunately, it must be noted that the terminology is reversed depending on the point of view when considering one and the same topic. Also it is to be stated that in the course of the computer and software history terminologies developed in different way. One will find therefore in a modern manual no terminologies, which always satisfy 100% each expert opinion.

2 Overview of ASTORINO

The ASTORINO is a 6-axis learning robot developed specifically for educational institutions such as schools and universities. The robot design is based to be 3D printed with PET-G filament. Damaged parts can be reproduced by the user using a compatible 3D printer.

Programming and control of the robot is done by the "astorino" software.

The latest software version and 3D files can be downloaded from the KAWASAKI ROBOTICS FTP server:

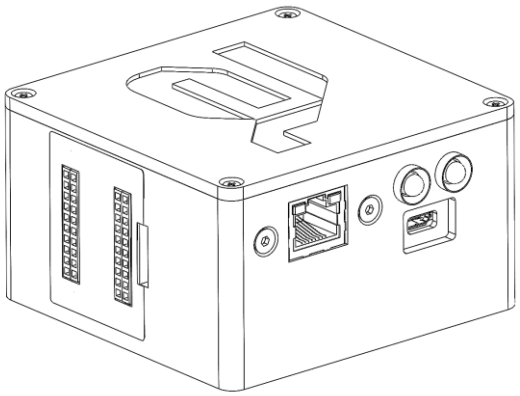
https://ftp.kawasakirobot.de/Software/Astorino/

Just like Kawasaki’s industrial Robots the ASTORINO is programmed using AS language. Providing transferable programming skills from the classroom to real industrial applications.

3 Technical specifications

Characteristics		SimBox
Working environment	Temperature	15–35°C
	Humidity	35–60%
Controller		Teensy 4.1
Inputs/Outputs		8/8 (PNP 8 mA, NPN 15 mA)
Max. current consumption		400 mA
Power supply		5V – microUSB
Weight		200 g
Material		PET-G
Colour		Black
Communication with a PC		USB/Ethernet
Communication protocols		MODBUS TCP, TCP/IP, UDP
Options	Module I/O 24V	8 × inputs/outputs
	7^{th} axis	Linear track – software option
	Adapter IO 3.3V	Yes

