

About This Manual Trial

AuboStudio is a teaching software developed by Aubo specifically for AUBO robotic arms. It is designed for controlling and programming of robotic arms.

This manual applies to ARCS 0.31. For instructions on how to check software version, please refer to [Version Check](#).

The applicable controller models are as follows: iS Series, S Series, C Series, ES Series and Advanced. For further information regarding the controller model, please refer to [Hardware User Manual](#).

The robot arm model's size and color used in the manual are for reference only. Please refer to the actual product for details.

Manual usage

This Manual should be referenced when you use AuboStudio to control AUBO physical or virtual machines.

Intended readers

This Manual is prepared for AuboStudio users, including customers, programmers, technical support personnel, and operators.

Preconditions for operation

For readers wishing to operate according to the instructions in this manual, there is no prerequisite of prior knowledge.

Safety Information

The AuboStudio enables the configuration of numerous safety features of the AUBO robot, such as safety I/O interfaces, along with additional electrical interfaces for connecting to various devices and protective devices. All safety features and I/O interfaces are designed according to ISO 13849-1:2015 (PL=d, CAT 3). The power and force limitation safety features meet the impact force limit requirements in ISO/TS 15066-2016, with the force value lower than the maximum thresholds specified for transient and quasi-static contact between humans and robots.

Safety instructions

Danger

"Danger" indicates a highly dangerous situation that, if not avoided, could result in serious injury or death.

Warning

"Warning" indicates a dangerous situation that, if not avoided, could result in injury.

Tip

"Tip" indicates a potential risk of equipment damage or data loss.

Info

"Info" indicates information that requires special attention.

Personnel safety

Before entering the working area of robot, ensure that all safety rules are strictly followed.

Please remain vigilant at all times and ensure your own safety!

Safety rules

To ensure the safe operation of the robot and personal user safety, adherence to the following safety guidelines is mandatory:

1. Please familiarize yourself with the operating procedures and safety information related to the hardware prior to use.
2. When the robot is operational, do not stretch your hands or other objects into the robot's operational area to prevent potential injury.
3. Before activating the robot power supply, please ensure the surrounding area is safe and free from any obstacles, which could interfere with robot operation.
4. In the event of an abnormal operation or unusual noises, please stop the robot immediately and contact the technical support team for inspection and maintenance.

Adhering the above safety guidelines will ensure safety operation of robot to the greatest extent possible, while also protect the personal safety of the user.

Network security

AuboStudio is compatible with communication protocols including TCP/IP, VNC and X. Prior to using the network, users must first conduct a risk assessment of network security.

Tip

1. In order to ensure optimum network security, it is strongly recommended to active SSH key authentication, refrain from using unsecure software to connect a device. This will enhance the security of identity authentication and reduce the risk of unauthorized remote access.
2. Operate the server with caution, avoid downloading any unverified software or performing any unconventional operations, thereby minimizing potential security risks, preventing the spread of malicious software, and creating a clean and standard server environment.
3. If the LAN to which the robot is accessed is connected to an external network, ensure that identity control authentication is enabled in the firewall to prevent unauthorized access and potential security threats.

AUBO and its affiliates shall not be liable for any data loss, service interruption, privacy breach or other losses arising from safety attacks, device vulnerabilities, user errors, third-party service failures, or other reasons.

Password security

In the interest of ensuring operational safety, the software has been designed to incorporate a password setting function. Users are able to manage and control different operations through

the setting of distinct passwords. Please refer to "[User Authority](#)" for further information.

To ensure optimal password security, please follow these recommendations:

1. Set a complex password that contains upper/lower-case letters, numbers, and special characters, and avoid using common words or personal information.
2. Passwords should ideally have a minimum of 8 characters, since longer passwords are more difficult for others to guess.
3. To avoid unexpected disclosure of password, set a unique password for each operation, and do not share a password for two or more operations.
4. To reduce the risk of password guessing or cracking, it is advisable to change passwords every three months.

Tip

Please keep Admin password in mind. The system does not support password recovery functionality. If you forget the password, you may be unable to use the device. Furthermore, it is imperative to safeguard your password from unauthorized access.

Start Here

Thanks for choosing AuboStudio. Aubo Studio is a specialized teaching software developed by AUBO exclusively for AUBO robot arms. The AUBO robot arm features a simple interface and easy operation, allowing users to quickly master its use.

This Manual describes the operation method of AuboStudio software in controlling AUBO robot arms. This chapter introduces the basic operations required before using the AuboStudio.

Warning

Please read the operating procedures and safety information related to hardware before using this software.

AuboStudio teach pendant software

Wired teach pendant

AuboStudio has been installed in the controller of AUBO robot arms before delivery. To use the AuboStudio software, simply connect the wired teach pendant to the controller, and then turn on both the controller and the wired teach pendant.

Client

Tablet operating system (OS) : Android 10 or above

Installing client (APP)

Please download the latest AuboStudio APP installation package from [the official website](#), and install it on a tablet that meets the requirements specified in [Appendix: Client Installation Environment Requirements](#).

To ensure the proper functioning of the app, please complete the connection between tablet and robot by following the instructions in "[Wireless Connection](#)" before using the app to control robot.

ARCS System 0.32.1 Version Update Details

This document describes the major new features of ARCS System `arcs 0.32.1-beta.1`, with the corresponding AUBO STUDIO version: `0.10.1-beta.1`.

Change Type	Description	Related Document
Feature Update	Added supported languages for AUBO teach pendant.	Language Setting
Feature Update	Added support for configuring [Value Format] when adding Modbus signals.	Modbus Signal Management
Feature Update	Added [Timeout Termination] function to basic programming module.	Settings
Feature Update	Added [Activate] function to settings module.	Activate
Feature Update	Added [Gripper] function to configuration module.	Gripper
Feature Update	Added [EtherNet/IP] configuration to configuration module.	EtherNet/IP
Feature Update	Added trigger action selection support for input/output signals in I/O configuration module.	I/O Configuration
Feature Update	Updated [Tool RS485] function in configuration module.	Tool RS485
Feature Update	Added [External Axis] configuration to basic mode in movement module.	Basic Mode
Feature Update	Added force-controlled dragging function in HandGuide mode.	Force-Controlled Dragging
Experience Optimization	Added notification reminder for robots exceeding their service life.	Robot Connection Description
Experience Optimization	Added explanation for ARCS system installation failure scenarios.	Online Software Update
Experience Optimization	Improved operation mode restriction prompts.	User Authority
Experience Optimization	iH series robots renamed to i series robots.	N/A
Experience Optimization	Optimized ARCS system I/O configuration page.	I/O Configuration

Experience Optimization	Added option to configure whether to stop the program when editing a program in programming settings.	Programming Settings
-------------------------	---	--------------------------------------

Power-On/Off and Connection

Power-on

Tip

Please stand outside the reach (work radius) of the robot arm.

AUBO-CB-M controller, AUBO-CB-iS controller

Once the robot arm and controller have been installed and wired, and the external power supply has been activated, please turn the switch above the controller power supply interface to "I"/"ON". Then, wait for the POWER indicator light and STANDBY indicator light to light up. Finally, press and hold the Power button in the upper right corner of the teach pendant to enter the system.

AUBO-CB-C controller

Method 1: Power on via the ON/OFF STATE button of the controller

1. After completing the installation and wiring of the robot arm and controller and turning on the external power supply, turn the switch above the controller power supply interface to "I"/"ON";
2. Press and hold the ON/OFF STATE button of the controller until the ON/OFF STATE button light flashes, indicating that the controller is started.

Method 2: Power on via the Power button on the control handle

1. After completing the installation and wiring of the robot arm and controller and turning on the external power supply, turn the switch above the controller power supply interface to "I"/"ON";
2. Press and hold the Power button on the control handle until the power indicator light on the control handle flashes, indicating that the controller is started.

AUBO-CB-S control box

Method 1: Power on via the POWER button of the controller

- After completing the installation and wiring of the robot arm and controller and turning on the external power supply, press and hold the POWER button of the controller until the POWER button light strip flashes, indicating that the controller is started.

Method 2: Power on via the Power button on the control handle

- After completing the installation and wiring of the robot arm and controller and turning on the external power supply, press and hold the Power button on the control handle until the power indicator light on the control handle flashes, indicating that the controller is started.

AUBO-CB-MINI-S-48V control box

After completing the installation and wiring of the robot arm, controller and external switch and turning on the external power supply, turn on the external switch, and wait for the POWER/STATE indicator light to light up, indicating that the controller is started.

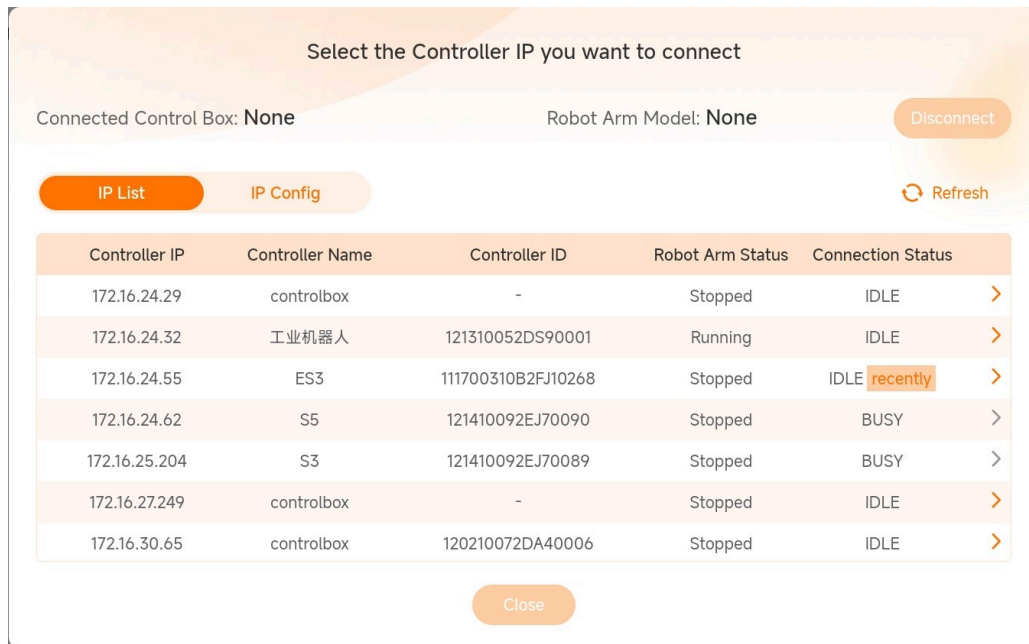
Wireless connection

Tip

Controller models that support wireless connectivity include AUBO-CB-C, AUBO-CB-S, and AUBO-CB-MINI-S-48V.

Search on the tablet and connect to the robot's WiFi. The default WiFi name is "(AAAA-CCCC-YYYY-)XXXX", and the default password is "12345678". Here, AAAA is a fixed brand name; CCCC is the controller model; YYYY is the production year, month and batch; and XXXX is the last 4 digits of the controller's serial number.

After connection via Wi-Fi, open the AuboStudio APP, select the entry in the [IP List] (or enter the controller's IP address) to connect the controller.

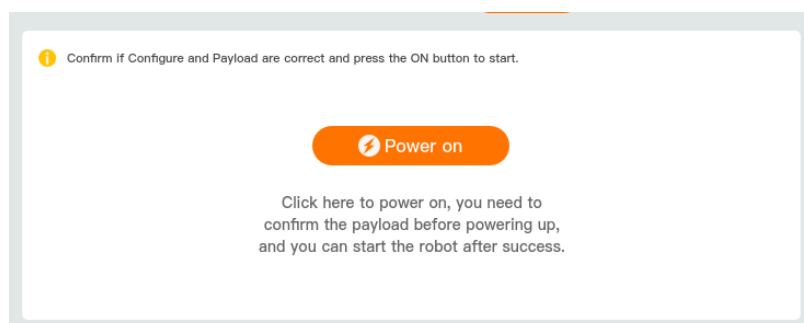


Upon successful connection, the software interface will refresh and enter the [Home] page. Click the [Disconnect] button at the top to disconnect the currently connected robot.

After the software is successfully connected, users can modify the Wi-Fi name and password. For settings of the built-in Wi-Fi, refer to "[System Settings - Wi-Fi](#)".

Robot power-on/off

1. Click the [Power On] button on the [Home] page.



2. Select the payload mode and click [Continue]. Payload has two preset modes:

- Payload: this is the default mode, and in this mode, the payload is fixed to 0 kg.
- Custom load: Enter the value of payload. The value shall not exceed 1.2 times the robot's rated load.

Active Payload is used to temporarily overwrite the Configure Payload.

Active Payload: ☒ Payload

Payload: Kg

3. The system begins initialization, and the robot will perform the following:

- Robot power-on.
- Load the robot drive.
- Activate the robot, and then the robot enters active state.

4. After successful system initialization, click the [Start] button, then the robot will release the brakes and enter operable state.

Warning! Stand clear of the robot when starting it!

Click here to release the brakes and run the robot. Click here to power off.

5. Click [Power Off] to power off the robot.

0.000Kg

Payload

Power-off

Normal power-off

Recommended power-off method:

1. Click "Home > Shutdown" in the AuboStudio teach pendant software to shut down the system.
2. Power off remotely.
3. Press and hold the [Power] button of the wired teach pendant for 3 seconds.
4. Press and hold the [Power] button on the control handle for 2 seconds.

Forced power-off

Danger

Please avoid using forced power-off.

In the event of a forced power-off, please save any ongoing tasks and files before powering-off, if possible. And perform necessary system checks and maintenance afterwards.

Forced shutdown method:

1. In any state, press and hold the [Power] button of the teach pendant for 10 seconds.
2. In any state, press and hold the [Power] button on the control handle for 10 seconds.
3. Disconnect the power supply directly.

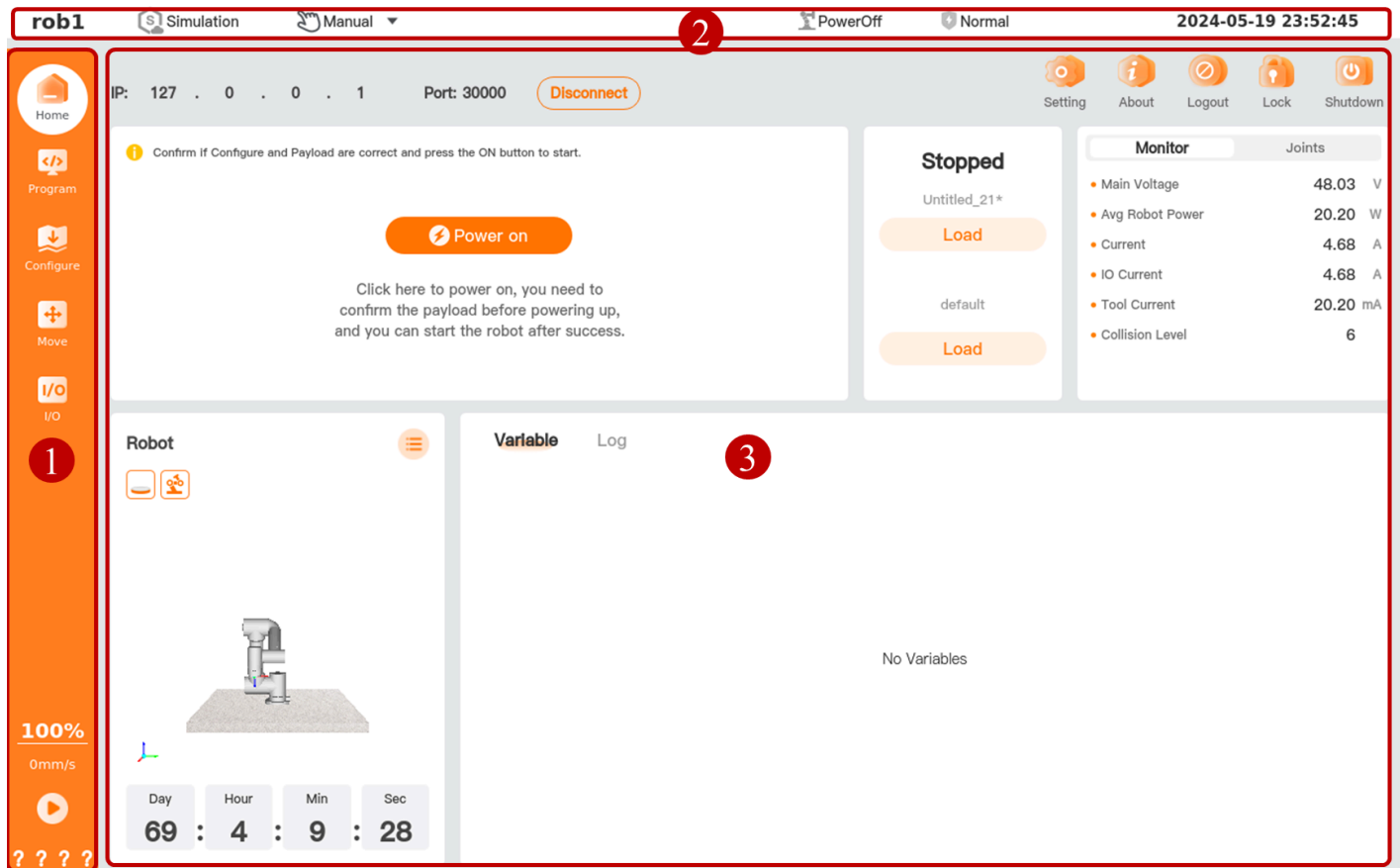
Danger

1. Methods 1 and 2 are only available for system shutdown under abnormal conditions.
2. Avoid directly cutting off the power supply to shut down the device, as this may cause serious system damage.

Introduction to Teach Pendant Software

Interface introduction

The AuboStudio interface is structured as follows:



Serial number	Name	Description
1	Menu bar	The upper part of the menu bar contains function menus, including: Home, Program, Configure, Move, and I/O. The lower part contains program menus, including: speed, run, safe parameter.
2	Status bar	The top status bar is always at the top of the interface while the software is running.
3	User interface	The user interface provides different views for managing and monitoring the robot under different function menus, allowing the user to control the robot easily and flexibly.

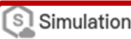
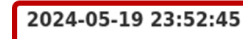
1. Function menus

Icon	Name	Description
	Home	Display basic information and program progress of the robot. For more information about "Home", refer to Home Tab .
 Program	Programming	Create or modify the robot program. For more information on "Programming", refer to Visual Programming .
	Configuration	Set up the AUBO robot and external device. For more information about "Configuration", refer to Configure Tab .
	Move	Control or adjust the movement of the AUBO robot. For more information about "Move", refer to Move Teaching .
	I/O	Monitor or set the controller's real-time input and output information. For more information on "I/O", refer to I/O .

2. Program menus

Icon	Name	Description
	Speed control	Speed Ratio: Set the robot's maximum running speed ratio. Click it to bring up the movement speed adjustment bar  , and drag the slider  to set the maximum speed ratio. TCP Speed: Display the current terminal speed of the robot.
   	Run/Resume Stop Pause Step	Program Run/Resume/Stop/Pause/Step button. * Only when all the nodes in the program tree comply with the predefined logic, that is, there are no nodes highlighted in yellow in the program tree, can you use the [Run] button to run the program.
 	Safe parameter	Display a four-digit hexadecimal security check code. If the parameters in "Configure > Safety" change, the system will update the check code. * Before the system is powered on, no safe parameters are accessible to the software, and question marks are displayed here. * Click here to view details of safe parameters.

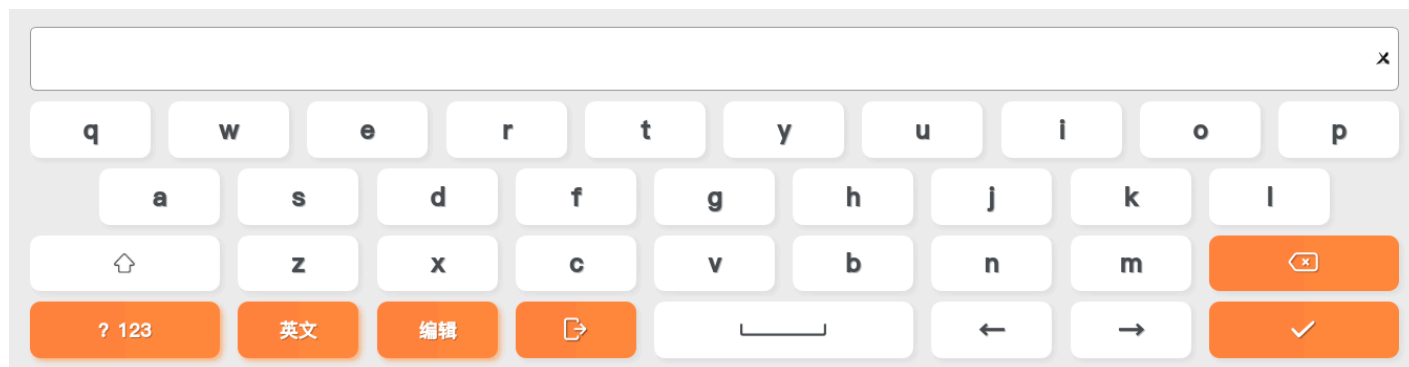
3. Top status bar

 rob1	 Simulation	 Manual	 PowerOff	 Normal	 2024-05-19 23:52:45
1	2	3	4	5	6

Serial number	Name	Description
1	Robot arm name	Display the name of the currently connected robot arm.
2	Simulation	Allow you to switch between a simulated robot arm and a real robot arm. When you turn on the Simulation button, the color of the simulation icon changes, and the system enters the simulation mode. * For detailed information on simulation mode, refer to Simulation Mode .
3	Mode switching	Allow you to switch between Manual, Auto, Linkage, and Local modes. * For detailed information on each mode, refer to Operating Mode . * Before switching the operating mode, please set an "Operating mode password". For details, refer to User Authority .
4	Robot mode	Display the current robot mode. This includes Power off, Power on, Idle, and Running modes. For details, refer to Robot Mode and Safety Mode .
5	Safety mode	Display the current system mode. This includes Undefined, Normal, etc. For details, refer to Robot Mode and Safety Mode .
6	Date	Display the current date of the controller.

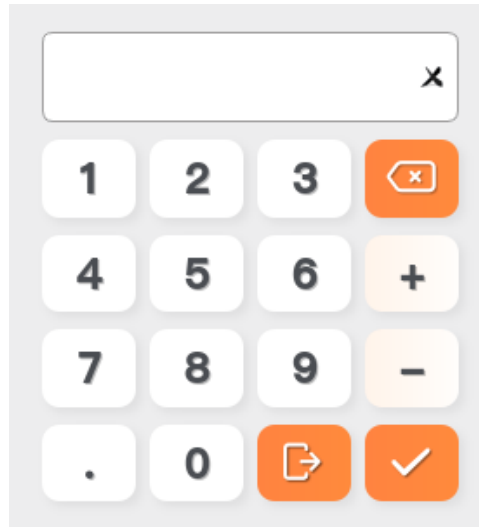
System soft keyboard

1. Text editor



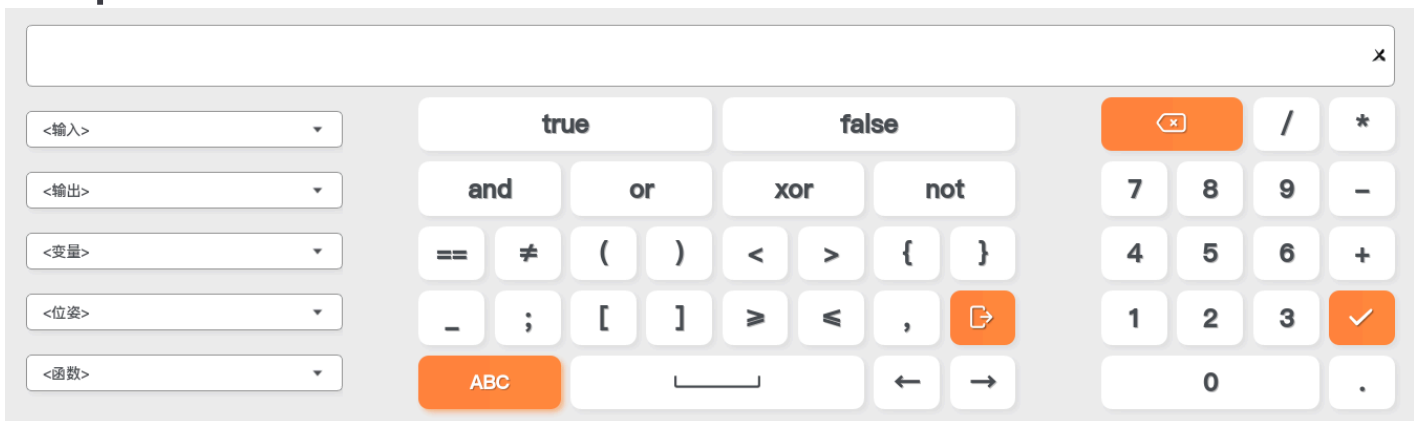
[Text Editor] Supports language switching between Chinese and English. Users can use [Text Editor] to edit and insert text.

2. Number editor



[Number Editor] Supports editing both positive and negative numbers. Users can use [Number Editor] to edit positive and negative numbers.

3. Expression editor



[Expression Editor] Include buttons, a special symbol keyboard, and a numeric keyboard. Click **ABC** to switch to [Text Editor]. Users can use [Expression Editor] to edit expressions.

- [Input] & [Output]: the drop-down menu contains all I/O interfaces.
- [Variables]: the drop-down menu contains all the defined variables.
- [Pose]: the drop-down menu contains all the defined coordinates.
- [Function]: the drop-down menu contains some special functions.

4. Considerations about editors

- When setting a digital signal via an expression, the value is set to true or false and the Modbus to 1 or 0. Otherwise, unpredictable errors will occur in the robot.
- When inputting a string via the expression editor, enclose the string in single quotes (') or double quotes (").

- When setting a position/pose for a variable via the expression editor, enclose the position/pose in curly braces ({}).

HandGuide

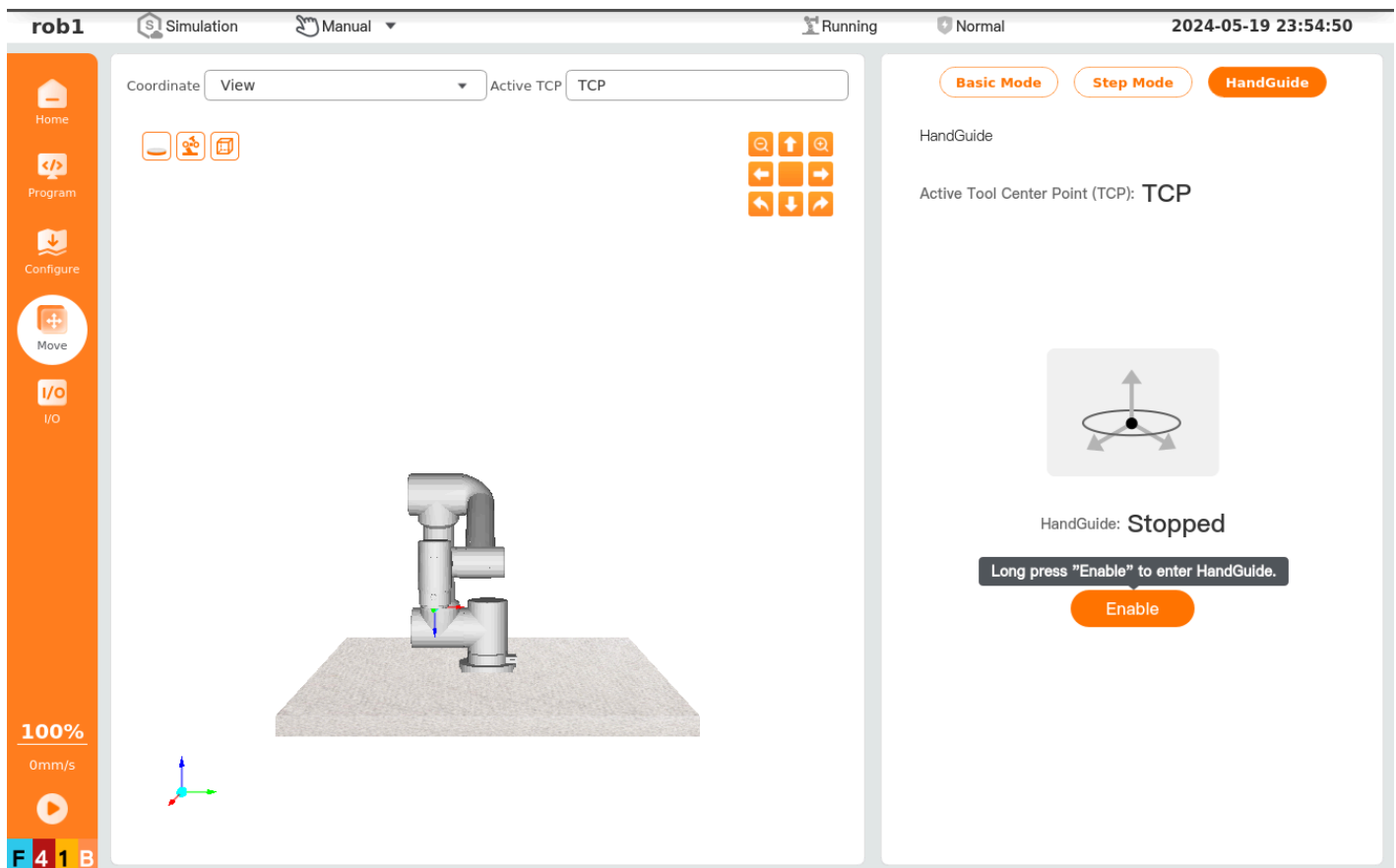
In HandGuide mode, you can manually drag the robot to the target position/pose. If the robot's position/pose is close to the joint limit or safe plane, the robot will produce a repulsive force.

When HandGuide mode is enabled, a "HandGuide" prompt will pop up in Top status bar. For specific usage methods and parameter settings, refer to "[Move Teaching - HandGuide](#)".

How to enter HandGuide mode?

Page [Move]

Click "Move > HandGuide" to enter page [HandGuide], press and hold [Enable]. When the icon is ungrayed, it indicates that the robot enters the HandGuide mode, and you can drag the robot to the target position/pose. To exit HandGuide mode, click [Disable] until the icon is grayed out, and then the robot stays in the current position/pose.



I/O actions

After the user sets an I/O for HandGuide, the robot enters the HandGuide mode when this I/O receives a valid signal, and then you can drag the robot to the target position/pose. When the valid signal to this I/O disappears, the robot exits the HandGuide mode. For the I/O setting method, see [I/O Setup](#).

Teach pendant button

Press and hold the button on teach pendant to enter HandGuide mode, and then you can drag the robot to the target position/pose. To exit HandGuide mode, release the button, and then the robot stays in the current position/pose. This function can be set through "Configure > Safety> ThreePosition". Refer to [Three-Position Switch](#) for details.



End-Of-Arm button

Press and hold the end-of-arm teach pendant button to enter HandGuide mode, and then you can drag the robot to the target position/pose. To exit HandGuide mode, release the button, and then the robot stays in the current position/pose.

提示

The end-of-arm HandGuide button is available on some models only. Please refer to the Hardware User Manual for details.

What is a Program?

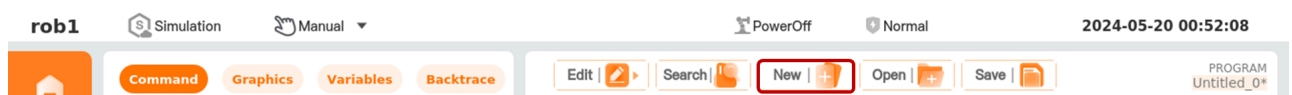
A program is a set of commands that a robot can recognize and execute. The robot is programmed to achieve the movement and control requirements in a specific application. In AuboStudio, users can write programs using the [Programming] interface of the system or write scripts using the interface of the system. This section will introduce how to program using the [Programming] interface. For instructions on using scripts, refer to "[Lua Script Programming](#)."

On page [Program], a program consists of a program block and program nodes, in which the program block is composed of multiple nodes for the convenience of code management. The robot's movement path is composed of a series of waypoints, which can be set by teaching or configuration items in [Command]. To control the movement and I/O signal sending/receiving, you can create and write one or more programs or set variables to run commands such as "ELSEIF" and "Loop."

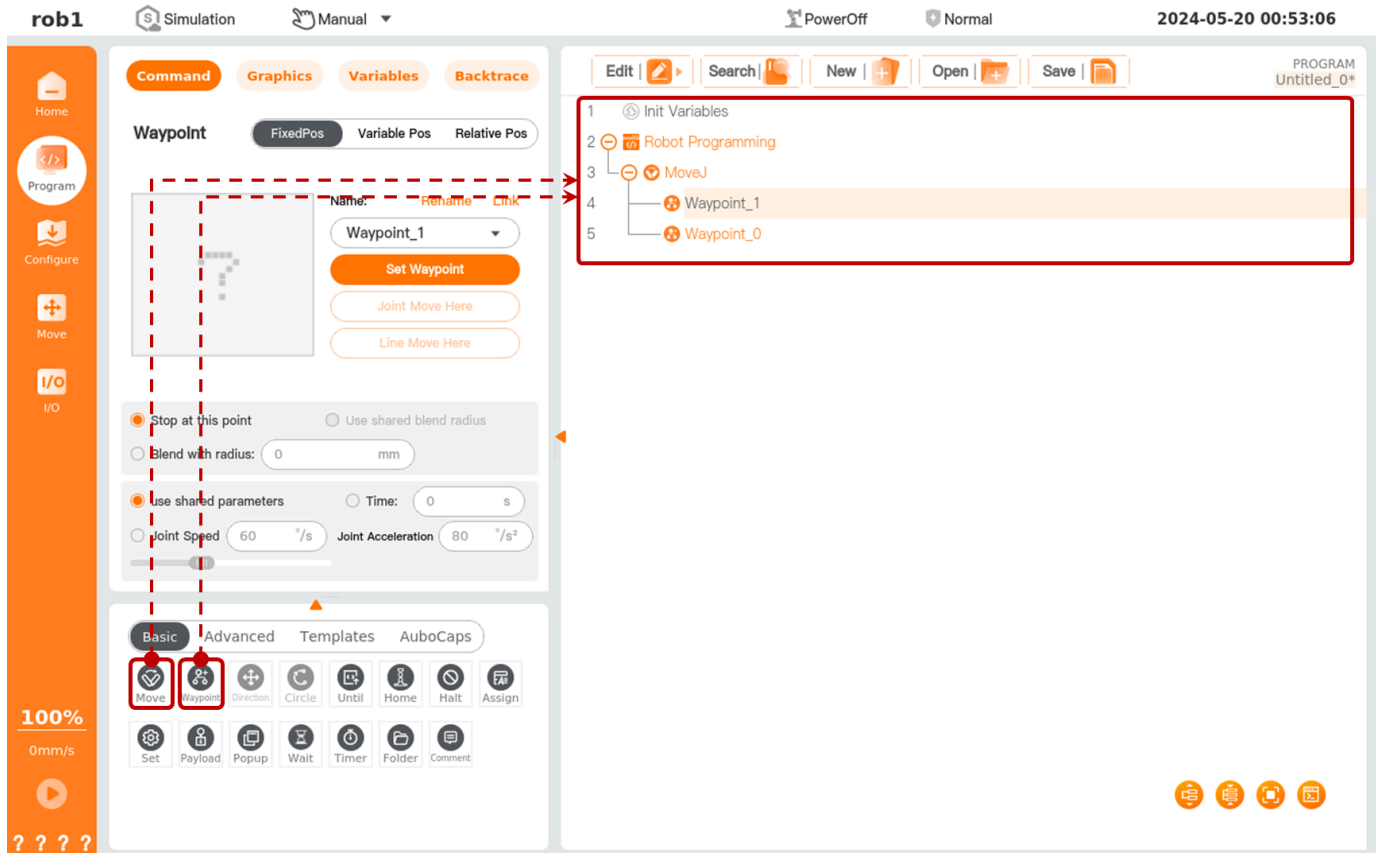
First program

The following is a simple program demonstrating how to write programs in AuboStudio. This example program enables the robot to move back and forth between two waypoints.

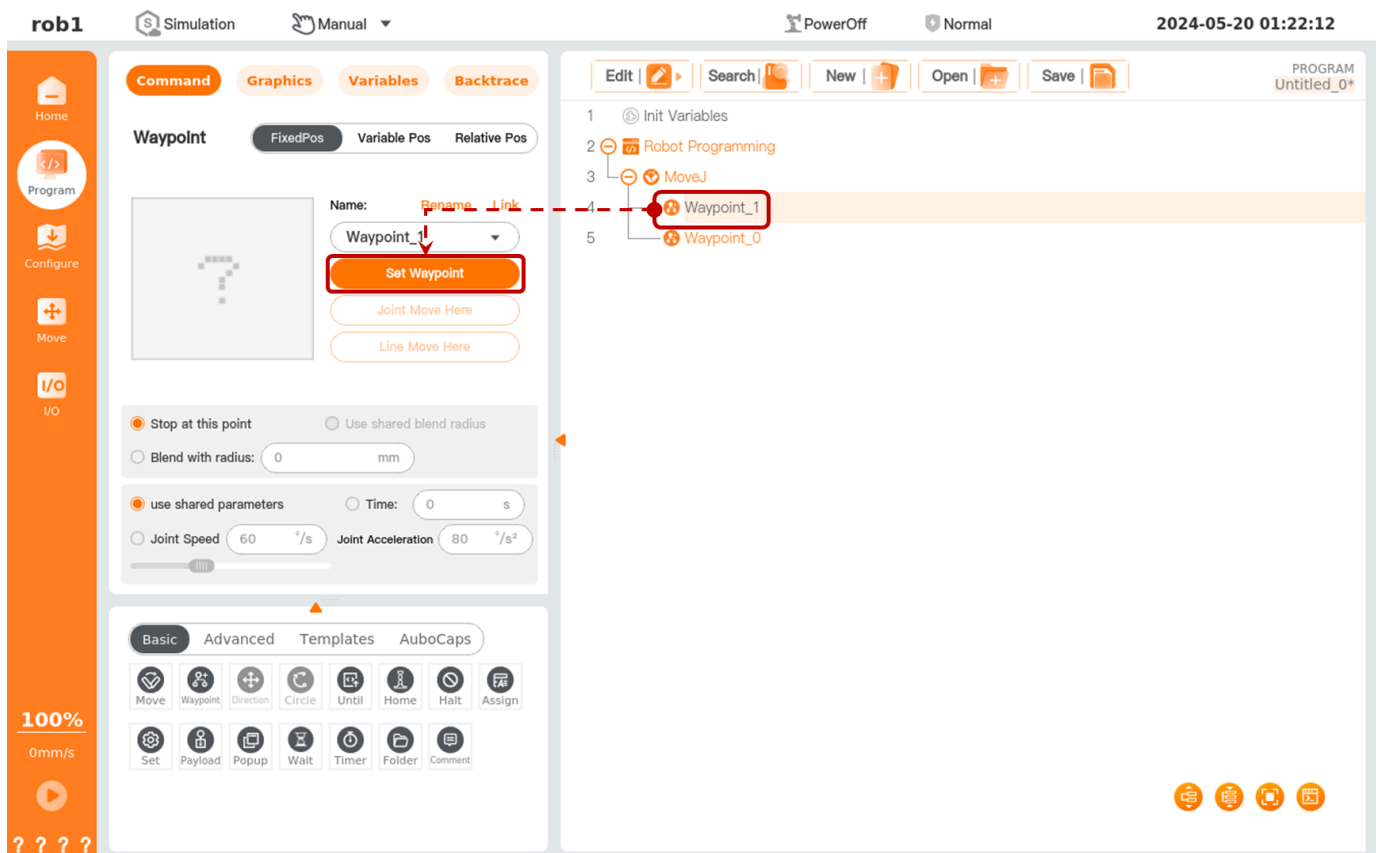
1. In page [Program], click [New] in the program manager tool bar to create a new program file.



2. Under the [Basic] tab, click [Move] to add the [Move] node to the program tree. Then click [Waypoint] to add a [Waypoint] node to the program tree. Now the program has a [Move] node, and two waypoints under this node, namely [Waypoint _0] and [Waypoint _1].



3. Click [Waypoint_1]. Then, click [Set Waypoint] under [Command] tab to enter page [Move].

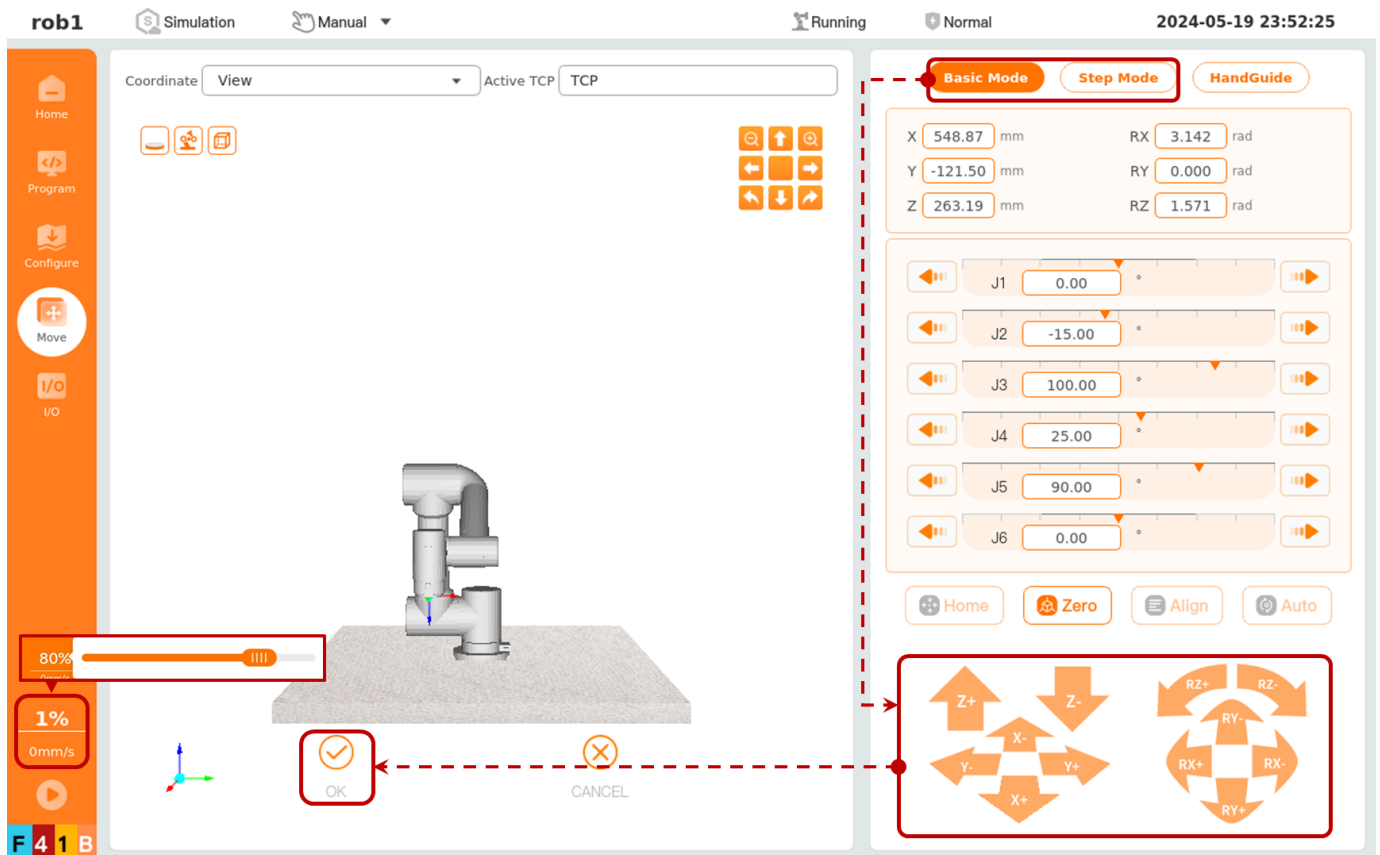


4. Users can select [Basic Mode] or [Step Mode]. Click [Position/Pose Control] to move the robot to the target position. After confirming the position, click [OK] to exit page [Move],

ending the position/pose settings of [Waypoint_1].

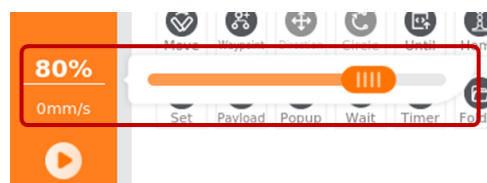
Tip

Before moving the robot arm, click the [Speed] button on the left menu bar, and drag the [Speed Bar] slider to adjust the movement speed of the robot arm.