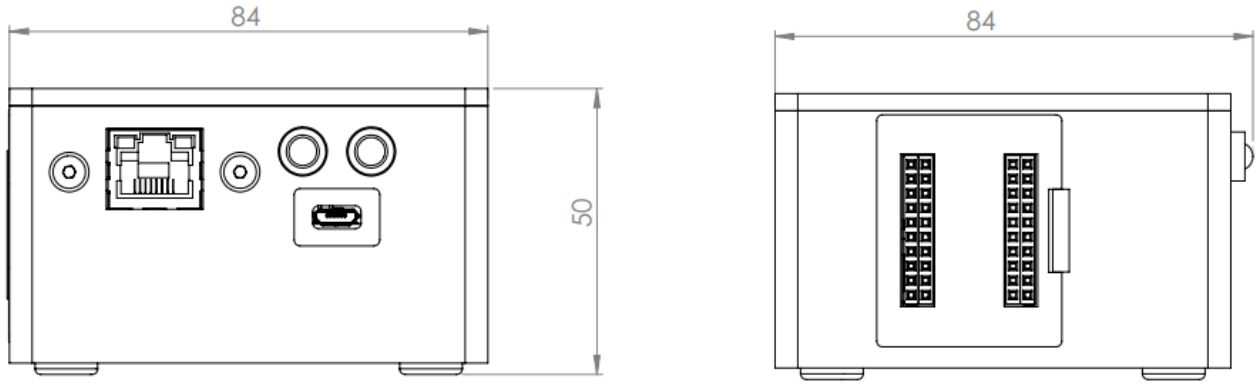
4 SimBox package contents



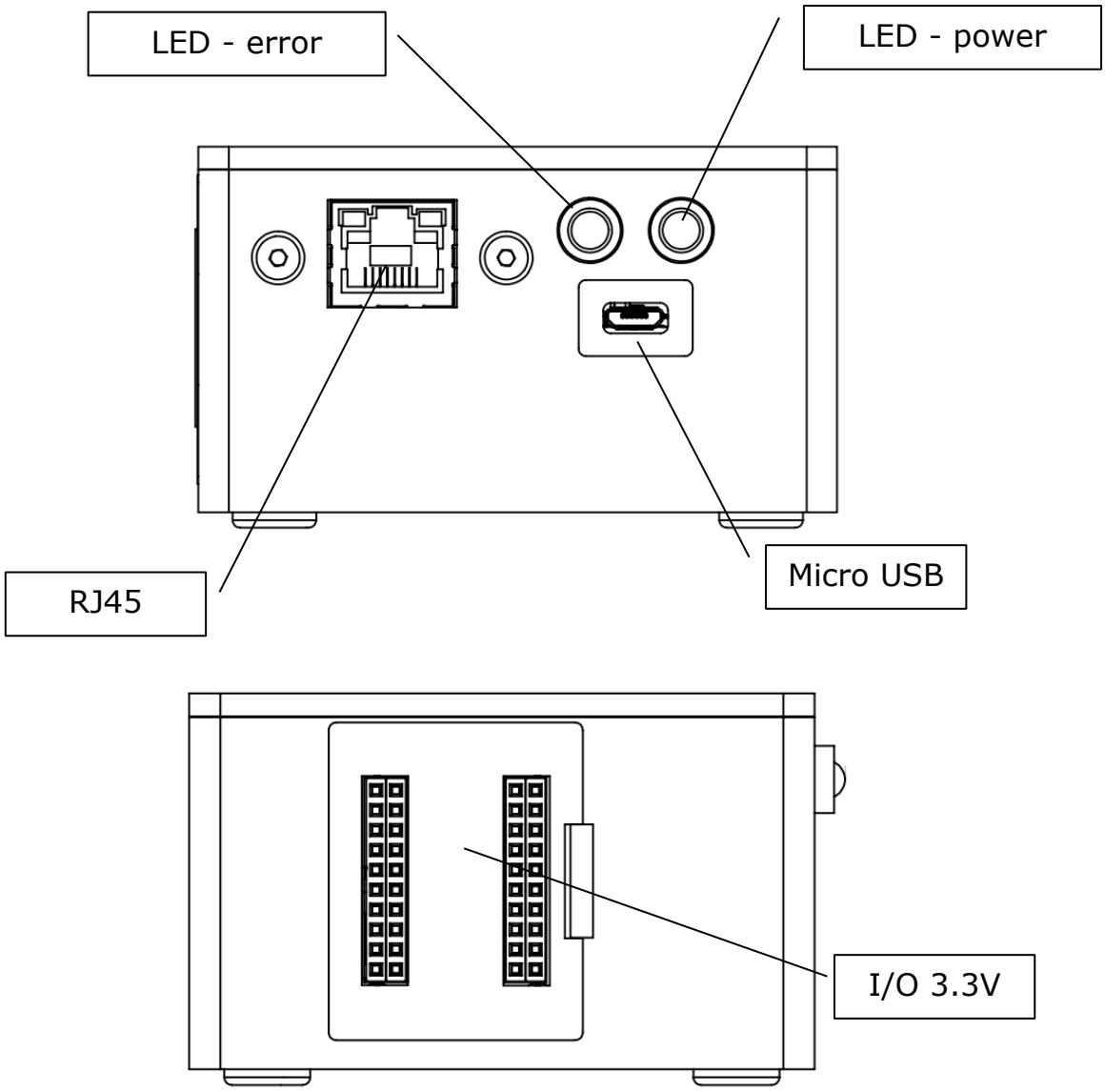
SimBox

Micro USB cable

5 Dimensions



6 Connectors

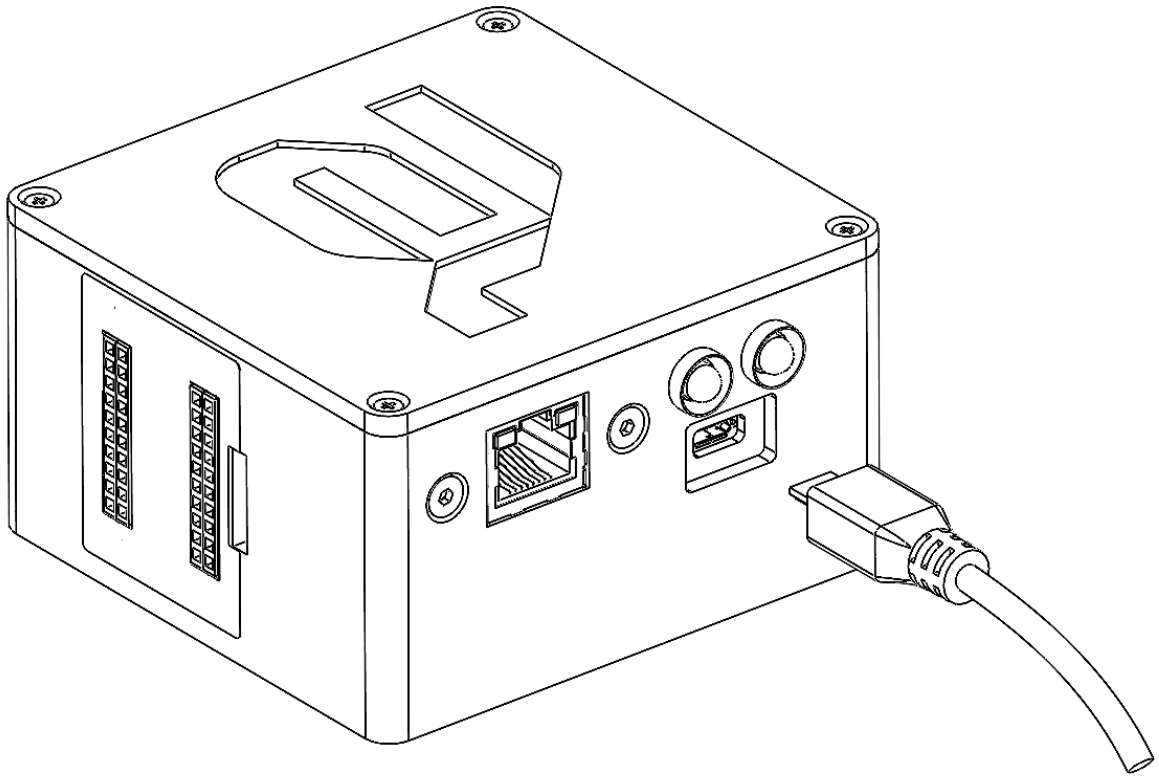


7 Unboxing and starting-up

Once the SimBox is removed from the packaging, place it on a solid surface.

7.1 Connection

- Connect the USB cable to your computer or to a 5V USB power adapter if you are using Ethernet only.



7.2 System Requirements

Before installing astorino software, ensure that the computer meets the following hardware and software requirements.

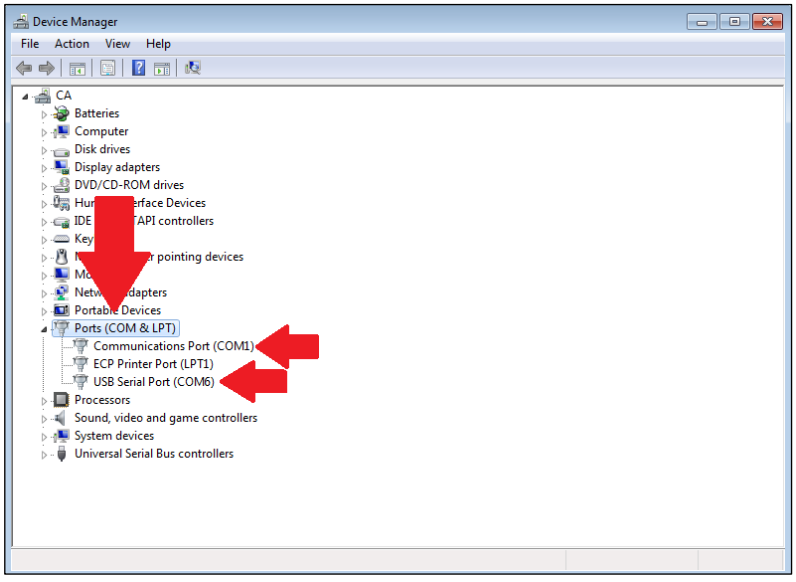
Part	Requirements
CPU	2.0 Ghz or faster processor
Memory	4 GB minimum
Disk	100 MB free space
Graphics card	Any
Display settings	1280 x 720 pixels minimum resolution, 100 % display scaling recomended
Mouse	Three-button mouse

System	Version
Windows	7, 8, 8.1, 10, 11

7.3 Driver installation

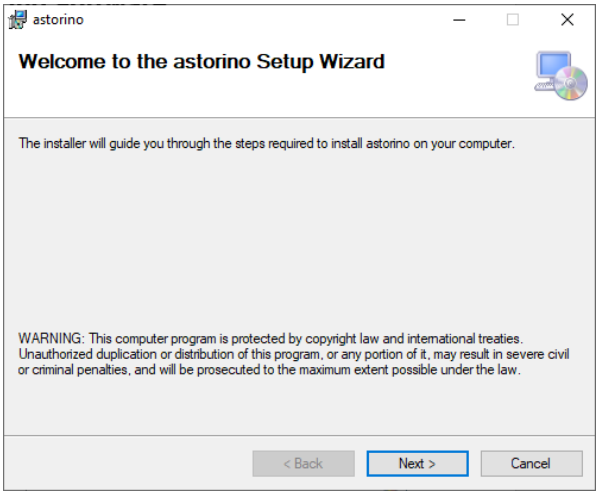
The required drivers install automatically since Windows 8. After successful installation, the robot will appear in the Device Manager as <USB Controller>. If using Windows 7 install the drivers before connecting the robot to the PC (downloaded from Kawasaki FTP server or from USB stick).

Open device manager via <**Windows + R**> ⇨ devmgmt.msc
or by clicking the icon in the selection menu via <**Windows + X**>.