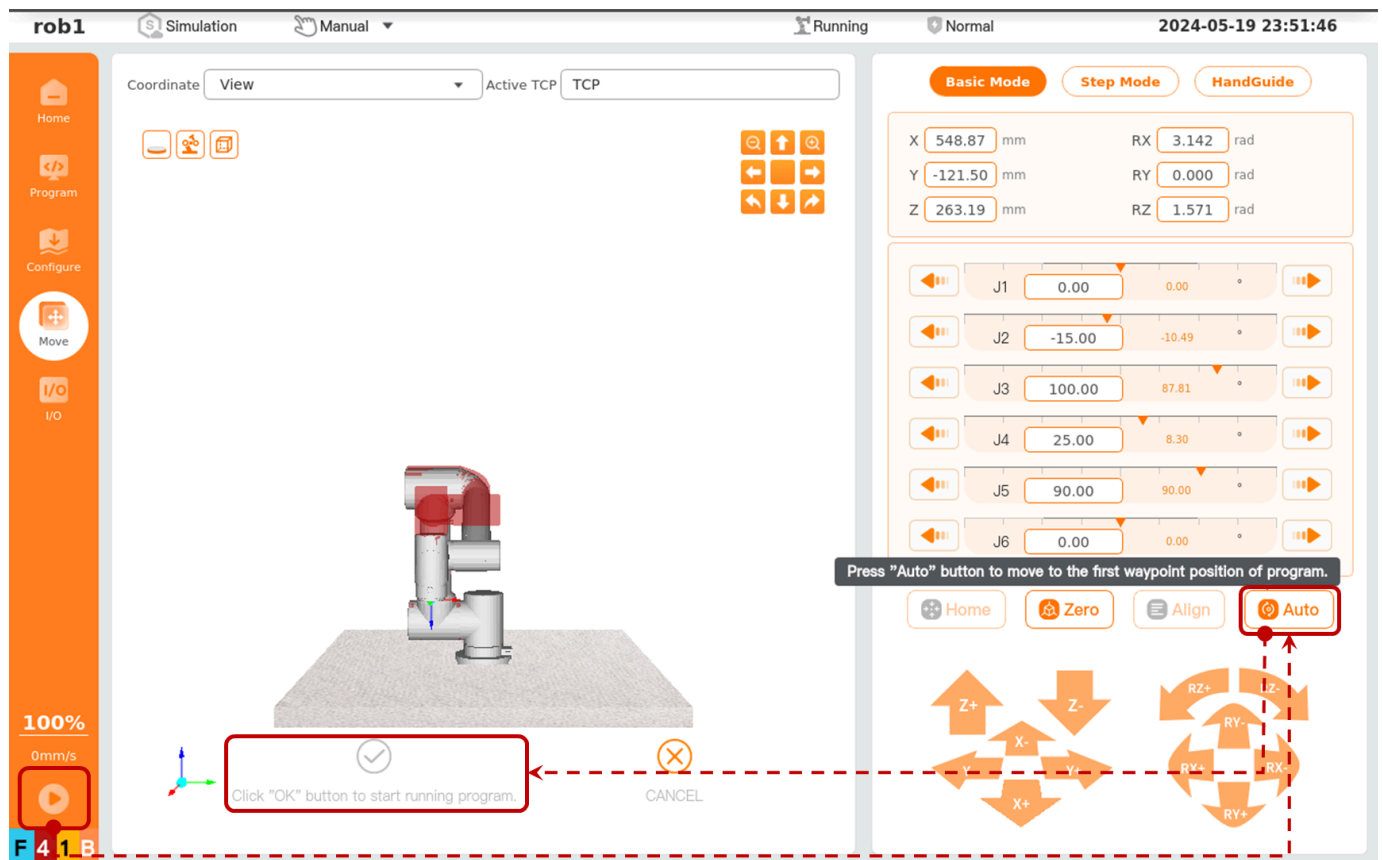


5. Repeat steps 3 to 4 to assign a value to [Waypoint_0].

6. Click [Speed] on the left menu bar to pop up [Speed Bar] slider, and drag the slider to adjust the speed.



7. Click the [Run] button on the left menu bar, select [Run Program from Start Point] to enter the [Move] interface. Then press and hold [Auto] until this button is grayed out to move the robot arm to the start position of the program. At this time, click the [Confirm to Start Program] button to run the program, then the robot moves between [Waypoint_0] and [Waypoint_1].



Congratulations! You have completed the first robot program! This program enables the robot to move between [Waypoint_0] and [Waypoint_1].

Quick Start Configuration

After robot is started for the first time, users are advised to complete the following key configuration items first to enable robot to be put into use more quickly.

Start

Please refer to [Configure tab - General - Start](#).

TCP

Please refer to [Configure tab - General - TCP](#).

Payload

Please refer to [Configure tab - General - Load](#).

Coordinate system

Please refer to [Configure tab - General - Coordinate system](#).

Collision

Please refer to [Configure tab - General - Collision](#).

Version Check

Software and client versions

Click "Home > About" to view software version, SDK version, plugin version, build time, copyright, company website, and other information.



Firmware version

Click  in the [Robot] functional area to view the serial number and version number of the current robot arm and controller.

The firmware package version is derived comprehensively based on the versions of joint and interface boards of the robot arm, including the robot firmware package version and the control box firmware package version. The rules to determine the version are as follows:

1. If only the firmware version of the end-of-arm tool is not consistent with the known one, then the firmware package version is marked with a gray star.
2. If the firmware versions of both the joints and base are not consistent with the known one, then the firmware package version is marked with a red star.



Robot:

Details	Joint 1	Joint 2	Joint 3	Joint 4	Joint 5	Joint 6	Tool	Base
Firmware	v1.2.3	v1.2.3	v1.2.3	v1.2.3	v1.2.3	v1.2.3	v3.2.6	v1.2.3
Hardware	v3.2.1	v3.2.1	v3.2.1	v3.2.1	v3.2.1	v3.2.1	v1.2.5	v1.2.3
UUID	ABCD	EF01	2345	6789	FECB	A987	1324	1243

DH parameter	Theory Real Errors					
/	Joint 1	Joint 2	Joint 3	Joint 4	Joint 5	Joint 6
Theta	3.14159	-1.5708	0	-1.5708	0	0
Beta	0	0	0	0	0	0
D	0.122	0.1215	0	0	0.1025	0.094
A	0	0	0.408	0.376	0	0
Alpha	0	-1.5708	3.14159	3.14159	-1.5708	1.5708

Control Box:

Details	Master board firmware	Slave board firmware
Firmware	v3.4.25	v1.5.27
Hardware	v1.5.26	v3.4.25
UUID	1234	4321

Operating Mode

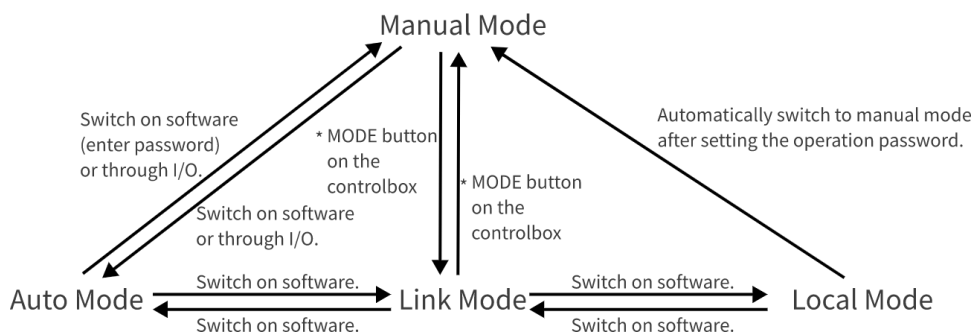
AuboStudio is designed with four operating modes: Manual, Automatic, Link, and Local, which can be selected by operating the [Mode Switch] / /  button in the [Status Bar].

Before operating the [Mode Switch] button for mode switching, please set a mode password first. For the setting method of the mode password, please refer to [User Authority](#).

Table: Function of Operating Modes

	Manual mode	Automatic mode	Link mode	Local mode
HandGuide	√			√
[Move]	√			√
[Configure]	√			√
[Program]	√			√
Execute program	√	√		√
Execute a program through I/O			√	

The switching logic between operating modes is as follows:



*: This function is only supported by some controller models. For details, please refer to the Hardware Manual.

2.3.1 Manual mode

In Manual mode, all the functions of the software are available for users to use, but users cannot control the robot to execute a program through I/O. This mode is usually suitable for applications with one robot only.

2.3.2 Automatic mode

In Automatic mode, users can neither edit and set programs and installation profiles, nor move the robot arm. That is, functions including [Program], [Configure] and [Move] are disabled, and the robot only executes the defined programs.

2.3.3 Link mode

In Link mode, users can neither edit and set programs and installation profiles, nor move the robot arm. That is, functions including [Program], [Configure] and [Move] are disabled, but the program is able to control through I/O. Link mode can only be switched from Automatic mode.

Tip

To unplug the wired teach pendant, first press the "TEACH PENDANT ENABLE/DISABLE" button on controller and then unplug teach pendant cable; otherwise, emergency stop will be triggered.

2.3.4 Local mode

The Local mode is activated when neither mode safety input I/Os nor mode switching password is set for the system.

Local mode is not available when the authority is Advanced.

Robot Mode and Safety Mode

1 Robot mode

Name	Description
No controller	Currently not connected to controller.
Not connected	Not connected to robot arm body.
Safety confirmed	Robot is performing safety configuration.
Power off	Robot arm body is in power-off status.
Powering on	Robot arm body is being powered on and initialized.
Power on	Robot arm body is successfully powered on, brake is locked, and the initial state of joint is not acquired.
Idle	Robot arm body is successfully powered on, brake is locked, motor is not powered, and the initial state of joint has been acquired.
Brake release	Robot arm body is successfully powered on, and brake is being released.
Operation	Robot arm body is successfully powered on, brake is released, and the software can control robot arm normally.
Powering off	Robot arm body is currently being powered off.
Reverse drive	Robot arm body is powered on, brake is released, and motor is not powered. Users can manually drag robot arm to a designated position, but robot arm cannot maintain position/pose.
Maintenance	Robot is in maintenance mode, such as firmware upgrade, parameter writing, etc.
HandGuide	Robot is in HandGuide mode.

2. Safety mode

Name	Description
Undefined	Robot has not obtained safety mode or has not obtained the correct safety mode.

Normal	Robot is in normal operating status and users can operate robot normally.
Safe stop	Robot is in a stop state triggered by safety I/O. At this time, robot arm does not maintain the track, brake is applied, and power is not off.
Protective stop	Robot is in a software-triggered stop state. At this time, robot arm maintains the track, brake is not applied, and power is not off.
Reduced mode	Robot is in reduced operation mode. For reduced parameters, please refer to "Configuration > Safety > Joint limit".
Recovery mode	Robot is in recovery mode. When robot arm is in an over-limit position, it enters recovery mode when it is powered on again. Exit this mode by dragging robot arm within the safety limit range.
Over-limit	Robot is in an over-limit state. The current robot arm speed or other safety parameters exceed the settings of safety configuration.
System emergency stop	Robot is in system emergency stop state.
Robot emergency stop	Robot is in robot emergency stop state.
Fault	Robot is in fault state.

Situations that trigger emergency stop

Safety mode	Trigger conditions	I/O status
Robot emergency stop	Teach pendant emergency stop button, control box emergency stop button, fixed emergency stop input	Internal emergency stop
System emergency stop	Safety input - Emergency stop	External emergency stop

Simulation Mode

In simulation mode, the robot status is simulated by its simulation model, and users can debug the program, check the operating status of the robot by simulation model, and verify the configuration.

To switch to simulation mode, click [Simulation] on the top status bar, and then the color of the [Simulation] icon will change. To exit simulation mode, click [Simulation] again, and then the color of the [Simulation] icon will return to normal.

rob1

Simulation

Manual

PowerOff

Normal

2024-05-19 23:52:59

Home

Program

Configure

Move

I/O

100%

0mm/s

???

IP: 127 . 0 . 0 . 1

Port: 30000

Disconnect

Confirm If Configure and Payload are correct and press the ON button to start.

Power on

Click here to power on, you need to confirm the payload before powering up, and you can start the robot after success.

Stopped

Untitled_21*

Load

default

Load

Monitor

Joints	
Main Voltage	48.04 V
Avg Robot Power	20.20 W
Current	4.87 A
IO Current	4.87 A
Tool Current	20.20 mA
Collision Level	6

Robot

100%

0mm/s

Day

Hour

Min

Sec

69

:

4

:

9

:

42

Variable

Log

No Variables

31 / 232

Coordinate System

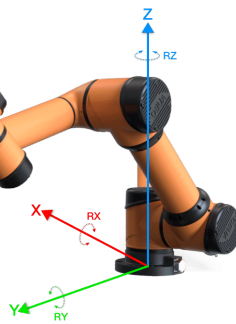
Users can employ a coordinate system to depict various objects or boundaries. For example, a robot may move relative to an object, which could be a table, workpiece, blank, another machine, or a boundary around the robot. Under such circumstances, it is necessary to employ a custom coordinate system to depict these objects and name the custom coordinate system for utilization.

The system has two preset coordinate systems, the position/pose of which is defined by the robot's configuration. Users can also customize new coordinate systems by using points, lines, planes, etc. in "Configure > General > Coordinate System."

Predefined coordinate system

Base coordinate system

The origin of the base coordinate system is at the center of the robot base.



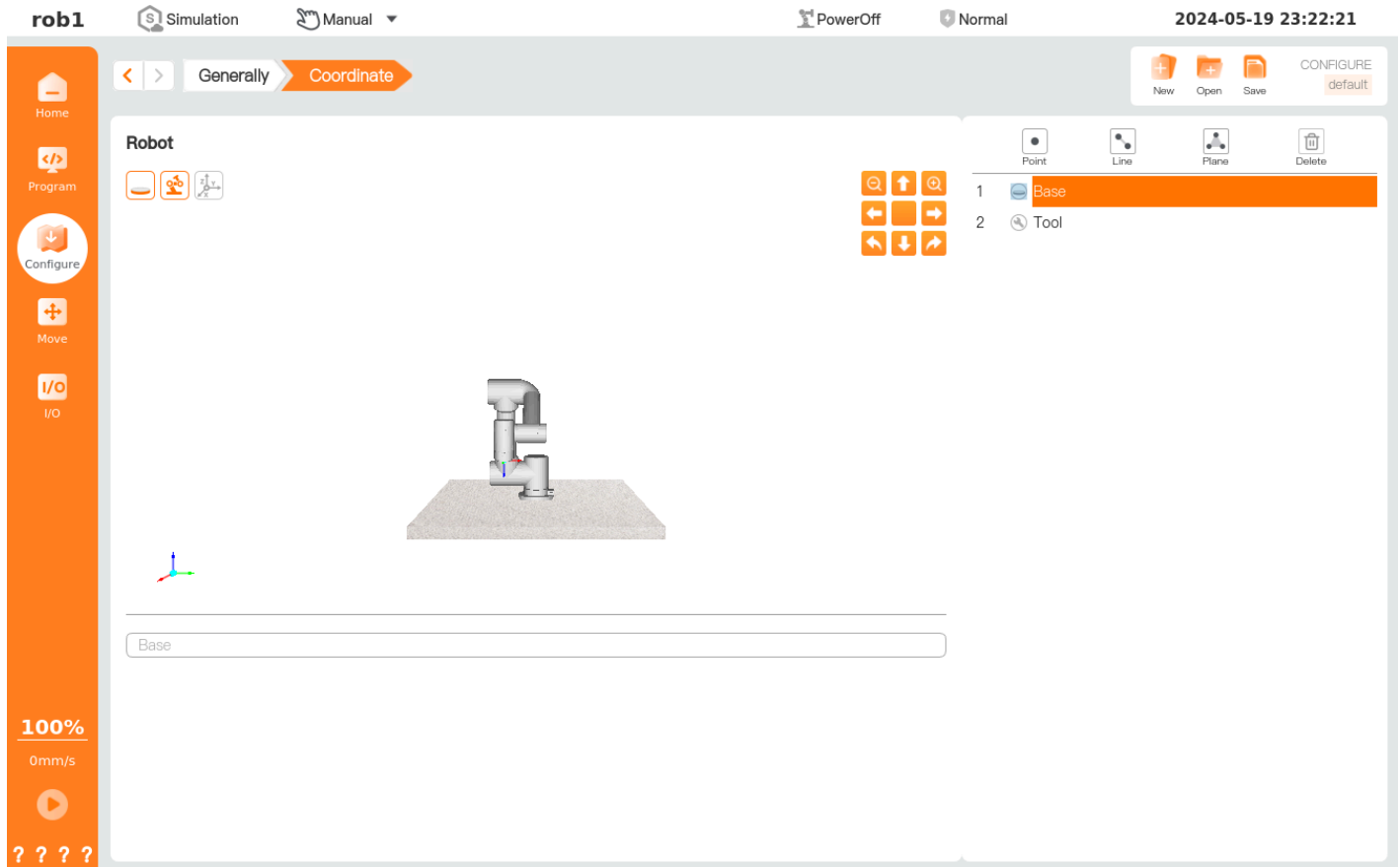
Tool coordinate system

The origin of the tool coordinate system is at the center of the current TCP.



Custom coordinate system

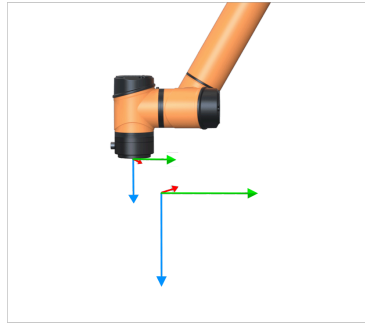
On the "Configure > General > Coordinate System" interface, users can create a new coordinate system in the form of a point, line, or plane. The new coordinate system is positioned by the position of TCP. After selecting the method to create a new coordinate system, users can show the position of the coordinate system by moving TCP to the target position/pose.



-  Point: Create a new coordinate system with points.
-  Line: Create a new coordinate system with lines.
-  Plane: Create a new coordinate system with planes.
-  Delete: Delete the coordinate system.

1. Create a new coordinate system with points

Creating a new coordinate system with points, i.e., showing the position of a new coordinate system origin through TCP, with the coordinate system axes orienting in the same directions as those of TCP. This method is typically used to define a security boundary or global primary configuration for a robot.



- Click [Point] ☐ to create a new coordinate system.
- Click the input box below to rename the newly created coordinate system. Click [Edit], control the robot using the position/pose control button and complete necessary settings, and then click [OK] to save the data and return to the [Coordinate System] page. Pose information used to set the point is displayed below the simulation model.
- Click [MoveJ to Here] or [MoveL to Here] to enter the [Move] page, and long press [Auto] to move the robot back to the position/pose used in creating the current coordinate system.

rob1
Simulation
Manual
PowerOff
Normal
2024-05-19 23:23:00

Home
Program
Configure
Move
I/O
100%
0mm/s

Generally
Coordinate

New
Open
Save
CONFIGURE
default*

Point
Line
Plane
Delete

1 Base
2 Tool
3 Point_0

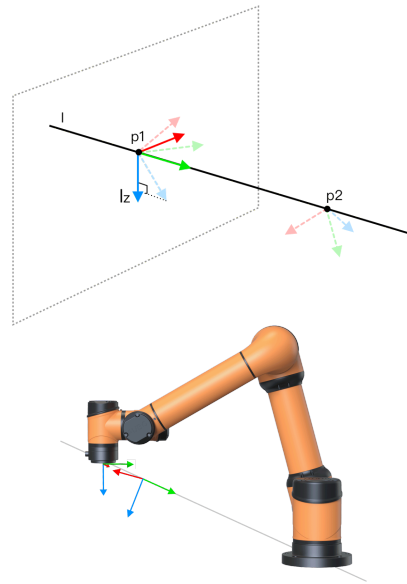
X nil mm
Y nil mm
Z nil mm
RX nil rad
RY nil rad
RZ nil rad
Joint 1 nil
Joint 2 nil
Joint 3 nil
Joint 4 nil
Joint 5 nil
Joint 6 nil

Point_0
Edit
Joint Move Here
Line Move Here

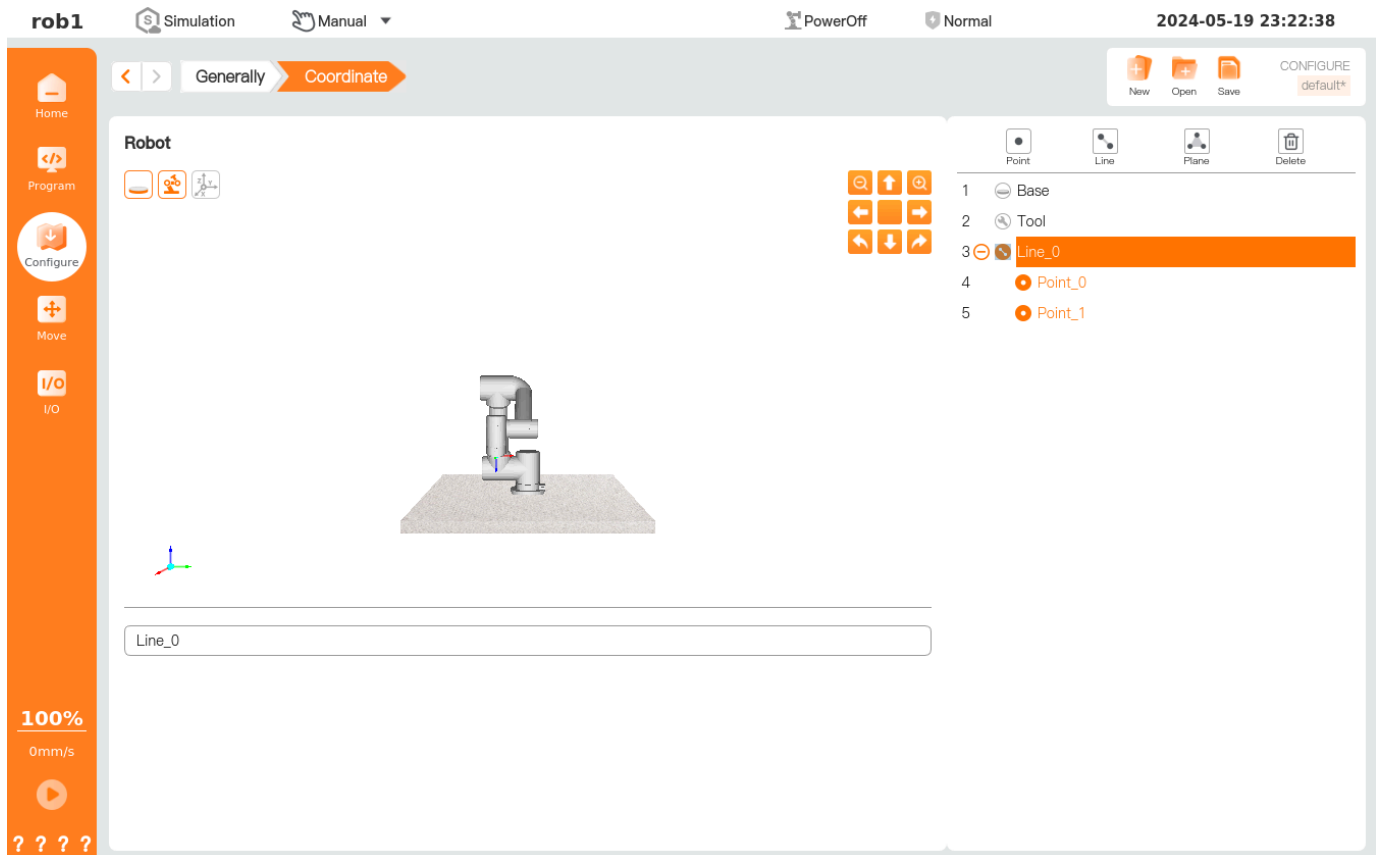
2. Create a new coordinate system with lines

Create a new coordinate system with lines, with the first point as the system origin; The Y axis is a line from the first point to the second point; the Z axis is the projection of the Z axis of the

first point on a plane perpendicular to the line. This method defines the moving lines that the robot needs to follow.

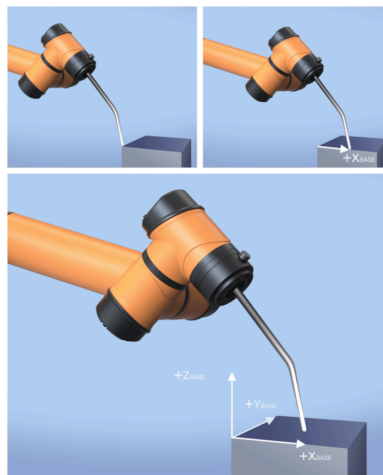


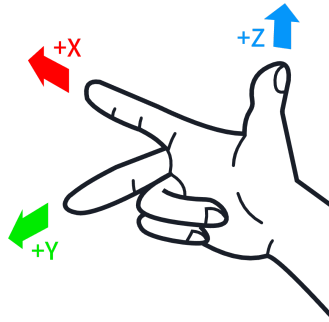
- Click [Line]  to create a new coordinate system.
- Click the input box to rename the newly created coordinate system.
- Click  to expand the list, select waypoints one by one, click [Edit], control the robot using the position/pose control button, complete necessary settings, and then click [OK] to save the data and return to the [Coordinate System] interface.
- Click [MoveJ to Here] or [MoveL to Here] to enter the [Move] page, and long press [Auto] to move the robot back to the position/pose used in creating the current coordinate system.



3. Create a new coordinate system with planes

Creating a new coordinate system with planes, also referred to as creating a coordinate system with the three-point plot method, involves taking the first point as the system origin; the X axis is a line from the first point to the second point. The third point falls on the XY plane, with the Z axis perpendicular to the XY plane, satisfying the right-hand rule. This method is generally adopted when a high-precision frame is required, such as when moving the robot relative to a table.





- Click [Plane]  to create a new coordinate system.
- Click the input box to rename the newly created coordinate system.
- Click  to expand the list, select waypoints one by one, click [Edit], control the robot using the position/pose control button, complete necessary settings, and then click [OK] to save the data and return to the [Coordinate System] interface.

rob1

Simulation

Manual

PowerOff

Normal

2024-05-19 23:22:49

Home

Program

Configure

Move

I/O

100%

0mm/s

???

Generally

Coordinate

New

Open

Save

CONFIGURE default*

Point

Line

Plane

Delete

1 Base

2 Tool

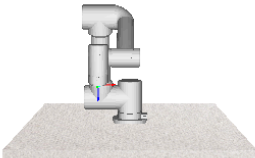
3  Plane_0

4  Point_0

5  Point_1

6  Point_2

Robot



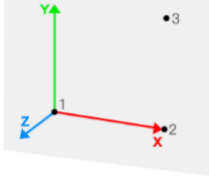
Plane_0

instance:

1. origin

2. positive X direction

3. XY plane



Payload and TCP

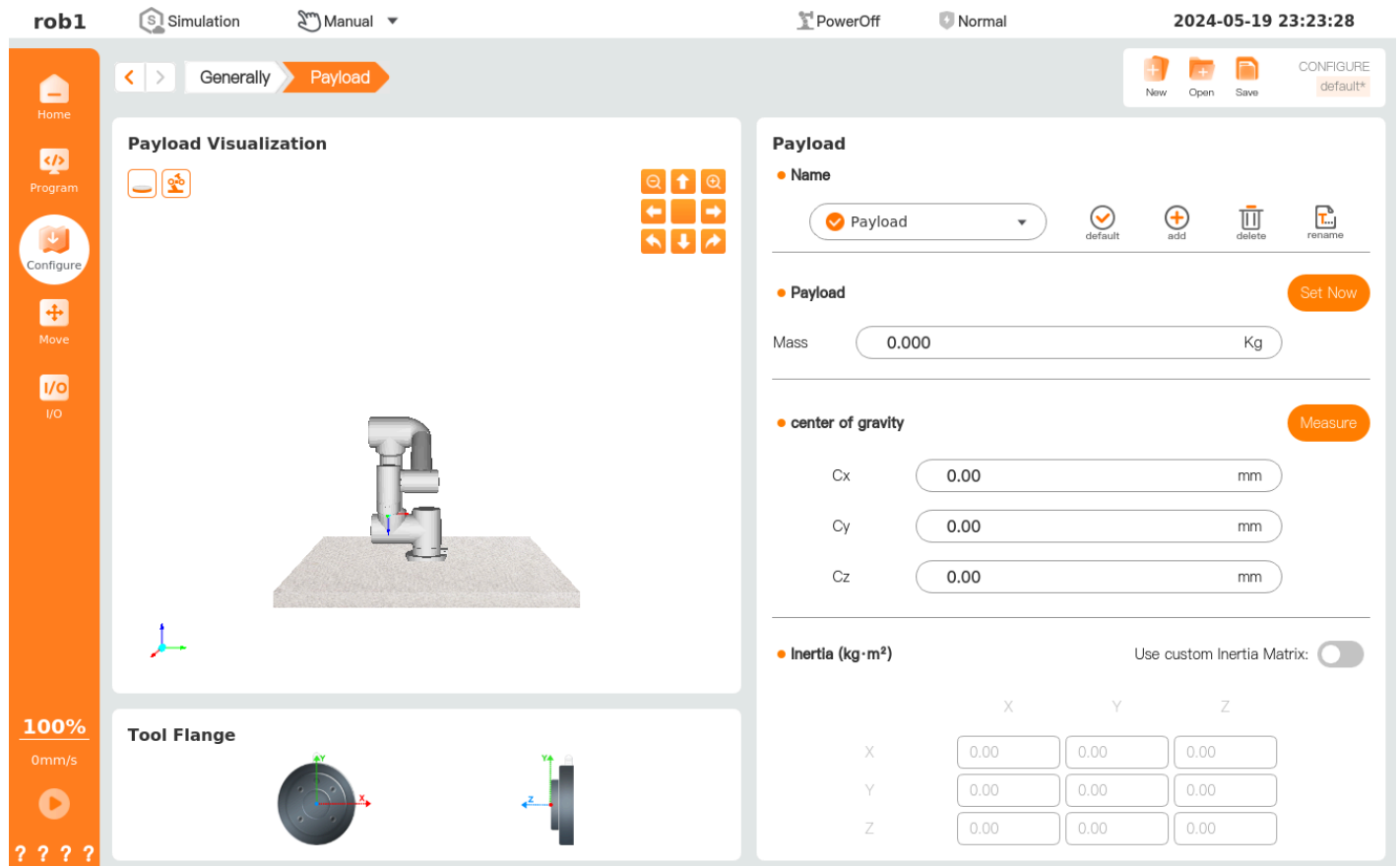
Payload

The payload encompasses the mass of the end load as well as its center of gravity (CoG). When the robot is loaded and unloaded, the robot end load and the CoG will change. To optimize the robot's performance, it is necessary to set the end load and CoG accordingly. The moment of inertia is measured in a specific coordinate system originating at the CoG of the payload, with the axis aligned with the axis of the tool flange. The default inertia can be calculated based on the inertia of a sphere with a user-specified mass and a mass density of 1 g/cm³.

The [Payload] page allows defining or switching a payload. You can define more than one payload, at least one. Each payload must contain parameters such as load mass and CoG to maximize the performance of the robot.

Info

- After the setting is completed, it is recommended to click the [Set Now], [Default], and [Save] buttons successively to ensure that the currently set payload parameters can be used during the move teaching and programming.
 1. The payload parameters determined as [Default] are to be used during programming.
 2. The payload parameters determined as [Set Now] are to be used during move teaching.
 3. The payload parameters determined as [Default] in the installation profile are to be used after shutdown and restart.
 - Payload setting priority: Payload set in [Programming] > Payload set in "Configuration > General > Load" > Payload set in "Home > Robot status" functional area.
-
- Load visualization: Simulate the status of the robot with the current settings.
 - Payload: Set up payload, CoG, etc.
 - Tool flange: Display the center point of the tool flange.



- Current default: The current payload.
- Default: Set to the default payload.
- Add: Add a new payload.
- Rename: Rename the currently selected payload.
- Delete: Delete the selected payload.
- Set Now: Make the currently set payload effective immediately.
- Measurement: Payload identification wizard, which can guide the user to have the robot automatically calculate the payload and CoG.

Payload setting

- Method 1: Manually enter values for the payload and CoG.
- Method 2: Use the payload identification wizard to determine the payload and CoG.

TCP

Tool Center Point can be abbreviated as "TCP." Due to the varying shapes and sizes of tools fixed at the end of the robot, such as a welding torch or gripper, it is necessary to select a point representative of the entire tool and take this point as the origin of the tool coordinate system.

When moving the robot to a point in space, the essence is to move the TCP to that point, so it can be said that the movement of the robot is equivalent to that of the TCP.

The four-point plot method is usually used to plot four positions to determine the tool coordinate system, including the origin of the system, by changing the joint angle of the robot while the position of the EOAT remains unchanged.



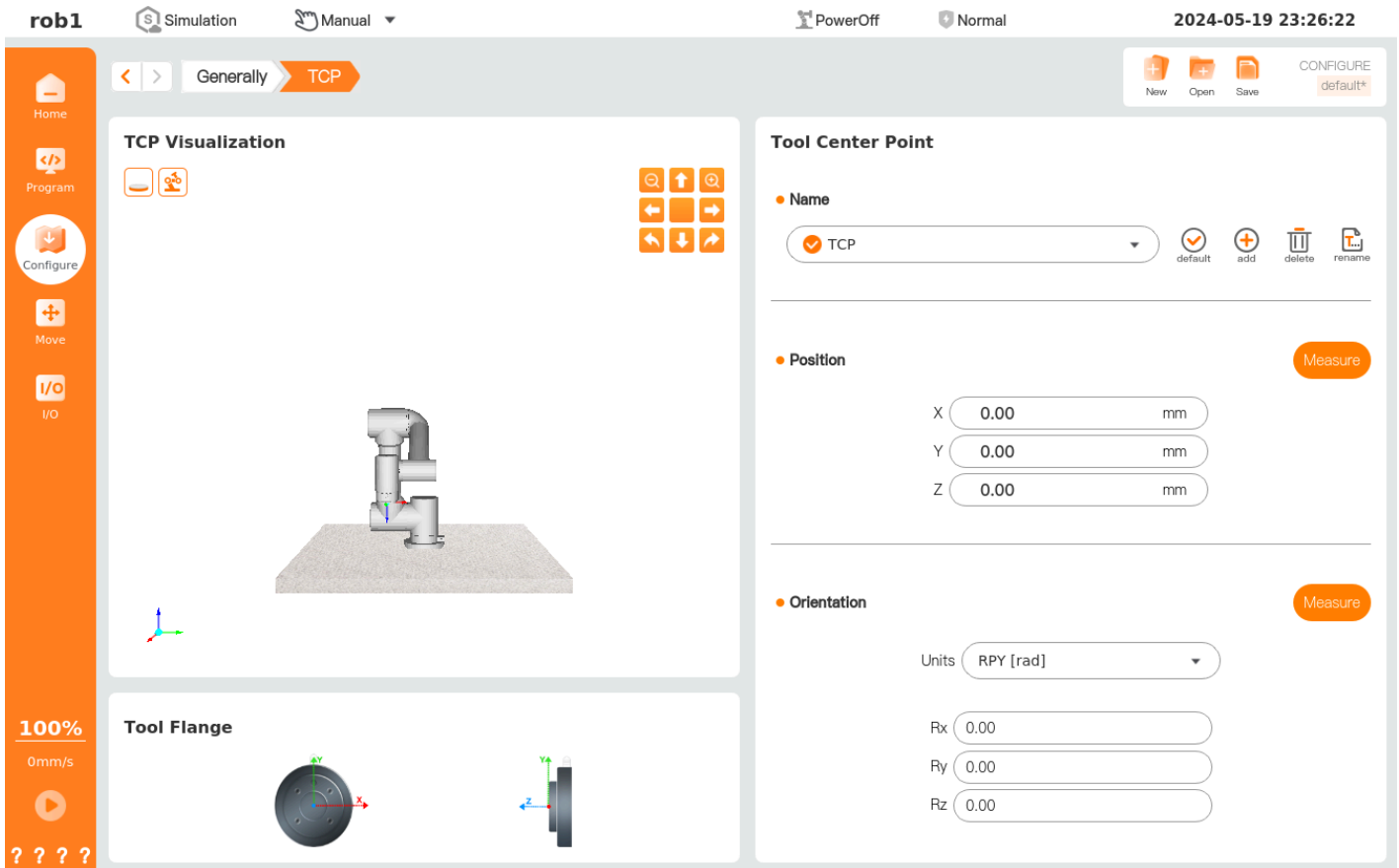
The direction of the tool coordinate system is plotted using the coordinate system of the target and the point-on-plane method.



[TCP] page allows setting of tool coordinate system. Each TCP contains translation and rotation parameters set relative to the center of the tool flange: X, Y, and Z coordinates specify the TCP position; RX, RY, and RZ coordinates define the TCP direction. When all values, including direction, are zero, TCP coincides with the center point of the tool flange.

- TCP visualization: Simulate TCP with the current settings.

- TCP: Set up and manage TCP.
- Tool flange: Display the center point of the tool flange.



- Current default: The current default TCP.
- Default: Set to the default TCP.
- Add: Add a new TCP.
- Rename: Rename the selected TCP.
- Delete: Delete the selected TCP.
- Location-Measurement: TCP location recognition wizard, which can guide the user to have the robot automatically calculate the TCP location.
- Direction-Measurement: TCP direction recognition wizard, which can guide the user to have the robot automatically calculate the TCP direction.

TCP setting

- Method 1: Manually enter values for positions X, Y, and Z and directions Rx, Ry, and Rz.
- Method 2: Use the TCP location/direction identification wizard to determine the location and direction of TCP.

User Authority

[Authority] provides users with the function of managing the authorities of each function module in the software, which is divided into two modes: basic and advanced modes:

- Classic mode: Log in to the software without a password. This mode provides three password settings and modification functions: admin, operating mode, and security. Different passwords manage the authorities of different functions.

Note:

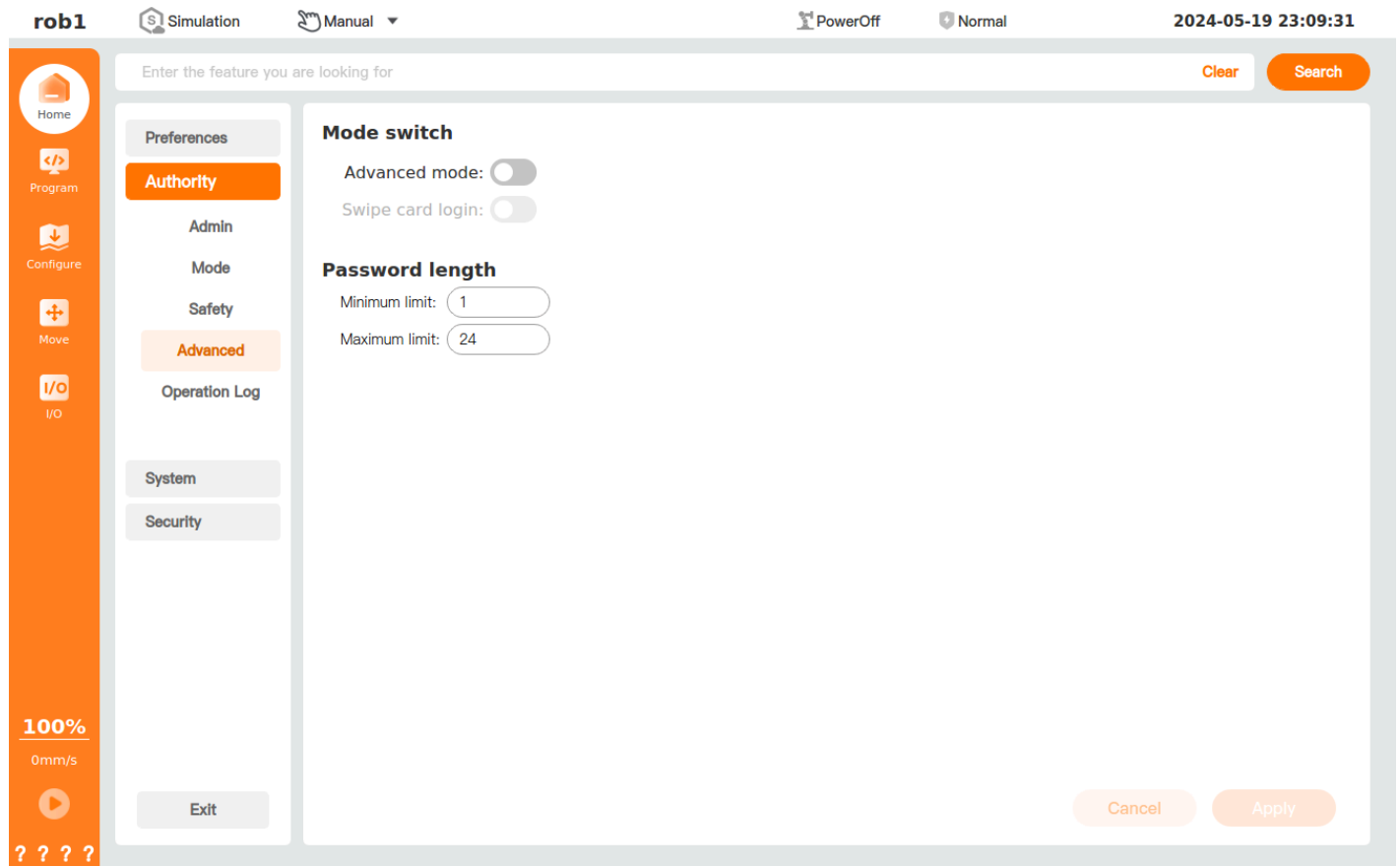
When the operating mode password has not been set:

- In local mode, only [Manual mode (Limited)] is available; [Manual mode] and [Automatic mode] cannot be used.
 - The system will display a [Operating mode password not set] prompt to the user.
- Advanced mode: Provides more detailed user management functions, allowing users with different authorities to log in and to operate the software within their authorities. Admin can create, set, or cancel accounts after logging in, and the authorities of other accounts are assigned by the admin account.

AuboStudio works in classic mode by default. You can switch to the advanced mode in the [Advanced] page after setting the admin password. When using this feature, please ensure that the user authority is properly configured and managed according to the instructions in the relevant documentation or official guides.

Mode switching

The [Advanced] page allows users to turn on or off the advanced mode and the swipe card login.



- Advanced mode: Click   to turn on or off the "Advanced mode." You must enter the admin password each time you switch the mode. The mode will be enabled after the admin account is authenticated successfully.
- Swipe card login: Click   to turn on or off the "Swipe card login." This function needs to be used in conjunction with the card reader. If you turn on the "Swipe card login" function, you can simply swipe your card to log in to the software.

Software Update

Software update interface

AuboStudio supports both online and offline software updates. For offline updates, first import the installation package into the controller, then access this interface for installation and upgrades.

In the [Software Update] page, you can switch software versions, manage software packages, and update software, etc.

The screenshot displays the 'Update robot software' interface. At the top, there's a header with 'rob1', 'Simulation', 'Manual', 'PowerOff', 'Normal', and a timestamp '2024-05-19 23:17:10'. Below the header, a search bar is present. The left sidebar contains navigation icons for Home, Program, Configure, Move, I/O, and a status indicator '100% 0mm/s'. The main content area is titled 'Update robot software' and includes a 'Version Number' dropdown set to '0.29.1-rc.7', a 'Switch' button, and a 'Search' button. A table lists software packages with columns for Status, Name, Download, and Local. The table contains 10 rows of data, each with a 'Cloud' status, a 'Software' type, and a specific package name. Each row has a 'Download' link and an 'Install' button. Below the table is a 'Description' section with a 'Patch update log' link. The bottom of the sidebar has an 'Exit' button.

Status	Name	Download	Local
Cloud	Software	arcs_update-0.29.0-alpha.113+f151200.nightly-x86_64.run	Download Install
Cloud	Software	arcs_update-0.29.0-alpha.110+b674f6c.nightly-x86_64.run	Download Install
Cloud	Software	arcs_update-0.29.0-alpha.109+3b39c9e.nightly-x86_64.run	Download Install
Cloud	Software	arcs_update-0.29.0-alpha.108+7357936.nightly-x86_64.run	Download Install
Cloud	Software	arcs_update-0.29.0-alpha.101+eacc0f1.nightly-x86_64.run	Download Install
Cloud	Software	arcs_update-0.29.0-alpha.93+6be7c1c.nightly-x86_64.run	Download Install
Cloud	Software	arcs_update-0.29.0-alpha.92+07df539.nightly-x86_64.run	Download Install
Cloud	Software	arcs_update-0.29.0-alpha.84+00076ac.nightly-x86_64.run	Download Install
Cloud	Software	arcs_update-0.29.0-alpha.83+2a65463.nightly-x86_64.run	Download Install

- Switch: Select the software version number in the drop-down menu, click [Switch], and restart the system to switch the software to the target version.
- Refresh: Refresh the list of software packages.
- Search: Input in the search box to search the software package.
- Filter
 - Status: Search the software package by status.
 - Name: Search the software package by name.

- Pre-Release: Display the software version under test. The pre-release provides the latest functional experience, but there may be some bugs, so please use it with caution.
- Download: Download the software package.
- Install: Install the software package.
- Delete: Delete the software package.
- Uninstall: Uninstall the software package.
- Patch update log: If a patch pack is installed, click here to view the update log for the current patch pack.

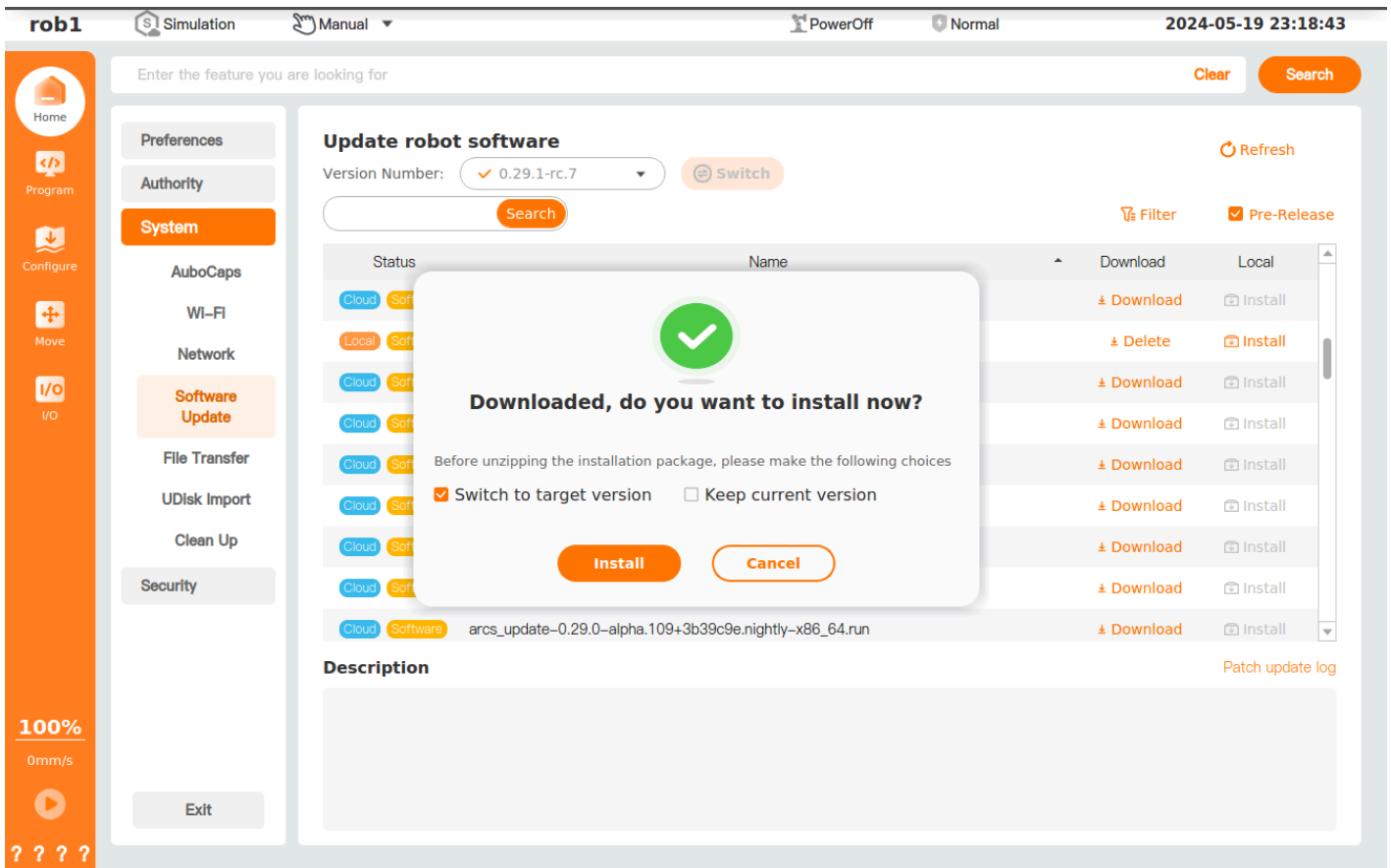
Software version online update

For online updates, you must ensure that the controller is plugged in with a network cable that can connect to the external network. Otherwise, the software can only be updated offline.

Note:

Before installing a new version, the system will automatically perform a resource check. If the local memory does not meet the minimum requirements for software operation, the installation will be terminated and a [System memory insufficient, unable to install] prompt will be displayed.

1. In the [Software Update] page, all currently updateable installation packages are displayed in the installation package list. If the installation package of the target version is not found, you can try to click [Refresh] to refresh the list.
2. Click [Download] of the target software package (`.run` file) to enter the [Download] page to download the installation package.
3. After downloading the software package, the [Downloaded] prompt box pops up. "Switch to target version" means that after the installation is completed, you can restart the controller to switch to the new version; "Keep current version" means that you can keep the current version. After choosing as required, click [Install] to enter the [Install] page for installation.
4. If "Switch to target version" is chosen, you can restart the controller to switch to the new version. Click "Home > About" to view the current version information. If the version number is correct, the version update is successful.



Software version offline update

1. Download the software package (`.run` file) on the official website. Download link:
 - C Series / iS Series / Advanced controller: <https://download.aubo-robotics.cn/arcs/>
 - S Series controller: <https://download.aubo-robotics.cn/arcs/arm64/>
2. Copy the installation package to a USB flash drive first, and then import it into the control box. For detailed instructions, please refer to [Import from USB Flash Drive](#).
3. Enter the [Software Update] page, click [Filter], add [Local] as the [Status] filter condition in [Filter], and click [OK]. Find the uploaded software package (`.run` file), and click [Install] to enter the [Install] page for installation.
4. If "Switch to target version" is chosen, you can restart the controller to switch to the new version. Click "Home > About" to view the current version information. If the version number is correct, the version update is successful.

Switching software version

1. After the software installation is completed, you can select the software version in the "Switch" drop-down menu, click [Switch], and the system pops up a window to prompt you to shut down or restart.
 - Click [OK] to restart the controller. After restart, the target version is applied.
 - Click [Cancel] to exit the pop-up window, and the version switching is canceled.
2. After switching the version and restarting the controller, you can click "Home > About" to view the current version information. If the version number is correct, the version update is successful.

Software Security Password

Configurations on the [Security] page include security I/O, joint limit, robot limits, safe home, etc., which can be unlocked only with the security password. For the setting method of the security password, please refer to "[Classic mode](#)."

Tip

1. Any changes made to the configurations on the [Security] page will not take effect until users click on [Apply] and subsequently re-power the robot following a power-off.
2. All configuration in [Safety] is saved in the installation profile. Users must click the [Save] button to save all configuration.

Security configuration unlock/lock

1. Click [Configure] in the menu bar on the left side of the screen, and click [Security] to enter the page with an [Unlock] popup.
2. Enter the security password in the "Security password" input box and click [Unlock] to enter the [Security] function configuration page.

After the page is unlocked, all configurations under the [Security] menu are unlocked.

3. When exiting the [Security] configuration page, all configurations under the [Security] menu will be locked.

rob1

Simulation

Manual

PowerOff

Undefined

2024-05-19 18:59:06

Home

Program

Configure

Move

I/O

100%

0mm/s

?

?

?

?

Generally

Safety

Fieldbus

Plug-In

apply

New

Open

Save

CONFIGURE

default_4*

I/O

I/O

Joint Limits

Joint Limits

Robot Limits

Robot Limits

Safe Home

Safe Home

ThreePosition

ThreePosition

Tool Position

Tool Position

Planes

Planes

ObstacleMarker

ObstacleMarker

Change Safety Configuration

Safety configuration settings must be changed based on the integrator's risk assessment results.

Safety configuration change process:

1. Ensure that changes comply with the risk assessment performed by the integrator.
 2. Adjust safety configuration in accordance with risk assessment requirements.
 3. Confirm that the new safety configuration has been applied (which can be verified using a safety checksum).
-

Apply safety configuration

The robot will be powered off when the safety configuration is changed.

After changing the safety configuration, click the [Apply] button. The system will then display a pop-up window prompting users to confirm the changes and show the current safety configuration information for users to check. Once confirmed, the robot will be powered off, and the changes will be automatically saved to the current safety configuration file.

Safety verification code

The verification code is displayed in the lower left corner of the AuboStudio interface. The verification code is updated when safety configuration changes. Click the verification code to view the details of the currently valid safety configuration.

rob1

Simulation

Local Mode

PowerOff

Undefined

Home

Welding

Configure

Move

I/O

IP: 127.0.0.1

Port: 30000

Disconnect

Confirm if Configure and Payload are correct and press OK

Robot Limits

Joint Position

Joint Speed

Planes

ObstacleMarker

Tool Position

I/O

Safe Home

Limit	Normal	Reduced
Power	800.00 W	800.00 W
Momentum	8.00 kg · m/s	8.00 kg · m/s
Stopping Time	500.00 ms	500.00 ms
Stopping Distance	400.00 mm	400.00 mm
Tool Speed	2000.00 mm/s	2000.00 mm/s
Tool Force	200.00 N	200.00 N
Elbow(joint3) Speed	1000.00 mm/s	1000.00 mm/s
Elbow(joint3) Force	100.00 N	100.00 N

Close

Robot

100%

0mm/s

Day

Hour

Min

Sec

0

:

00

:

00

:

00

?

?

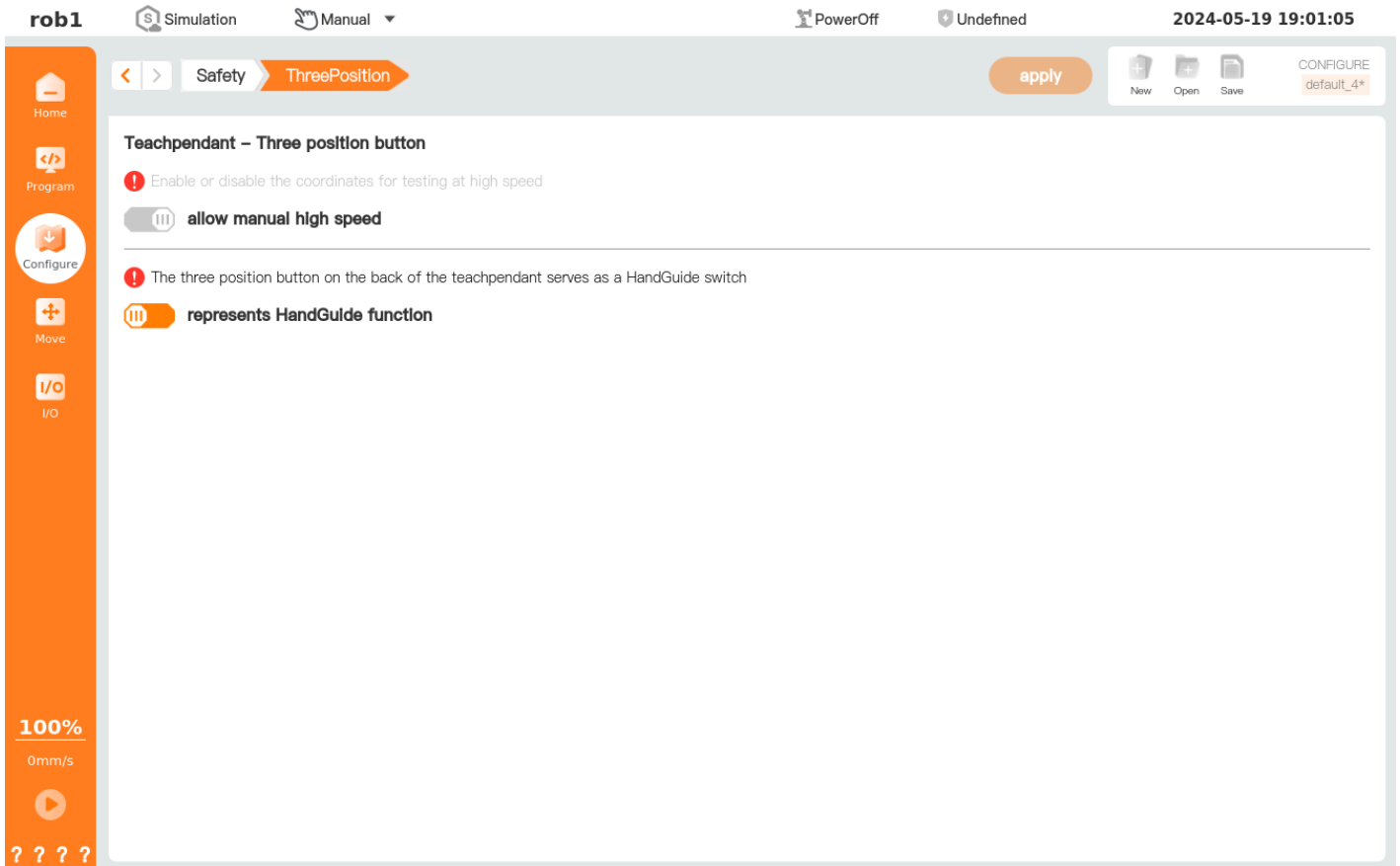
?

?

Safety Restriction

ThreePosition

In the [ThreePosition] page, users can set the function of the three-position button.



Joint limit

In the [Joint Limits] page, users can set the maximum limit that each joint of the robot arm can reach so as to limit the movement of each joint of the robot arm in the joint space. On this page, users can set the rotation angle of each joint and the joint max speed limit in "normal mode" or "reduced mode."

1. When the "reduced mode" and "normal mode" are set at the same time, the value in "reduced mode" shall not be greater than that in "normal mode"; otherwise, the system will prompt a save failure.
2. The default values of joint maximum value, joint minimum value, and joint max speed limit are the maximum ranges and can only be modified by the user within the default ranges.

rob1

Simulation

Manual

PowerOff

Undefined

2024-05-19 18:57:42

Home

Program

Configure

Move

I/O

100%

0mm/s

???

Safety

Joint Limits

apply

New

Open

Save

CONFIGURE default_4*

Joint Limits

Joint	Range		Normal Mode		Reduced Mode		
	Minimum	Maximum	Minimum	Maximum	Minimum	Maximum	
Joint1	-360.00°	360.00°	-178.00	178.00	-178.00	178.00	+2°/-2°
Joint2	-175.00°	175.00°	-175	175	-175	175	+2°/-2°
Joint3	-162.00°	162.00°	-162	162	-162	162	+2°/-2°
Joint4	-175.00°	175.00°	-175	175	-175	175	+2°/-2°
Joint5	-175.00°	175.00°	-175	175	-175	175	+2°/-2°
Joint6	-360.00°	360.00°	-178.00	178.00	-178.00	178.00	+2°/-2°

Max Speed

Joint	Maximum	Normal Mode	Reduced Mode
Joint1	147.00°/s	147	147 -11°/s
Joint2	147.00°/s	147	147 -11°/s
Joint3	147.00°/s	147	147 -11°/s
Joint4	178.00°/s	178.00	178.00 -11°/s
Joint5	178.00°/s	178.00	178.00 -11°/s
Joint6	178.00°/s	178.00	178.00 -11°/s

Robot limit

On the [Robot Limits], users can set the limits for robot hardware parameters, including power, momentum, stopping time, stopping distance, tool speed, tool force, elbow (joint 3) speed, and elbow (joint 3) force. The system provides several preset values, and the users can also use custom settings as needed. However, the safety configuration parameters are different from those defined in the risk assessment, which may lead to hazards not being reasonably eliminated or risks not being sufficiently reduced.

rob1

Simulation

Manual

PowerOff

Undefined

2024-05-19 18:58:49

Home

Program

Configure

Move

I/O

100%

0mm/s

?

?

?

?

Safety

Robot Limits

apply

New

Open

Save

CONFIGURE default_4*

Robot Limits

Use of Safety Configuration parameters different from those defined by the risk assessment can result in hazards that are not reasonably eliminated or risks that are not sufficiently reduced.

Most Restricted

More Restricted

Less Restricted

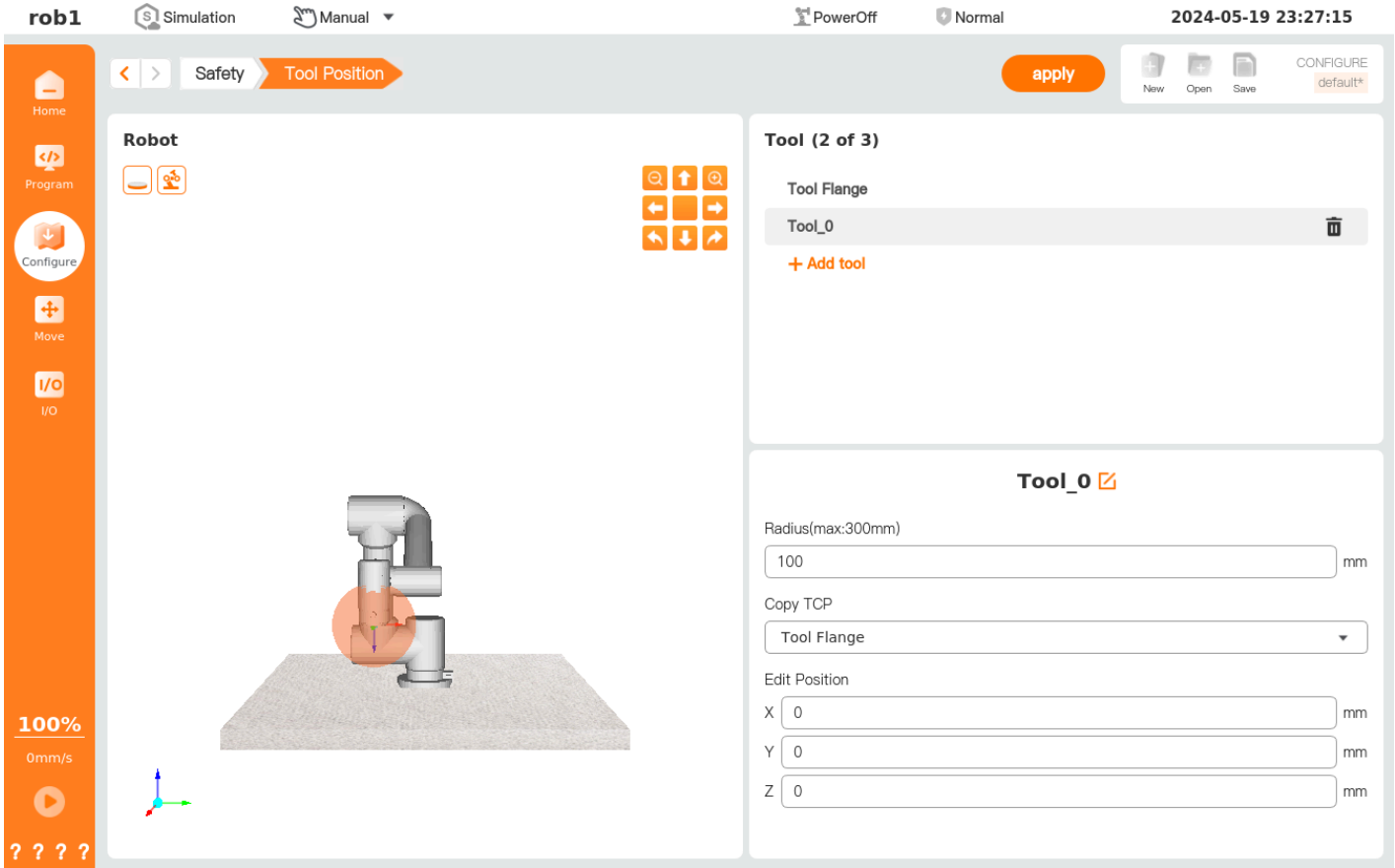
Least Restricted

Custom

Limit	Normal		Reduced	
Power	800.00	W	800.00	W
Momentum	8.00	kg·m/s	8.00	kg·m/s
Stopping Time	500.00	ms	500.00	ms
Stopping Distance	400.00	mm	400.00	mm
Tool Speed	2000.00	mm/s	2000.00	mm/s
Tool Force	200.00	N	200.00	N
Elbow(joint3) Speed	1000.00	mm/s	1000.00	mm/s
Elbow(joint3) Force	100.00	N	100.00	N

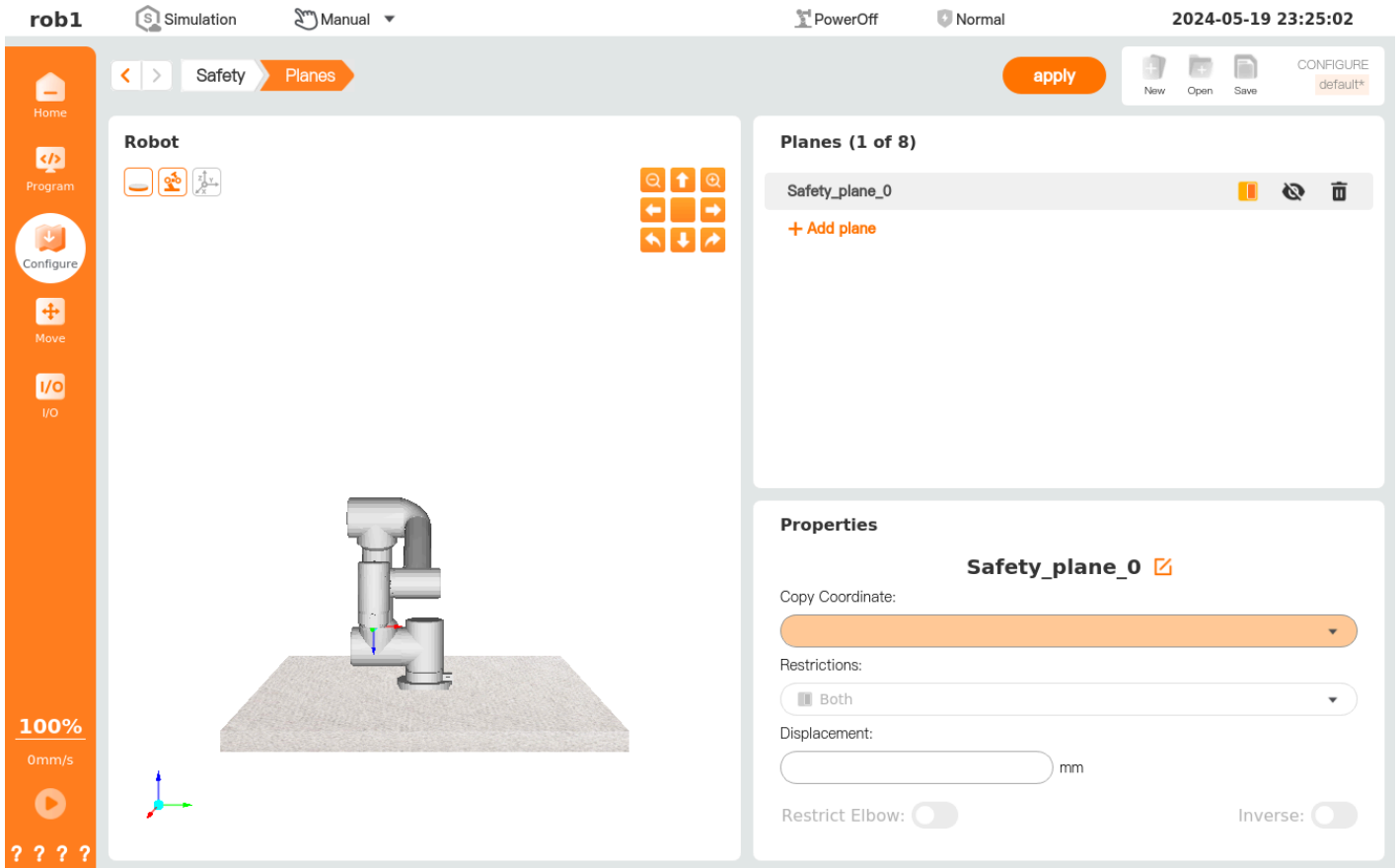
Tool position

In the [Tool Position] page, users can set the spherical envelope of end-of-arm tooling (EOAT) and simulate the shape of the tool through the spherical envelope. This feature is designed to protect the end-of-arm tooling (EOAT) by using the spherical envelope to monitor collisions as TCP approaches a safety plane or an obstacle, thereby protecting the integrity of the end-of-arm tooling (EOAT).



Plane

In the [Planes] page, users can set the safety plane.



The safety plane supports the following five restriction conditions. Upon selecting a corresponding condition, the robotic arm will trigger different safety protection behaviors based on the current operating mode (Normal Mode/Reduced Mode). Specific details are as follows:

1. **Disable:** The current safety plane is not active and provides no protection.
2. **Normal:**
 - The robot arm is currently in normal mode: when the robotic arm moves to the negative direction of the safety plane, a pop-up window appears indicating safety protection has been triggered, and the robotic arm stops moving.
 - The robot arm is currently in reduced mode: when the robotic arm moves in the negative direction of the safety plane, it can operate normally without triggering safety protection.
3. **Reduced**
 - The robot arm is currently in normal mode: when the robotic arm moves to the negative direction of the safety plane, it can operate normally without triggering safety protection.
 - The robot arm is currently in reduced mode: when the robotic arm moves to the negative direction of the safety plane, a pop-up window appears indicating safety protection has

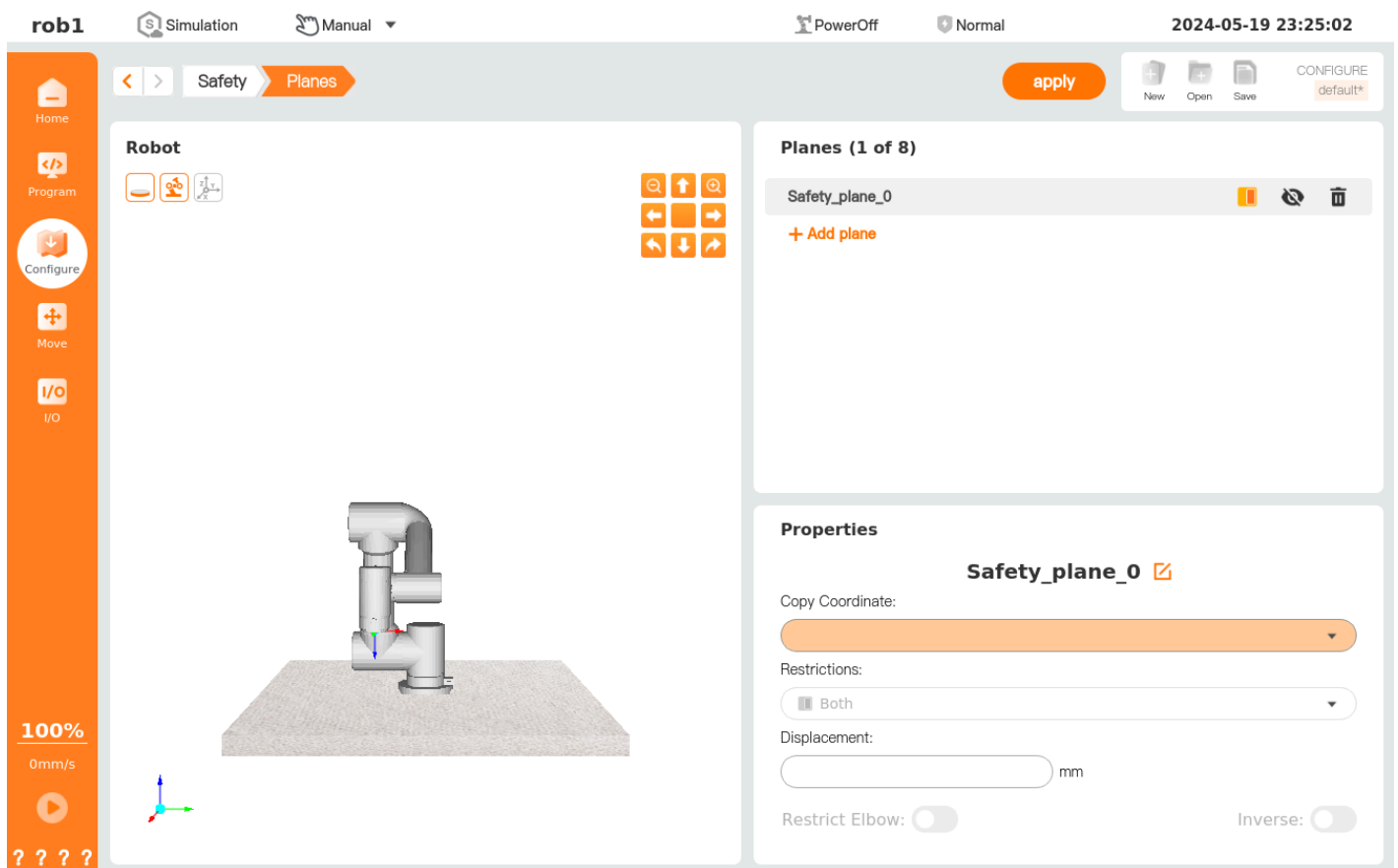
been triggered, and the robotic arm stops moving.

4. Both

- The robot arm is currently in normal mode: when the robotic arm moves to the negative direction of the safety plane, a pop-up window appears indicating safety protection has been triggered, and the robotic arm stops moving.
- The robot arm is currently in reduced mode: when the robotic arm moves to the negative direction of the safety plane, a pop-up window appears indicating safety protection has been triggered, and the robotic arm stops moving.

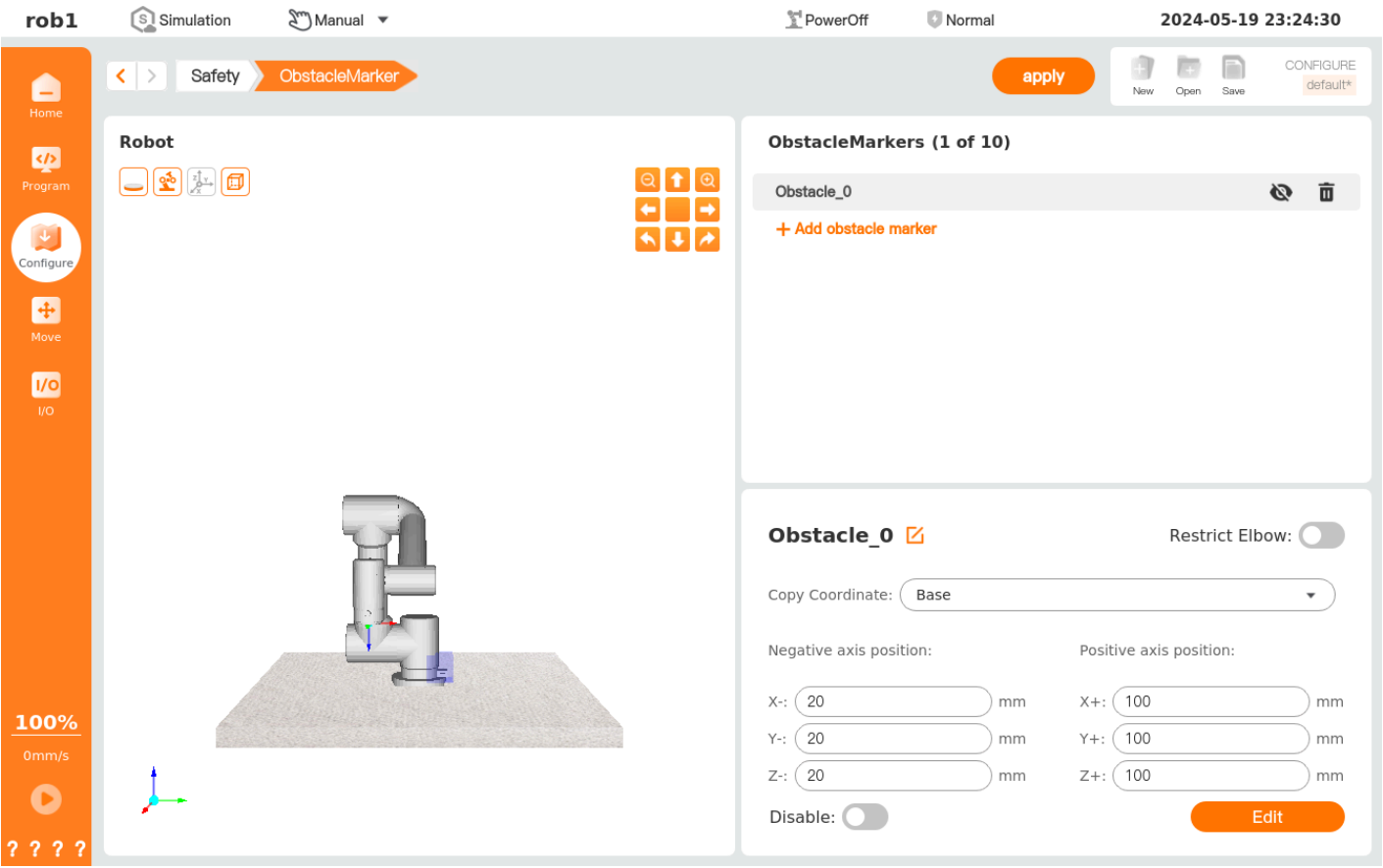
5. Reduced Trigger Mode

- The robot arm is currently in normal mode: when the robotic arm moves to the positive direction of the safety plane, the robotic arm enters reduced mode and continues to operate.
- The robot arm is currently in reduced mode: when the robotic arm moves to the negative direction of the safety plane, the robotic arm continues to operate in reduced mode.



ObstacleMarker

In the [ObstacleMarker] page, users can mark the three-dimensional space where the obstacle is located.



Safety I/O

This page allows setting of security I/O: click the security I/O drop-down menu to edit, and click [Save] in the configuration manager toolbar to save.

Info

1. Some controller models do not support security I/O customization. Please refer to the Hardware User Manual for the function definition of fixed safety I/O.
2. C-series and iS-series controller types support using unassigned safety I/O ports as ordinary I/O ports.
3. Dual-channel safety I/O ports must receive the enable signals simultaneously to activate the safety I/O function.
4. The number of controller I/Os may affect the use of some safety I/O functions. Refer to the actual product for details.

Description of security I/O functions

Function	Trigger Mode	Description	Notice
Not assigned	-	-	-
Emergency stop	Logic low	Level triggering. The level signal typically remains logic high, and "external emergency stop" will be triggered when the signal is set to logic low.	-
Safe stop	Logic low	Level triggering. The level signal typically remains logic high, and "safe stop" will be triggered when the signal is set to logic low.	It can be used together with "safe resetting."
Safe resetting	Rising edge	Edge triggering. After the "safe stop" signal returns to logic high, the signal will be changed from logic low to logic high, the rising edge will trigger "safe resetting," and the robot will release the safe stop state and continue to move.	It must be configured together with "Safe stop."
Reduced mode	Logic high	Level triggering. It will be activated when the level signal is set to logic	The reduced parameters take effect at the same

		high. After activation, the robot will enter the reduced mode, and the robot arm will move according to the reduced parameters. For detailed reduced parameters, please refer to " Joint Limit ."	time as the safety limits.
Enabling device	Logic low	Level triggering. When the signal remains logic high, the robot will operate normally. When the signal is set to logic low, the page will prompt to trigger "safe stop." When the logic high signal is restored, the robot will release the safe stop and continue to move.	* After the "Enable device" is configured, the dual I/O on the controller must receive valid signals at the same time before the robot can be controlled normally. * It is valid in manual mode.
Operating mode	Logic high: Manual mode Logic low: Automatic mode	Level triggering. The operating mode of the robot can be switched.	* If the "Operating mode" is not configured in I/O, the operating mode can be switched on the software interface. * After the "Operating mode" is configured in I/O, manual mode can only be entered via the operating mode in I/O.
HandGuide	Logic high	Level triggering. When the logic high is set, the robot will enter the HandGuide mode.	-
Safe stop of automatic mode	Logic low	Level triggering. In automatic mode, this signal typically remains logic high, and safe stop will be triggered when the signal is set to a logic low.	* It must be configured together with "Enable device." * Usually used in conjunction with "safe resetting of automatic mode." * It is valid in automatic mode.
Safe resetting of automatic mode	Rising edge	Edge triggering. In automatic mode, after the "safe stop of automatic mode" signal returns to logic high, the signal will be changed from logic low to logic high, the rising edge will trigger safety reset, and the robot	* It must be configured together with "Safe stop of automatic mode." * It is valid in automatic mode.

		will release the safe stop state and continue to move.	
--	--	--	--

Function	Output signal	Description
High when internal emergency stop is triggered	Logic high	When the robot's emergency stop device is triggered and the robot enters the emergency stop state, logic high will be set.
Low when internal emergency stop is triggered	Logic low	When the robot's inherent emergency stop device is triggered, the robot enters the robot emergency stop state and is set to logic low.
Robot movement	Logic high	When the robot arm position/pose changes, logic high will be set.
The robot does not move	Logic high	When the robot arm position/pose has no changes, logic high will be set. * When the motor speed (excluding reducer) is within ± 5 RPM, the robot arm position/pose will be deemed to have no changes.
Reduced mode	Logic high	When the robot is in the reduced mode, logic high will be set.
	Logic low	When the robot is not in the reduced mode, logic low will be set.
Non-reduced mode	Logic high	When the robot is not in the reduced mode, logic high will be set.
	Logic low	When the robot is in the reduced mode, logic low will be set.
The robot does not stop	Logic high	When the system emergency stop, safe stop, or protective stop is triggered, the output port will be set to logic high from the time the system receives the stop signal until the signal is released.
Safe home	Logic high	When the robot is in the safe home position, logic high will be set.

Safe Home

In the [Safe Home] page, you can set the safe home. When the robot is in the safe home position and the safe home I/O is defined, the output signal of the safe home will be valid. You can define a "home" as a "safe home." Please refer to [Home](#) for the settings of "home."

The screenshot displays the AUBO Developer software interface for configuring the 'Safe Home' position. The top navigation bar shows the robot name 'rob1', mode 'Simulation', and a 'Manual' dropdown. Status indicators for 'PowerOff' and 'Normal' are present, along with a timestamp of '2024-05-19 23:26:06'. The left sidebar provides quick access to 'Home', 'Program', 'Configure', 'Move', and 'I/O' functions. The main workspace is split into two sections. The 'Robot' section on the left features a 3D visualization of the robot arm. The 'Safe_home' section on the right contains configuration options. It includes a note that deleting the safe home position will unassign any safe home output. A red warning icon indicates that the 'Safe home position not synchronized with home position'. Below this, six joint position inputs (Joint1 through Joint6) are shown, each with a value of '0.00'. A 'Sync from home' button is located at the bottom of this section. The bottom of the interface shows a '100%' zoom level and a '0mm/s' speed indicator.

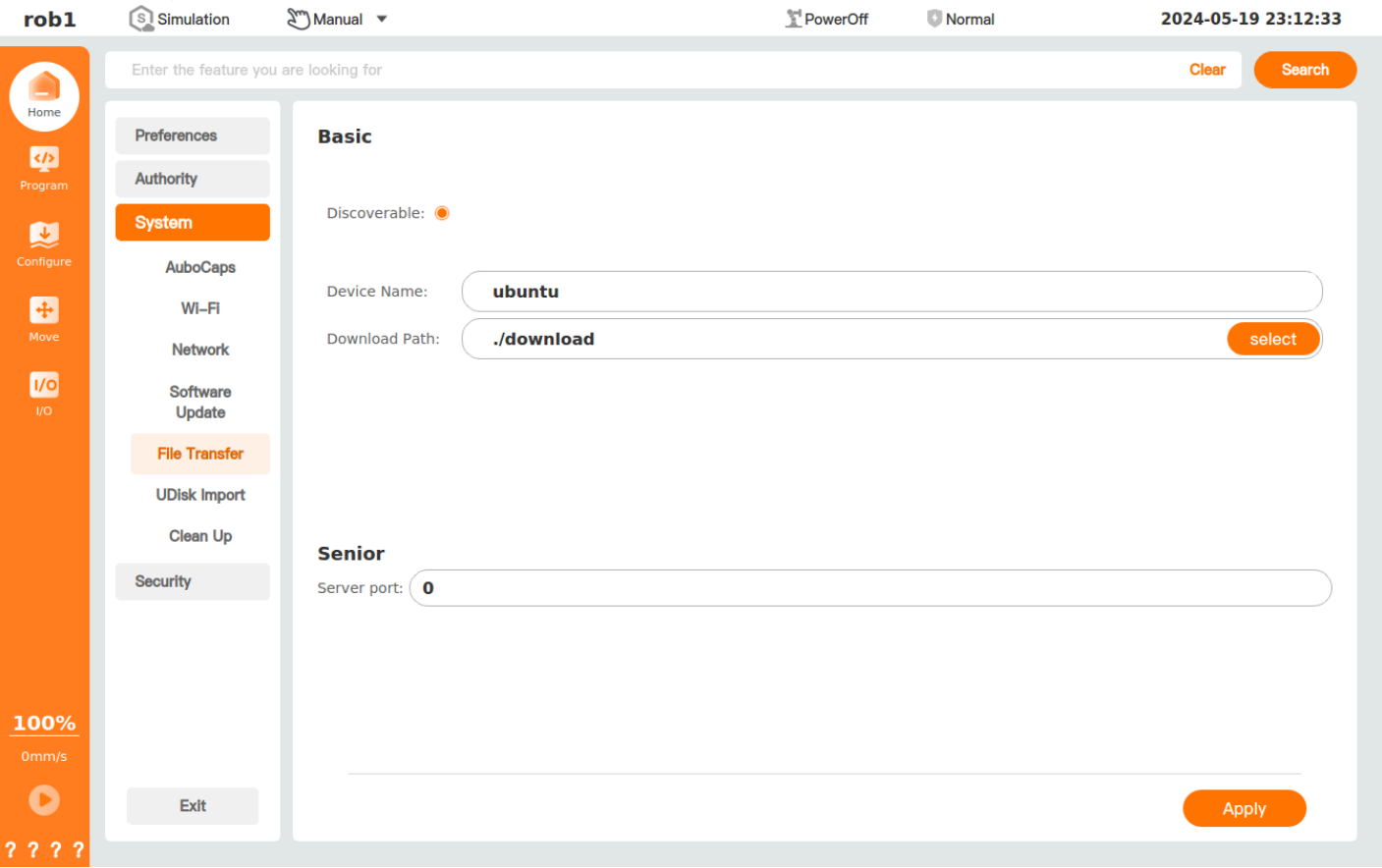
- Home sync: The current "home" is defined as the "safe home."

Program and File Management

File transfer

On the [File Transfer] page, you can configure the device name and download path so that the system can transfer files between different devices within the same LAN.

Before using this function, please turn on the [Network] function.

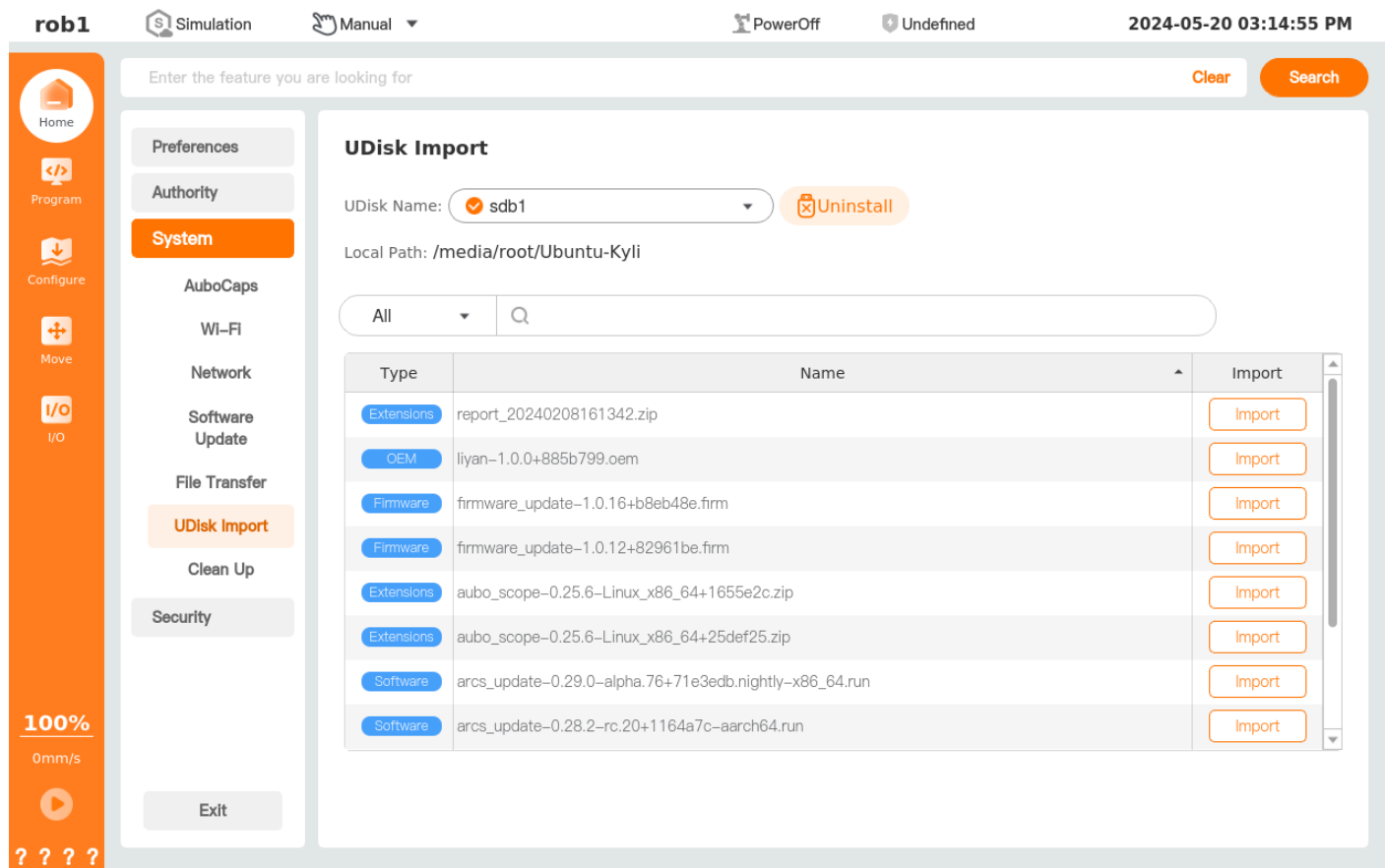


- Discoverable: Set whether this device can be discovered by other devices.
- Device Name: Set the device name for this system.
- Download Path: Set the download path for importing external device files into this system.
- Server port: Set the server port.
- Apply: Click to apply the settings.

UDisk import

On the [UDisk Import] page, users can import files into the system via a USB storage device. The system downloads the files to the corresponding directory according to the imported files.

For example, a software package is imported into the software package directory of the system, and users can view the imported software package in the installation package list in "Setting > System > Software Update."



- Uninstall/Mount: Mount and uninstall the U disk. After the U disk is inserted into the controller and recognized, you need to click [Mount] to operate on the files in the U disk. Before unplugging the U disk, you need to click [Uninstall] to safely unplug the U disk to avoid damaging the data by forcibly pulling the U disk.

The U disk cannot be uninstalled when it is in use.

- [All] dropdown box: Filter the files on the U disk according to the file type.
- Search: Search for files on the U disk.
- Import: Import the files of the U disk into the controller.