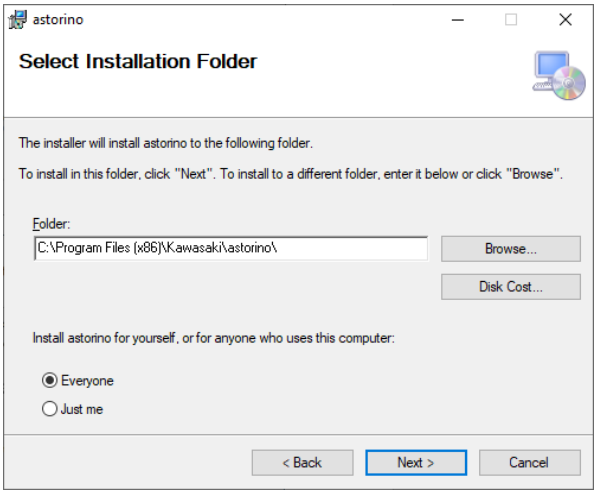


7.4 Installing the astorino Software

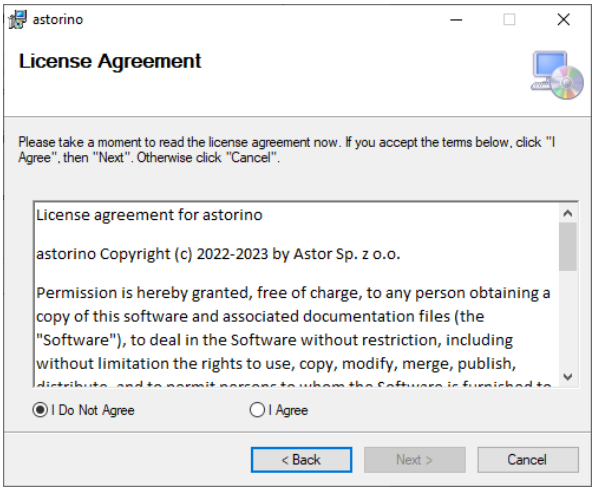
Run astorino_x.x.x.exe



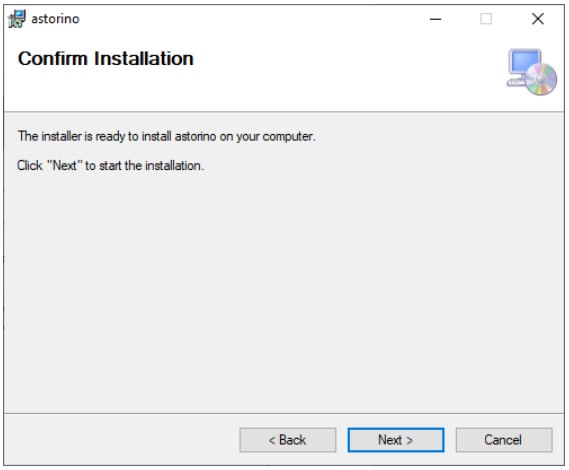
Confirm or specify installation directory



Read and accept the license agreement

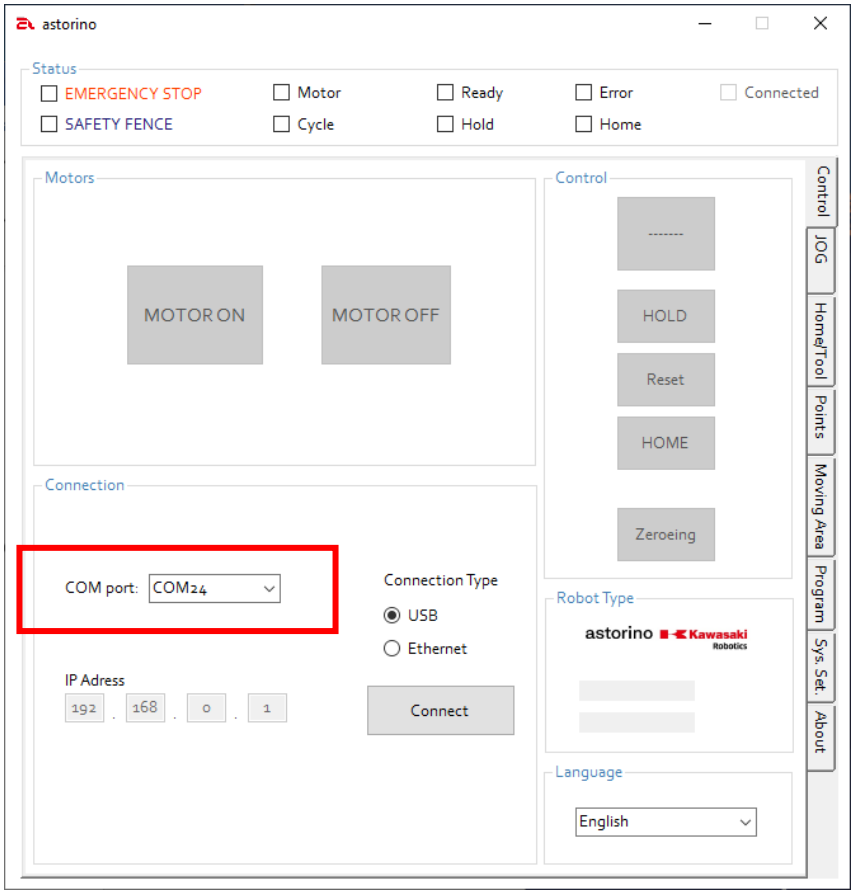


Start the installation



7.5 Making the astorino SimBox ready for operation

- Open the astorino software.
- The COM port to which the robot is connected will automatically appear in the drop-down list in the [Control]-menu Connection area.



- Once the controller is properly connected, it is ready to work, SimBox automatically switches on the drives and resets the robot.

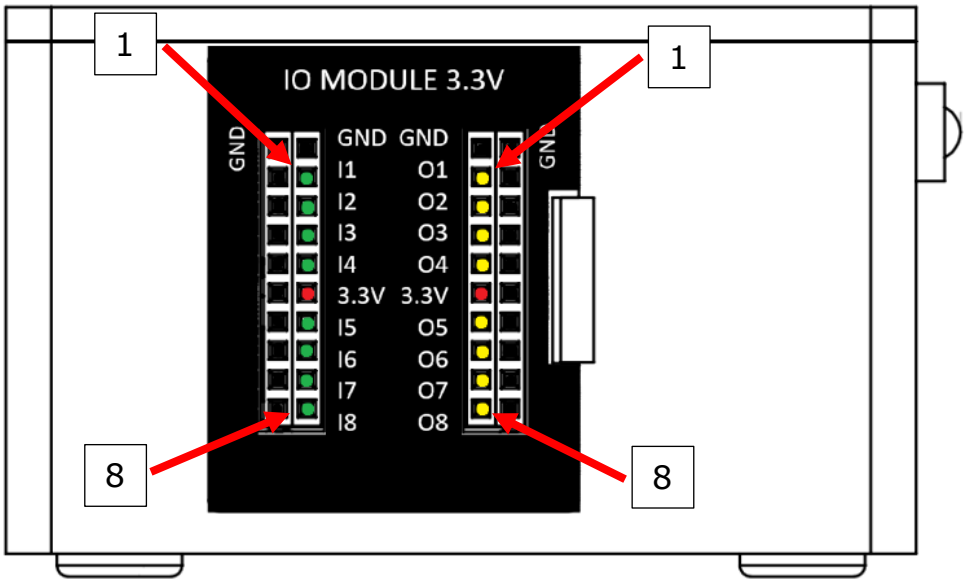
8 Operation

In order to learn how to use the Simbox (Astorino robot), you need to use the knowledge from the Astorino robot manual, as well as the AS language instructions.