UDisk import

1. After the U disk is inserted into the controller, the system automatically recognizes the U disk. Select a USB device from the drop-down menu on the [UDisk Import] page, and the

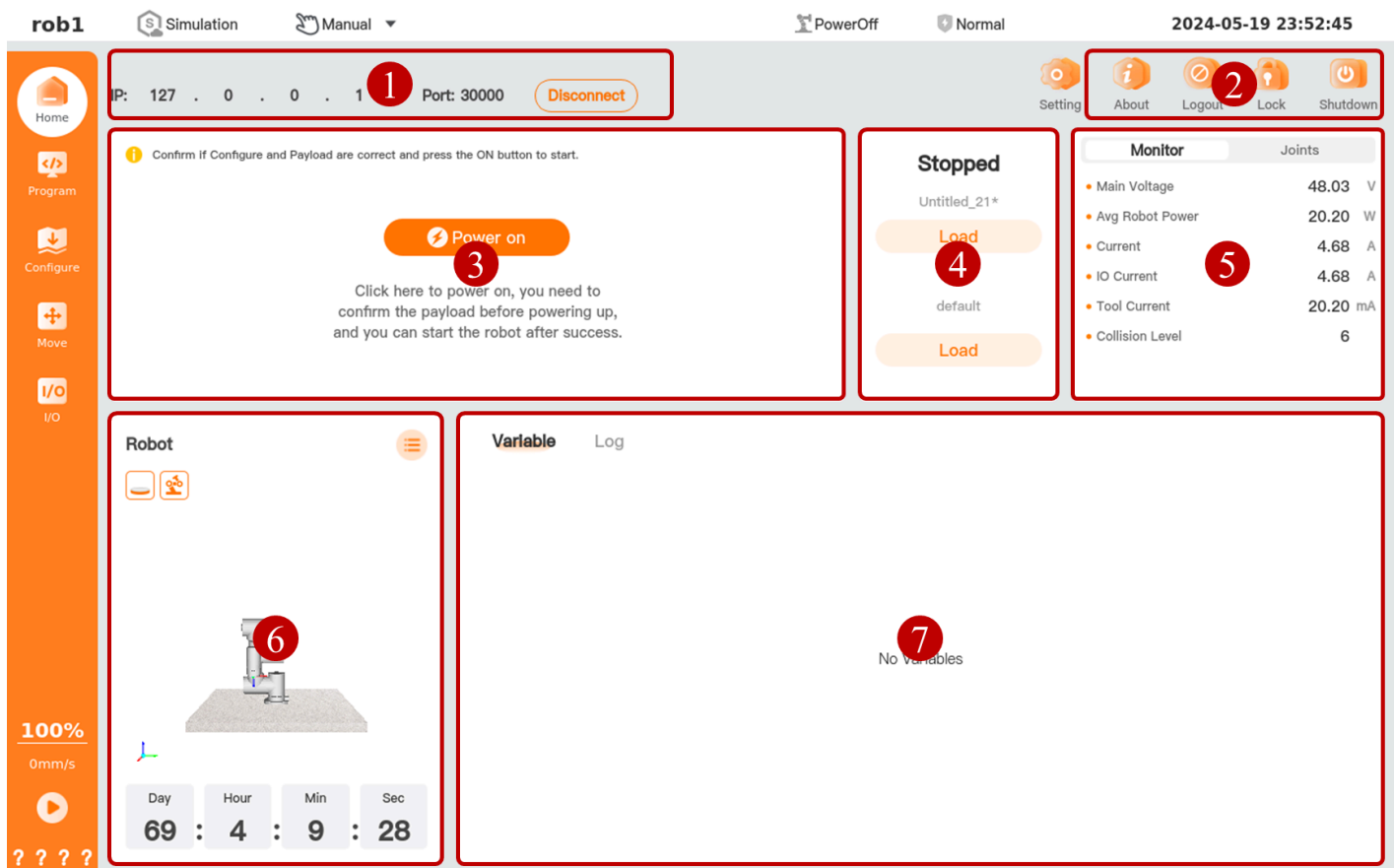
software package, installation configuration, and other files under the root directory of the U disk are displayed.

2. Click [Mount]. When the U disk is marked as , it indicates that the U disk is mounted.
3. Click [Import] corresponding to the file in the list to import the file into the controller.
4. Perform specific operations under the function corresponding to the imported file type.
 - In the [Program] page, users can click [Open] to display the imported project file.
 - In the [Configure] page, users can click [Open] to display the imported installation profile.
 - On the "Home > Setting > System > Software Update" page, the imported installation packages are displayed.
 -
5. After the file transfer is completed, please click [Uninstall] to uninstall the U disk and then pull it out.

Home Interface Description

In page [Home], you can quickly start and operate the robot. The page Home consists of 7 functional areas:

Robot Connection Description



Serial number	Name	Description
1	Robot connection	Display and switch the robot's connection. Note: When the robot arm usage exceeds the warranty period, users will be reminded [Robot arm warranty has expired].
2	System control buttons	Other settings and information.
3	Robot status	Set and display the robot status.
4	Profiles and programs	Load the running programs and installation profiles of the robot.

5	Monitor and joint	Display the status of the controller and the robot's joints.
6	Robot	Display the age and simulation model of the robot, and view robot version information.
7	Variables and log	Variables: display the variable value of the running program. Log: Display Controller System Log. For more information about logs, please refer to Log - View Log and Runtime Records .

Running Project

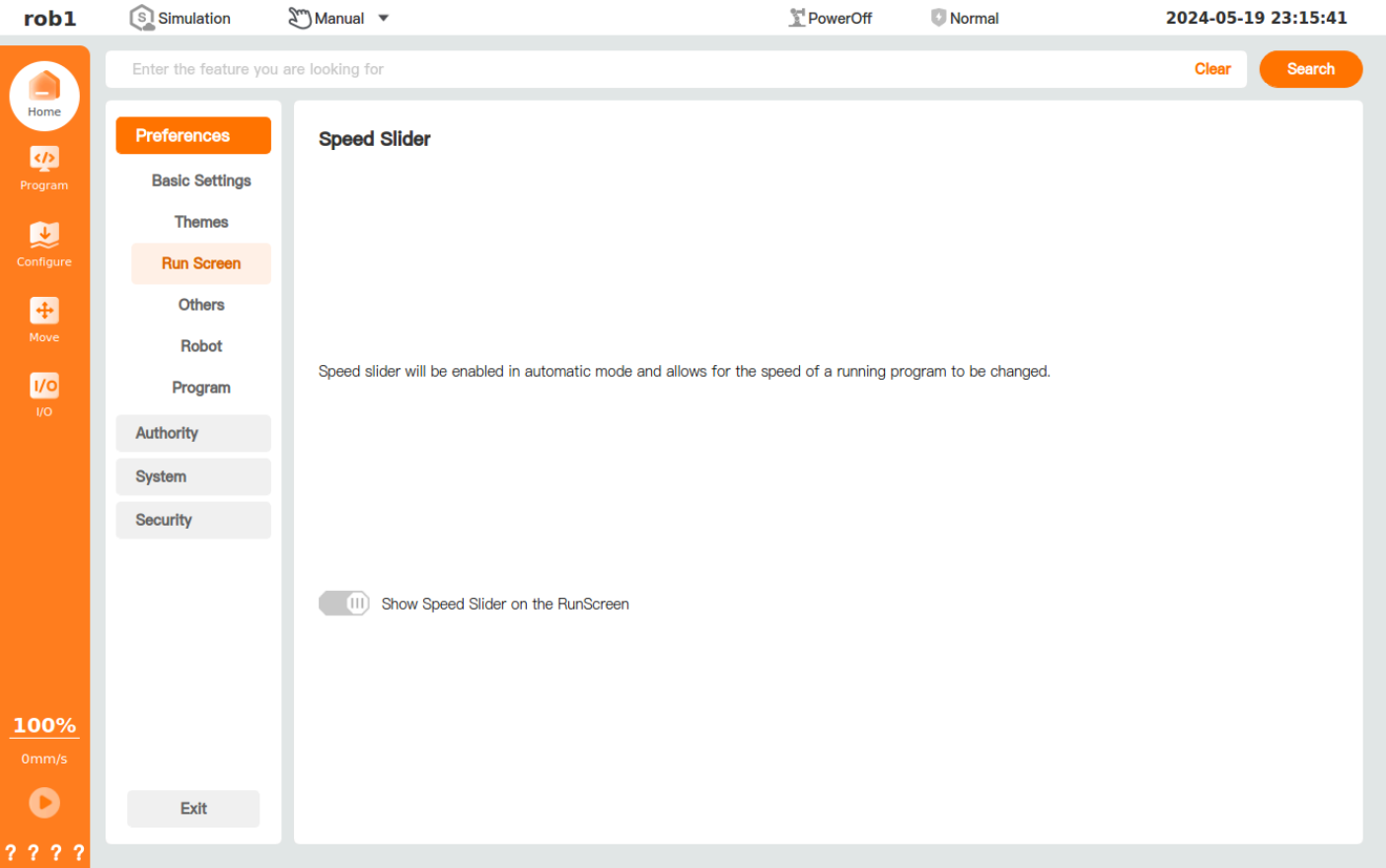
Run screen

In the [Run Screen] page, you can turn on or off the movement speed slider of [Speed].

In automatic mode, if this function is turned on, click [Speed]

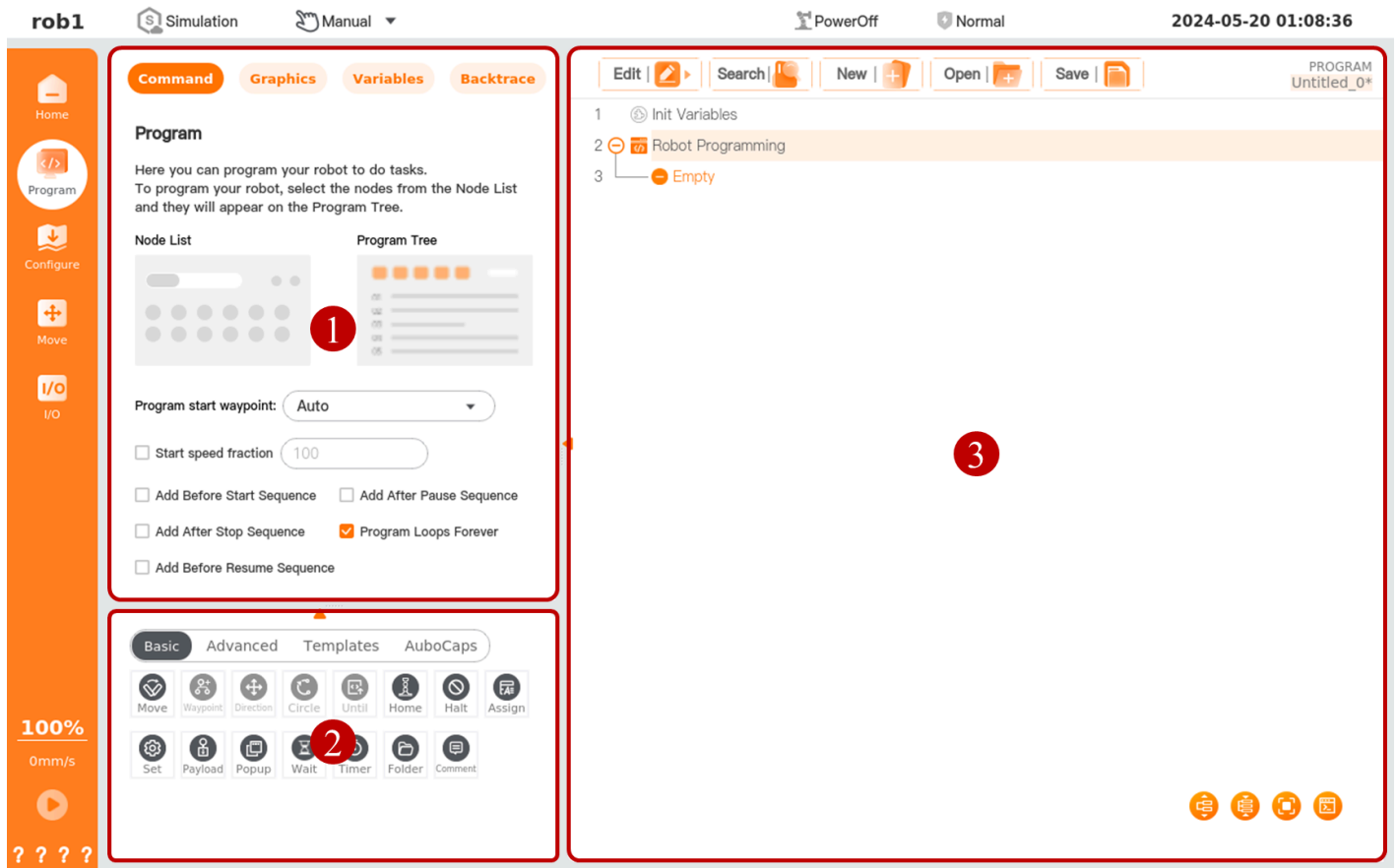
30%
0 mm/s

 at the bottom of the left menu bar, and the movement speed slider will appear; if this function is turned off, the movement speed slider will not appear, i.e., the movement speed cannot be adjusted.



Programming Interface Description

The robot program can be created or written through page [Program].

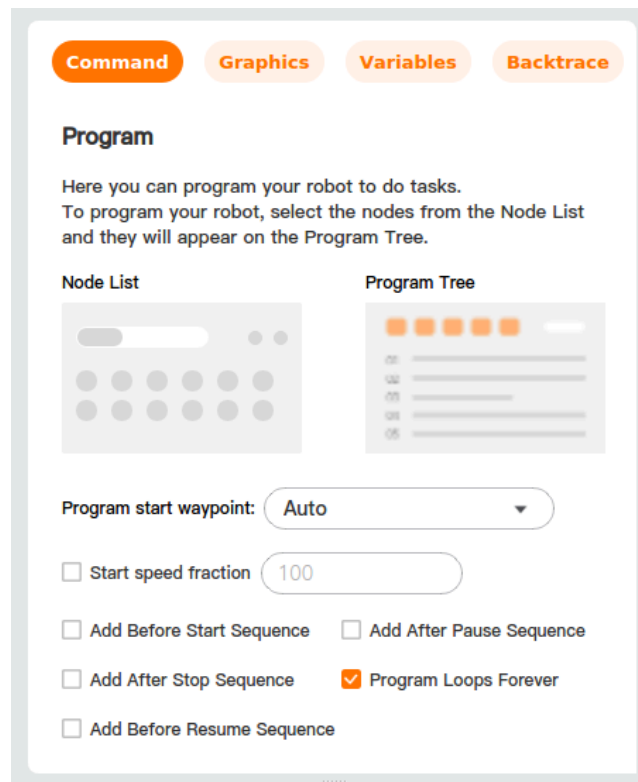


Serial number	Name	Description
1	Program Information Card	Include Configure tabs: Command, Graph, Variable, and Track.
2	Program Node Card	Containing node classification and node list.
3	Programming Card	Containing the program manager toolbar, program tree, and program tree toolbar.

Program information card

Command

The [Command] page is for you to configure nodes selected in the program tree. After a program node is selected, the configurable items of the node will be displayed/can be set on the page [Command].



- Program start waypoint: set the waypoint from which the program starts running.
 - Auto: Click [Run] to enter the [Move] interface, move the robot arm to the start waypoint, and then click [OK] to start running the program.
 - Ignore: Click [Run]. Then, the robot arm moves to the start waypoint, and the program starts running.
- Start speed fraction: Check to set the start speed fraction of the program.
- Add Before Start Sequence: Check to add a sequence to be executed before the master program starts running.
- Add After Pause Sequence: Check to add a sequence to be executed after the master program pauses.
- Add Before Resume Sequence: Check to add a sequence to be executed before the master program resumes running.
- Add After Stop Sequence: Check to add a sequence to be executed after the master program stops.

- Program Loops Forever: Check to enable the master program to run continuously.

Graphics

The page [Graphic] displays the 3D model of the current robot, whose actions are consistent with those of the real robot.

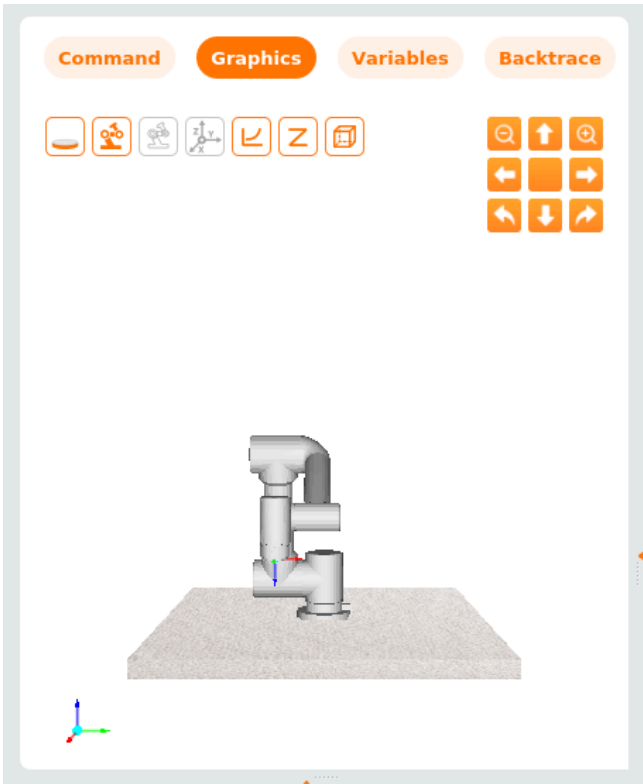


Table: Graphic symbols of model operation buttons

Icon	Name	Description
	Up	Move the robot upward.
	Down	Move the robot downward.
	Left	Move the robot left.
	Right	Move the robot right.
	Turn left	Turn the robot clockwise.
	Turn right	Turn the robot counterclockwise.
	Enlarge	Enlarge the robot.

	Micrify	Micrify the robot.
	Reset	Reset the robot's position and view.

Table: Graphic symbols of model display buttons

Icon	Name	Description
	Base	Hide/show base
	Robot arm	Hide/show robot arm
	Target position/pose	Hide/show position/pose of target waypoint
	User coordinate system	Hide/show user coordinate system
	Track	Hide/show track
	Waypoint	Hide/show waypoint
	Obstacle	Hide/show obstacles
	Tool envelope ball	Hide/show tool envelope ball
	Point cloud	Hide/show point cloud

Variables

The page [Variables] displays the base coordinate system, tool coordinate system, waypoint joint angle, and end-of-tool TCP position/pose of the running program, as well as all the variables and values in the programs and profiles. The information shown in this page is the same as that on the page called out by "Home > Variables."

For information on variable types and data types, please refer to "[Configuration - General - Variables](#)."

CommandGraphicsVariablesBacktrace

i_var_0	1
i_var_1	true
i_var_2	"a"
i_var_3	"a"
i_var_4	0.5
i_var_5	{0, -15, 100, 25, 92, 0}
i_var_6	{403.82, -119, 211.54, 3.142, 0, 1.571}
Base	{0, 0, 0, 0, 0, 0}
Tool	{0, 0, 0, 0, 0, 0}
Waypoint_0_p	{0.550227, -0.13211, 0.419111, -3.1415...
Waypoint_0_q	{2.66454e-15, -0.191616, 1.51194, ...
Waypoint_1_p	{0.550227, -0.13211, 0.313056, -3.1415...
Waypoint_1_q	{2.85339e-15, -0.261799, 1.74533, ...

Tracking

The page [Backtrace] displays the start waypoint of the current program and the time when the program runs the node. You can click [Variables] in the log list to view the detailed information, offering convenience for your program debugging.

CommandGraphicsVariablesBacktrace

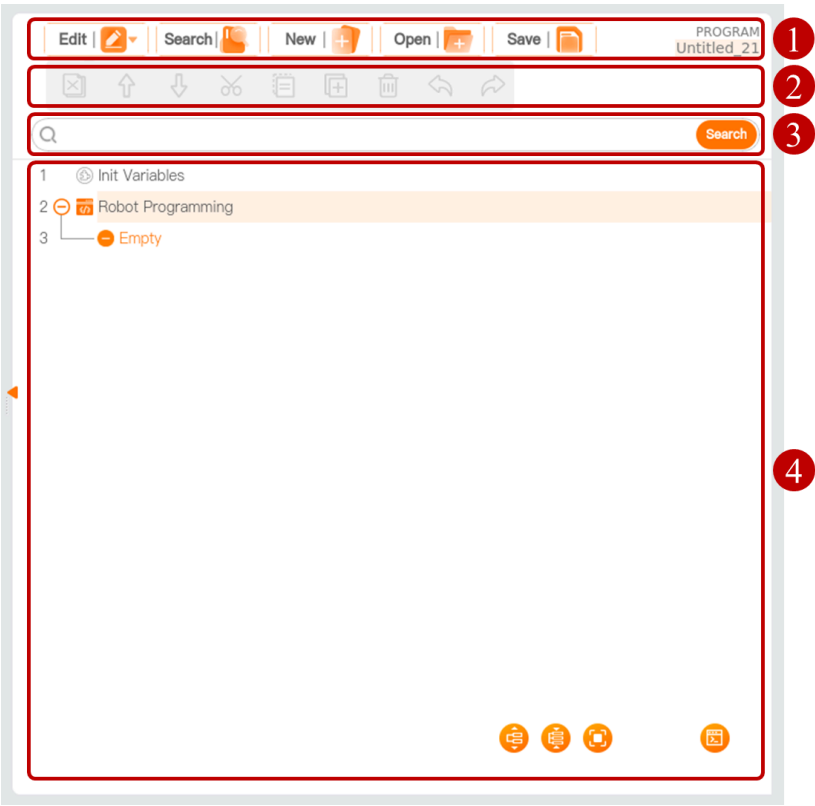
20:07:02 Stopped

22:53:29 Running

+0.197	Waypoint_0	Variables
+1.001	Waypoint_1	Variables
+0.999	Waypoint_0	Variables
+1.093	Waypoint_1	Variables
+1.000	Waypoint_0	Variables
+1.000	Waypoint_1	Variables
+1.099	Waypoint_0	Variables

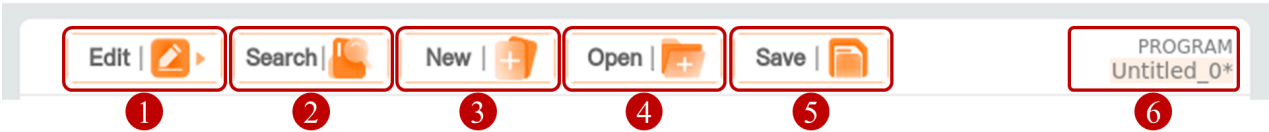
Programming card

The programming card displays or allows the user to edit the program tree, manage the program files, etc.



Serial number	Name	Description
1	Program manager tool bar	Manage program files.
2	Program tree toolbar	Edit the node in the program tree.
3	Search box	Enter keyword to search for nodes in the program tree.
4	Program tree	Display or edit programs.

Program manager tool bar



Serial number	Name	Description
1	Edit	Show/hide the program tree toolbar.
2	Search	Show/hide the search box, in which a keyword can be entered to search for nodes in the program tree.
3	New	Create a new program file.
4	Open	Open the program file.
5	Save	Save the program file.
6	Program name	Display the name of the current program file. Click the program name to view the name of the profile associated with the current program file.

Tip

1. If you use the I/O interface or the Start/Stop button on the control handle to control the start/stop of the program, please save the program file first. Otherwise, the program cannot be started. For detailed I/O function settings, please refer to [I/O Configuration](#). For the operation of the control handle, please refer to the Hardware User Manual.
2. After a program is written, if the associated configuration file is modified, it is required to save the configuration file first and then the program file. Otherwise, a program runtime error may occur.

Program tree

- Graphically programmed commands are added to the program tree, which can be called "program nodes" or "nodes" for short. Some nodes may contain other nodes, for example, the [Move] node, which can contain the [Waypoint] node. Such nodes can be called "program blocks". In other words, a program block is composed of multiple program nodes.
- Folding/unfolding a program block
 - Click  or  before the program block to expand it or make it collapse. -Double-click the program block name to unfold/fold this program block. (Recommended)
 - Click  or  to expand all program blocks in the program tree or make them collapse.
- Initial variables: the first value assigned to the program variables when the program is started. The [Init Variables] node is added by default before the master program. In the [Init Variables] command page [Init Variables] for short), click the [Variables] dropdown box,

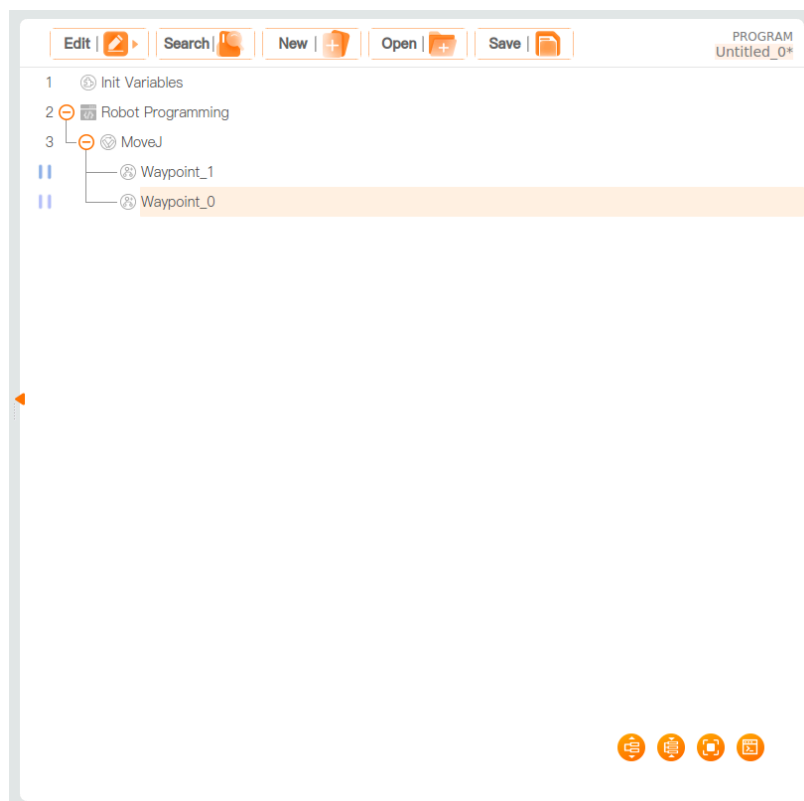
select initial variables, and click the expression input box to assign the initial variables a value.

- **Last-Run Value:** when this is checked, the value of the initial variables is replaced with the value of the initial variables when the program is stopped. If a new program is loaded after the current program pauses, the value of the initial variables when the current program was stopped cannot be used and will be reset to the user setting.

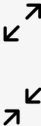
Tip

To use this function normally, you are required to create a variable first. If you do not set the initial variables, the master program runs directly.

- When the node in the program tree does not conform to the preset logic, the system will highlight associated nodes in yellow. When the node is configured in accordance with the preset logic, the system displays the node in black normally.
- **Breakpoint:** To add a breakpoint to a line in the program tree, click the Line No. The newly set breakpoint is displayed in a darker color. After running to this line, the program will pause directly. At this time, click [Step] in the menu bar. Each time you click it, the program will run for one line to realize single-line running.



Description of graphical symbols in program tree tool bar

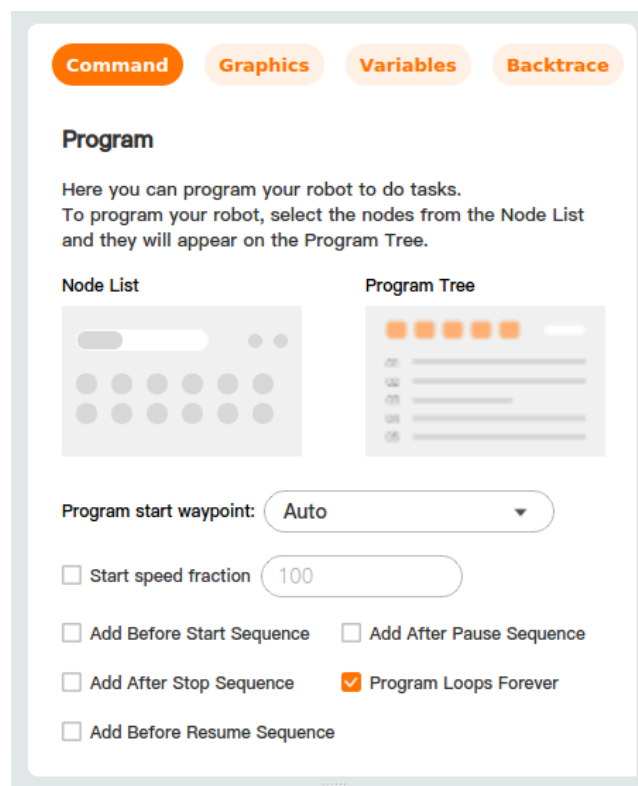
Icon	Description
	Up, move a node upward.
	Down, move a node downward.
	Undo, undo changes to the command.
	Redo, redo changes to the command.
	Cut, cut a node, and allow it to be pasted elsewhere in the program tree.
	Copy, copy a node and allow it to be pasted elsewhere in the program tree.
	Paste, paste a cut or copied node.
	Delete, delete a node.
	Inhibit, inhibit a node. The program will directly skip the inhibited node during running. An inhibited node can also be released from inhibition. [Inhibit] allows one to quickly change a program without destroying the original content.
	Unfold all nodes.
	Fold all nodes.
	Full screen: Display the program tree in full screen. Restore, restore the program tree to default display.
	View scripts. View graphical program.

Program Information Card

Configurable tabs include [Command], [Graphics], [Variables], and [Backtrace]. You can view the contents under each tab when necessary.

Command

The [Command] page is for you to configure nodes selected in the program tree. After a program node is selected, the configurable items of the node will be displayed/can be set in the page [Command].



- Program start waypoint: set the waypoint from which the program starts running.
 - Auto: Click [Run] to enter the [Move] interface, move the robot arm to the start waypoint, and then click [OK] to start running the program.
 - Ignore: Click [Run]. Then, the robot arm moves to the start waypoint, and the program starts running.
- Start speed fraction: Check to set the start speed fraction of the program.
- Add Before Start Sequence: Check to add a sequence to be executed before the master program starts running.

- Add After Pause Sequence: Check to add a sequence to be executed after the master program pauses.
- Add Before Resume Sequence: Check to add a sequence to be executed before the master program resumes running.
- Add After Stop Sequence: Check to add a sequence to be executed after the master program stops.
- Program Loops Forever: Check to enable the master program to run continuously.

Graphics

The page [Graphic] displays the 3D model of the current robot, whose actions are consistent with those of the real robot.

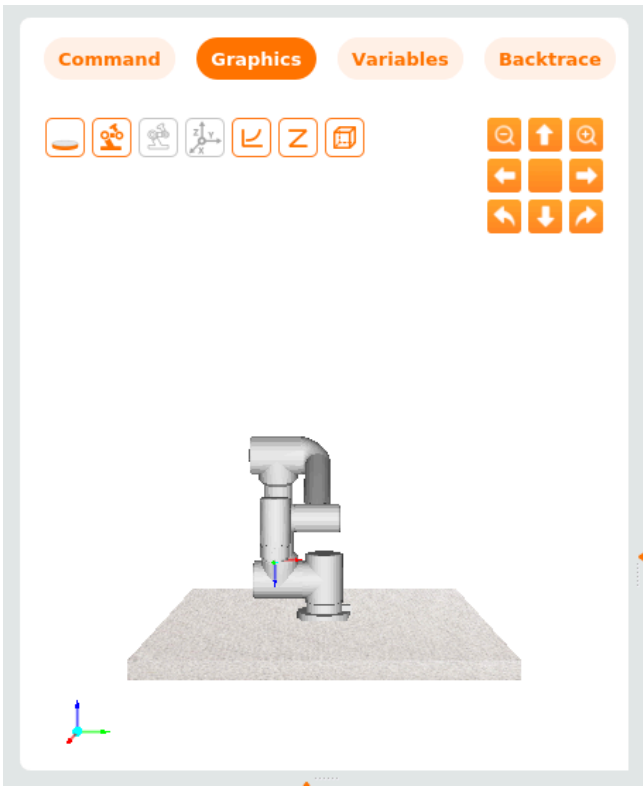


Table: Graphic symbols of model operation buttons

Icon	Name	Description
	Up	Move the robot upward.
	Down	Move the robot downward.

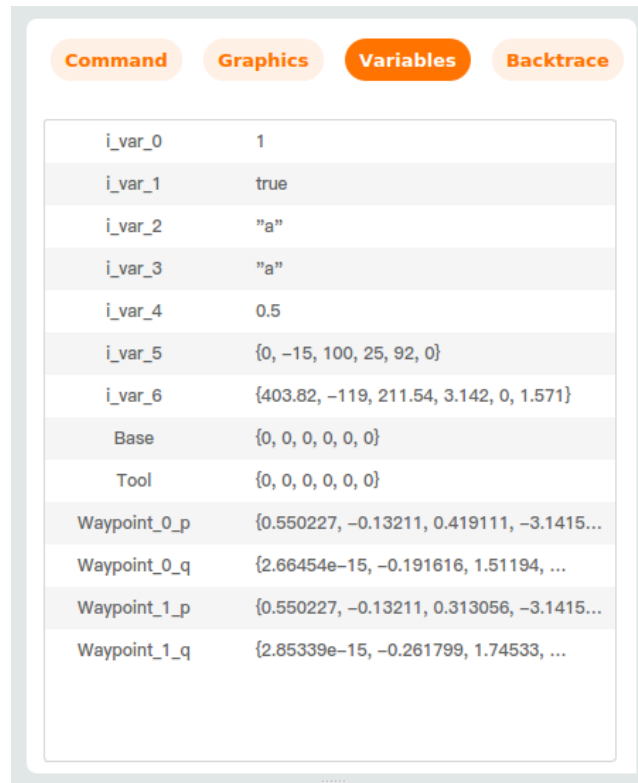
	Left	Move the robot left.
	Right	Move the robot right.
	Turn left	Turn the robot clockwise.
	Turn right	Turn the robot counterclockwise.
	Enlarge	Enlarge the robot.
	Micrify	Micrify the robot.
	Reset	Reset the robot's position and view.

Table: Graphic symbols of model display buttons

Icon	Name	Description
	Base	Hide/show base
	Robot arm	Hide/show robot arm
	Target position/pose	Hide/show position/pose of target waypoint
	User coordinate system	Hide/show user coordinate system
	Track	Hide/show track
	Waypoint	Hide/show waypoint
	Obstacle	Hide/show obstacles
	Tool envelope ball	Hide/show tool envelope ball
	Point cloud	Hide/show point cloud

Variables

The page [Variables] displays the base coordinate system, tool coordinate system, waypoint joint angle, and end-of-tool TCP position/pose of the running program, as well as all the variables and values in the programs and profiles. The information shown in this page is the same as that on the page called out by "Home > Variables."



i_var_0	1
i_var_1	true
i_var_2	"a"
i_var_3	"a"
i_var_4	0.5
i_var_5	{0, -15, 100, 25, 92, 0}
i_var_6	{403.82, -119, 211.54, 3.142, 0, 1.571}
Base	{0, 0, 0, 0, 0, 0}
Tool	{0, 0, 0, 0, 0, 0}
Waypoint_0_p	{0.550227, -0.13211, 0.419111, -3.1415...
Waypoint_0_q	{2.66454e-15, -0.191616, 1.51194, ...
Waypoint_1_p	{0.550227, -0.13211, 0.313056, -3.1415...
Waypoint_1_q	{2.85339e-15, -0.261799, 1.74533, ...

Types of variables

- Program variables: Such variables apply only to a running program, and their values are lost after the program is stopped. Therefore, they are general variables. Please refer to "[Assignment](#)" for more details.
- Configure variables: Such variables can be applied to more than one program, and their names and values persist until the robot is decommissioned. Therefore, they are persistent variables. Please refer to "[Variables](#)" for more details.
- Script variables: Such variables come from script files and can be assigned to different variable types. Script variables are not displayed in page [Program] or page [Configure]. They are used by a robot program to store information about the waypoint.

Based on value treatment mechanism (saved by system or not) and application scope (one program or more than one program), variables are divided into the following two types:

- Persistent variables: These variables can be applied to more than one program, and their names and values persist until the robot is decommissioned. Values of persistent variables

are retained after AuboStudio or the controller is restarted. For example, the configured variables are of this kind.

- General variables: These variables apply only to the running program, and their values are lost after the program is stopped. For example, the program variables are of this type.

Data type

- Bool: Boolean variables, whose value is true/false.
- Integer (Int): Integer variables whose value is an integer.
- Float: Floating-point variables, whose value is a floating-point number.
- String: String variables, whose value is a string value enclosed in single quotes (') or double quotes (").
- Pose: Pose variables, whose value is waypoint data containing 6 parameters, as described below:
 - In case of MoveL, the first 3 parameters are pose parameters measured in mm, and the last 3 parameters are pose parameters measured in rad.
 - In case of MoveJ, all the 6 parameters are joint angles, which are measured in rad.

Backtrace

The page [Backtrace] displays the start waypoint of the current program and the time when the program runs the node. You can click [Variables] in the log list to view the detailed information, offering convenience for your program debugging.

CommandGraphicsVariablesBacktrace

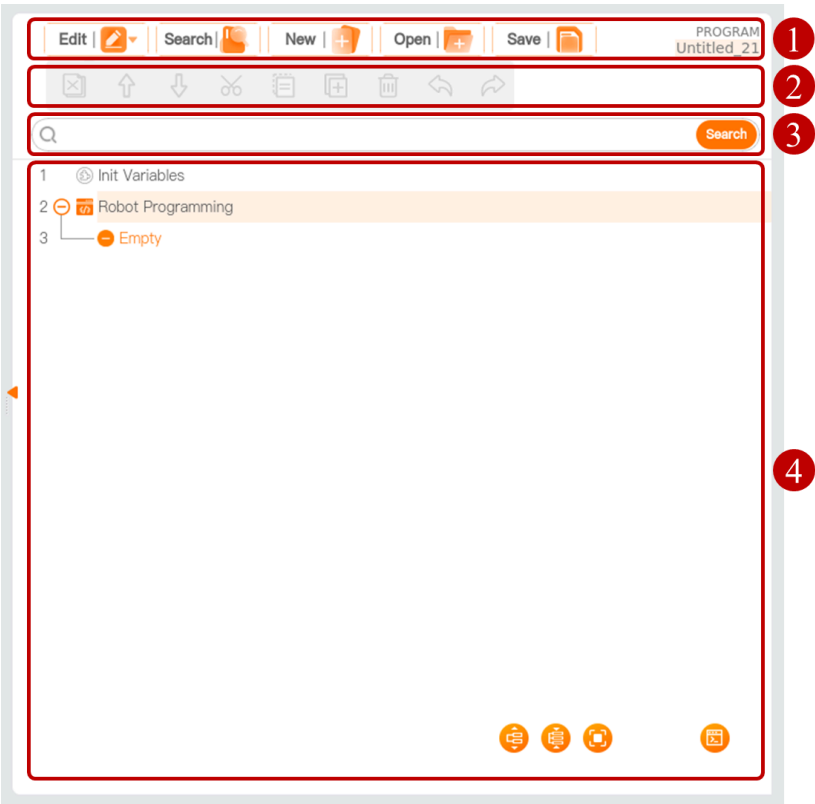
20:07:02 Stopped

22:53:29 Running

+0.197	Waypoint_0	Variables
+1.001	Waypoint_1	Variables
+0.999	Waypoint_0	Variables
+1.093	Waypoint_1	Variables
+1.000	Waypoint_0	Variables
+1.000	Waypoint_1	Variables
+1.099	Waypoint_0	Variables

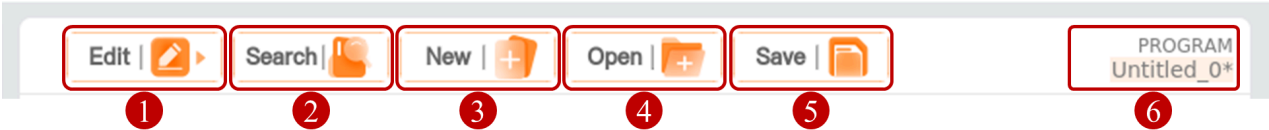
Programming Card

The Programming Card displays or allows the user to edit the program tree, manage the program files, etc.



Serial number	Name	Description
1	Program manager tool bar	Manage program files.
2	Program tree tool bar	Edit the node in the program tree.
3	Search box	Enter a keyword to search for nodes in the program tree.
4	Program tree	Display or edit programs.

Program manager tool bar



Serial number	Name	Description
1	Edit	Show/hide the program tree tool bar.
2	Search	Show/hide the search box, in which a keyword can be entered to search for nodes in the program tree.
3	New	Create a new program file.
4	Open	Open the program file.
5	Save	Save the program file.
6	Program name	Display the name of the current program file. Click the program name to view the name of the profile associated with the current program file.

Tip

1. If you use the I/O interface or the Start/Stop button on the control handle to control the start/stop of the program, please save the program file first. Otherwise, the program cannot be started. For detailed I/O function settings, please refer to [I/O Configuration](#). For the operation of the control handle, see the Hardware User Manual.
2. After a program is written, if the associated configuration file is modified, it is required to save the configuration file first and then the program file. Otherwise, a program runtime error may occur.

Program tree

- Graphically programmed commands are added to the program tree, which can be called "program nodes," or "nodes" for short. Some nodes may contain other nodes, for example, the [Move] node, which can contain the [Waypoint] node. Such nodes can be called "program blocks". In other words, a program block is composed of multiple program nodes.
- Folding/unfolding a program block
 - Click  or  before the program block to expand it or make it collapse. -Double-click the program block name to unfold/fold this program block. (Recommended)
 - Click  or  to expand all program blocks in the program tree or make them collapse.
- Initial variables: the first value assigned to the program variables when the program is started. The [Init Variables] node is added by default before the master program. In the [Init Variables] command page ([Init Variables] for short), click the [Variables] dropdown box,

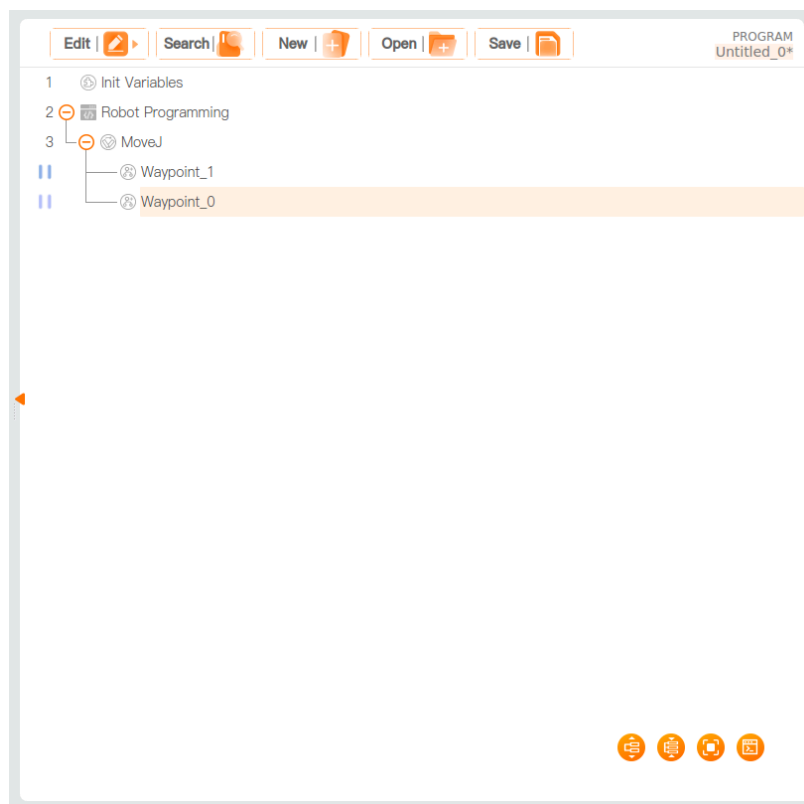
select initial variables, and click the expression input box to assign the initial variables a value.

- **Last-Run Value:** when this is checked, the value of the initial variables is replaced with the value of the initial variables when the program is stopped. If a new program is loaded after the current program pauses, the value of the initial variables when the current program was stopped cannot be used and will be reset to the user setting.

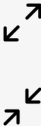
Tip

To use this function normally, you are required to create a variable first. If you do not set the initial variables, the master program runs directly.

- When the node in the program tree does not conform to the preset logic, the system will highlight associated nodes in yellow. When the node is configured in accordance with the preset logic, the system displays the node in black normally.
- **Breakpoint:** To add a breakpoint to a line in the program tree, click the Line No. The newly set breakpoint is displayed in a darker color. After running to this line, the program will pause directly. At this time, click [Step] in the menu bar. Each time you click it, the program will run for one line to realize single-line running.

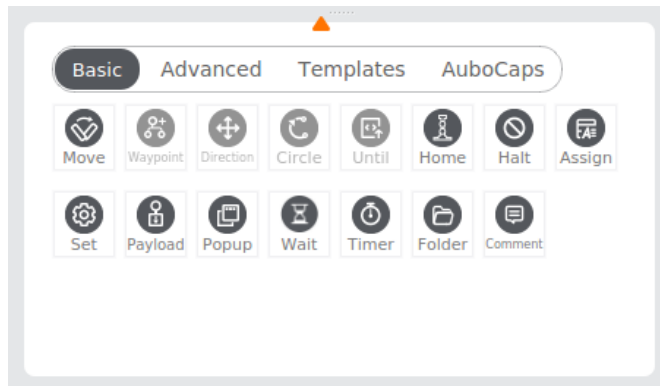


Legend for program tree tool bar

Icon	Description
	Up, move a node upward.
	Down, move a node downward.
	Undo, undo changes to the command.
	Redo, redo changes to the command.
	Cut, cut a node, and allow it to be pasted elsewhere in the program tree.
	Copy, copy a node and allow it to be pasted elsewhere in the program tree.
	Paste, paste a cut or copied node.
	Delete, delete a node.
	Inhibit, inhibit a node. The program will directly skip the inhibited node during running. An inhibited node can also be released from inhibition. [Inhibit] allows one to quickly change a program without destroying the original content.
	Unfold all nodes.
	Fold all nodes.
	Full screen: Display the program tree in full screen. Restore, restore the program tree to default display.
	View scripts. View graphical program.

Basic Program Node

Basic program node, namely, the basic node that makes up the program, includes node commands indicating waypoint, direction, comment, folder, etc. For example, the commands indicating waypoint and direction can be used to determine the motion track of the robot. And the commands indicating comment and folder can assist users in managing the program, and their proper use enables good readability of the program.



Move

The [Move] node allows you to add a waypoint for the movement of the robot and set the shared parameters of other nodes under the node.

The screenshot displays the AUBO Developer software interface. On the left is a vertical sidebar with icons for Home, Program, Configure, Move, and I/O. The main workspace is divided into several panels. The top panel shows tabs for Command, Graphics, Variables, and Backtrace. The Command panel is active, showing the 'Move' section with buttons for MoveJ, MoveL, and MoveS. Below these, there are settings for TCP (Use Active TCP), Joint Speed (60 %/s), Joint Acceleration (80 %/s²), and Coordinate (Base). A 'reset' button is at the bottom. The right panel shows a program snippet with the following steps:

```

1 Init Variables
2 Robot Programming
3 MoveJ
4   Waypoint_0

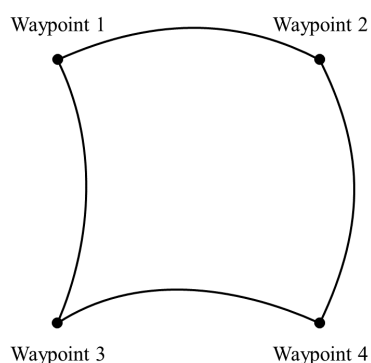
```

At the bottom of the interface, there is a status bar showing '100%' and '0mm/s'.