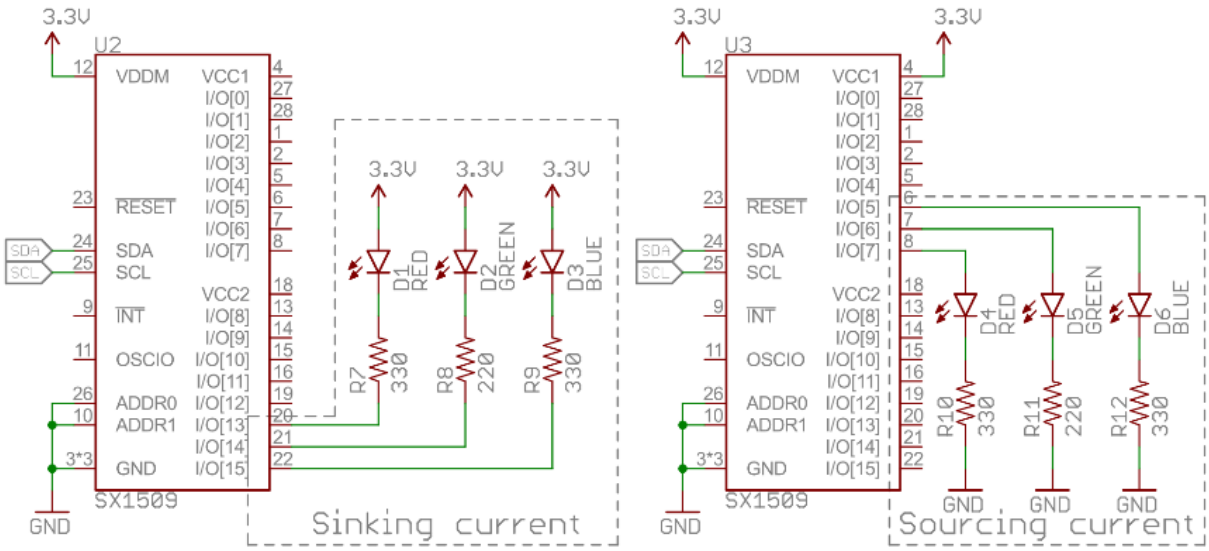
9 Inputs/Outputs – 3,3V

SimBox has 8 inputs and 8 outputs based on 3.3V DC.

Colour	Function
	Output
	Input
	3,3 V/DC
	0V (GND)



The system normally operates in PNP switching mode (sourcing current). PNP means positive switching (mainly used in Europe and North America). A module therefore switches positive potential to its output.



The operation can be changed to NPN by using the following commands in the terminal:

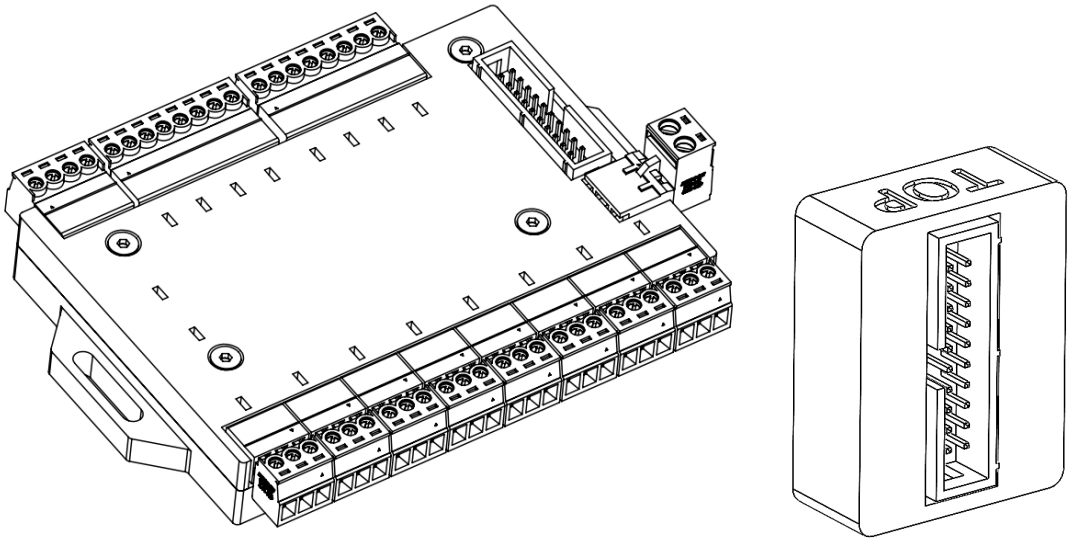
- Z_OUTSOURCE 1 – SOURCE OUTPUT
- Z_OUTSOURCE 0 – SINK OUTPUT
- Z_INPULL 1 - activates pulling the inputs to 3,3V
- Z_INPULL 0 - deactivates pulling the inputs to 3,3V

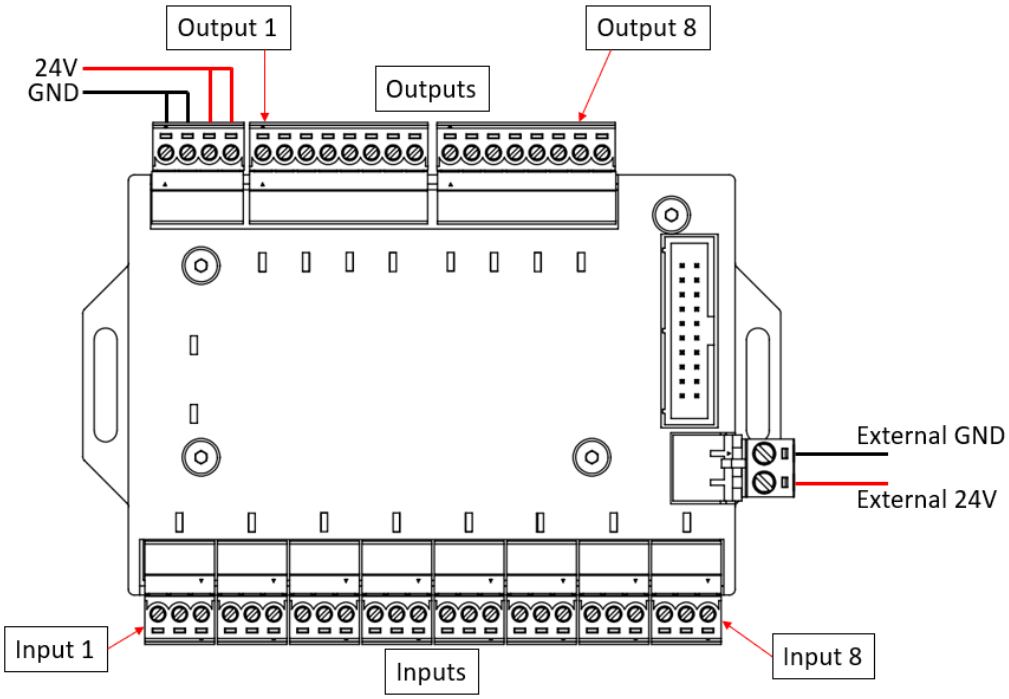
WARNING!