Move type

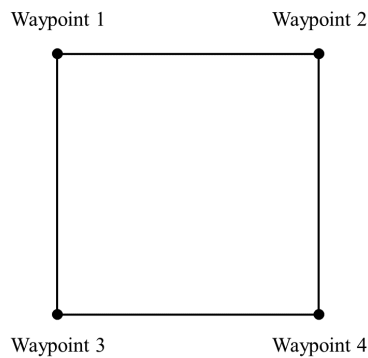
The robot can move in three modes: MoveJ, MoveL, and MoveS.

- MoveJ refers to the movement of the robot by controlling its joint. MoveJ provides a curved path for the tool center point (TCP). If you want the robot to move between waypoints quickly, regardless of the movement path of the TCP between these waypoints, MoveJ should be selected preferentially. MoveJ is suitable for movement in the fastest way in a sufficiently wide environment. This motion mode is as shown in the figure below.



- MoveL refers to the movement of the tool center point (TCP) of the robot in a linear way to the target waypoint. This means that each joint moves in a more complex way to keep the

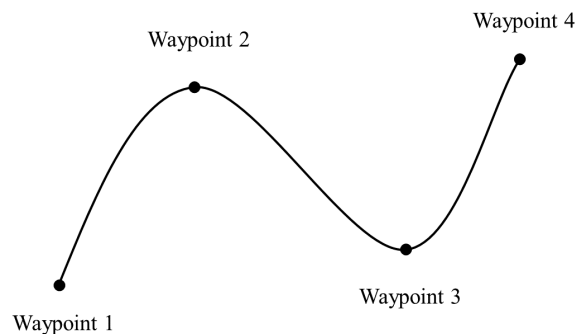
TCP on a straight path, and whether the maximum tool speed can be reached and maintained depends on the linear displacement and maximum acceleration parameters. This motion mode is as shown in the figure below.



- MoveS refers to the smooth movement of the robot through all given waypoints at a constant speed according to a spline curve path established based on the given waypoints. More waypoints given on the intended curve will result in a more desirable curve path.

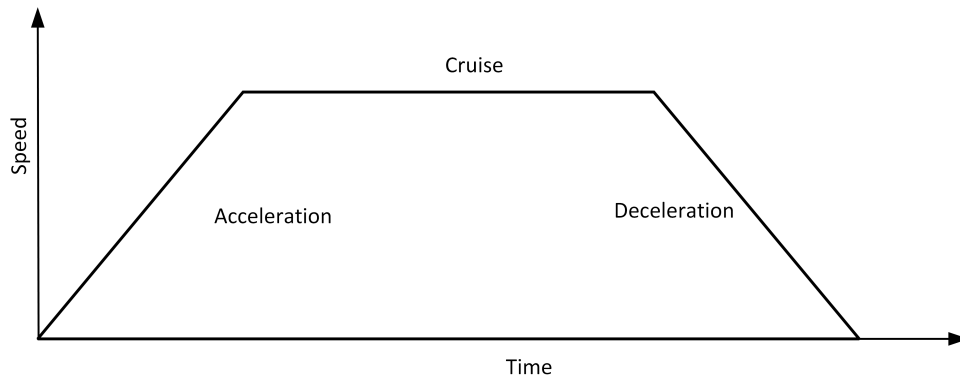
Tip

1. Operator's or I/O operations are not supported during MoveS. Operator's or I/O operations during MoveS may trigger the protective stop of the robot.
2. Only [Waypoint] nodes can be included in [MoveS] nodes.



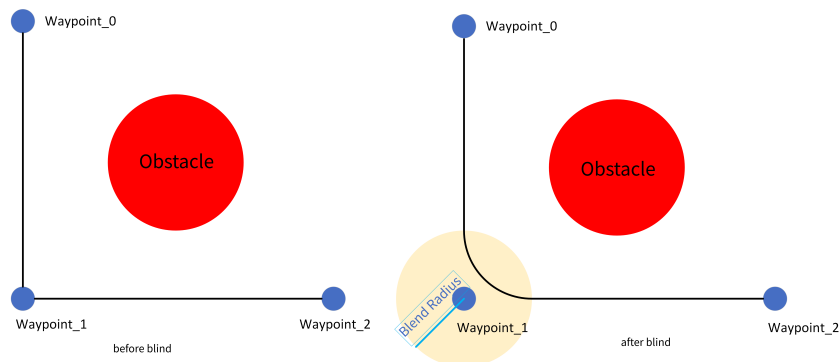
Movement speed

The movement speed curve of the robot is divided into three parts: acceleration, cruise, and deceleration. The movement speed parameter determines the cruise, and the acceleration parameter determines the acceleration and the deceleration gradient, as shown in the figure.

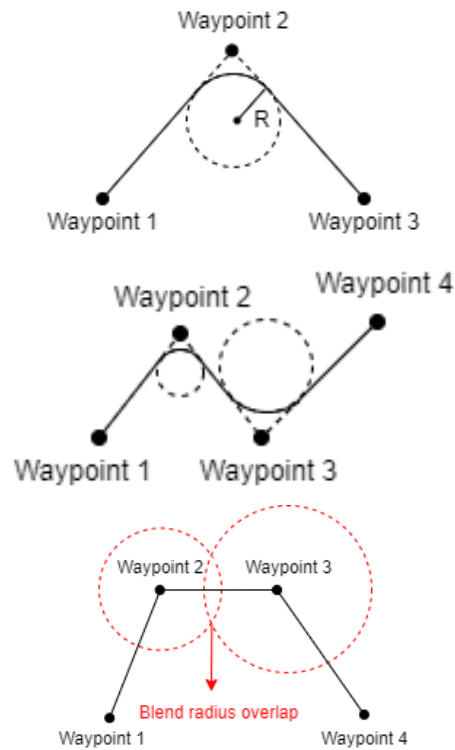


Blend

When the robot is moving, if you have set the blend parameters, a smooth transition between the two tracks can be achieved so that the robot can move faster and more stably. For example, in the following figure, the robot is expected to reach [Waypoint_2] from [Waypoint_0], but because there are other items between the two waypoints, the robot has to start from [Waypoint_0] and go through [Waypoint_1] to reach [Waypoint_2]. Since [Waypoint_1] only exists to bypass other items, it is preferable for the robot to expend minimal energy and time at [Waypoint_1]. At this time, adding the blend parameters will enable the robot to smoothly transition from the first track to the second track, eliminating the need for deceleration and acceleration at the intermediate point. This reduces the time and energy required for the robot to turn, thereby enhancing the movement efficiency.



The main blend parameter is the blend radius, that is, the turning radius of the robot. When the robot is within the blend radius of a waypoint, the blend feature may be activated, allowing the robot to deviate from its original path and move continuously without stopping at the waypoint. The smaller the blend radius, the larger the path angle; and the larger the blend radius, the smaller the path angle. Note, however, that overlaps shall be avoided in the blend, as shown in the figure below.

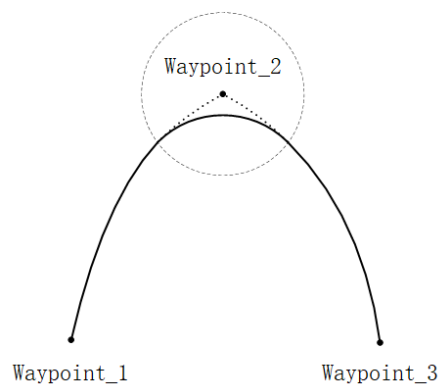


Other parameters that affect the blend track:

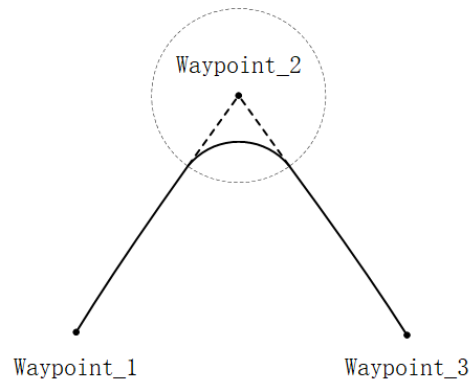
- Movement type (MoveJ, MoveL, MoveS) at the starting and ending positions of blend.
- Robot movement time on the track.
- Robot speed (joint speed/joint acceleration, tool speed/tool acceleration).

Blend track

- Blend in [MoveJ]: After the blend parameters are set, the system will generate a smooth curve in the joint space.



- Blend in [MoveL]: After the blend parameters are set, an arc path will be planned at the blend position, following a direction blended with the smooth interpolation between the two tracks.



Shared parameter

The shared parameter feature enables bulk setting of parameters in this program block. For example, after setting [Shared parameter] in the [Move] node, all nodes added to the [Move] program block will, by default, apply the values set in the [Move] node for any parameters included in [Shared parameters], unless those parameters are specifically set.

Coordinate system

As the robot moves, the software calculates the position of the TCP within the selected coordinate system, so the motion track of the robot may vary with the coordinate system selected. Two coordinate systems, for the base and one for the tool are predefined, and users can also select a custom coordinate system here. Please refer to "[Coordinate system] (../02_basic_operation/05_coordinate.md#Custom coordinate system)" for the description and settings of the coordinate system.

Tool center point (TCP)

Select a different tool center point (TCP) to adjust the movement mode of the robot between waypoints. The TCP for [Waypoint] can be set in the [Move] node. See [TCP](#) for the description and settings of TCP.

- Use active TCP.
- Ignore active TCP, allowing adjustment of the movement in relation to the tool flange.
- Custom TCP.

Use joint angles

If this option is checked, the joint angles will be recorded in the scripts when the program is running, replacing the location information of the robot. And the system will disable the [Set TCP] and [Coordinate] here. If this feature is activated, the waypoint information will remain unchanged during project migration, allowing the robot to continue using the data without adjustment.

Add [Move] node

1. Click the [Move] icon to add a [MoveJ] node to the program tree, including a [Waypoint] node beneath it.
2. The [Move] node contains movement types such as MoveJ, MoveL, and MoveS. The display name of the [Move] node in the program tree is determined by the movement type you select.
3. Select the [MoveJ/MoveL/MoveS] node to set the node.

Add [MoveS]

1. Click the [Move] command to add a [Move] node to the program tree.
2. Select [MoveS] in the [Move] command interface.
3. Set the relevant parameters of the MoveS: TCP, coordinate, joint speed, joint acceleration, etc.
4. Under the [MoveS] node, add several [Waypoint] nodes and configure the location for each waypoint to finalize the [MoveS] addition.

Waypoint

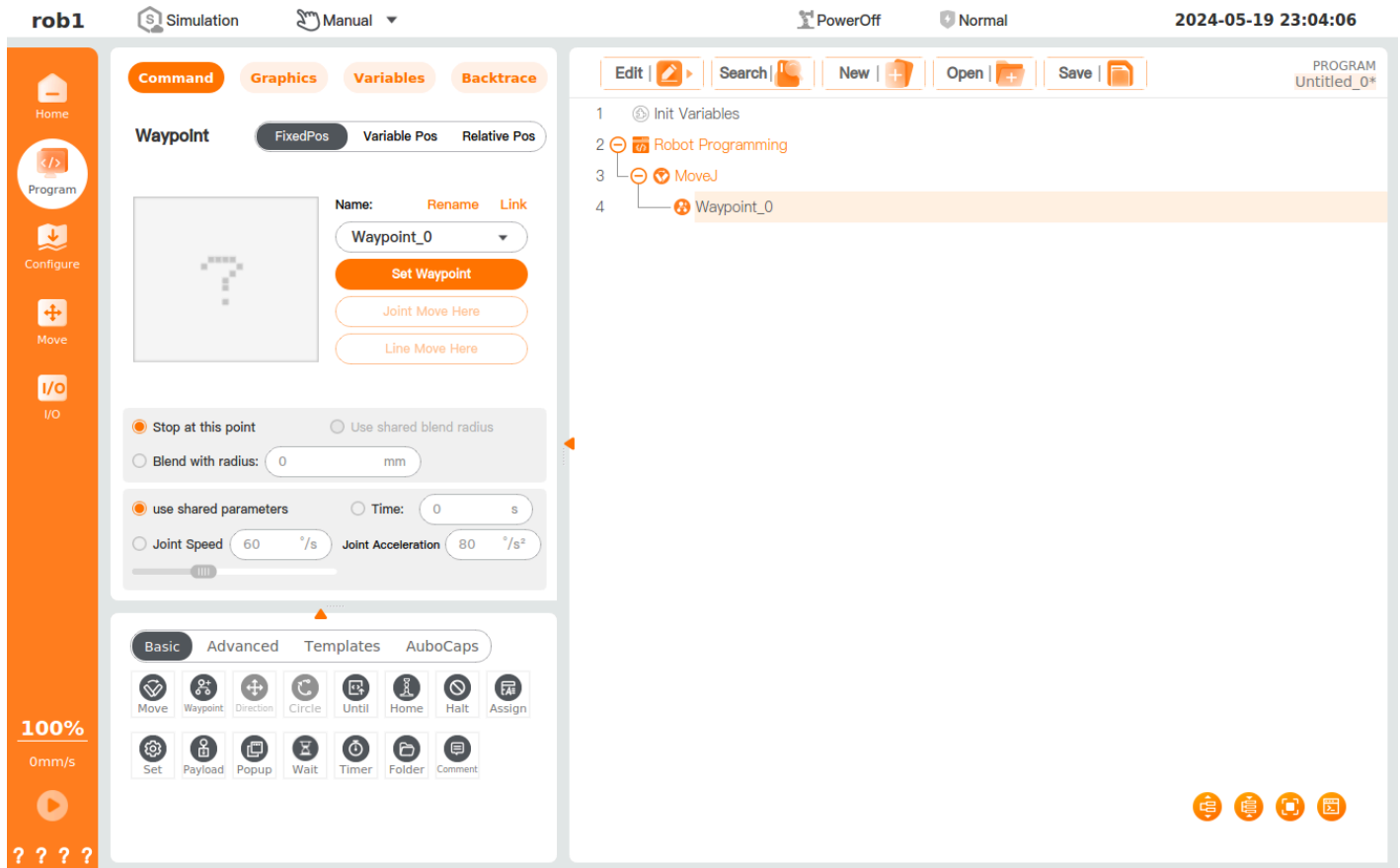
The [Waypoint] command indicates the target position for the tool center point (TCP) of the robot. By configuring the waypoint, the precise location of the robot can be established. Generally, the motion track of the robot's tool center point (TCP) includes two or more waypoints. The [Waypoint] node enables the configuration of [FixedPos], [Variable Pos,] and [Relative Pos.]

Tip

The [Waypoint] node must be placed under the [Move] node.

1. Fixed pos

The [Fixed Pos] feature enables configuration of the position/pose of the [Waypoint] by operating the robot's [Move] interface or dragging the robot.

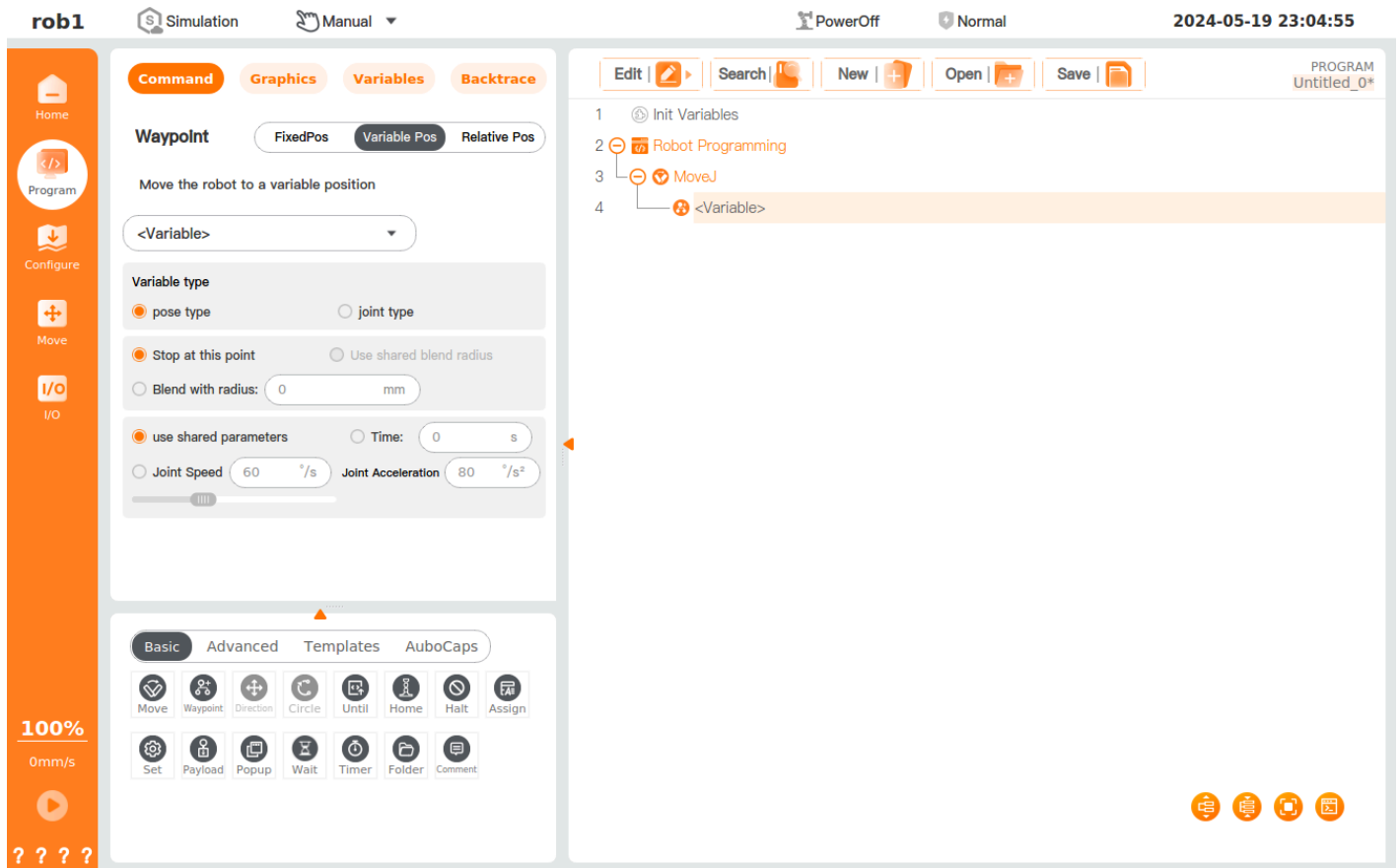


- Teach: You can guide the robot to assume a specific position/pose in the current coordinate system. Below is the procedure for teaching a waypoint to a robot:
 1. Add a [Move] program node to the program tree.
 2. Set TCP via the [Set TCP] drop-down menu in the [Move] program node; select a coordinate system via the [Coordinate] drop-down menu.
 3. Insert a [Waypoint] node under the [Move] program node. After insertion, select the waypoint type in the upper right of the [Waypoint] node interface.
 4. Click [Set Waypoint] in the [Fixed Pos] interface to access the robot teaching interface, where you can use the [Position/Pose Control] to move the robot or enter the HandGuide mode to manually guide the robot to the target waypoint. After setting, click [OK] to save the position/pose of the [Waypoint.]
- [Waypoint] naming and switching

- A waypoint is named "waypoint_n" by default, with n starting at 0 and incrementing by 1 for each new waypoint.
- Rename waypoint: Click [Rename] to change the waypoint name. Waypoints with duplicate names are not allowed.
- Link: Click the icon ▼ or [Link] and select the waypoint in the program to use the position/pose of the target waypoint. After successful linking, the [Link] button will change to [Cancel Link]. Click [Cancel Link] to restore the original waypoint.
- Blend parameters
 - [Stop at this point]: You can set the blend radius to 0 mm to stop the robot at this waypoint position.
 - Use shared blend radius: You can set the waypoint to use a shared blend radius (this option is not selectable in MoveJ and MoveS modes).
 - Blend with radius: You can set the blend radius for the waypoint.
- Speed
 - Use shared parameters: You can set the waypoint to use shared speed parameters (joint/tool speed, joint/tool acceleration).
 - Time: You can set the total movement time (in seconds) from the last waypoint to this waypoint, which affects the speed of the robot. An unreasonable total movement time may cause out-of-range motion parameters of the robot, which will be optimized by AuboStudio, and the operating parameters of the robot shall prevail.
 - Joint/tool speed, joint/tool acceleration: You can set the joint/tool speed and joint/tool acceleration of the waypoint.

2. Variable pos

The [Variable Pos] option allows you to set the position/pose of [Waypoint] by configuring variables. The [Variable Pos] option is often used for bulk setting of the parameters for identical waypoints to save programming time.



- Select a variable: Click the [Variables] dropdown box to select the variable. Please refer to "[Assignment](#)" and "[Variables](#)" for the settings of variables.
- Variable type: You can set the type of the variable.
 - **Joint type:** The six parameters of the variable correspond to the **joint angles** of the robot, measured in **radians (rad)**.
 - Example: `i_variable_1 = {0, -0.261796, 1.74532, 0.436316, 1.5708, 0}` .
 - **Pose type:** The six parameters of the variable represent the **Cartesian pose** of the robot's end-effector, formatted as {x, y, z, roll, pitch, yaw}:
 - The first 3 parameters (x, y, z) are the **position coordinates** of the end-effector in the world coordinate system, with the unit of **meters (m)**;
 - The last 3 parameters are the **end-effector orientation angles** (Roll, Pitch, Yaw, i.e., RPY Euler angle representation), measured in **radians (rad)**.
 - Example: `i_variable_1 = {0.548871, -0.1215, 0.263199, 3.14159, 0, 1.5708}` .

Pose expression notation:

The pose uses **Roll-Pitch-Yaw (RPY) Euler angle** representation.

Rotation sequence (fixed axis): X→Y→Z, which means: first rotate around the **X-axis** of the fixed coordinate system by **roll** (roll angle), then around the **Y-axis** by **pitch**

(pitch angle), and finally around the **Z-axis** by *yaw* (yaw angle).

The corresponding total rotation matrix is:

$$R = R_z(yaw)R_y(pitch)R_x(roll)$$

Where R_x , R_y , and R_z are the basic rotation matrices about the X, Y, and Z axes respectively.

Rotation around fixed axis $X \rightarrow Y \rightarrow Z$ and rotation around moving axis $Z \rightarrow Y \rightarrow X$ result in mathematically equivalent total rotation matrices.

3. Relative pos

The [Relative Pos] feature defines the position/pose of the [Waypoint] by the gap between two given locations (source waypoint and target waypoint).

Tip

In programming, it is essential to include at least one waypoint in addition to the [Relative Pos] waypoint; failing to do so may result in unpredictable errors in the robot's operation.

The screenshot displays the AUBO Developer software interface. On the left is a vertical sidebar with icons for Home, Program, Configure, Move, and I/O. The main area is divided into two panels. The left panel, titled 'Waypoint', shows the 'Relative Pos' tab selected. It includes fields for 'Distance 0 mm' and 'Angle 0 °', and buttons for 'Set From Waypoint' and 'Set To Waypoint'. Below these are options for 'Stop at this point' and 'Blend with radius'. The right panel, titled 'Program', shows a sequence of steps: 1. Init Variables, 2. Robot Programming, 3. MoveJ, and 4. Waypoint_from_0 Waypoint_to_0. The bottom of the interface features a '100%' status bar and a row of icons for various functions like Move, Waypoint, Direction, Circle, Until, Home, Halt, Assign, Set, Payload, Popup, Wait, Timer, Folder, and Comment.

- Distance: the Cartesian distance between the source and target waypoints.

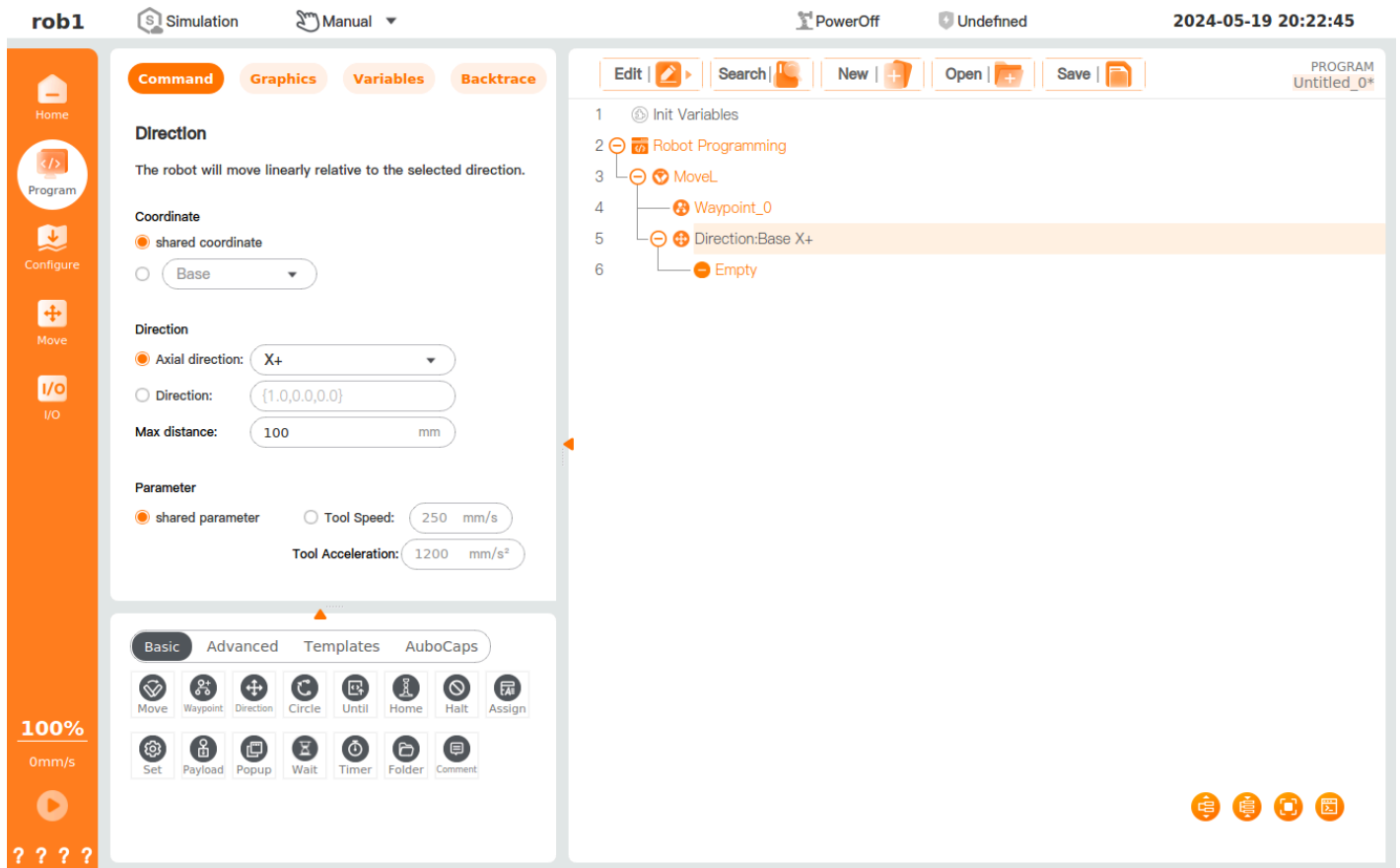
- Angle: the length of the rotational vector for direction change between the source and target waypoints.
 - [Waypoint] rename and selection
 - Click [Rename] or the waypoint name to rename the source waypoint or target waypoint.
 - Click [Link] or the icon ▼ to expand the waypoint list and select a waypoint. After successful linking, the [Link] button will change to [Cancel Link]. Click [Cancel Link] to restore the original waypoint.
 - Set source waypoint and target waypoint: Click to access the robot teaching interface, where you can use the [Position/Pose Control] to move the robot or enter the HandGuide mode to manually guide the robot to the target waypoint. After setting, click [OK] to save the position/pose of the source or target waypoint.
 - Move here: click to access the [Move] interface, where you can press and hold [Auto] to move the robot to the position/pose of the waypoint.
-

Direction

The [Direction] command specifies the motion relative to the coordinate system or TCP so that the robot can move relative to the specified direction until the stop condition of the [Until] command is met or the maximum distance set in [Direction] is reached.

Tip

1. The [Direction] feature must be used under the [MoveL] command.
2. The [Direction] command should be used with the [Until] command.



Settings

- **Coordinate:** You can set the coordinate system used for motion at this node. You can use a shared coordinate system for the [Move] node or a custom coordinate system.
- **Direction:** You can set the movement direction of the robot under the specified coordinate system, which can be the axial direction of the coordinate system or the custom direction.
- **Max distance:** You can set the maximum distance the robot can run in the direction.

Tip

If the distance for the stop condition of the [Until] command is greater than the [Max distance], the [Max distance] will be the maximum extent of the motion.

- **Shared parameter:** using the relevant parameters set in the [MoveL] node. Users can also define the values of [Tool Speed] and [Tool Acceleration] in the [Direction] node.

Add [Direction]

1. Click [Move]  under the [Basic] program node to add the [Move] node to the program tree and set the [Move] node to [MoveL].

2. Click [Direction]  under the [Basic] program node to add a [Direction] node under the [MoveL] node.
 3. Select [Direction]  in the program tree, and set the parameters in the [Command] interface.
 4. Users can add [Until]  to set the stop condition for movement in the direction. Please refer to "[Until](#)" for the [Until] command.
-

Circle

The [Circle] command positions the TCP at the end of the robot, enabling it to move in an arc or a circle at a constant speed. The arc or circle is defined by a three-point method, with the robot moving sequentially from the starting waypoint to the ending waypoint to establish the Cartesian trajectory planning. When the robot moves in an arc or a circle, the starting and ending points will affect the pose change of the robot, and the maximum speed and acceleration in the movement process are similar to those in linear motion.

Tip

1. The [Circle] node needs to be used under the [MoveL] command.
2. The three points that form a circle must not be collinear; otherwise, the system will display an error message in a pop-up window.

The screenshot displays the AUBO Developer software interface. On the left is a vertical sidebar with icons for Home, Program, Configure, Move, and I/O. The main window is divided into two panes. The left pane shows the configuration for the 'CircleMove' node, which includes instructions on how to use waypoints and options for orientation (Fixed, Unconstrained, Via Point) and trajectory (Full Circle, Number of Turns). The right pane shows a program tree with steps: 1. Init Variables, 2. Robot Programming, 3. MoveL, 4. Circle (highlighted), 5. Waypoint_1, 6. Waypoint_2, and 7. Waypoint_0. At the bottom of the main window is a toolbar with various icons and a status bar showing '100%' and '0mm/s'.

Settings

- The [Circle] node requires setting of three waypoints: the starting point (waypoint_0), the ViaPoint (waypoint_1), and the EndPoint (waypoint_2). By default, all waypoints use the same blend radius. Please refer to "[Move](#)" for a description of the blend. As TCP passes through each waypoint at a constant speed, the robot will not wait for I/O or the operator's operations. At that time, I/O, or the operator's operation, may cause a protective stop of the robot.
- Select a mode for orientation relative to the circle (the mode defines the orientation of the tool):
 - Fixed: The tool pose remains unchanged with respect to the arc path coordinate system.

Set position/pose	Actual position/pose

- Unconstrained: The tool pose changes linearly, rotating around the fixed axis of space, transitioning from the starting point pose to the target point pose.

Set position/pose	Actual position/pose
	

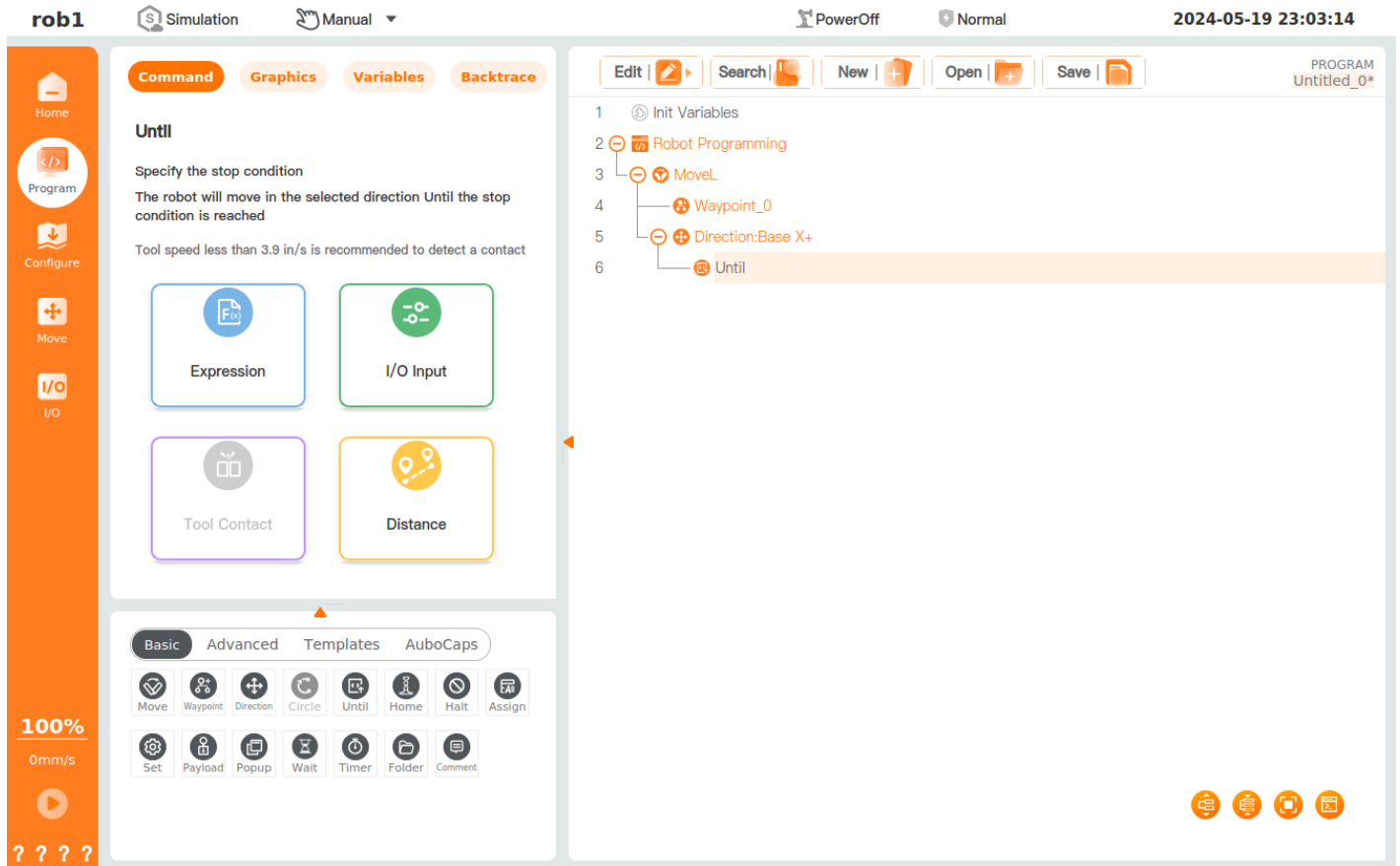
- Via Point: The tool pose starts from the starting point pose, passes through the via point pose, and changes to the target point pose.

Set position/pose	Actual position/pose
	

- Select the trajectory of the circle
 - Full Circle: When this option is checked, the robot will move in a complete circle determined by the three-point method. When this option is unchecked, the robot moves in a circle between the starting point, the via point, and the end point and returns to the starting point in a straight line after reaching the end point.
 - Number of Turns: You can set the number of turns for a complete circle motion.

Until

The [Until] command sets the motion stop condition. Multiple stop conditions can be added for a single action. When the first stop condition is met, the motion stops. Users can add the [Until] node under the [Direction] node or the [Waypoint] node to set the stop condition of the robot.



Set stop condition

- **Expression:** I/O, a variable or a script function is defined as the stop condition, and the robot will stop when the condition of the expression is met.
- **Tool Contact:** The robot moves until an impact on the tool is detected and stops at a defined deceleration.

Tip

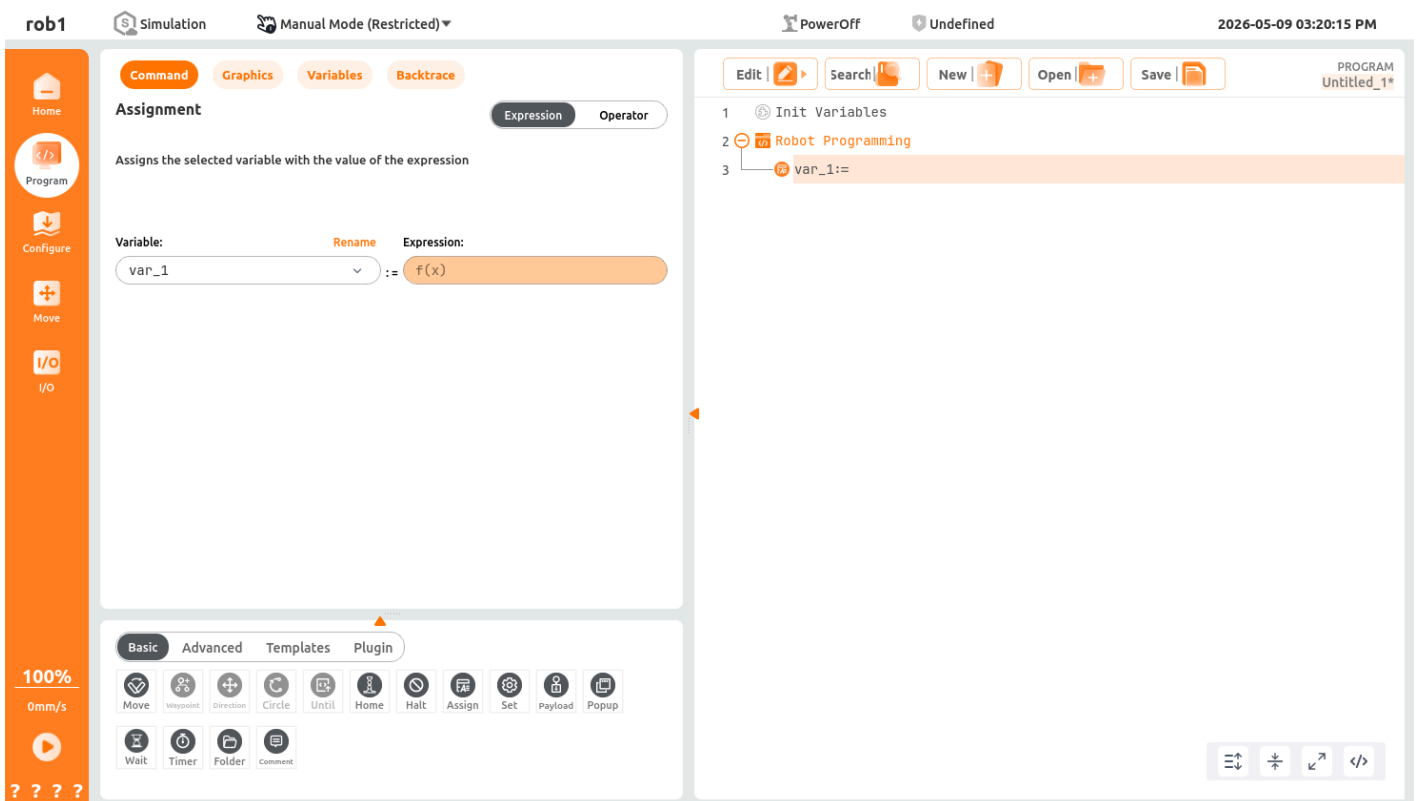
The default movement speed is too fast for contact detection. A faster movement speed can cause the robot to collide, triggering a protective stop, before the tool contact condition takes effect. - To avoid triggering a protective stop, a tool speed of less than 3.9 mm/s^2 is recommended.

- **I/O Input:** The [Until] stop feature of the robot is triggered by a digital input or an analog input. For example, if [Until] digital input DI02 is set at a low level, the robot will stop when it receives a low-level signal of DI02.
- **Distance:** The robot moves along the specified direction and stops at [Distance].
- **Stop at this point:** The robot will stop after reaching the waypoint.

- Blend with radius: If there are other waypoints after the node, the robot will not stop but run smoothly to the next node.
- Reach waypoint: This option appears when [Until] is inserted directly under the [Waypoint] node, and other actions can be added to the [Until Reach] node. When the robot reaches the [Waypoint] node, it performs the actions in the [Until Reach] node.
- Add action: When this option is checked, you can add an action when the [Until] condition is met. If this option is not defined, the program executes the next node in the program tree.

Assignment

The [Assignment] command assigns a program variable with a value. It assigns the selected variable with the value of the expression or with the value entered by the machine operator.

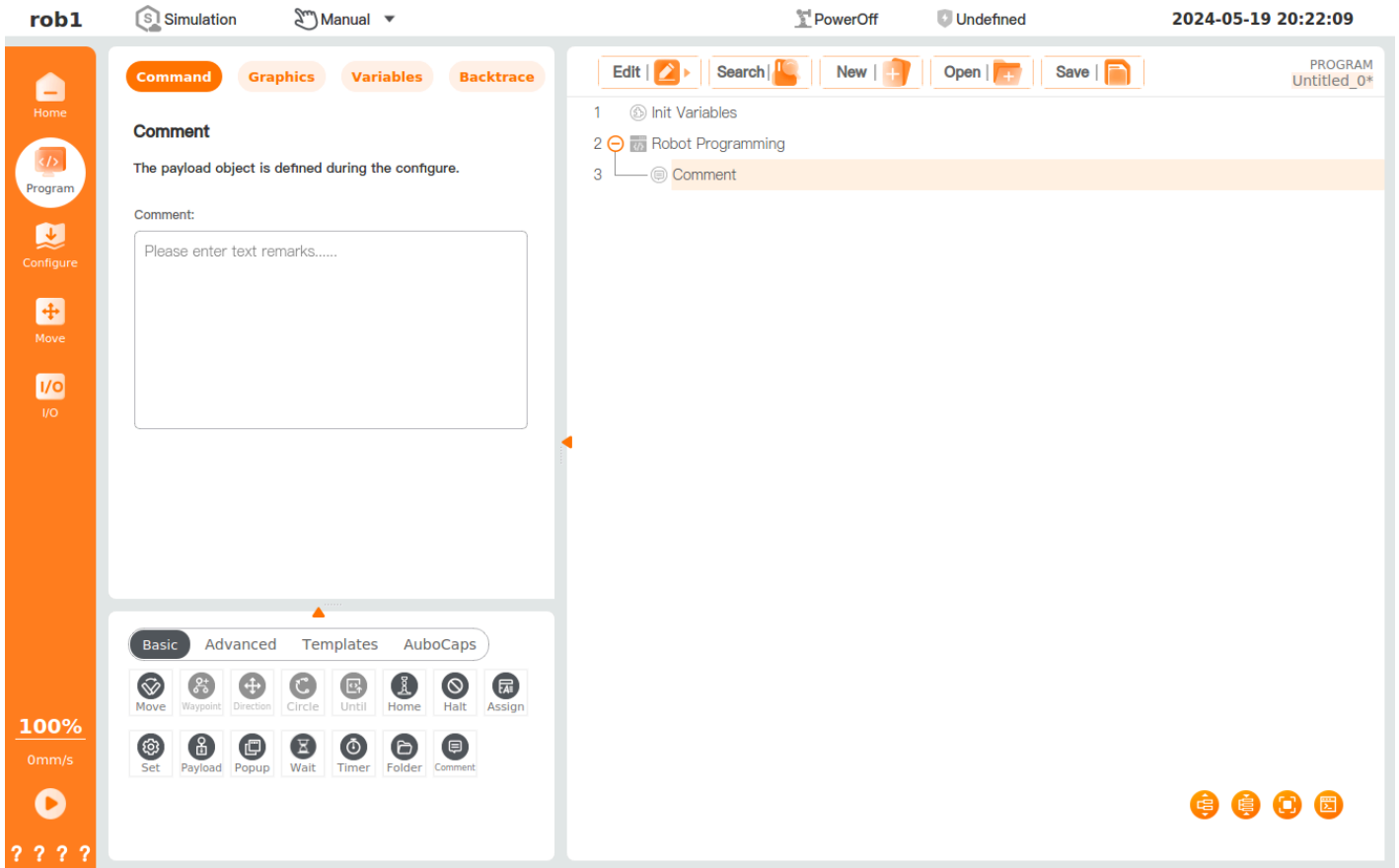


Set assignment type

- Expression: You can set the variable to an expression or create a new variable.
- Operator: You can set the type of input variable; when the program runs to this node, the operator needs to manually input the variable for value assignment.
 - Operator prompt: You can add prompts to help the operator enter the correct variable.