Each OUTPUT provides 8mA of current. Please do not exceed the limit as this may damage the motherboard.

10 24V I/O-Module

The 24V I/O module is available as an option and is sold independently of the astorino robot or the astorino robot Kit. Both inputs and outputs work in PNP mode.

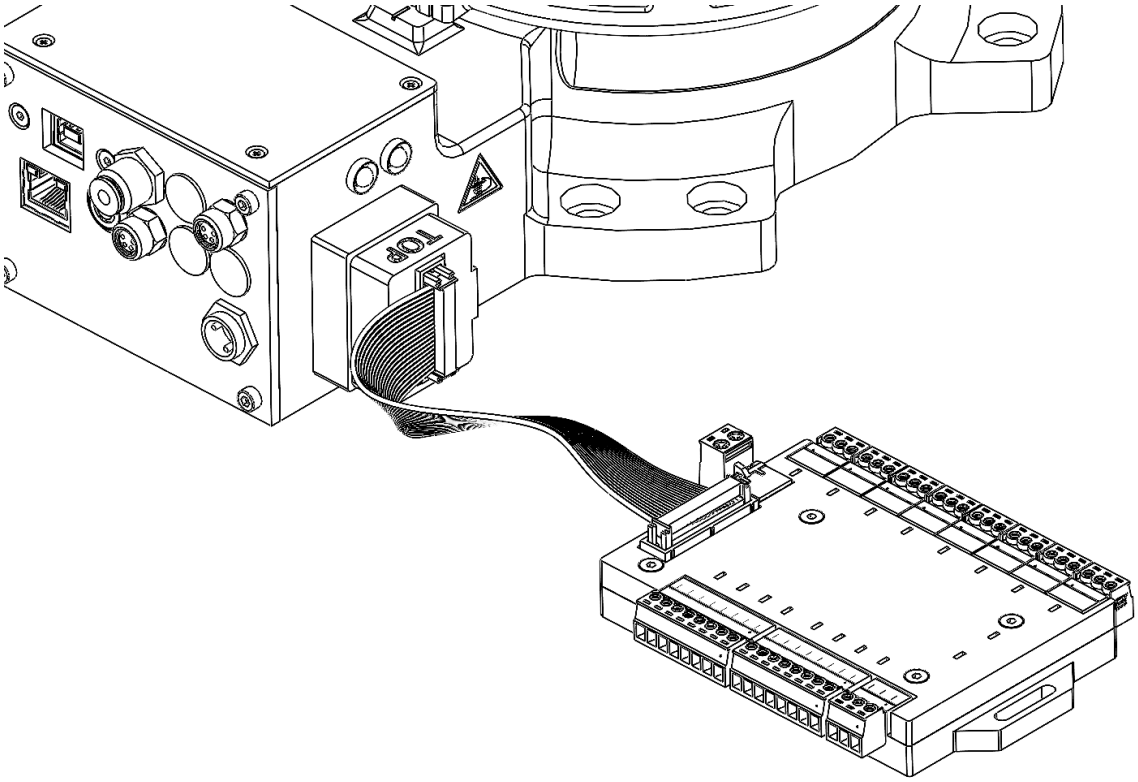




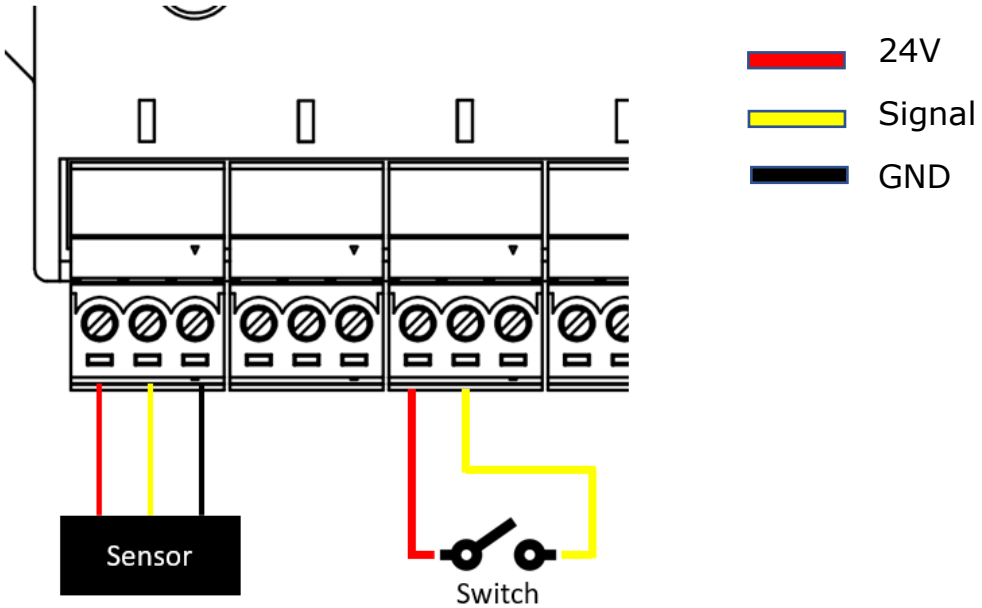
A total of 8 24V inputs and 8 24V outputs can be connected.
Each output provides 300mA current (approx. 7.2W output power).

10.1 Connection to the robot

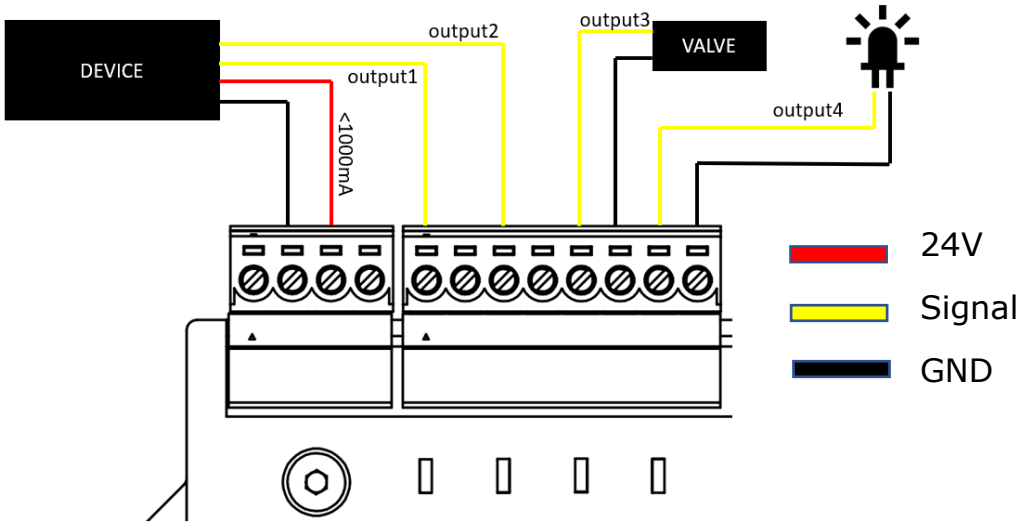
The module is connected via the supplied 30cm ribbon cable.