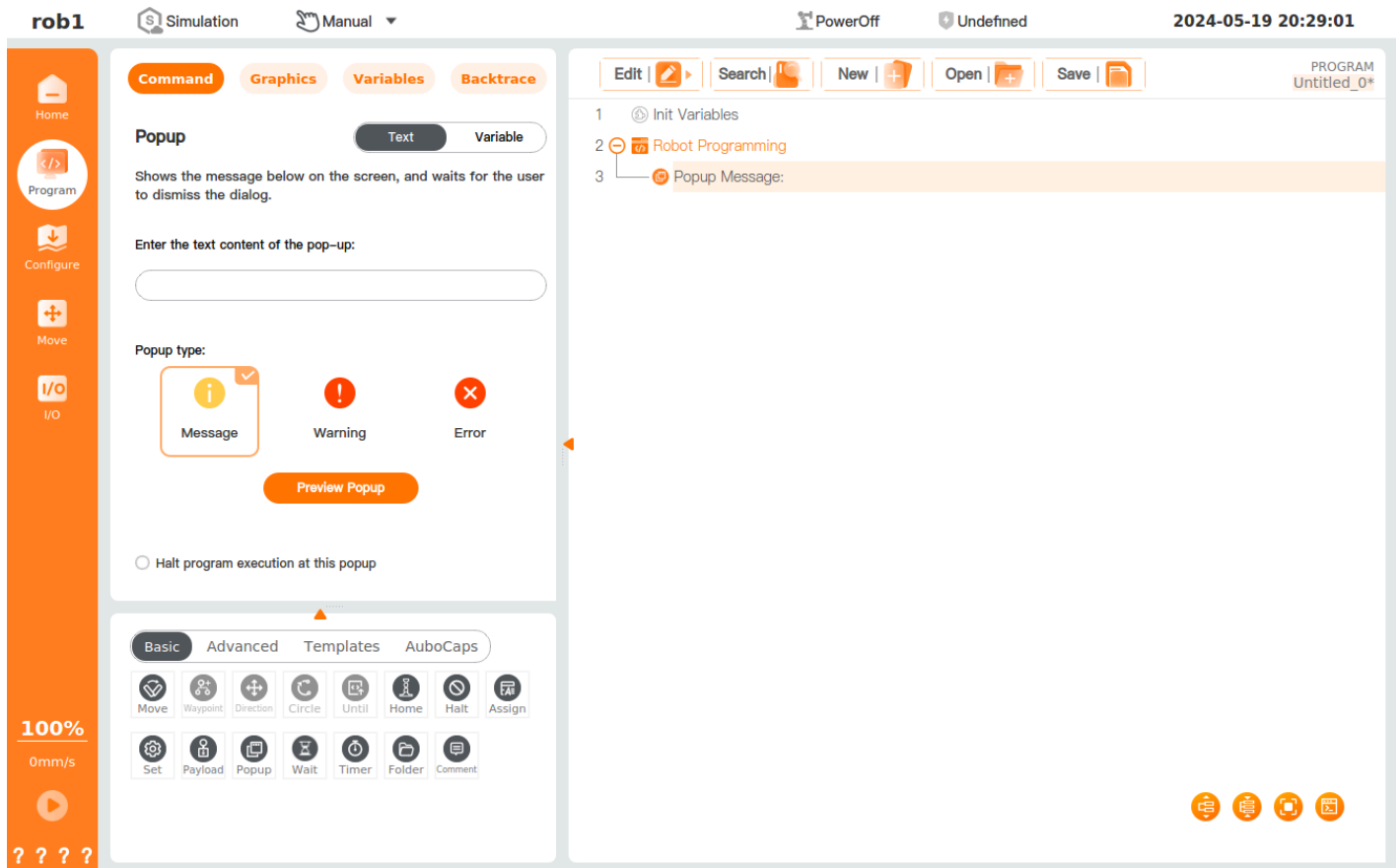
Comment

The [Comment] command is designed to help users record the comment information, and the comment text will not be displayed when the program runs.



Popup

The [Popup] command allows you to specify a message, which will be displayed on the screen when the program runs to the node.

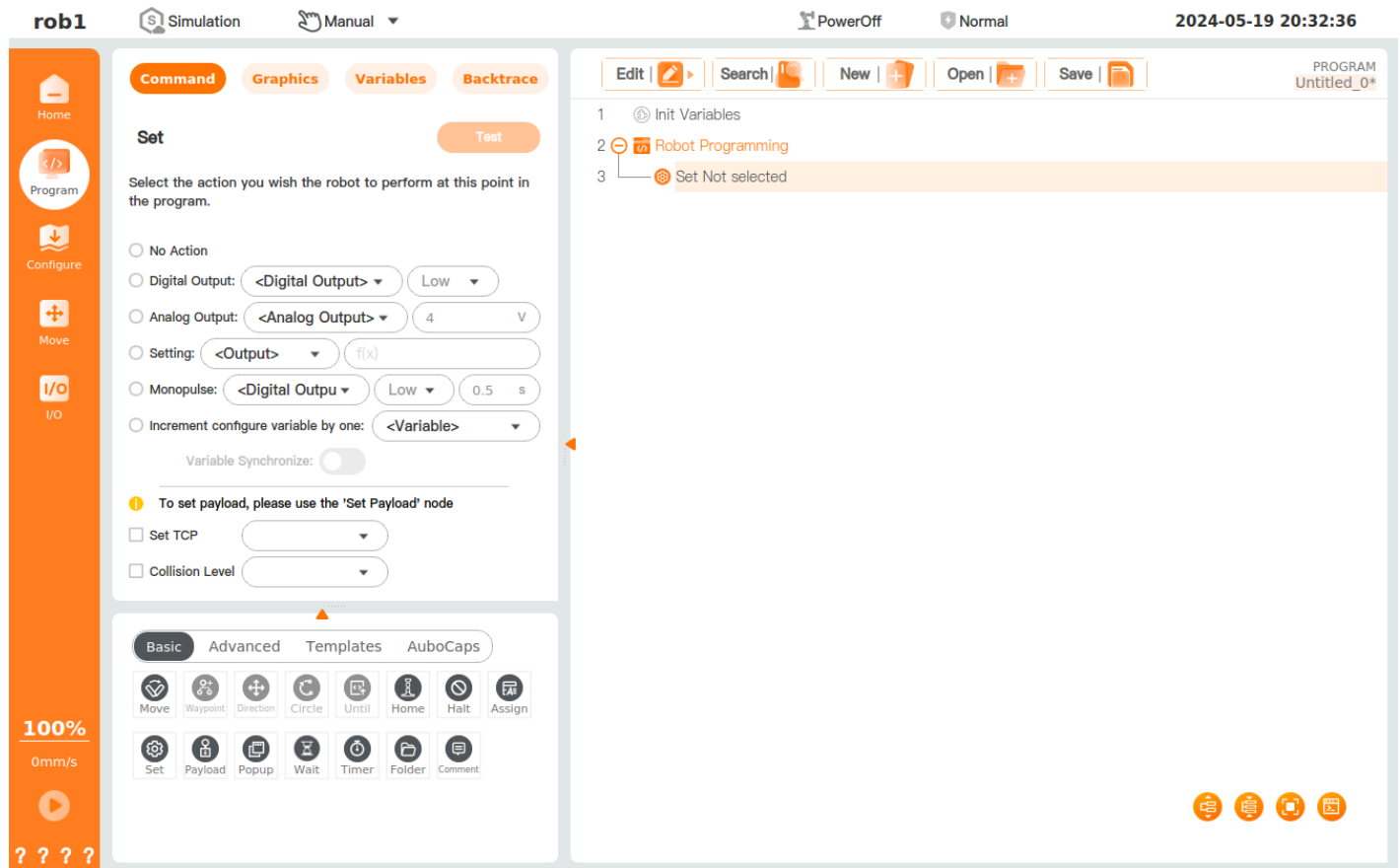


Settings

- **Text:** Inputting the text information and clicking [Preview Popup] to view the popup effect. The text popup is divided into three types: message, warning, and error.
- **Variable:** You can select the variables, and when the program runs to the node, the value of the selected variable will be given in a popup, waiting for the user to dismiss the dialog. The variable popup is divided into three types: message, warning, and error.
- **Halt program execution at this popup:** If this option is checked, when the program runs to the [Popup] node, a message will be displayed in a popup on the screen, and the program execution will be halted.

Settings

The [Set] command allows you to add the action you wish the robot to perform at this point in the program.



- No Action: No operation.
- Digital Output: You can set the digital output signal here.
- Analog Output: You can set the analog output signal here.
- Setting: You can set the expression of the digital or analog output. Only true/false or 1/0 is supported by the digital output port, where only 1/0 is supported by the Modbus port.
- Monopulse: You can set the time for the digital output port to continuously output high/low levels.
- Increment configure variable by one: The configure variable is incremented by one automatically. Please refer to "[Variables](#)" for the settings of the configured variable.
- Variable Synchronize: Please refer to "Synchronization" in "[Assignment](#)".
- Set TCP: You can select the TCP here. Please refer to "[TCP](#)" for the creation and settings of TCP.
- Collision Level: You can set the collision level here.

- Test: The settings of this node are performed once. When [Setting] or [Increment configure variable by one] is selected, the node cannot be tested.
- Timeout Termination:
 - When this option is checked and a non-zero time is set, if the bus write operation is not completed within the set time, the program will be terminated immediately and a runtime error pop-up will be displayed.
 - When unchecked or the time is set to `0.00s`, the operation returns directly without checking whether the write was successful.

Note

- Usage: This extended option [Timeout Termination] is only available when the output is set to a Modbus address.
- Value Range: `0 ~ 86400.00s` (maximum timeout of 24 hours).

Payload

The [Payload] command allows you to select an existing payload or select [Custom Load] to enter a value. See "[Payload](#)" for the settings of the existing payload.

The screenshot displays the AUBO Developer software interface. On the left is a vertical orange sidebar with icons for Home, Program, Configure, Move, and I/O. The main area is divided into two panels. The left panel, titled 'Select Payload', contains a 'Command' tab and a 'Payload' section. The 'Payload' section has a 'Set total payload' dropdown set to 'Payload', a 'Mass' input field set to '0.000 Kg', and a 'Center of Gravity' section with 'Cx', 'Cy', and 'Cz' input fields all set to '0 mm'. A 'Set Now' button is at the bottom. The right panel shows a program editor with a list of steps: 1. Init Variables, 2. Robot Programming, and 3. SetPayload: Payload. The 'SetPayload: Payload' step is highlighted. The top of the interface shows 'rob1', 'Simulation', 'Manual', 'PowerOff', 'Undefined', and the date/time '2024-05-19 20:28:40'. The bottom of the interface has a '100%' zoom indicator and a '0mm/s' speed indicator.

Settings

- Set Now: click to execute the setting of the node once.

Tip

Payload setting is required for picking up and releasing materials by the robot.

Wait

The [Wait] command allows you to set the waiting time or digital input signal.

The screenshot displays the AUBO Developer software interface. On the left is an orange sidebar with navigation icons for Home, Program, Configure, Move, and I/O. The main window is divided into several sections. At the top, there are tabs for 'Command', 'Graphics', 'Variables', and 'Backtrace'. Below these, the 'Wait' command configuration is shown, with the instruction 'Please select what should trigger the robot's next action.' The options are:

- ☐ No Wait
- ☐ Wait: 0.01 s
- ☐ Wait for Digital Input: <DI.Input> Low
- ☐ Wait for: <An.Input> 4 V
- ☐ Wait for: f(x)

 Below the configuration options is a 'Basic' tab with a grid of icons for various commands: Move, Waypoint, Direction, Circle, Until, Home, Halt, Assign, Set, Payload, Popup, Wait, Timer, Folder, and Comment. The 'Wait' icon is highlighted. On the right side of the interface, a program editor shows a sequence of steps: 1. Init Variables, 2. Robot Programming, and 3. Wait. The 'Wait' step is currently selected and highlighted in orange. At the bottom right, there are four small circular icons representing different robot states or actions.

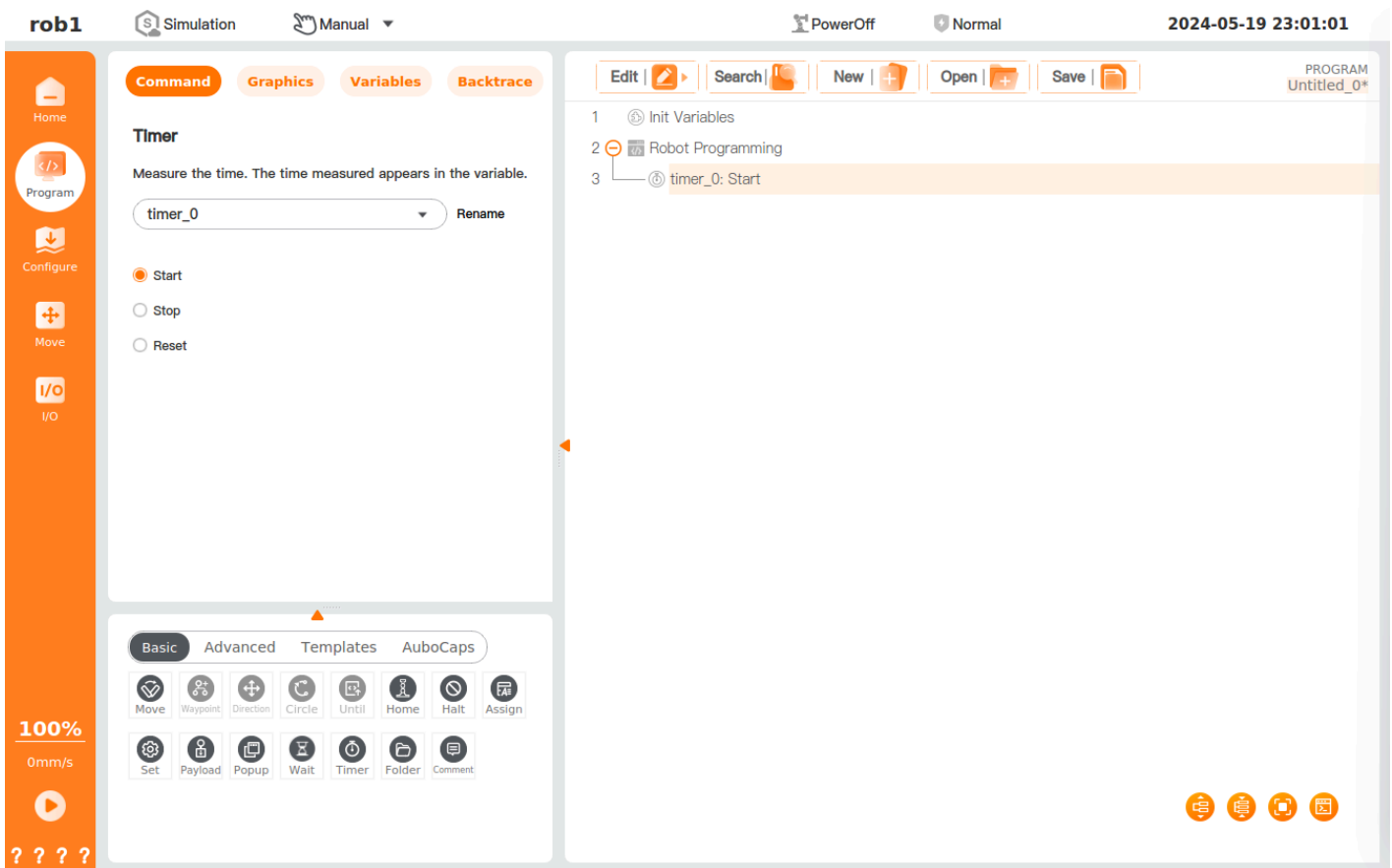
Settings

- No wait: When this option is checked, the robot will not wait.
- Wait: The robot is programmed to wait for the set duration before resumed operation.
- Wait for Digital Input: The robot is programmed to wait for the set digital signal before resuming operation.
- Wait for Analog Input: The robot is programmed to wait for the set analog signal before resuming operation.

- Wait for the expression: The robot is programmed to wait for the expression to be established before resuming operation.

Timer

The [Timer] command is designed to record the execution time of the program, which will be displayed in the [Variable] interface.

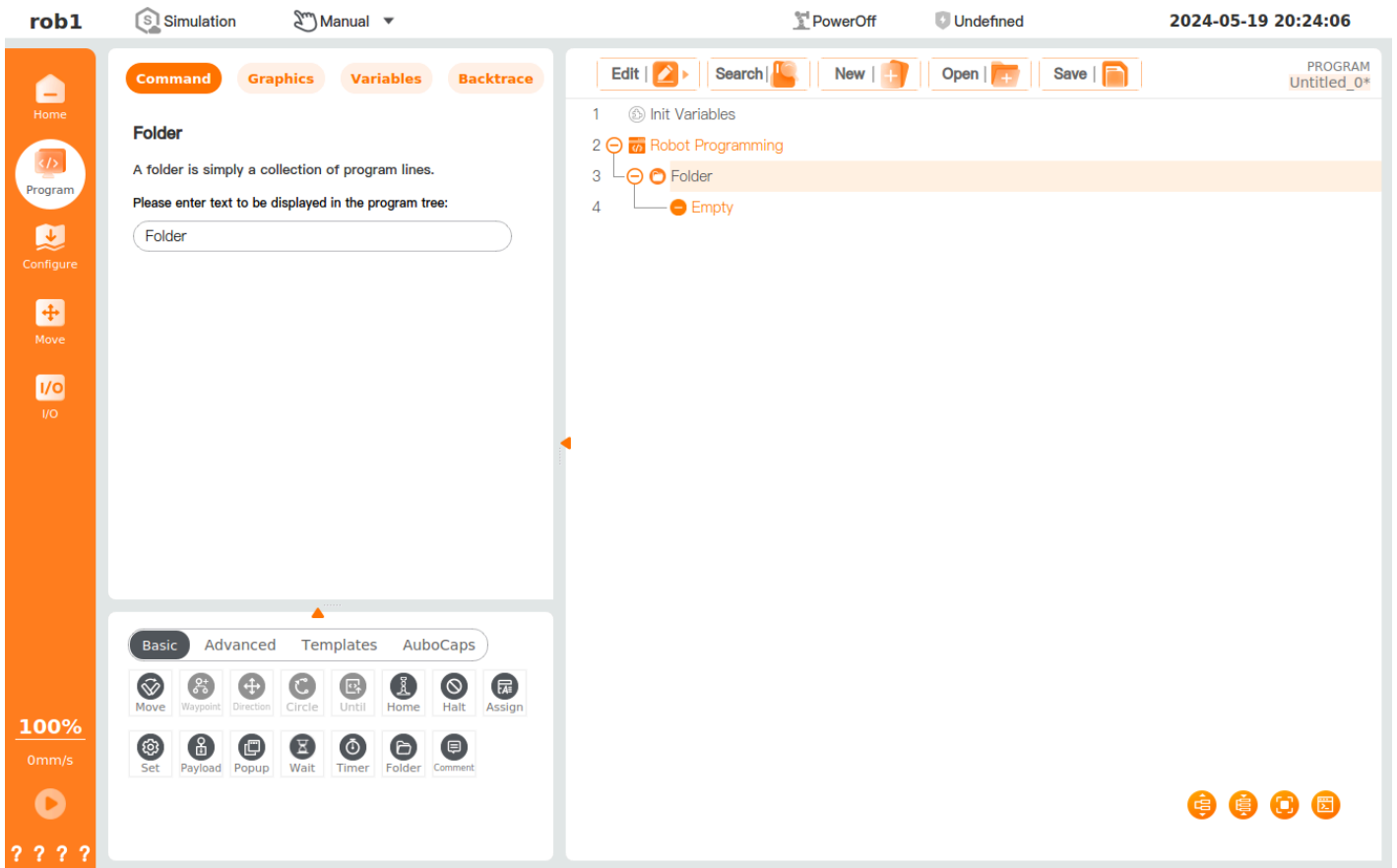


Settings

- You can select a timer in the dropdown box in the [Timer] interface or select [New] in the dropdown box to create a new timer.
 - The timer is named "timer_0" by default, and the default value is incremented by 1 each time a new one is created.
 - Rename: click to customize the timer name.
- [Timer] status
 - Start: You can start the timer here.
 - Stop: You can stop the timer here.
 - Reset: You can reset the timer here.

Folder

The [Folder] command is used to organize the program and allows you to add annotations to a program to enhance the clarity of the program tree, making it more readable and navigable. The folder has no effect on the program and its execution.

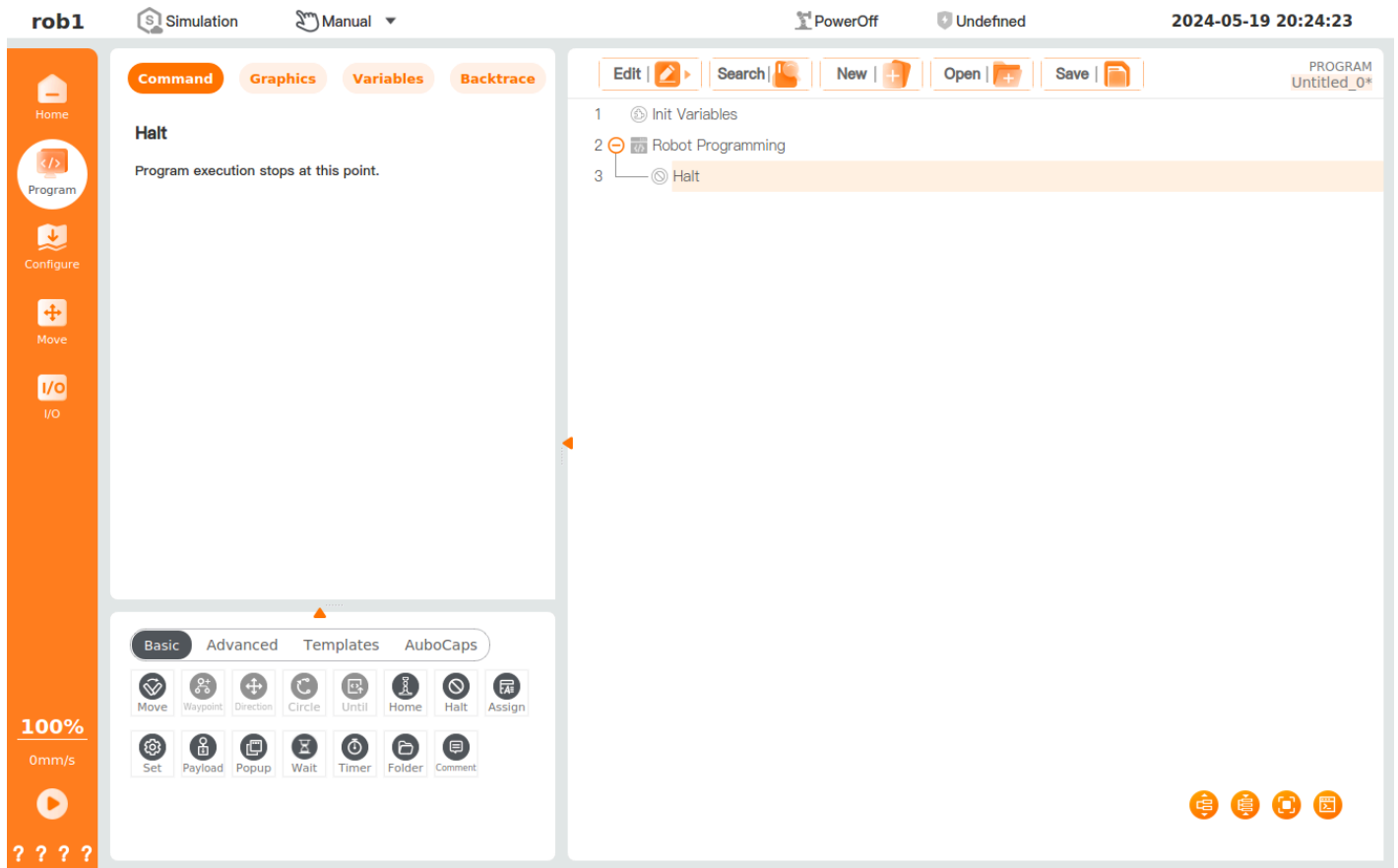


Settings

- Textbox: Enter text in the textbox, and the text will be displayed in the program tree.

Halt

The [Halt] command is designed for stopping the program's execution. The program stops after reaching the [Halt] node.



Home

The [Home] command is designed to move the robot to the home node. Please refer to "[Home](#)" for position setting of the home node. If the corresponding safety I/O is set: when the robot moves to the Safe Home position and keeps stopping, the safety I/O will output a "1" signal; if the robot does not stop at the Safe Home position, the safety I/O output is "0." See [I/O Configuration](#) for the settings of safety I/O.

rob1 Simulation Manual PowerOff Normal 2024-05-19 23:56:30

Command Graphics Variables Backtrace

Home

Move to the Home Position defined in the configure.

joint speed 60 °/s

joint accele 80 °/s²

time 0 s

☒ Use joint angles

Joint Move Here

Line Move Here

Assign Output to the Safe Home position in the Safety Configuration

Safe Home	Outputs
Not configured	Not assigned

Basic Advanced Templates AuboCaps

Move Waypoint Direction Circle Until Home Halt Assign

Set Payload Popup Wait Timer Folder Comment

1 Init Variables

2 Robot Programming

3 Home

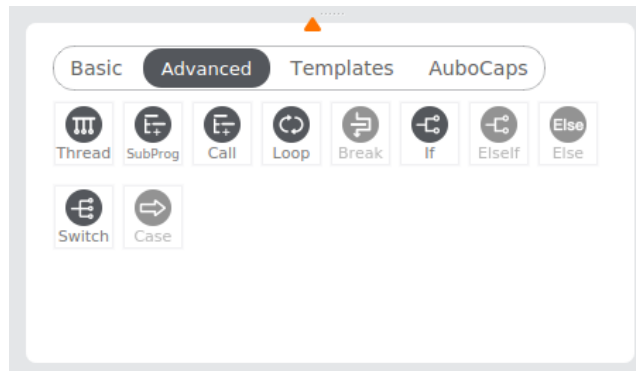
PROGRAM Untitled_0*

Settings

- MoveJ.
 - Joint speed/joint acceleration: You can set the joint speed/joint acceleration for movement to the [Home] position.
 - Time: You can set the time for movement to the [Home] position.
- Joint Move Here/Line Move Here: click to access the [Move] interface, and press and hold [Auto] to move the robot to the [Home] position.

Advanced Program Node

Advanced program nodes can realize functions such as loop, determination, call, multi-thread, etc., improve programming efficiency, reduce the probability of program errors, and make the program more flexible and convenient.

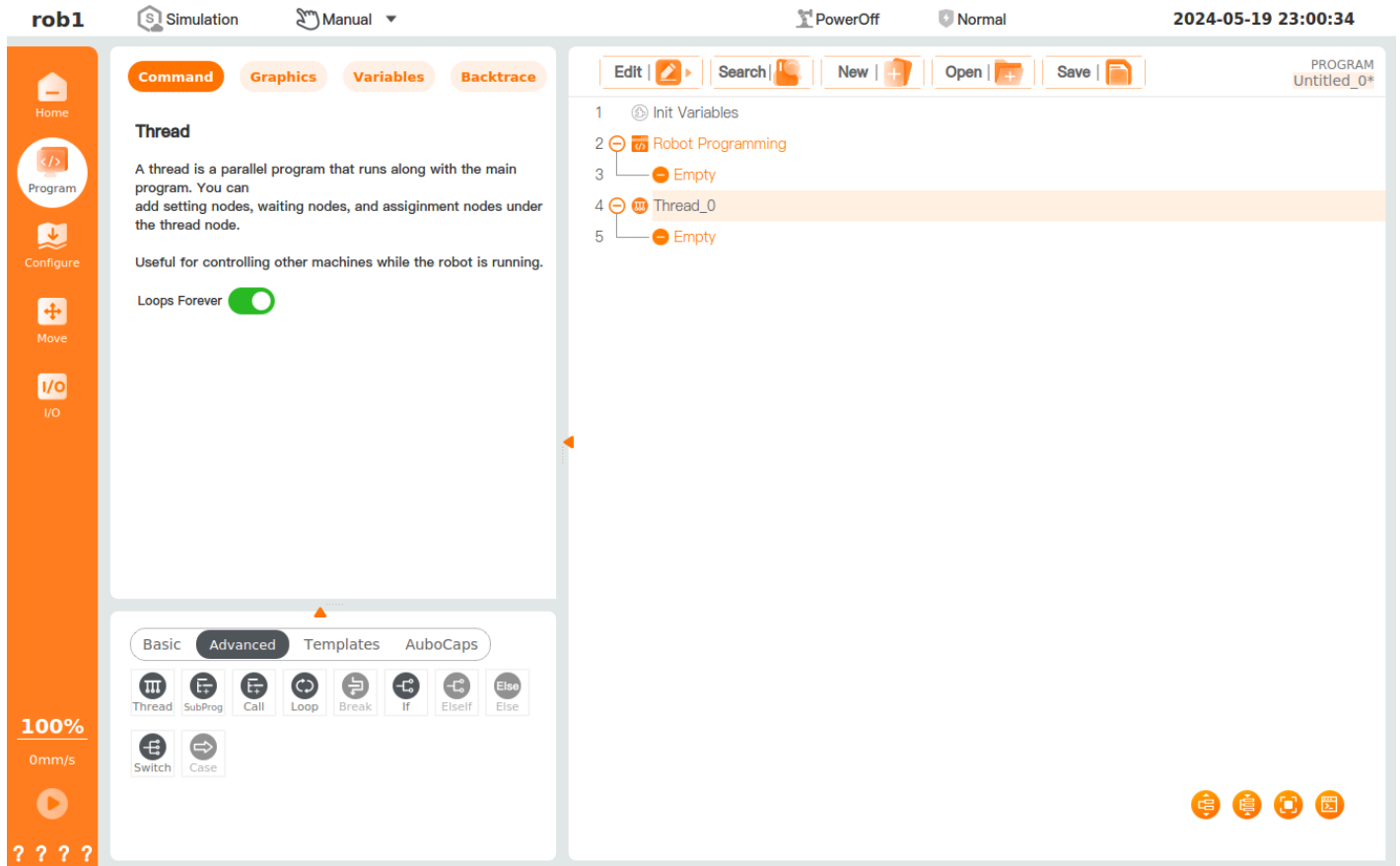


Thread

The [Thread] command is designed to create a multi-thread, that is, a program that can be parallel with the master program. The thread can be used to control other external devices that are not related to the robot arm.

Tip

1. Under the **Thread** node, you can add nodes not related to motion, such as **Set**, **Wait**, **Assignment**, etc.
2. Adding nodes related to motion under the **Thread** node, such as **Move**, **Waypoint**, etc., may cause robot operation conflicts and trigger other safety risks.
3. It is recommended to avoid the use of multi-thread whenever possible. If multi-threading must be used, please pay attention to the parallel logic and timing matching of the primary thread and secondary thread.



Settings

- Infinite loop: When this option is checked, the program in the thread can run in an infinite loop until stopped.

Subprog

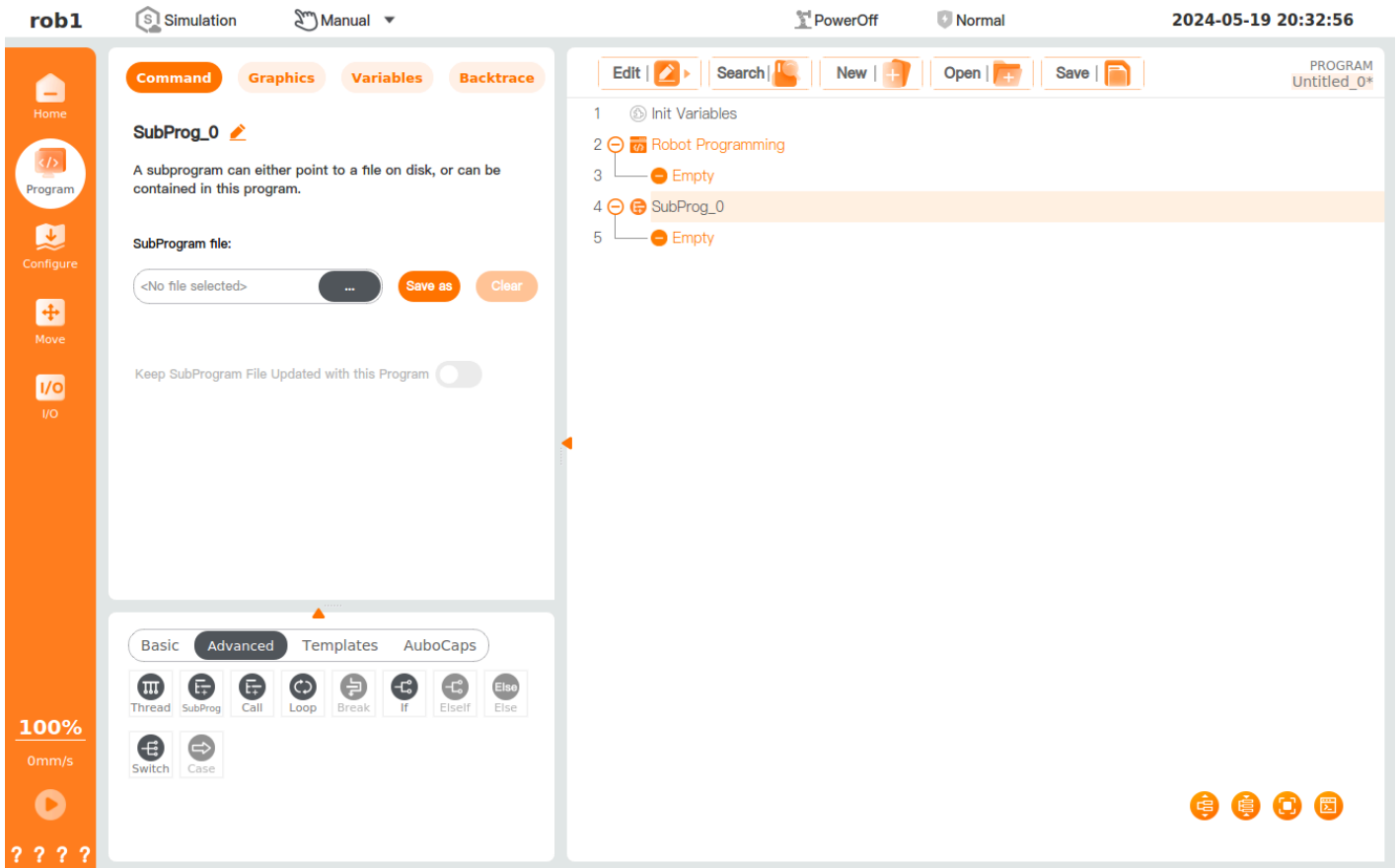
The [SubProg] command is designed to create a subprogram, which is also known as a subproject. It is often used in conjunction with the [Call] node.

Characteristics of subprogram:

- The subprogram can be either a control program or a controlled program.
- The data contained in the subprogram only represents the information present when the subprogram is activated within the master program.
- The subprogram can be called and run at multiple positions within multiple programs through the [Call] node.

Tip

When the subprogram is called, you need to write a program in the [SubProg] node or click  to load the written program and then use the [Call] node to call the subprogram.



Settings

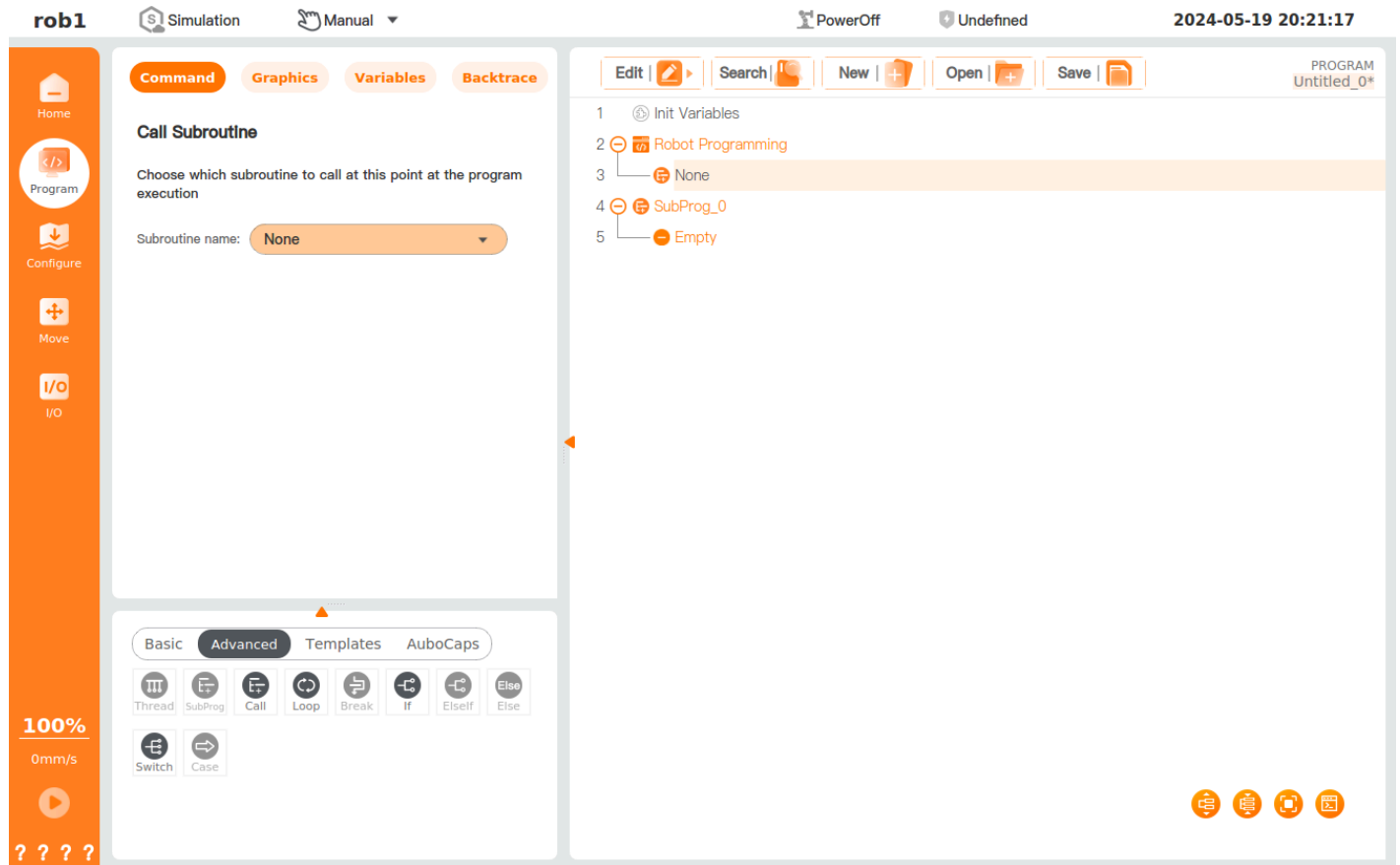
- : Rename the subprogram in the master program.
- : Load the program file.
- Save as: You can save the subprogram as a separate file on the disk.
- Clear: You can clear the open files within the subprogram.
- Keep Subprogram files updated with this program: When this option is checked, changes in the open subprogram will be automatically saved to the subprogram file.

Tip

This option is only available in a subprogram that has been saved as a file.

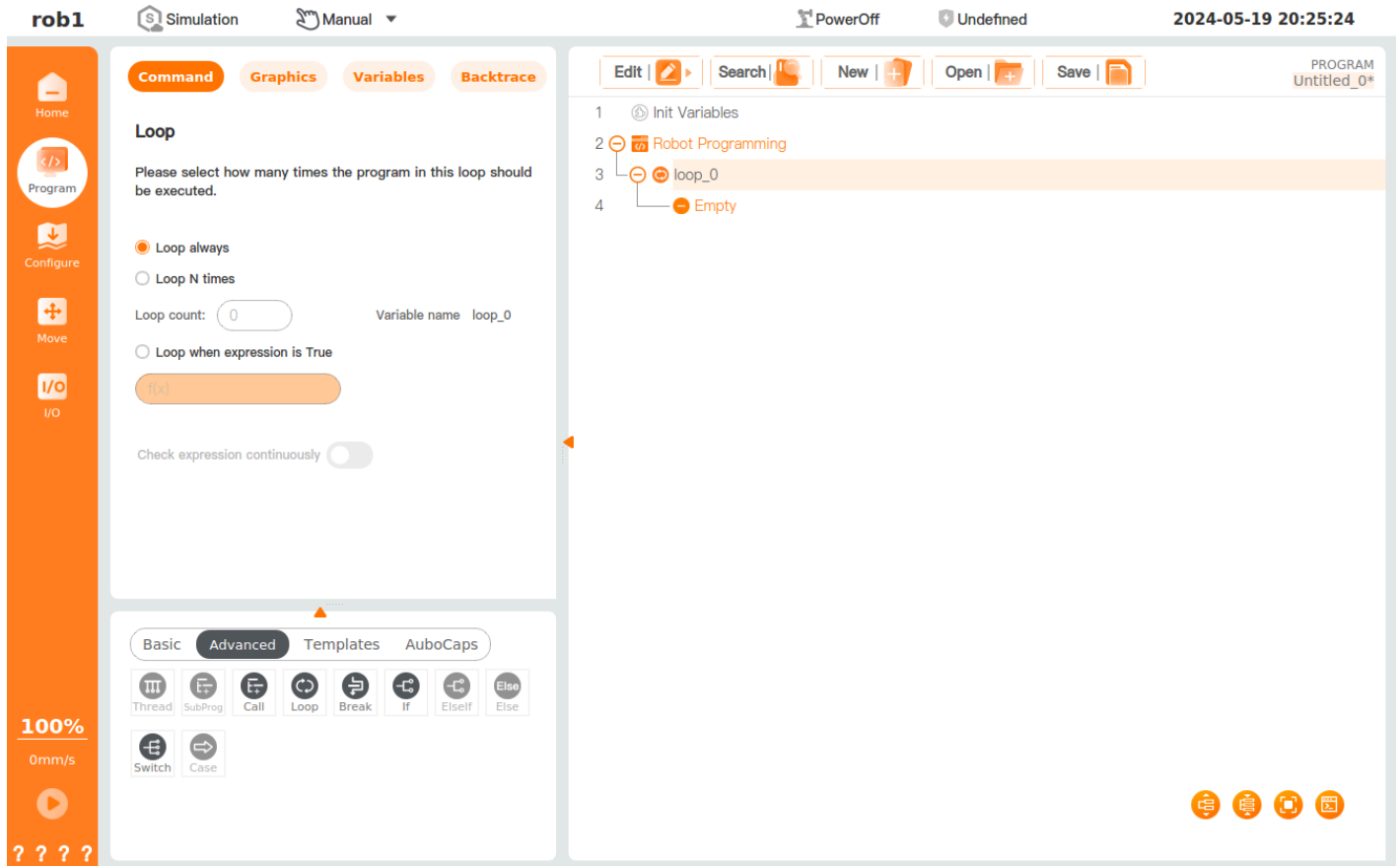
Call

The [Call] command is designed to call the subprogram. When the program executes to this node, it jumps to the called subprogram.



Loop

The [Loop] command is designed to create a loop program block.

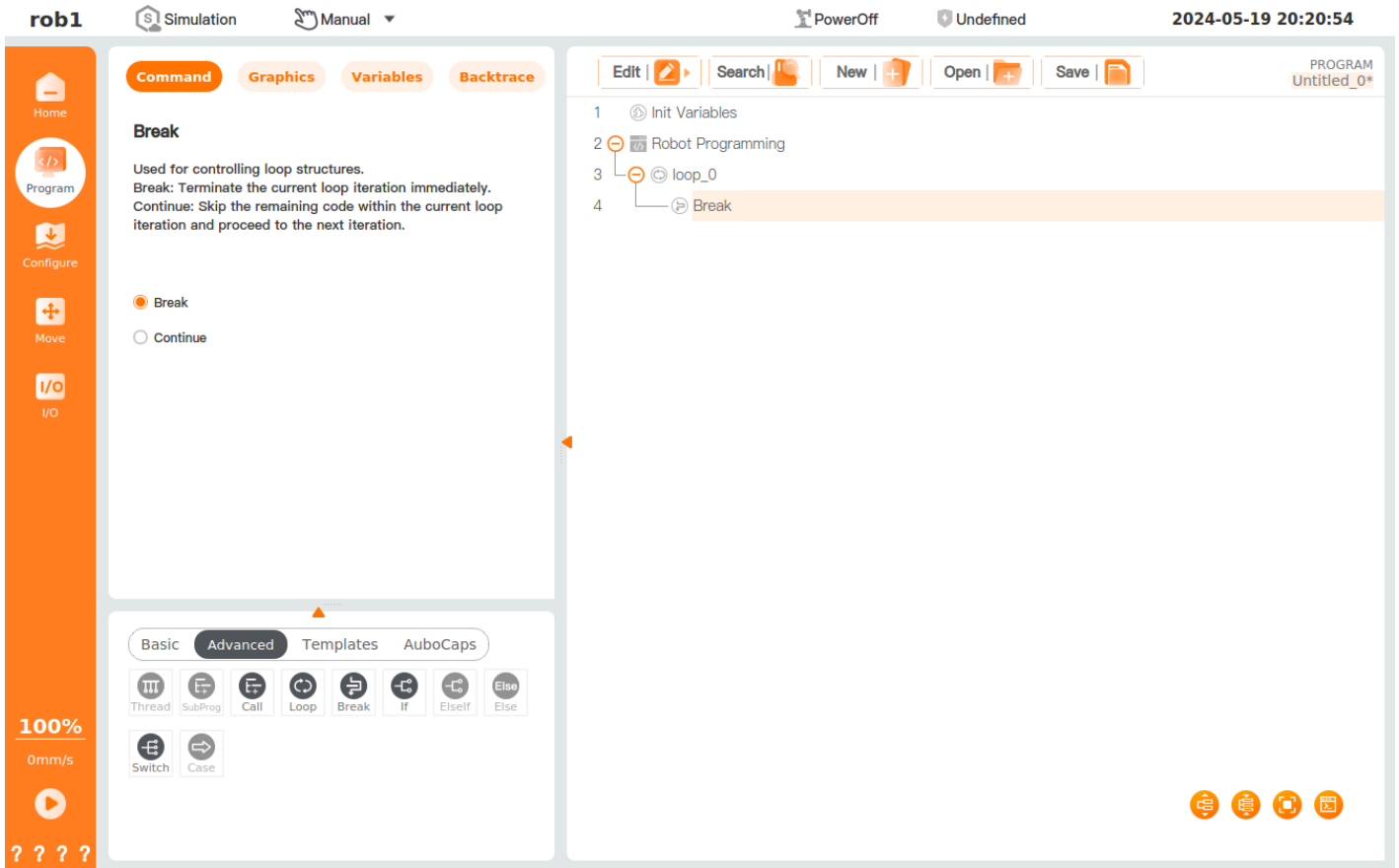


Settings

- **Loop always:** When this option is checked, the nodes in the program block will run in an infinite loop.
- **Loop N times:** If this option is checked, after setting of the "Loop count," where N is the number of loops, the nodes in the program block will run in a loop for N times. The program creates a special loop variable (*loop_0* as shown in the figure) that can be used in the expression inside the loop. After entering the loop, the loop variable starts counting from 0 to $N - 1$, ending the loop.
- **Loop when expression is true:** If this option is checked, after setting the expression, when the program runs to the node, whether the expression is true will be determined; if the expression is true, the loop will be entered and the node in the program block will be executed; if the expression is determined to be false, this loop will be skipped.
- **Check expression continuously:** If this option is checked, the expression will be continuously checked during execution of the loop program block, and once the expression is determined to be false, the [Loop] program block will be skipped at any time.

Break

The [Break] command is designed to break the loop.



Set break mode

- Break: Terminate the current loop iteration immediately, and the program runs the next program node of the **Loop** node.
- Continue: Skip the remaining code within the current loop iteration and proceed to the next iteration. If the **Loop** node does not require variable or expression determination, the next loop will be entered immediately. If the **Loop** node requires variable or expression determination, determination will be made first. If it is true, the next loop will be entered; if it is false, the loop will be ended.

If/Elself/Else

The [If] command is designed to create a condition for the determination of the program block.

The [ElseIf] command is designed to create a condition for the determination of the program block under the "If" node.

The [Else] command is designed to create a program block under the **If** or **ElseIf** node.

1. Settings

If setting:

- Click the expression $f(x)$ in the **If** interface to edit the condition expression.
- Check expression continuously: If this option is checked, the condition expression will be continuously checked during execution of the **If** program block, and once the expression is determined to be false, the **If** program block will be skipped at any time.

ElseIf setting:

- Click the expression $f(x)$ in the **ElseIf** interface to edit the condition expression.
- Check expression continuously: If this option is checked, the condition expression will be continuously checked during execution of the **ElseIf** program block, and once the expression is determined to be false, the **ElseIf** program block will be skipped at any time.

2. Application

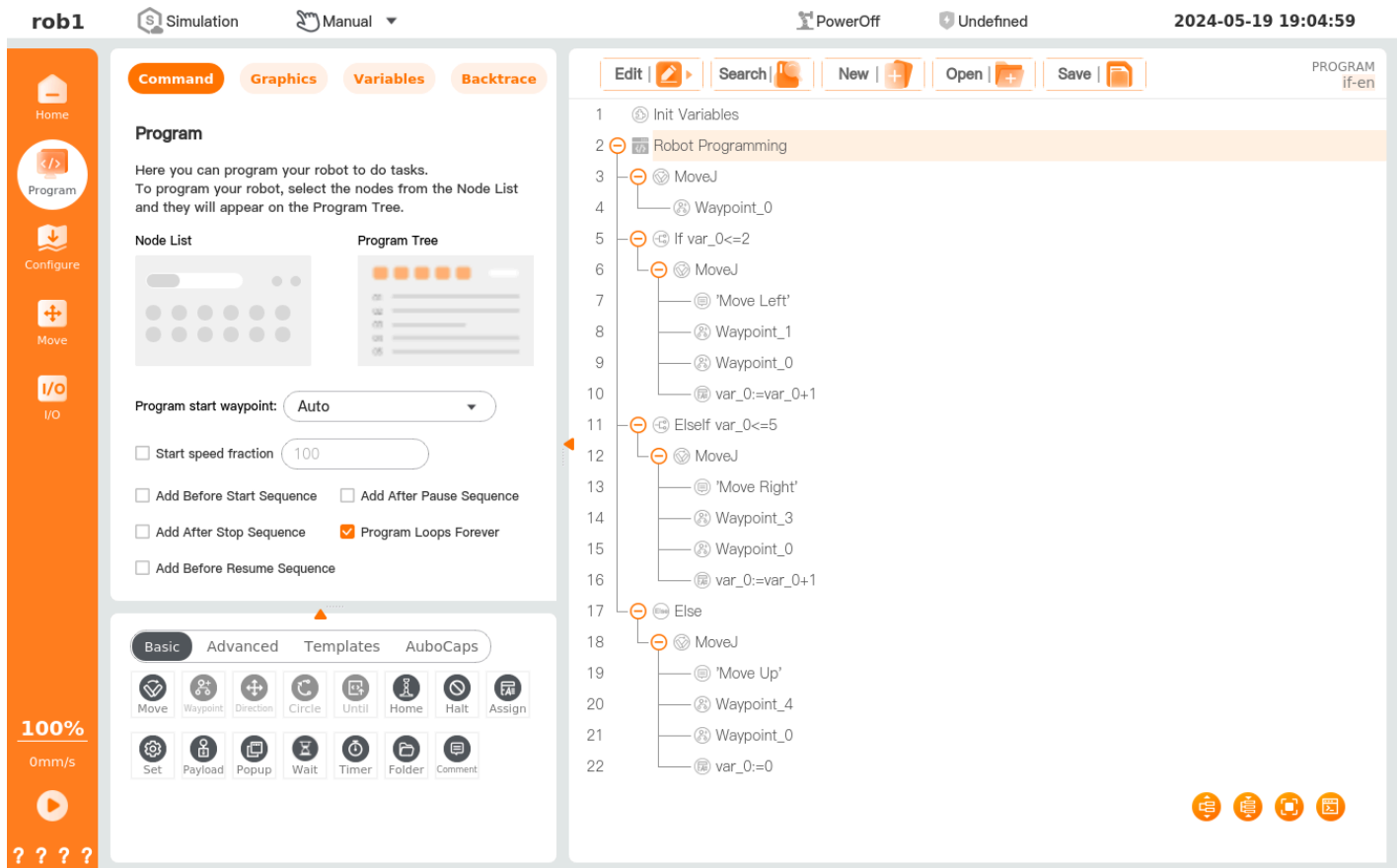
- If** : If the **If** condition is true, the **If** program block is executed.
- If...Else...** : If the **If** condition is true, the program inside this **If** node will be executed; otherwise, the program in the **Else** node will be executed.
- If...Else...If...Else...** : If the **If** condition is true, the program inside this **If** node will be executed; otherwise, whether the "ElseIf" condition is true will be determined. If the "ElseIf" condition is true, the program inside the "ElseIf" node will be executed; otherwise, the program inside the "Else" node will be executed.

Example:

Set an initial variable `variable_0` . The program starts execution and runs to `waypoint_0` .

Determine whether `variable_0` is less than or equal to 2. If `variable_0` is less than or equal to 2, execute **MoveJ** with the comment "left." If `variable_0` is greater than 2, determine

whether `variable_0` is less than or equal to 5. If `variable_0` is greater than 2 and less than or equal to 5, execute `MoveJ` with the comment "right." If `variable_0` is greater than 5, execute `MoveJ` with the comment "Up."



Switch/Case/Default

The [Switch] command is designed to create multiple switches and control the flow of the program according to the value of the variables. You can set whether to add a `Default` program block to the program tree. For "Synchronize," please refer to the description of "Synchronize" in [Assignment](#)."

The [Case] command is designed to create a `Case` program block under `Switch` .

A `Case` node can be inserted under a `Switch` node only.

Application

- `Switch...Case...Default...` : Execute different `Case` program blocks by determining the conditions. You can enter the selection and determination condition expression in the "Switch" interface and enter the matching value in the "Case" interface. When the program

runs the "Branch" node, it first calculates the value of the expression of the "Branch" node, which is then compared with the values of the "Condition" under the "Branch": if the same value is matched in all the conditions, the "Condition" program block will be executed; If no matching "Condition" value is found, the "Default" program block under that "Branch" will be executed.

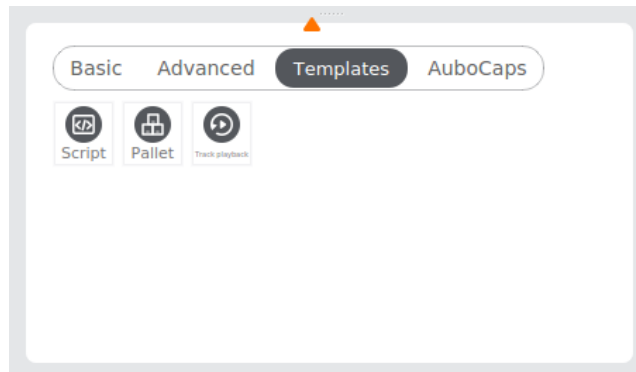
- A "Branch" can include 0 to several "Condition" program blocks.
- A "Branch" can include 0 to 1 "Default" program blocks, which can be added by checking the "Add default condition" in the "Branch."

Example:

Set an initial variable `variable_0` . Access the program and run it to `waypoint_0` . Determine whether `variable_0` is equal to 0. If `variable_0` is equal to 0, run `MoveJ` with the comment "Right"; if `variable_0` is not equal to 0, determine whether `variable_0` is equal to 1. If `variable_0` is equal to 1, run `MoveJ` with the comment "Left." If `variable_0` is neither equal to 0 nor equal to 1, run `MoveJ` with the comment "High."

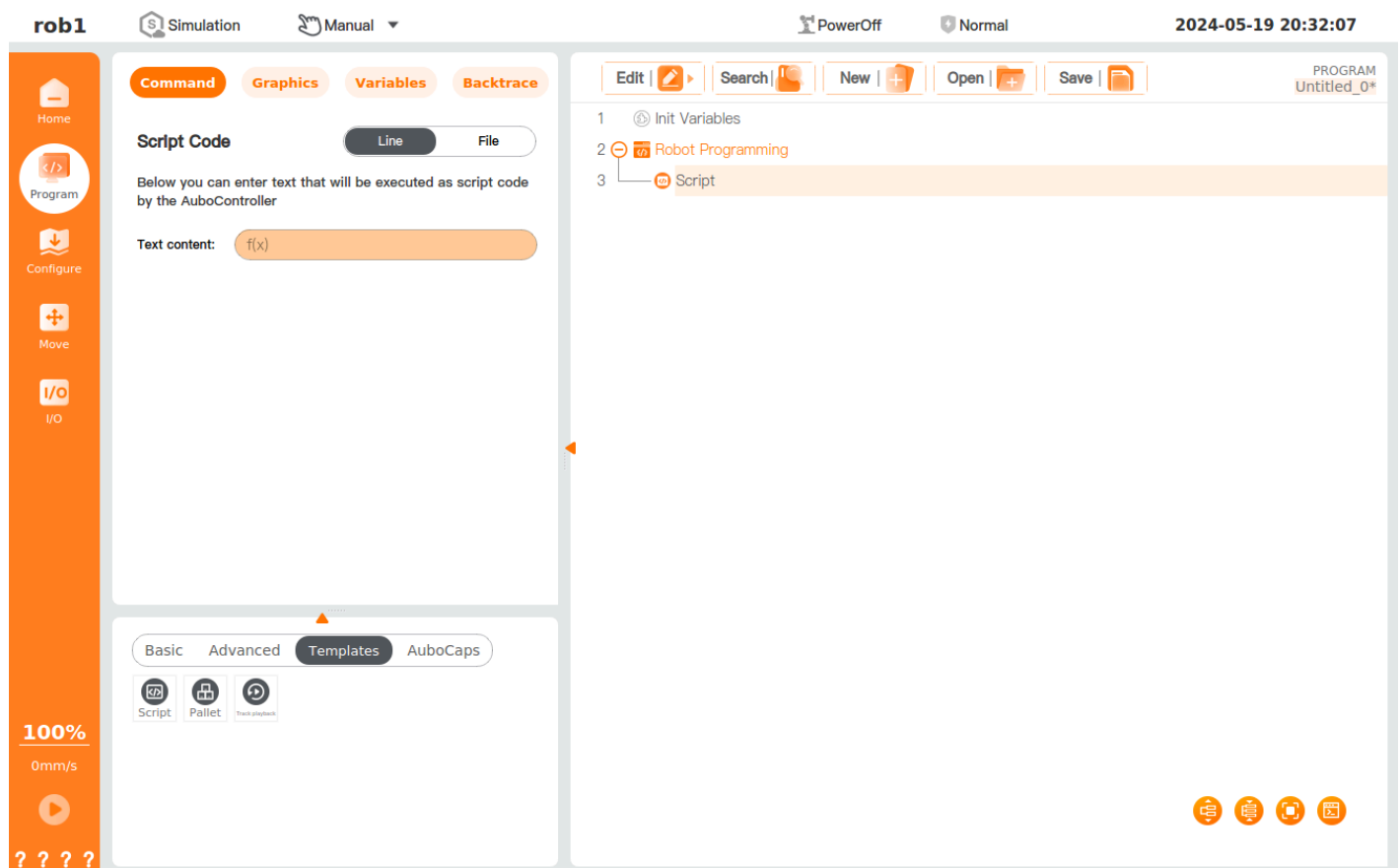
Template Program Node

The template program node is a built-in node designed to facilitate user access to complex functions, including scripts, palletes, track playback, and other functions. The template program node can be applied in some specific scenarios; for example, the "script" function can make it easier for users to write a script; and the "pallet" function is used for palletizing and depalletizing.



Scripts

The [Script] command allows the addition of a script line or a script file.



Settings

- Line: Allow entering the text to be executed as script code through the editor.
- File: Allow opening or saving the script file and editing, clearing, and saving the opened script file.

Tip

1. Before editing the scripts, please open the saved scripts or [Save As] the scripts, and then click [Edit] to edit the scripts. Failing to do so will prevent script editing.
2. The script editor comes with a syntax checker, which is in a white background if the script code entered is correct and in yellow if it is incorrect. To ensure the script is saved to the script file, the entered script code must be correct.

Pallet

The [Pallet] command allows the creation of a palletizing or depalletizing task. Users can accomplish palletizing and depalletizing tasks through parameter configuration. This command serves as a template that guides the robot to perform repeatable operations on different items in multiple layers of different patterns.

rob1 Simulation Manual PowerOff Undefined 2024-05-19 20:28:15

Command Graphics Variables Backtrace

Pallet

• Grid Pattern • CornerItems

Row: 2 Column: 2

row: from Corner 1 to Corner 2

column: from Corner 1 to Corner 4

Corner 1 Move Here

Corner 2 Move Here

Corner 3 Move Here

Corner 4 Move Here

After setting the four corners of the grid, the palletizing node automatically generates the operations to be performed at each point. The starting point of movement is the Tool Action point in the generated program tree.

Basic Advanced **Templates** AuboCaps

Script Pallet Track playback

100% 0mm/s

PROGRAM Untitled_0*

1 Init Variables

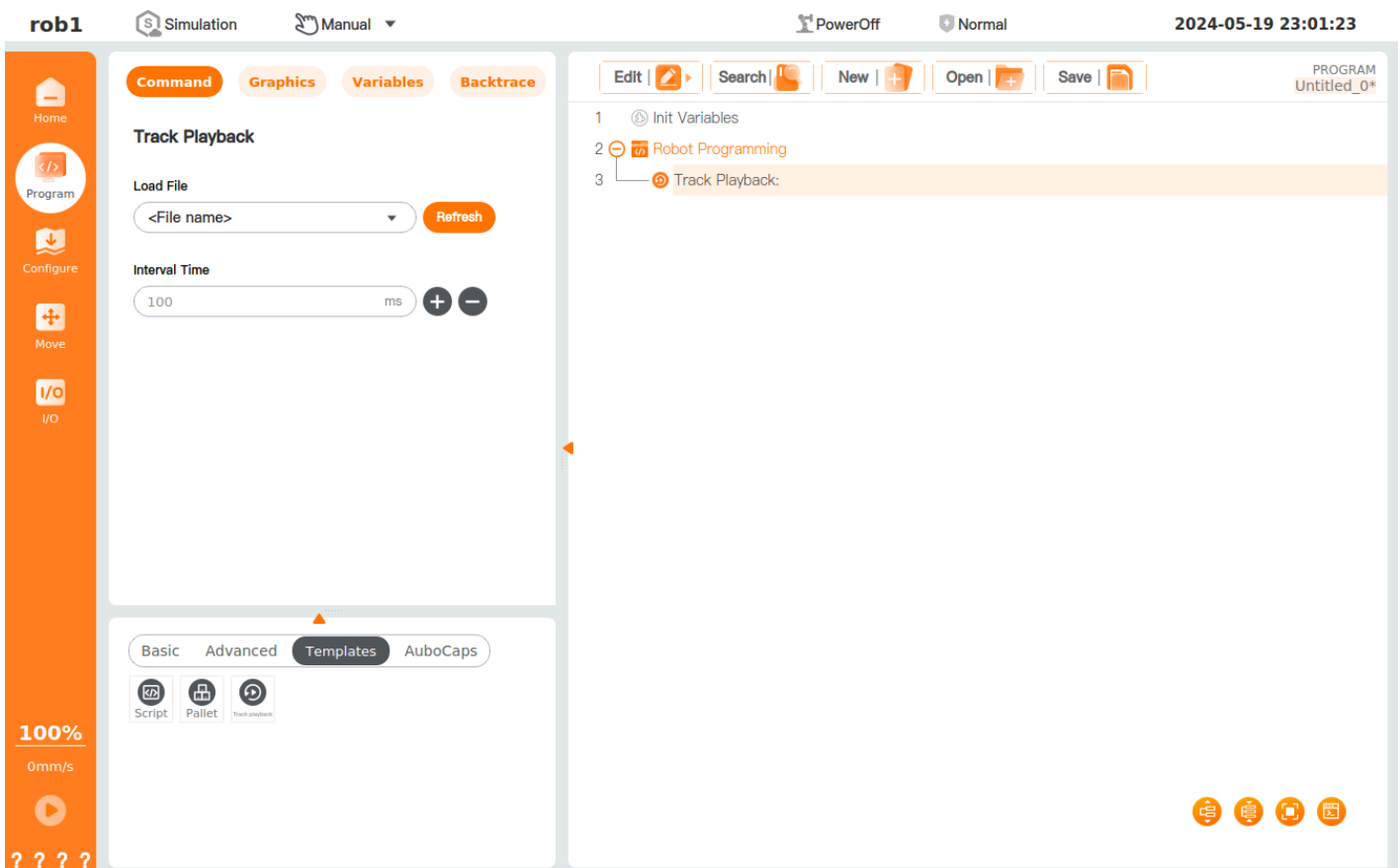
2 Robot Programming

3 Pallet_0

- Grid pattern: Showing row and column information for palletizing.
- Vertex: Four vertices of the palletizing pattern. Click [Corner 1] ~ [Corner 4] to set the position/pose of the four corners in turn.
- After setting the vertices of the palletizing pattern, the operation to be performed at each point will be generated in the [Pallet] node block. The starting point of robot palletizing is the working point.
 - Approach point: The starting point of the end-of-arm tooling (EOAT).
 - Working point: The first point of palletizing work.
 - Tooling actions: Actions of the end-of-arm tooling (EOAT).
 - Exit point: The end point of the end-of-arm tooling (EOAT).

Track playback

The [Track Playback] command allows insertion of the track file in the program. When the program runs to the "Track Playback" node, the track file will be played, and the robot will move along the track recorded in the file. The track file is a file that records the motion track of the robot and is generally stored in the installation profile. For the generation of the track file, please refer to "[Record](#)."



Settings

- Load file: Click "Load file" or ▼ the icon, and select the track file, then the program will start track playback after running to this node.
- Refresh: Click it to refresh the "Load" drop-down menu.
- Interval Time: Used to adjust the speed at which the track is running.

Constant force control

The [Constant Force Control] command can set the force or torque value to a flexible axis, enabling the TCP to move with constant force on a curved surface.

The screenshot displays the AUBO Developer software interface. On the left, a sidebar contains navigation icons for Home, Program, Configure, Move, I/O, and a 100% speed indicator. The main window is divided into two sections. The left section, titled 'Force Control', features tabs for Command, Graphics, Variables, and Backtrace. Below these tabs, a dropdown menu is set to 'Base'. A table allows users to configure the force control for various axes. The right section shows a program editor with a sequence of steps: 1. Init Variables, 2. Robot Programming, 3. Force, and 4. Empty.

	C-Axes	Force	Speed Limits
☐	X	N	mm/s
☐	Y	N	mm/s
☐	Z	N	mm/s
☐	RX	Nm	°/s
☐	RY	Nm	°/s
☐	RZ	Nm	°/s

Use "HandGuide" button with "Test" button to test force mode.
View compliant axis(es) status in "Graphics" tab.

Basic Advanced **Templates** AuboCaps

Script Pallet ImageForce Vision

- Coordinate system mode: After configuring the flexible axis and setting the force value, the robot will keep moving with constant force in the flexible axis direction and move according to the preset track in other directions.
- Motion: The force coordinate system is adjusted according to the motion of the tool center point (TCP). The X-axis of the force coordinate system is the projection of the TCP motion direction onto the X-Y plane of the selected coordinate system, and the Y-axis is perpendicular to the X-axis in the X-Y plane.
- Speed limit: The maximum movement speed of the end in the direction of the flexible axis (when there is no reference track). The actual movement speed may be less than or equal to this speed.
- Stiffness: Generally 0.

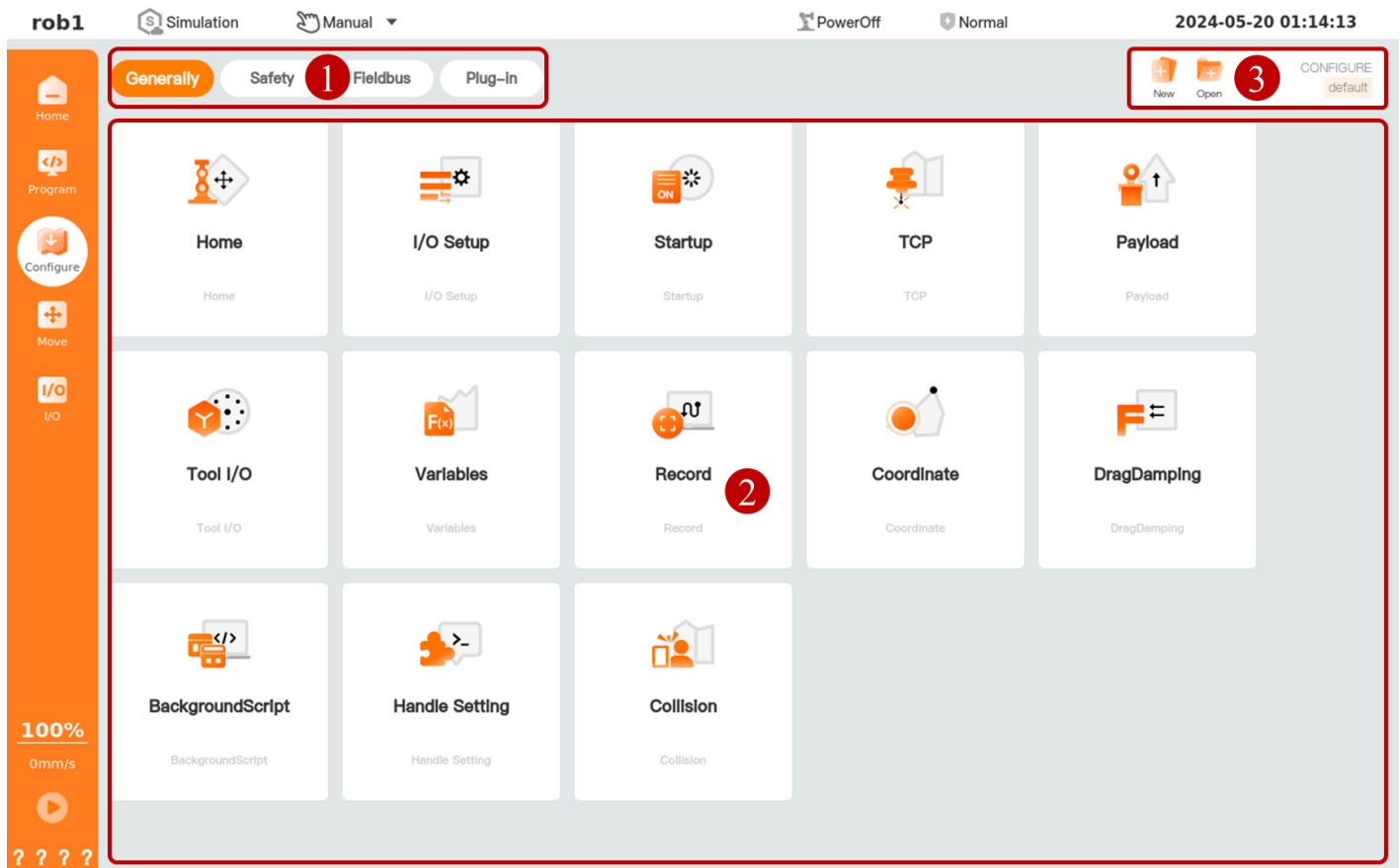
- Damping: This value will affect the actual movement speed of the end.
 - If the target force remains constant, the greater the damping, the lower the movement speed, and it may not move at all. The smaller the damping, the greater the movement speed, but the target force may overshoot after contact.
- Environmental stiffness: The deformability of tools and workpieces.
 - If this value is too small, the robot arm will shake; too large a value will affect system response speed, but it is relatively safe.

Configure Tab

The settings in the [Configure] tab will affect the overall performance of the robotic system. All settings are stored in the installation profile. Users can customize the configuration in the [Configure] tab or import the existing installation profile.

Tip

1. After completing the settings for all functions under the [Configure] tab, please save the installation profile promptly to avoid losing configuration information. To ensure that the new settings take effect in the program, write the program file after saving the profile.
2. If you do not save the installation profile, the system will alert you that the installation profile is not saved before shutdown. If you proceed with the shutdown, any unsaved installation configuration information will be lost after restart.
3. If the configuration file is modified and is bound with a program file, be sure to save the configuration file first and then the program file to avoid errors during program operation.



Serial number	Name	Description
1	Configure menu	-

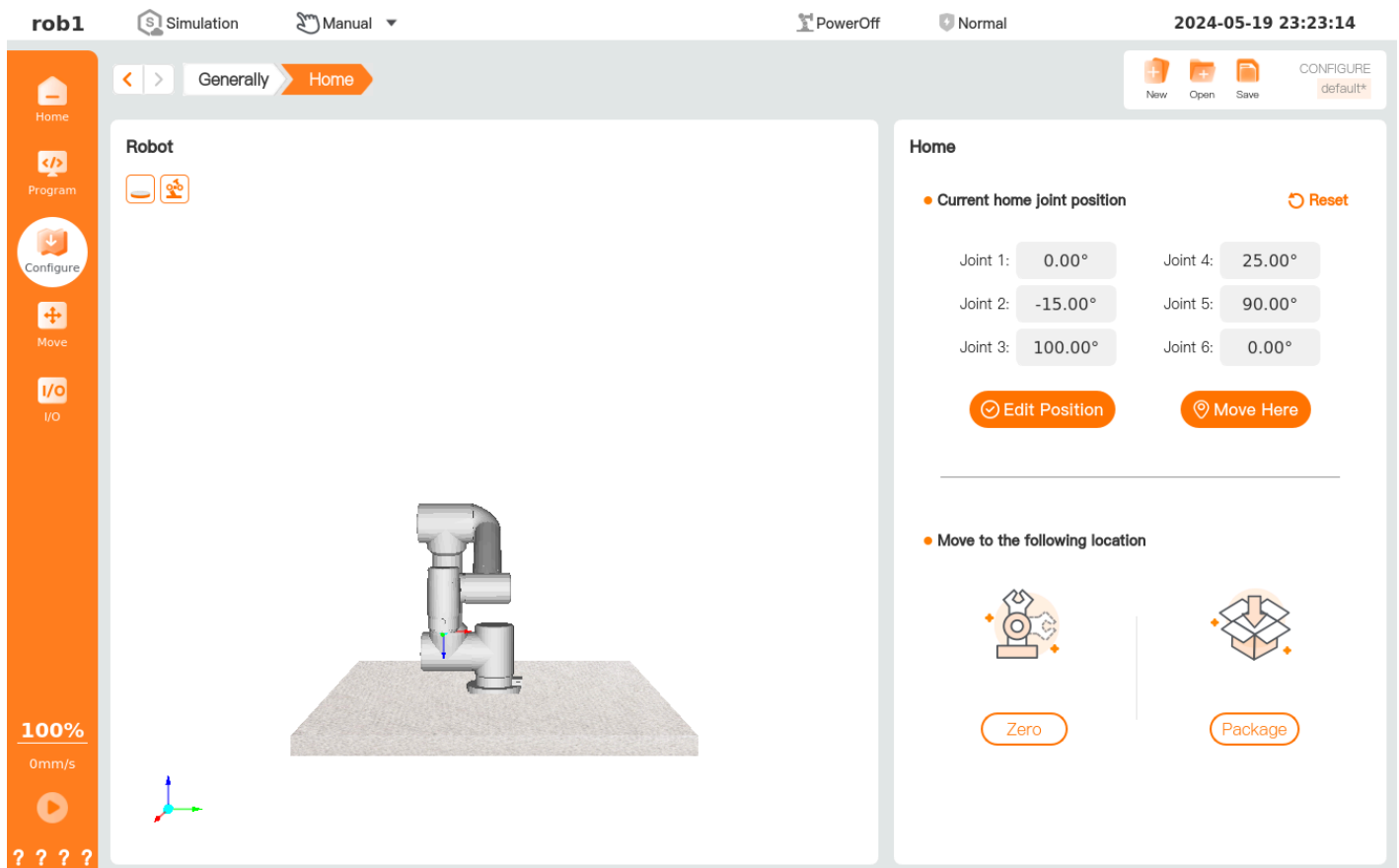
2	Configure settings page	-
3	Configuration manager tool bar	Manage the installation profile, with the name of the current installation profile displayed on the far right. * New: When creating a new installation profile, you can retain the current configuration in the new profile or reset all configurations. * Open: Open the installation profile stored in the controller. * Save: Save the installation profile in the controller. * Save > Save As: Rename the file and save it in the controller.

General

Home

The [Home] page contains the common positions/poses of the robot, where you can customize the home position/pose or quickly move the robot to the common positions/poses.

- Robot: Allow simulating the current robot status.
- Home: Allow viewing the position/pose parameters of the current "Home," setting the common positions/poses, or controlling the movement of the robot to the common positions/poses.



- Edit Position: Edit the user-defined home position/pose. Click [Edit Position] to enter the [Move] page and move the robot to set the position/pose. Click [OK] to save and return.
- Move Here: Move the robot quickly to the home position/pose. Click [Move Here] to quickly enter the [Move] page, where the orange numbers in the joint control box indicate the joint parameters of the current home position/pose. Long press [Home] or [Auto] to move the robot to the home position/pose, and then click [OK] or [Cancel] to return.
- Restore Default: Clear the user-defined home position/pose and restore the default home position/pose.

- Zero: Move the robot quickly to the zero position/pose.
 - Package: Move the robot quickly to the packaging position/pose. The packaging position/pose is used when the robot is being packaged for shipment.
-

I/O setup

For I/O configuration methods, please refer to [I/O Configuration](#), and for an introduction to configurable I/O functions, please refer to [I/O Function](#).

Start

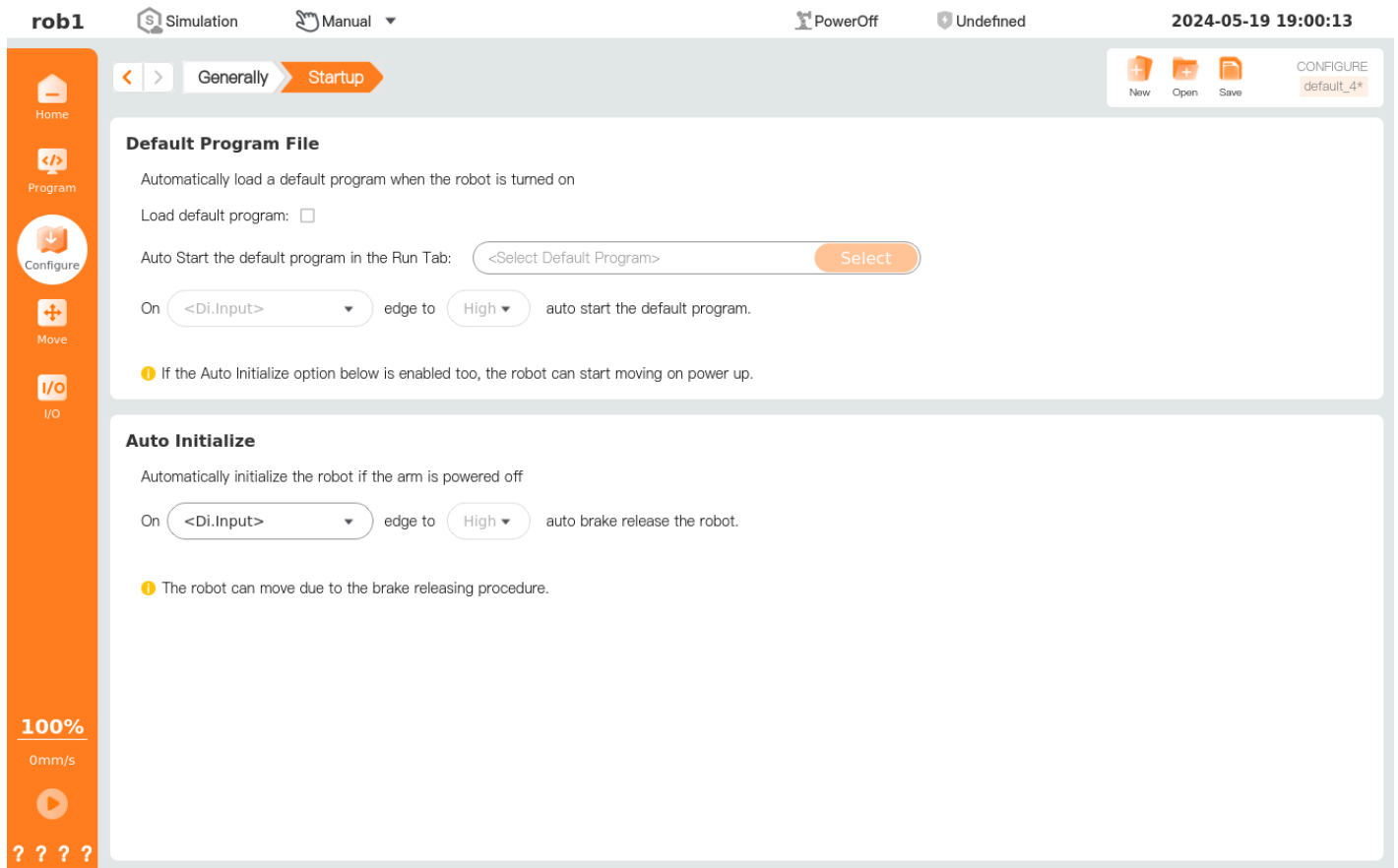
[Start] allows setting of the default program and auto initialization. A default program is a program loaded automatically upon boot of the controller, which is loaded automatically when the system starts running without loading any programs. Auto initialization, i.e., when the robot receives an initialization signal in the power-off state, it will start initialization to restore to the operable status.

Warning

1. When automatic robot connection, automatic power-on, and automatic loading are enabled, the robot runs the program immediately after the controller is powered on, provided that the input signal matches the selected signal level. For example, in the case of low level, it is no need to transit an edge level to the selected signal level.
2. Be careful when setting the signal level to a low level. The input signal is set at a low level by default, allowing the program to run automatically without interference from external signals.

Tip

Exiting the running program or clicking the "Stop" button in the teach pendant software will disable the auto-start feature until the "Run" button is pressed again.



TCP

Please refer to [Basic robot operations - Load and TCP](#).

Payload

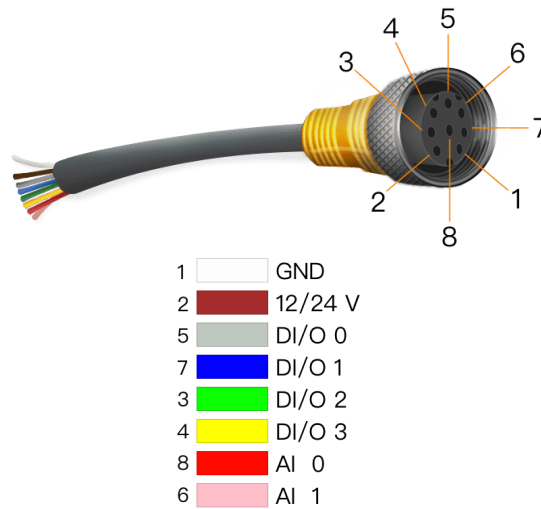
Please refer to [Basic robot operations - Load and TCP](#).

Tool I/O

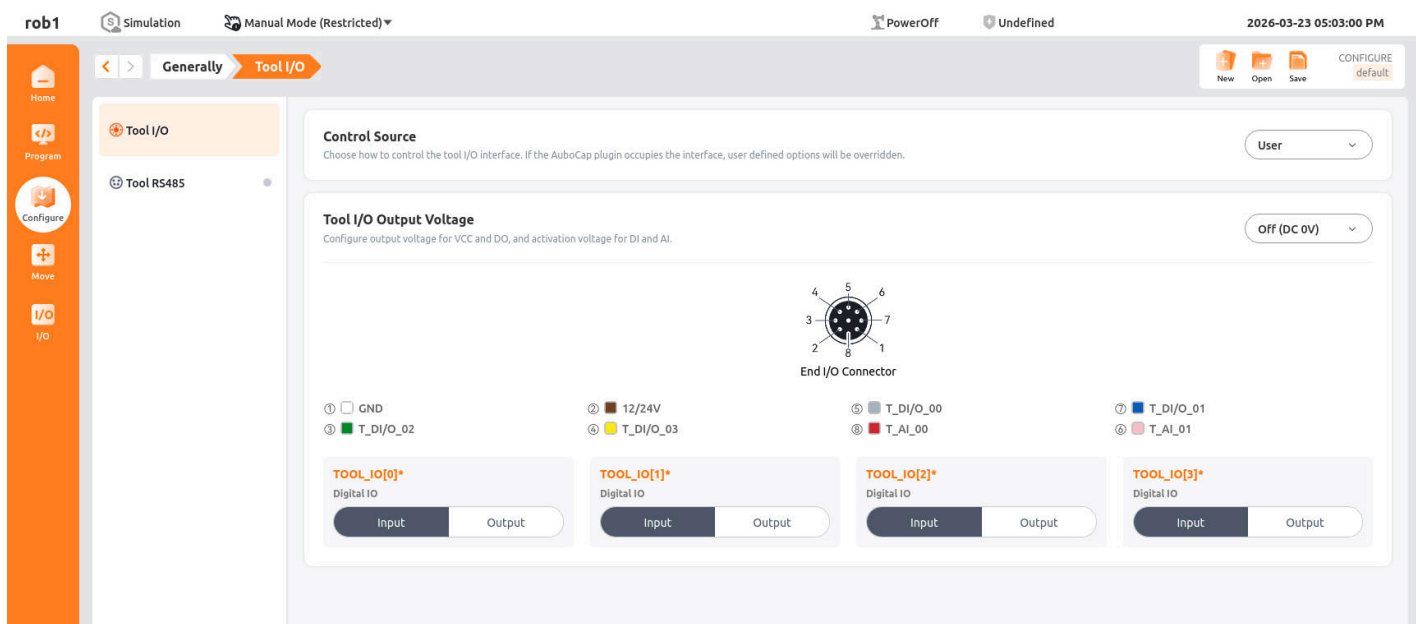
The robot end (joint 6) is equipped with two types of interface: an 8-pin tool I/O interface and a 4-pin tool RS485 interface, which can provide power supply and control signals to end-of-arm tooling (EOAT) (such as a gripper). The end interface configuration may vary for different models; please refer to the actual product and product specifications for details.

Tool I/O interface

The tool I/O interface is an 8-pin connector, as shown in the figure below. Users can configure 4 channels of digital I/O via pins 3/4/5/7, and configure analog input via pins 6/8, with the analog voltage of 0V ~ 10V; pin 2 can be configured with output voltage of 0 V, 12 V, and 24 V.



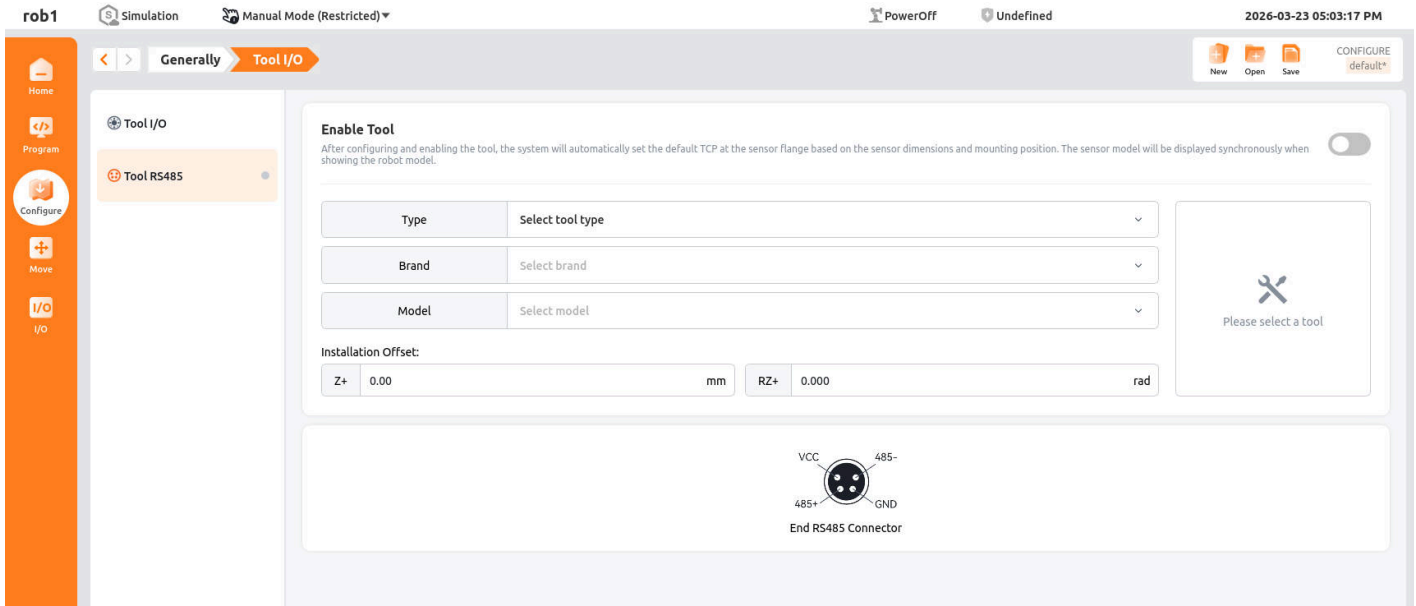
[Tool I/O] page allows settings of the EOAT control mode, power voltage and digital I/O communication direction. For the function settings of each I/O interface for EOAT, please refer to “[I/O Function.](#)” For the status monitoring of I/O interfaces, please refer to the “[I/O Tab.](#)”



- Control right: Control the switching of tool I/O interfaces.
- Tool I/O output voltage: Set the voltage level (in V) for the tool's digital I/O in output mode. After replacing tools or resetting system, please double-check to ensure that the voltage matches the tool's rated voltage.
- TOOL_IO[0] ~ TOOL_IO[3]: Set the communication direction of the tool's digital I/O.

Tool RS485 interface

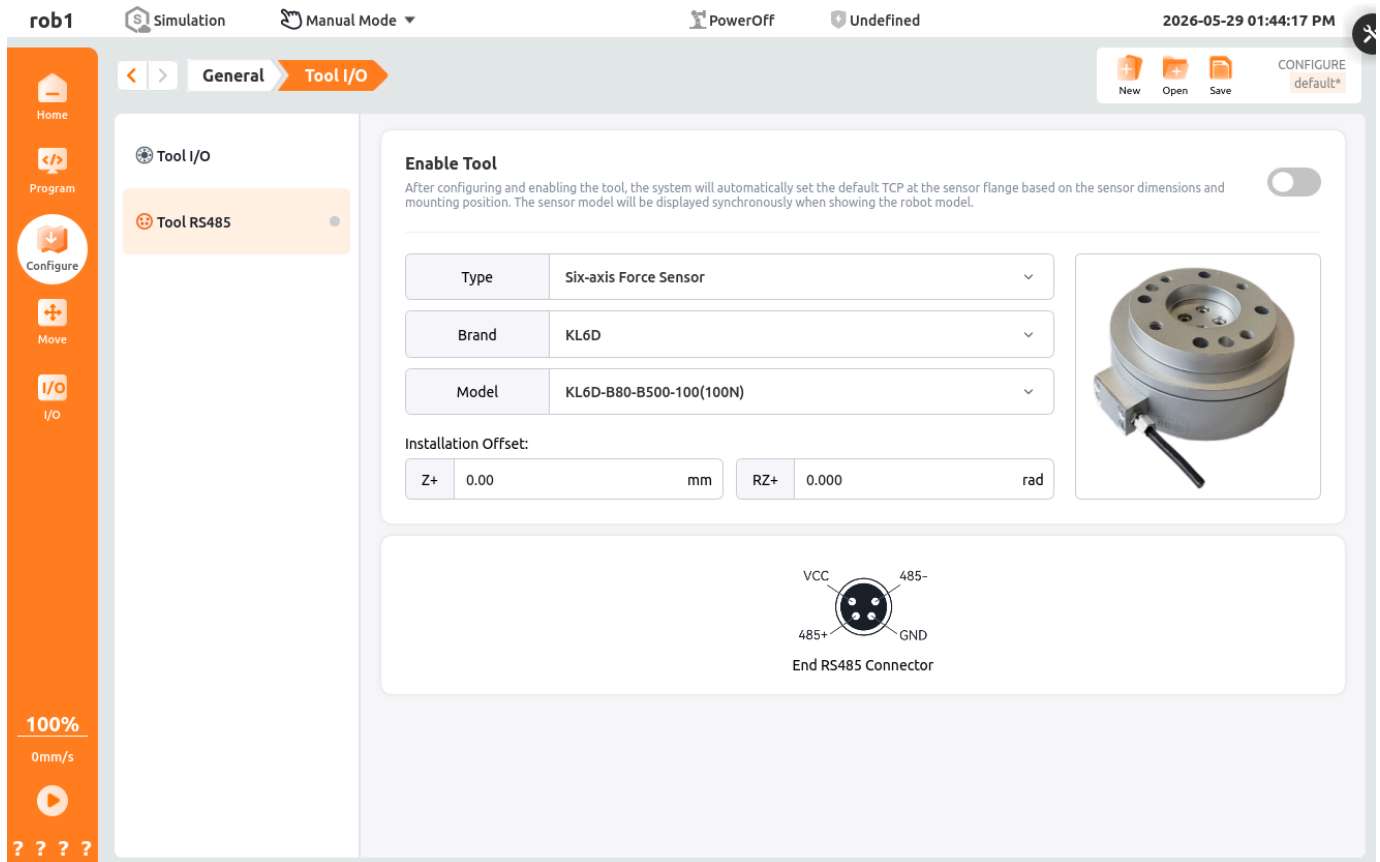
After configuring and activating a tool, the system automatically sets the default Tool Center Point (TCP) at the sensor flange based on the sensor dimensions and mounting position. Additionally, the sensor model will be synchronously displayed when showing the robot model.