10.2 Connecting inputs

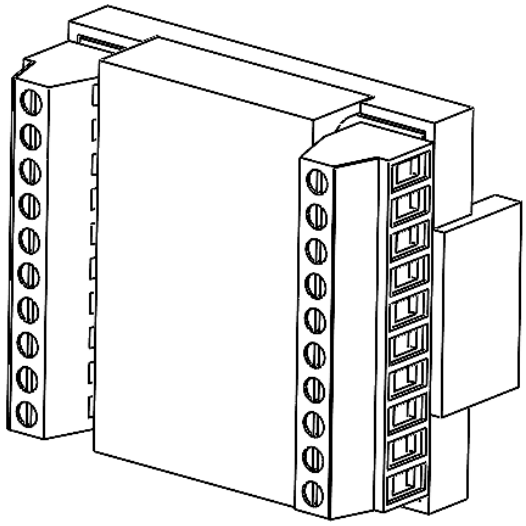


10.3 Connecting outputs

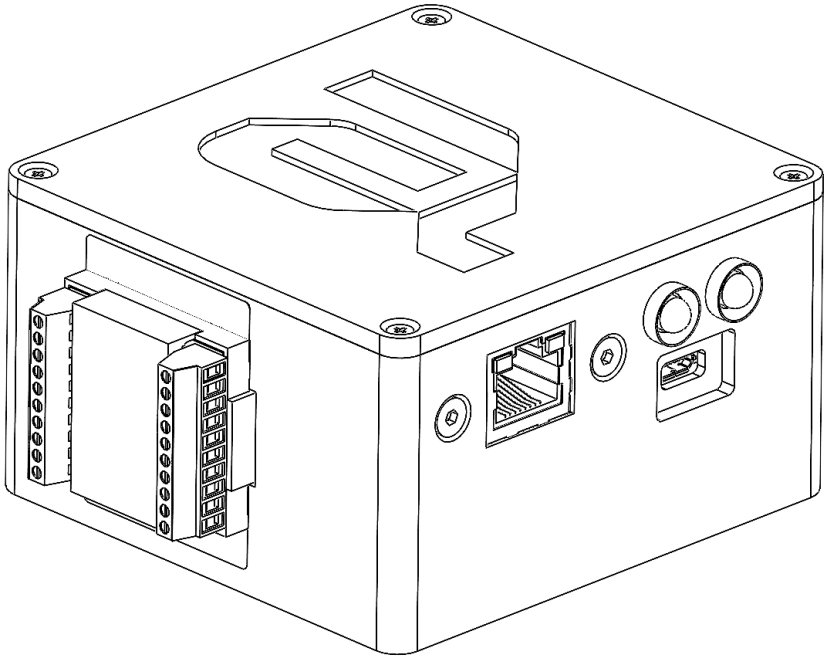


11 Adapter IO 3.3V

You can also connect a 3.3V IO adapter to the SimBox, thanks to the adapter we have easy access to 3.3V I/O through screw connectors



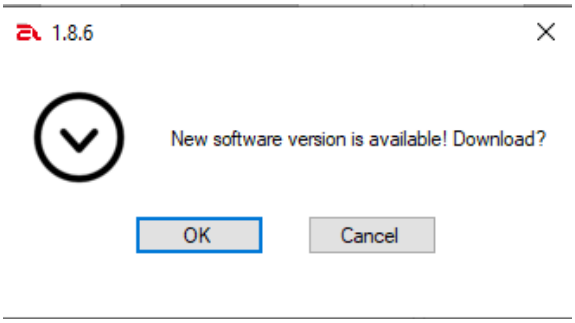
11.1 Connection to a SimBox



11.2 Firmware Update

11.2.1 Basic information

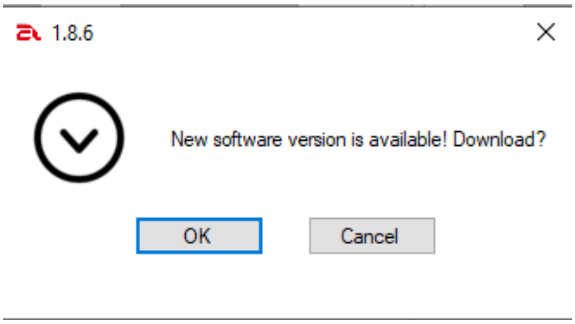
Astorino software after startup will automatically check if there is a new version available and if so, informs user.



Clicking [OK] button will download the new version to user specified location on the hard drive.

Then user needs to uninstall the old version from a PC and install a new one.

After coonecting to the robot astorino software will check if the firmware on the robot is up to date. If the new firmware is available, apllication will inform user about it.



Clicking [OK] button will download the new to user specified location on the hard drive.

The latest firmware version can also be downloaded from KAWASAKI ROBOTICS FTP server: https://ftp.kawasakirobot.de/Software/Astorino/

or contact technical support:
Tech-Support@kawasakirobot.de

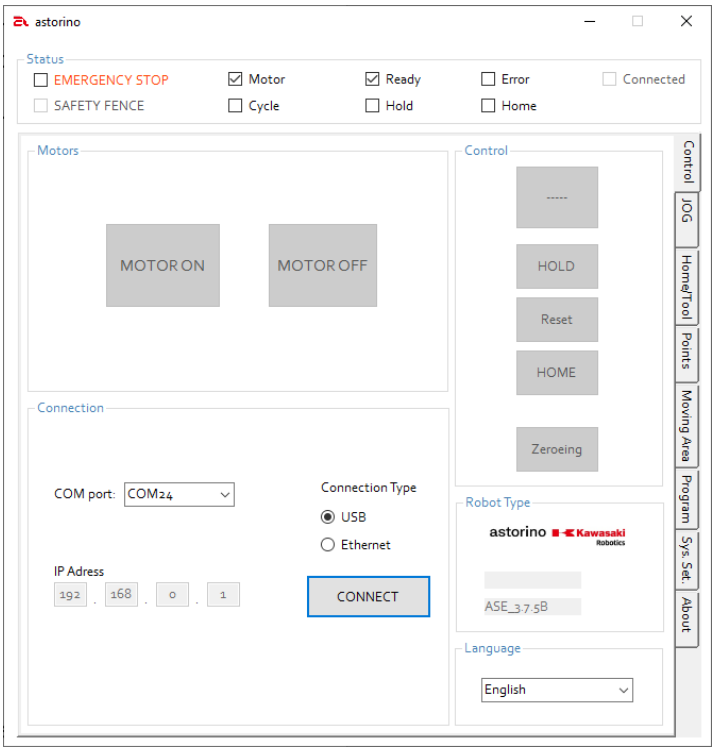
11.2.2 Update procedure

To update the firmware, start the astorino software.

[**ATTENTION**]

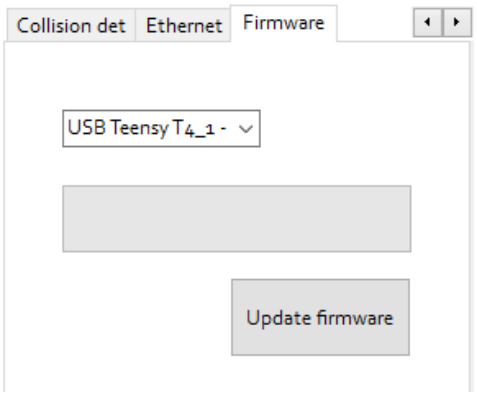
Make sure that the software is **not** connected to the robot. The motors must be switched off!

Interrupting the process might damage the CPU, please do not turn off the robot during update procedure!



Navigate to the [System Setting] Tab and to the IO configuration area.

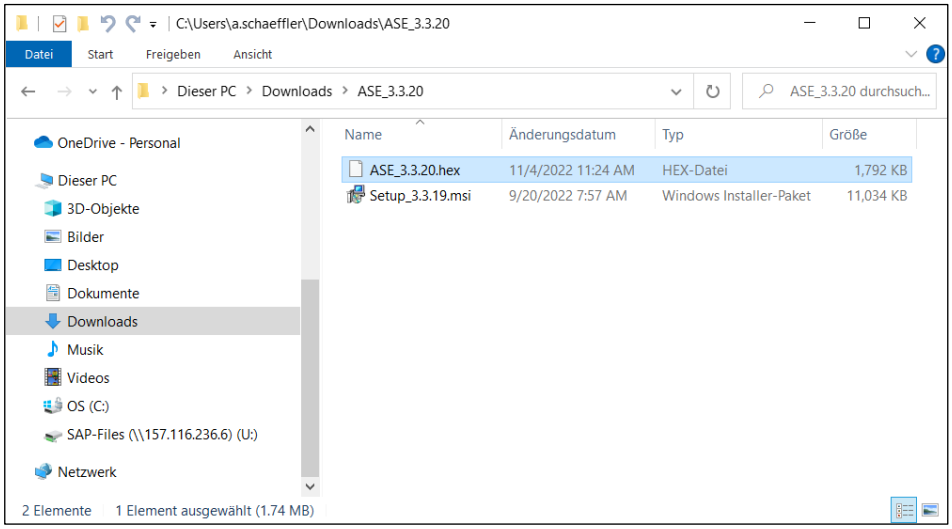
Click on the right arrow symbol until the [Firmware] subtab is visible.



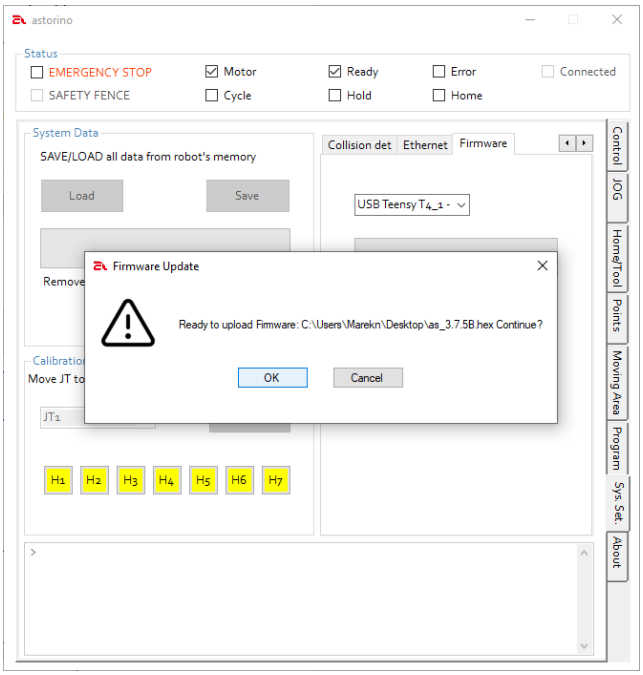
Press [Update Firmware] to open the file selection window.

Select the *.hex file which contains the new firmware.

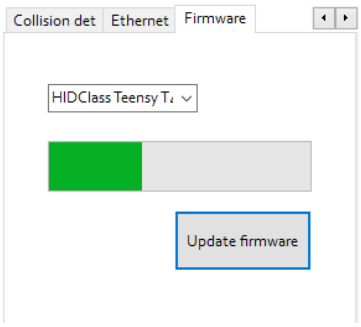
File selection window:



Confirm the upload to start loading the new firmware to the robot's memory:

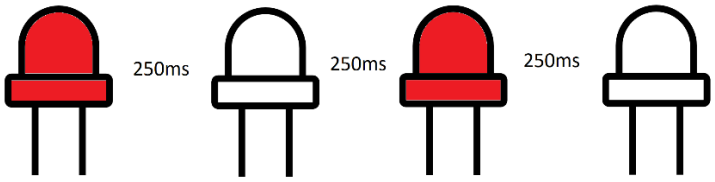


The firmware update is being performed.



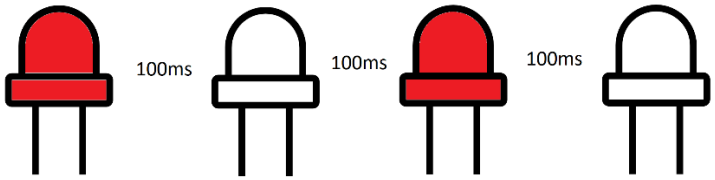
After the firmware is installed observe the red (Error) led on the robots base.

If the red led starts to flash slowly (around 2 times per second).

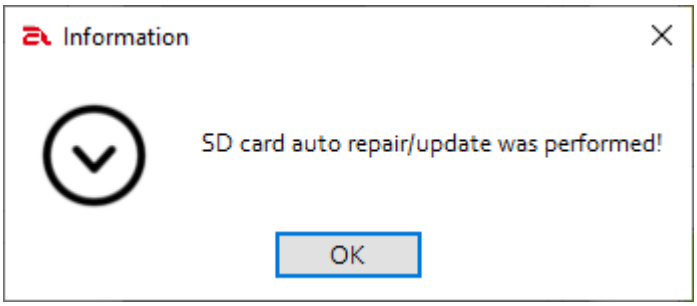


Turn off the robot and turn it on again. This is an error from the SD card inside robots base, CPU was not able to restart the card after firmware update. Resetting the power solves the problem.

If the red led starts to flash fast (around 5 times per second).

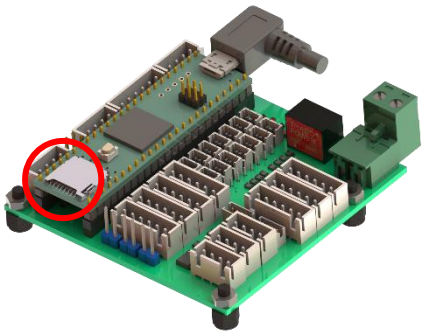


This means that the robot needs to update the data on the SD card. The procedure will perform automatically. When the procedure is complete, the red LED will turn off. And when connected to the computer, there will be a message that the SD card data repair/update procedure is complete

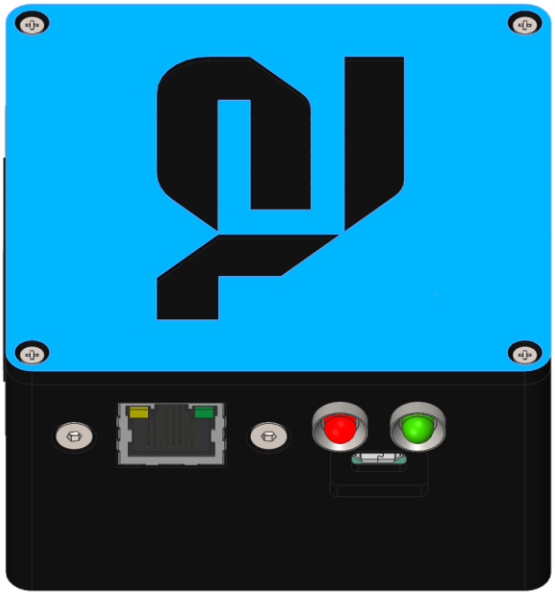


11.2.3 Update fail recovery

If the update process is interrupted then the CPU might not work and Windows will not detect the robot. Reset the CPU to the factory settings by pressing the white reset button on the CPU board for 13s to 17s.



The white button is located on the processor board inside the base of the robot. To access it, you need to unscrew and remove the back top cover.



Be very careful when doing this! Do not use any metal objects inside the SimBox while restoring the processor to factory settings!

12 Manufacturer information

For further questions, contact Kawasaki Robotics support.

Contact:

Kawasaki Robotics GmbH

tech-support@kawasakirobot.de

+49 (0) 2131 – 3426 – 1310

Kawasaki Robot
SimBox Operation Manual

2024-12: 2nd Edition

Publication: KAWASAKI Robotics GmbH

Appendix 1 – PET-G material

PETG is one of the most versatile 3D printing materials, strong and easy to print. Its popularity has increased over the last years as an alternative to PLA. PETG is the PET variant used in 3D printed. The G at the end means Glycol-modified. This change affects the chemical structure, making the material more transparent, less fragile and easier to process.

PETG has very interesting properties, and its closest competitors are PLA and ABS. The main properties you should consider are the following:

- **Rigidity:** Difficulty of the material to be deformed, including stretching and bending. PLA is more rigid than PETG, being PETG and ABS almost as rigid.
- **Resistance:** PETG is generally more difficult to break than PLA and ABS. Based on technical properties, PETG is not only more resistant than ABS, but the adhesion between layers is higher, giving an overall better resistance.
- **Heat resistance:** PETG softens at 80ºC, while PLA can start softening at 50ºC. However, ABS has the highest heat resistance, softening at 105ºC.
- **Odourless printing:** Unlike ABS, PETG does not produce an odour when printed.
- **Recyclable:** Due to its popularity, most cities have the required infrastructure to recycle PETG.

How to print PETG

Hotend temperature: PETG is usually printed at 220-250ºC, and it can be printed with almost any 3D printer, including all-metal hotends or those that use an inner PTFE tube.

- **Surface temperature:** In order to print PETG, it's necessary to use a heated bed at 60-90ºC. It's also recommended to add an adhesive such as paper glue to the print surface.
- **Enclosed 3D printers:** Even though it's not necessary to use enclosed 3D printers, we recommend to avoid room temperature variations.
- **Layer fan:** It's recommended to use a layer fan when printing PETG.
- **Warping:** PETG has a reduced thermal contraction, so it is not prone to warping and results in parts with good dimensional tolerances

Appendix 2 – PNP wiring

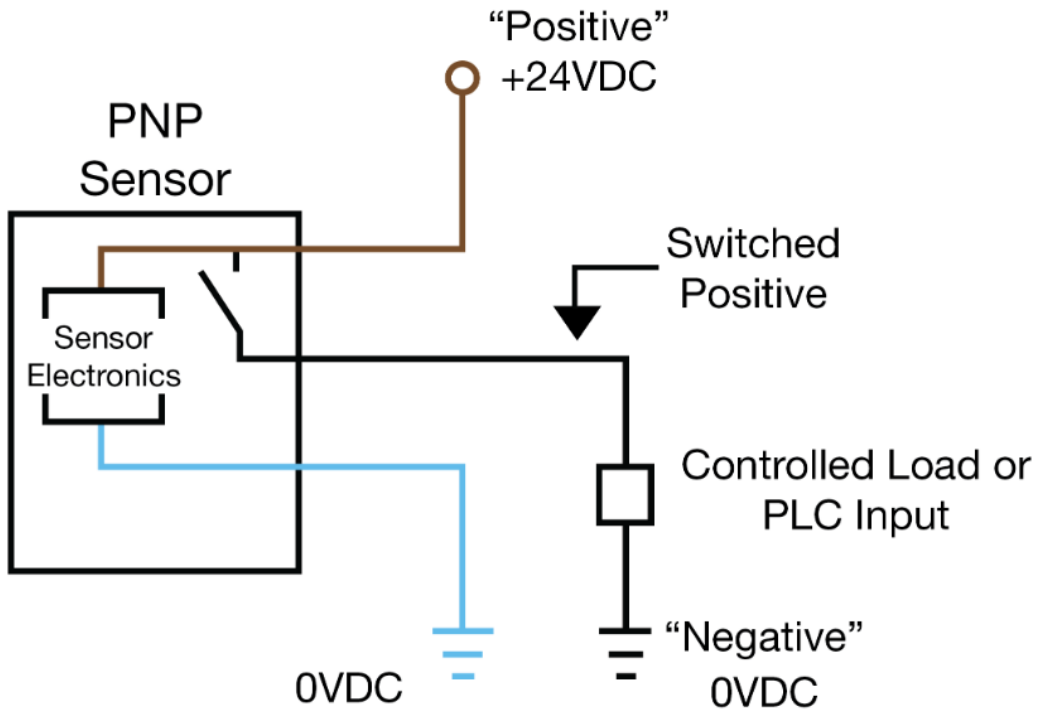
PNP stands for Positive, Negative, Positive. Also known as sourcing. On an IO Module, a PNP input, when undriven is pulled up to a high state e.g. +24V.

Most common in Europe is the 'sinking' type of input/output, these will be used with the PNP sensor or actuator. Less common nowadays are input cards that 'source', these were popular in Asia and require the NPN type of sensor in order to operate correctly.

Here's a simple way remember how to wire up a 3-wire DC PNP:

PNP = Switched Positive

"Switched" refers to which side of the controlled load (relay, small indicator, PLC input) is being switched electrically. Either the load is connected to Negative and the Positive is switched (PNP).



Appendix 3 – Teensy 4.1

Teensy 4.1 is the most powerful Arduino compatible microcontroller available today. Based on the NXP i.MX RT1062 ARM Cortex-M7 running at 600MHz with the ability to be overclocked. It is formatted into a very compact 'teensy' board outline for easy embedding into projects or for use with solderless breadboards. Perhaps best of all, it is compatible with the popular Arduino IDE programming environment as well as many of the existing Arduino libraries, so it is very easy to get up and running unlike many other advanced microcontrollers that are available.

The heart of the i.MX RT1060 microcontroller is an ARM Cortex-M7 CPU core that brings many powerful features to a true real-time microcontroller platform.

The Cortex-M7 is a dual-issue superscalar processor, meaning the M7 can execute two instructions per clock cycle, at 600MHz! Of course, executing two simultaneously depends upon the compiler ordering instructions and registers. Initial benchmarks have shown C++ code compiled by Arduino IDE tends to achieve two instructions per cycle about 40% to 50% of the time while performing numerically intensive work using integers and pointers.

For more information please visit PJRC webpage.

https://www.pjrc.com/store/teensy41.html