Activate Tool

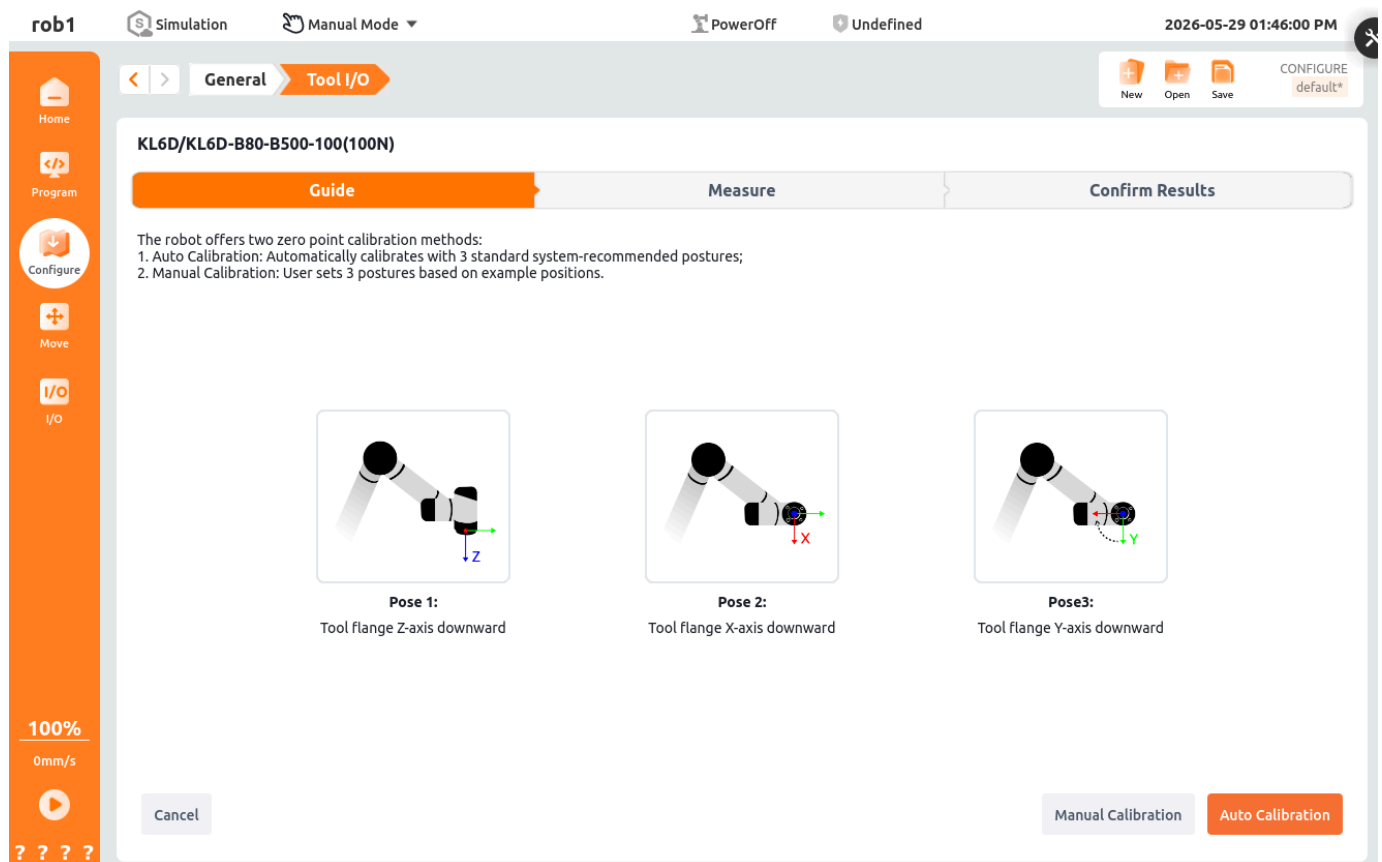
- Fill in the tool parameters, then toggle the button in the upper right corner of the page to activate the sensor.
 - Type: Select six-axis force sensor.
 - Brand: Currently supports Kunwei, Yuhong Minxin, Yuli, Zhili, Keli, Landian, and Xinjingcheng.
 - Model: The corresponding model under the selected brand.
 - Mounting Offset: Used to compensate for the offset between the sensor coordinate system and the flange coordinate system. Generally 0; currently only Yuli sensors require an offset of 45°.
- After toggling the button, the activation status can be identified by the activation indicator light. Upon successful activation, the [Zero Calibration] and [Sensor Debugging] function buttons will appear.
 - Activation indicator light:
 - Gray when not activated.
 - Green when activation is successful.

- Red when activation fails.

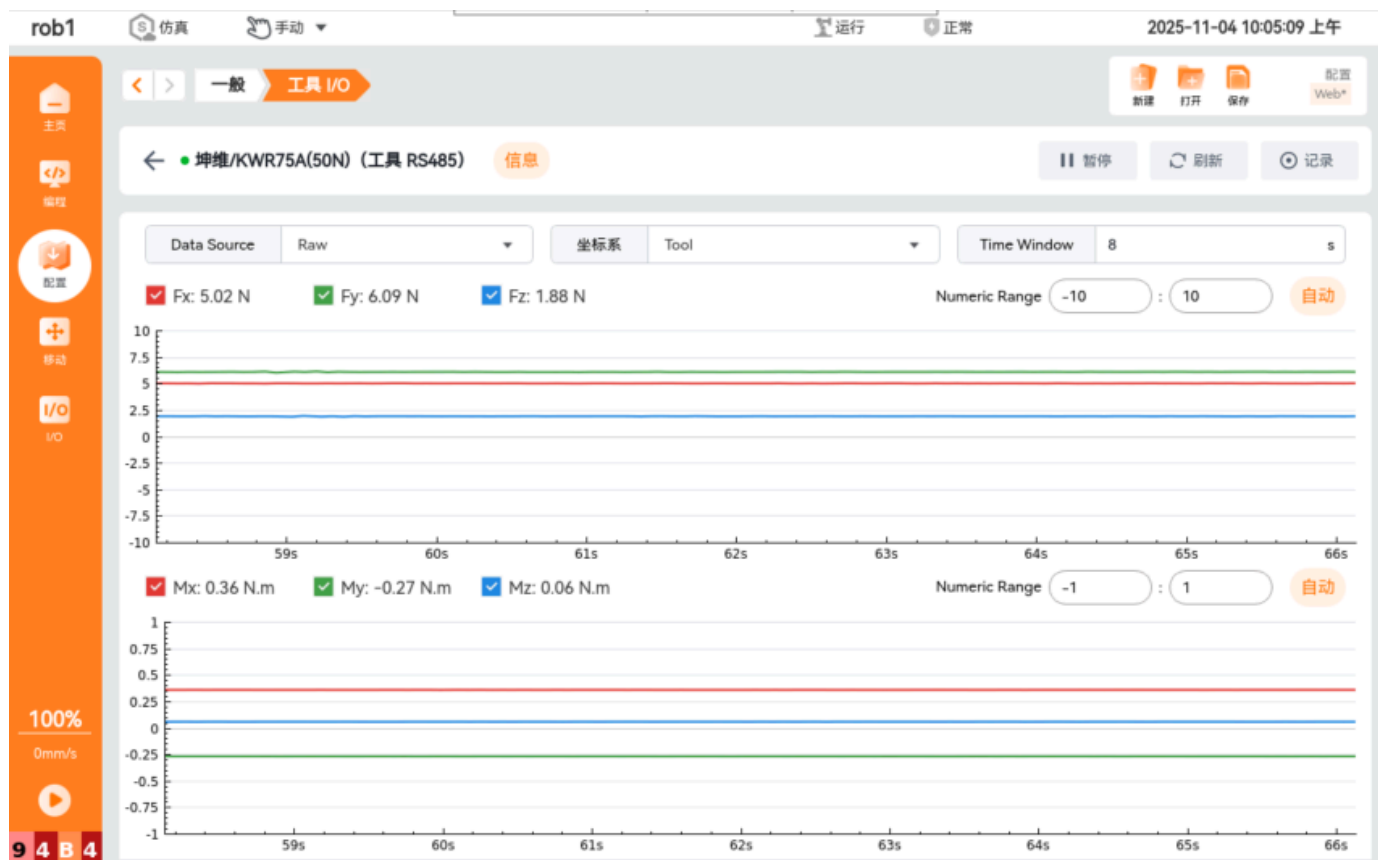


3. Zero Calibration The zero calibration home page displays the bias from the last calibration. To recalibrate, click the [Recalibrate] button in the lower right corner of the page to enter the recalibration screen. Recalibration includes both manual calibration and automatic calibration functions.

- Automatic calibration: Achieved by autonomously moving to three preset poses; be mindful of potential interference during use.
- Manual calibration: Requires teaching three poses according to prompts, then starting the calibration process.



4. Sensor Debugging The sensor debugging screen is primarily used to view sensor data curves and also provides capabilities for recording data, replaying data, selecting data types and coordinate systems, and setting the data display time.



Variables

The variables created on this page are called "configure variables," utilized in a manner akin to program variables. They are categorized into five types: Bool, Int, Float, String, and Pose. The difference between a configuration variable and a program variable is that the name and value of the configuration variable are stored in the installation profile, and the user can use the same configuration variables in multiple programs.

The [Variables] page allows the creation of configured variables as well as the setting of their initial values, which can be saved by clicking [Save] on the configuration manager toolbar. In the program running process, the status of configured variables can be updated on this page in real time.

rob1 Simulation Manual PowerOff Undefined 2024-05-19 19:03:53

Home Program Configure Move I/O

100% 0mm/s

Configure variables

Variable	Type	Value	Range
i_var_0	Integer	1	-10000 ~ 10000
i_var_1	Boolean	true	
i_var_2	Any	'a'	
i_var_3	Any	"a"	
i_var_4	Float	0.5	-10000 ~ 10000
i_var_5	Any	{0,-15,100,25,92,0}	
i_var_6	Any	{403.82,-119,211.54,3.142,0,1.571}	

Variable name: i_var_6 Value: {403.82,-119,211.54,3.142,0,1.571}

Type: Any Range: →

- Global persistence:

1. Types of variables

- Program variables: These variables only apply to the running program, and variable values will be lost after the program stops. They are regular program variables. For details, please refer to "[Program tab - Basic program nodes - Assignment.](#)"

- **Configure variables:** Such variables can be applied to more than one program, and their names and values persist until the robot is decommissioned. Therefore, they are persistent variables.
- **Script variables:** Such variables come from the script file and can be assigned to different variable types. Script variables are not displayed on page [Program] or page [Configure]. They are used by a robot program to store information about the waypoint.

Based on the value treatment mechanism (saved by the system or not) and application scope (one program or more than one program), variables are divided into the following two types:

- **Persistent variables:** These variables can be applied to more than one program, and their names and values persist until the robot is decommissioned. Values of persistent variables are retained after AuboStudio or the controller is restarted. For example, the configuration variables are of this kind.
- **General variables:** These variables apply only to the running program, and their values are lost after the program is stopped. For example, the program variables are of this type.

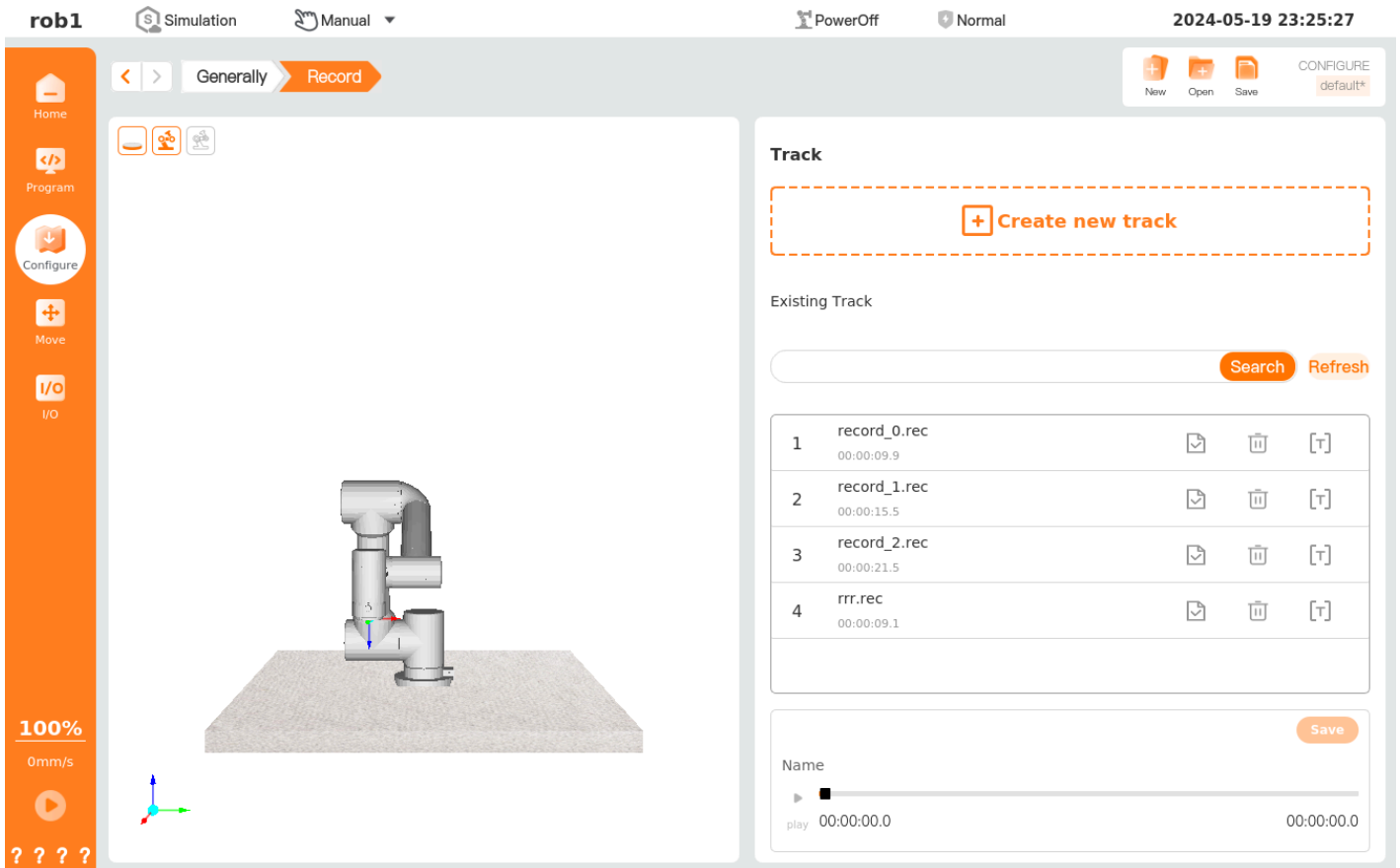
2. Data type

- **Bool:** Boolean variables, whose value is true/false.
- **Integer:** Integer variables, whose value is an integer.
- **Float:** Floating-point variables, whose value is a floating-point number.
- **String:** String variables, whose value is a string value enclosed in single quotes (') or double quotes (").
- **Pose:** Pose variables, whose value is waypoint data containing 6 parameters, as described below:
 - In case of MoveL, the first 3 parameters are pose parameters measured in mm, and the last 3 parameters are pose parameters measured in rad.
 - In case of MoveJ, all the 6 parameters are joint angles, which are measured in rad.

Record

The [Record] page stores the tracks of robot movement for a period of time, and the stored tracks can be added to the program.

- Simulation model: Simulate robot status.
- Track: Create/Edit/Save/Delete tracks.

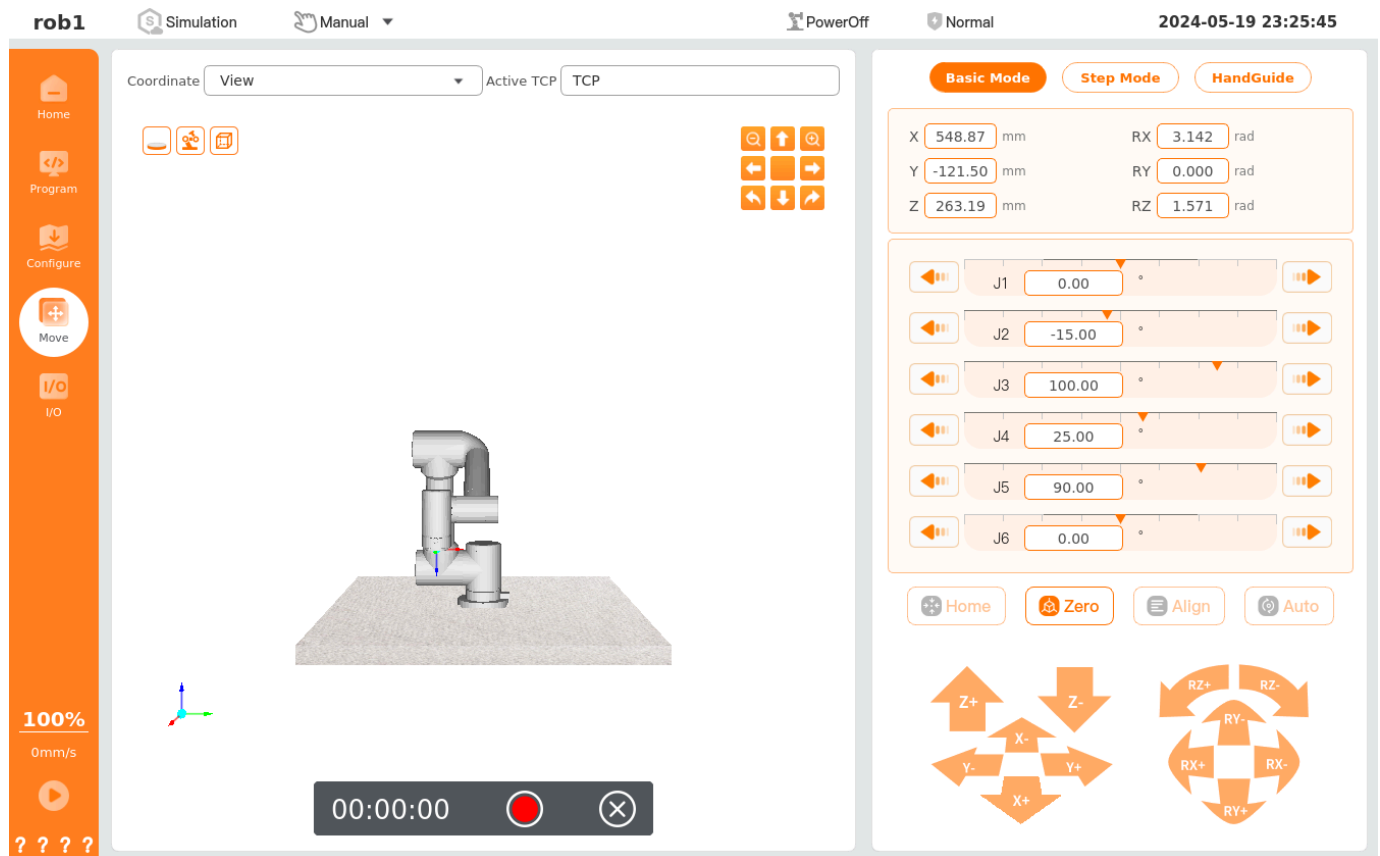


- Create Track: Click [Create Track] to enter the [Move] page, and control the robot using the position/pose control button; then the system will start recording the track of the robot. After the running is completed, click [OK] to name and save the track.
- Search: Search for a certain track by its name.
- Refresh: Refresh the list of tracks.
- Load: Load a track.
- Rename: Rename a track file.
- Delete: Delete the track.
- Play: Play the currently loaded track. Click [Play] , then enter the [Move] page to move the robot to the initial position, and click [OK] to start playing the moving track. At this time, clicking on [Play] will switch to [Stop] status .

Create a new track

1. Click [Create Track] to enter the [Move] page, then click to start recording. Control the robot using the position/pose control button. The system will record the robot's track and

time. Click  to exit the [Move] page. After recording, click  to display a pop-up prompting you to name the track file. After naming, click [Save] to return to the [Record] page.



2. Click  to rename the recorded track.

3. Select the saved track file from the track list and click [Load]  to load the track into the player.

Coordinate system

Please refer to [Basic robot operations - Coordinate system](#).

Drag damping

Please refer to [Move teaching - HandGuide - Drag damping](#).

Background script

[Background Script] allows setting of multiple scripts running in the background, tailored to users' needs.

Tip

The scripts added in [Background Script] are not allowed to contain robot motion commands.

rob1

Simulation

Manual

PowerOff

Undefined

2024-05-19 18:54:09

Home

Program

Configure

Move

I/O

100%

0mm/s

???

Generally

BackgroundScript

New

Open

Save

CONFIGURE default_4*

Background Script

Add script

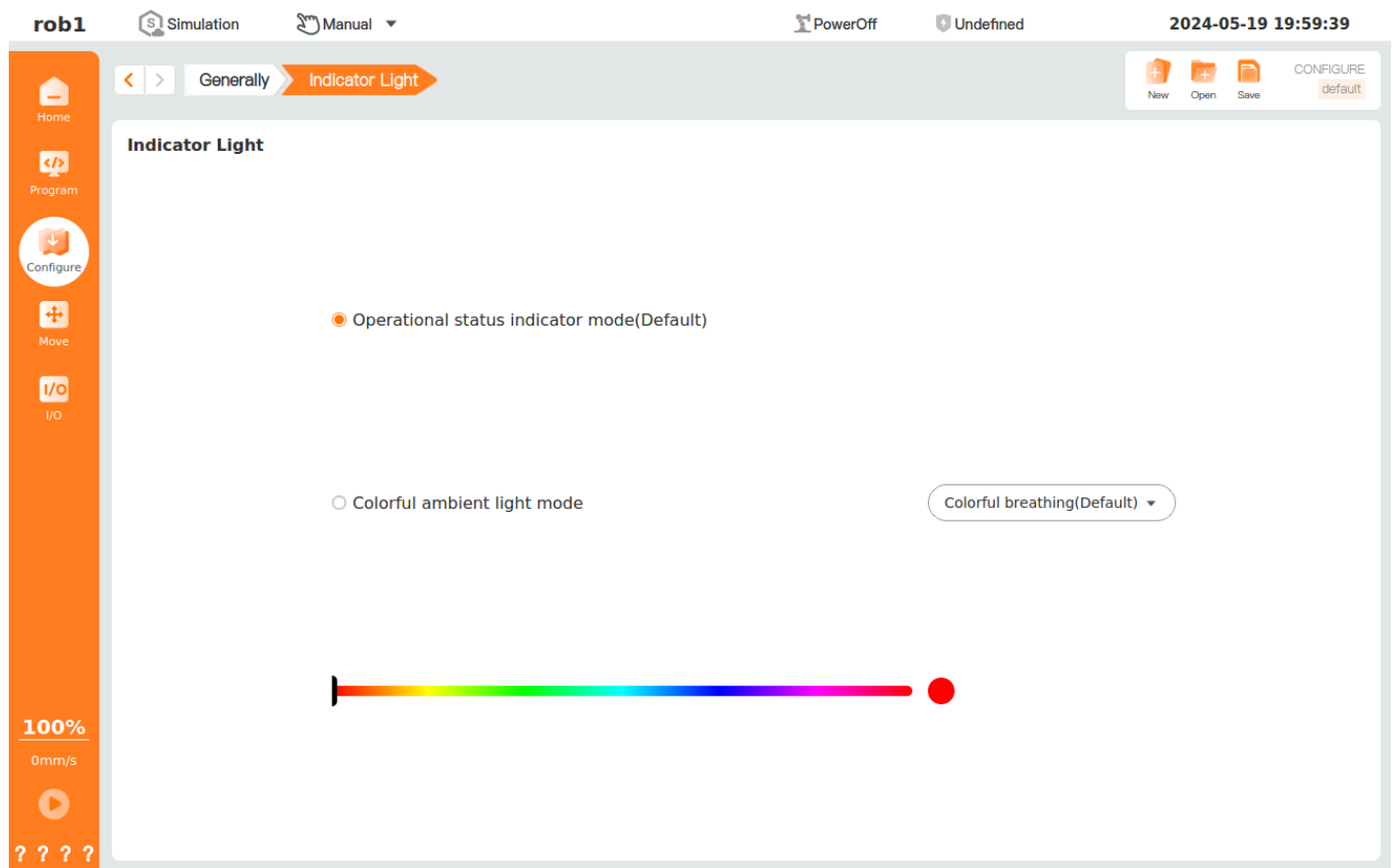
Delete

Stop working

Name	Path	Status	Operate	Auto Run
Untitled_6	/root/arcs_ws/program/Untitled_6.lua	Not running	Execute	☐

Indicator light*

[Indicator Light] allows setting of the mode of the robot indicator light. This function is only supported by some robot arm models. Please refer to the [Hardware User Manual](#) for more details.



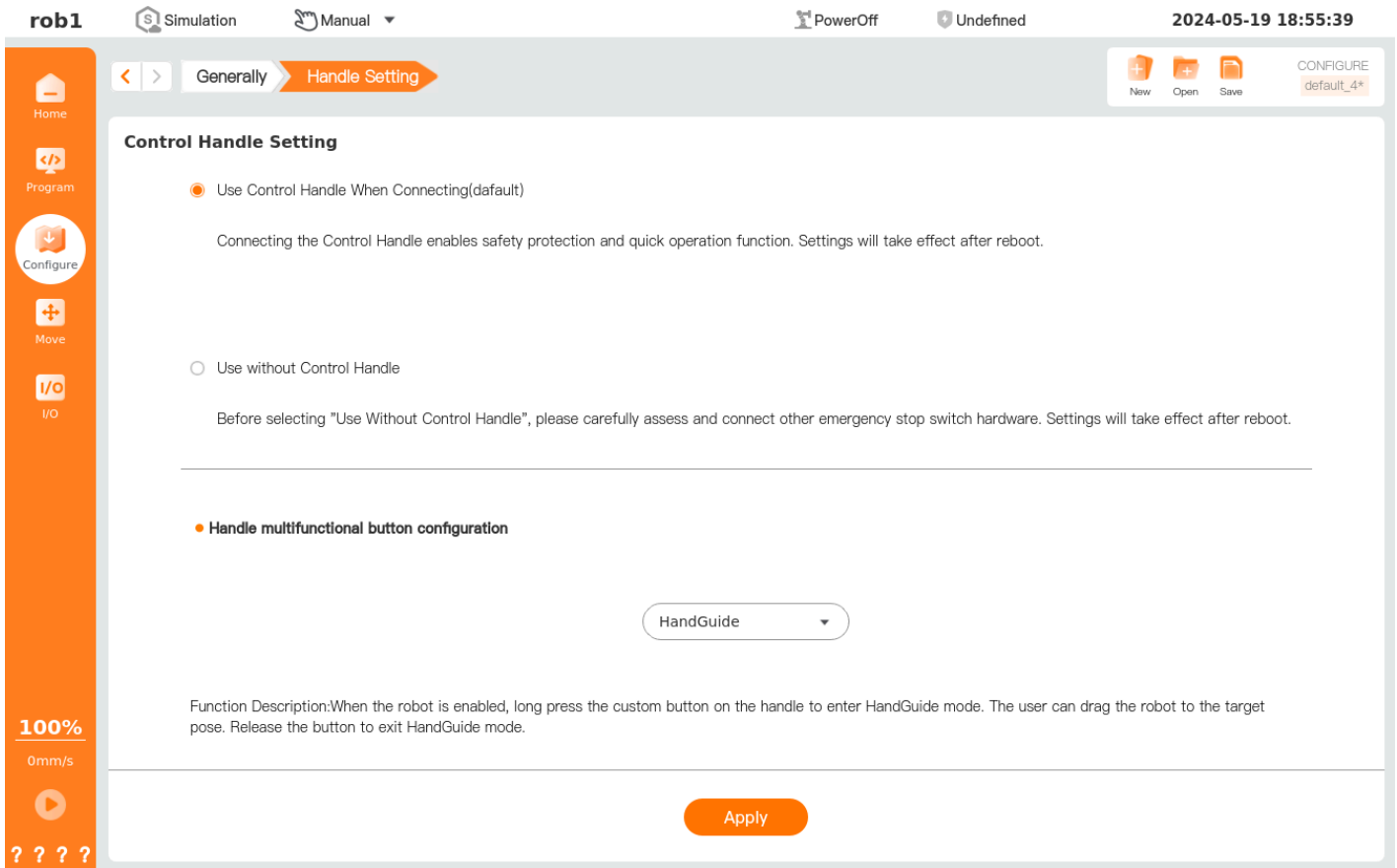
Handle setting*

[Handle Setting] allows setting of the control mode of the handle matched with the product and configuration of the functions of the custom button on the handle. This feature is only supported by some controller models. Please refer to the [Hardware User Manual](#) for more details.



Info

Clicking [Apply] will only make the handle settings effective temporarily. To ensure that the relevant settings can be found upon restart, click [Save] to save the current installation profile.



Gripper

The [Gripper] function supports the integration of commonly used electric gripper brands within the software, connecting to the robot arm end or control cabinet via RS485 communication interface, and providing complete configuration and online programming capabilities on the teach pendant.

1. Hardware Connection

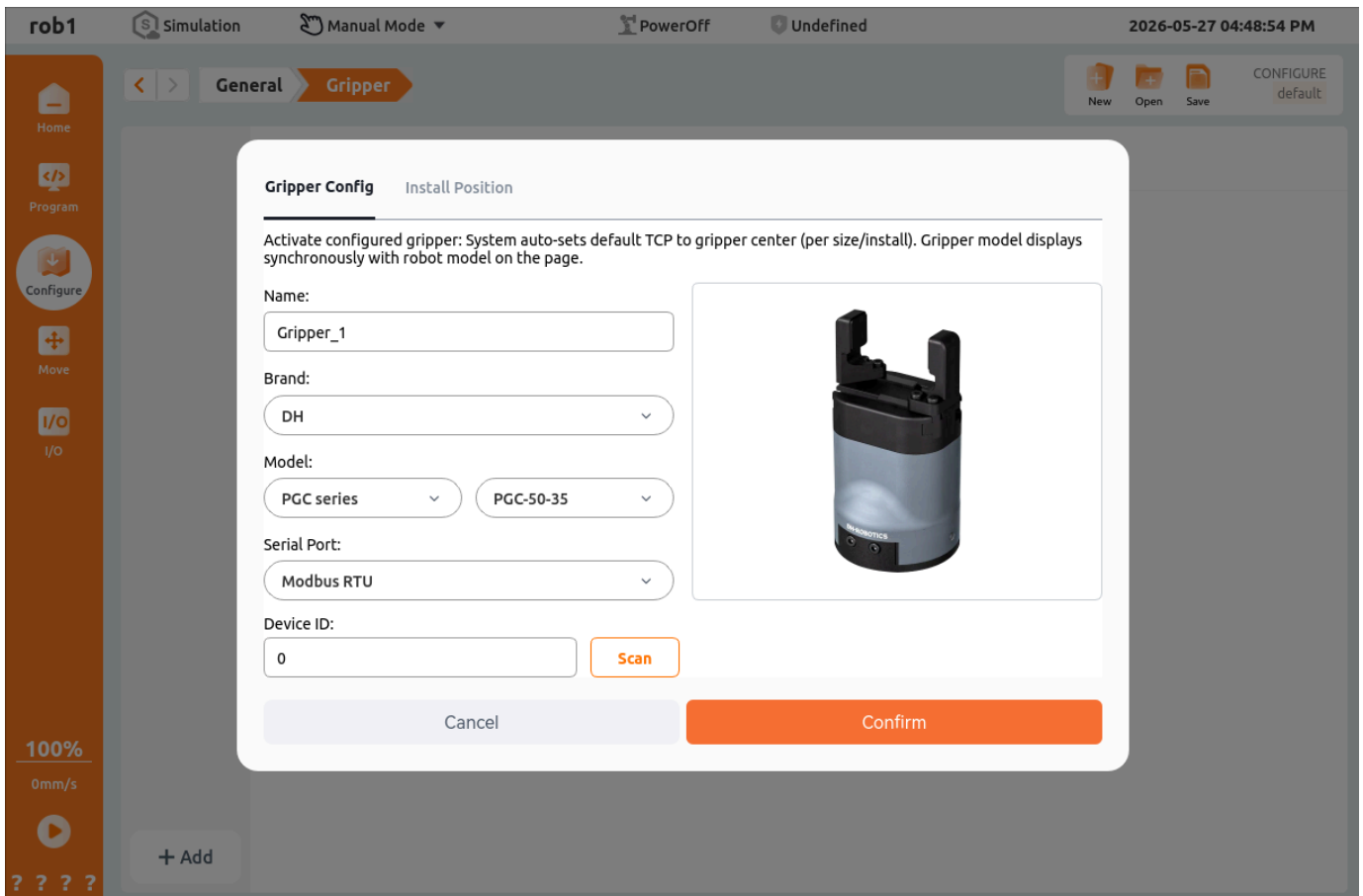
Connect the gripper to the robot arm end or control cabinet via the RS485 communication interface.

Note:

Before configuring the gripper, ensure that the gripper is physically connected correctly and the power supply system is functioning properly.

2. Software Configuration

1. On the teach pendant, navigate to "Configure > General > Gripper" in the left sidebar.
2. Click [Add] at the bottom left of the gripper configuration page to begin adding a new gripper device.



- Gripper Configuration

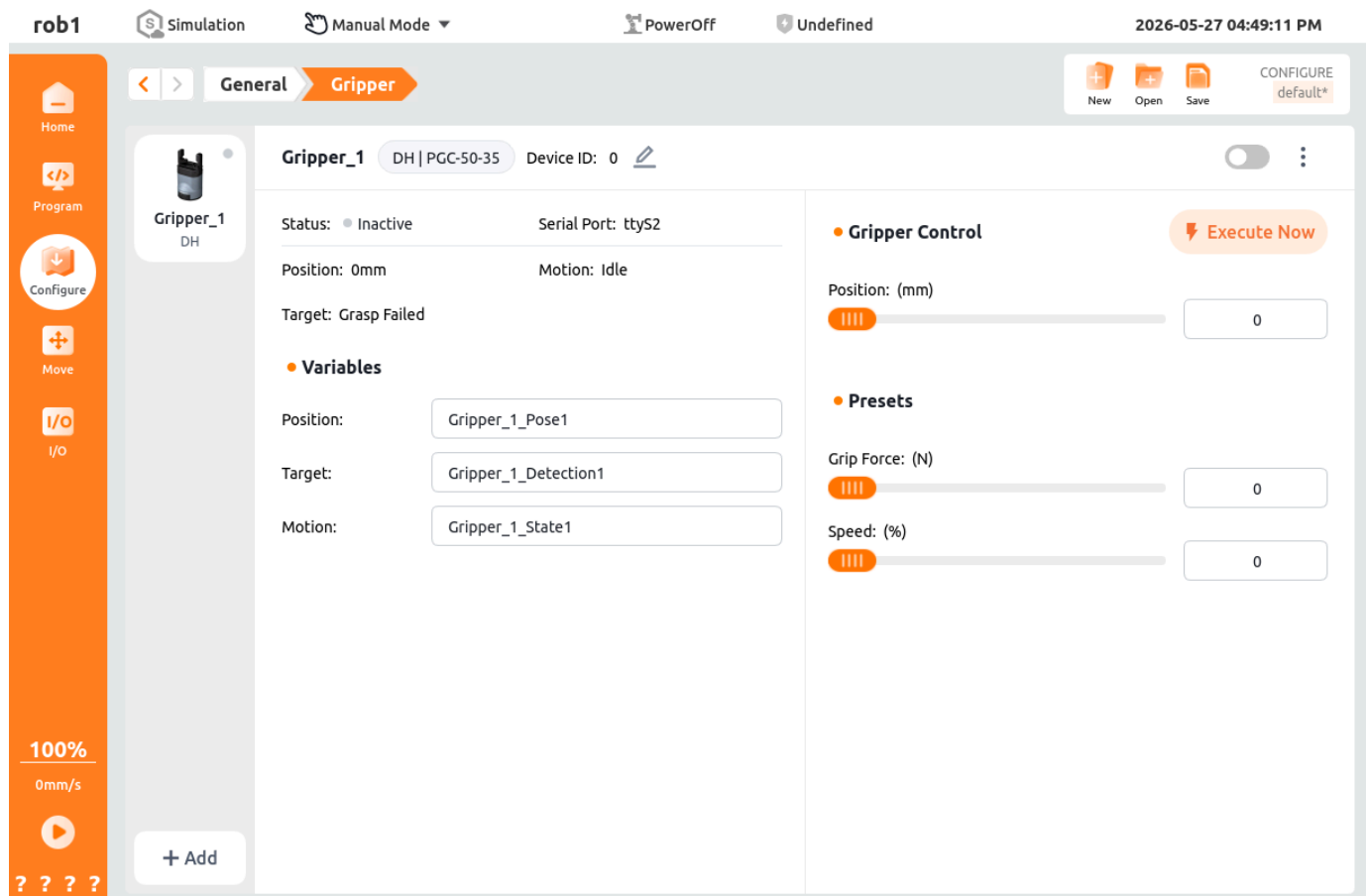
Activate configured gripper: The system will automatically set the default TCP to the gripper center (based on dimensions / mounting), and the gripper model will be synchronously displayed with the robot arm model on the page.

- Name: Gripper name.
- Brand: Gripper brand. Currently supports Da Huan, Rou Chu, and Jun Duo.
- Model: Select the specific gripper model under the corresponding brand.
- Serial Port: Select the communication serial port identified by the system to establish a data connection channel with the gripper.
- Device ID: Unique device identifier.
- Mounting Position

- Enable: Used to enable / disable the gripper mounting position configuration function.
- X/Y/Z (mm): Defines the linear offset of the gripper in the robot end flange coordinate system, in millimeters.
- RX/RX/RZ (°): Defines the attitude rotation angle of the gripper in the robot end flange coordinate system, in degrees, corresponding to rotation around the X, Y, and Z axes respectively.

After configuration, click [Confirm] to complete the addition of the new gripper device.

3. Once the gripper is added, it can be configured.

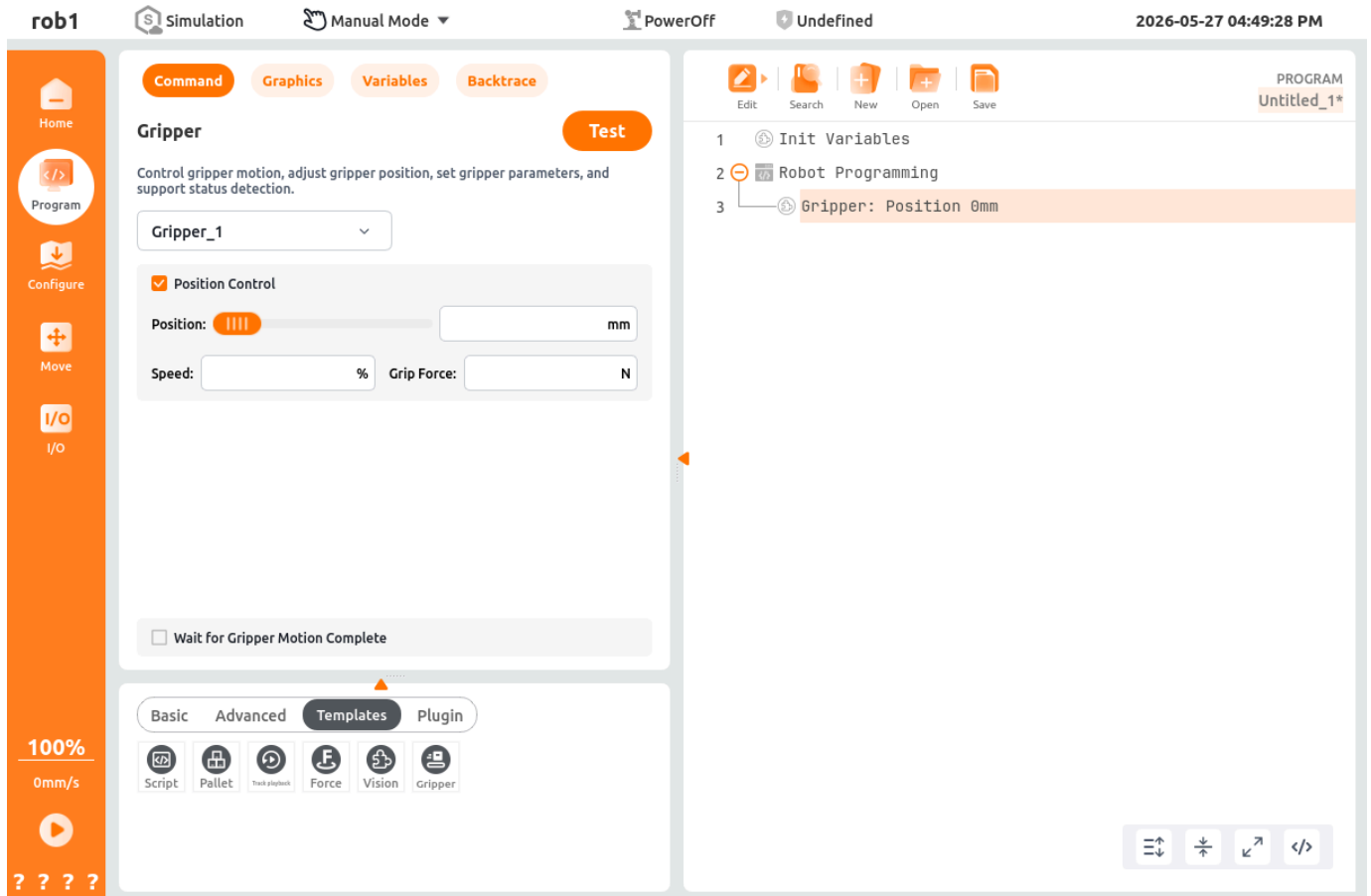


- Gripper Enable Button: Controls whether the gripper is enabled.
- Gripper Information:
 - Device ID: The device ID set when adding the gripper, used to distinguish between different grippers in multi-device scenarios; it can be modified.
- Gripper Status:
 - Not Started: The gripper is not enabled.
 - Normal: The gripper is enabled and functioning normally.
 - Communication Fault: The gripper cable connection is incorrect.
 - Gripper Fault: The gripper hardware is damaged or functionally defective.

- Gripper Over Temperature: The gripper temperature exceeds 85 °C.
- Gripper Voltage Abnormal: The gripper supply voltage is outside the normal range.
- Gripper Current Abnormal: The gripper supply current is outside the normal range.
- Gripper Enable Failed: The gripper enable process was not completed or the enable conditions are not met.
- Communication Serial Port: The serial port number set when adding the gripper.
- Gripper Position: The current opening / closing position of the gripper, in millimeters (mm).
- Movement Status: The movement status of the gripper, including [Idle] and [Moving].
- Object Detection: The built-in grasping status feedback function of the gripper, which determines whether the gripper has successfully grasped the target object through position closed-loop and current / force control feedback.
- Gripper Control
 - Gripper Position: The target displacement for gripper opening and closing, set within the device stroke range (0~26 mm). Once set, the gripper will move to the specified position.
- Parameter Presets
 - Gripping Force: The clamping force when the gripper closes, which directly affects grasping stability and workpiece safety. It should be adjusted according to the workpiece material and weight.
 - Gripper Speed: The movement speed of gripper opening and closing, expressed as a percentage of the device's maximum speed. Lower values result in smoother motion.
- Variables
 - Gripper Position: Retrieves the current position value of the gripper (0-26 mm), which can be used to determine whether the gripper has reached the position.
 - Object Detection: Retrieves the result of object detection (Success / Failure), which can be used for grasp logic decisions (e.g., trigger an alarm and retry upon failure).
 - Movement Status: Retrieves the movement status of the gripper (Idle / Moving), which can be used for process waiting (e.g., wait for the gripper action to complete before proceeding to the next step).

3. Programming Usage

1. In the programming tree, click "Templates > Gripper" to add a [Gripper] node.



2. Select the gripper added in [Configuration] and choose whether to enable [Position Control].

- Enable [Position Control]:
 - Control Logic: The gripper moves to the specified target position and then stops. If the gripping force reaches the set threshold during movement, it will also stop early and determine that the grasp is successful.
 - Applicable Scenarios: Standardized grasping where workpiece dimensions are known and a fixed opening is required, or scenarios requiring precise control of the gripper stroke.
 - Effective Parameters: Gripper Position (mm), Gripper Speed (%), Gripping Force (N).
- Disable [Position Control]:

The gripper will prioritize the set gripping force as the control target, performing **force-controlled adaptive grasping**.

- Control Logic: The gripper continuously moves in the closing direction until it detects that the set gripping force threshold has been reached (workpiece grasped) or the mechanical limit position is reached, after which it stops. The target position parameter will be ignored.

- Applicable Scenarios: Scenarios with unknown / variable workpiece dimensions, fragile workpieces, or flexible grasping, where position deviation could otherwise cause workpiece damage.
- Effective Parameters: Gripper Speed (%), Gripping Force (N).

3. Choose whether to enable [Wait for Gripper Motion Complete]

- Enable [Wait for Gripper Motion Complete]
 - Control Logic: The robot main program will enter a waiting state and will not execute subsequent instructions until the gripper action has completed (reached the target position / triggered the force control stop condition).
 - Applicable Scenarios: Processes such as grasping and placing that require confirmation that the gripper action has completed before moving, to prevent workpiece dropping or program logic confusion.
- Disable [Wait for Gripper Motion Complete]
 - Control Logic: The robot main program will not wait for the gripper action to complete and will immediately execute subsequent instructions. The gripper action and robot motion proceed asynchronously in parallel.
 - Applicable Scenarios: Scenarios with extremely high cycle time requirements where the gripper action and robot movement do not interfere with each other, improving program execution efficiency.

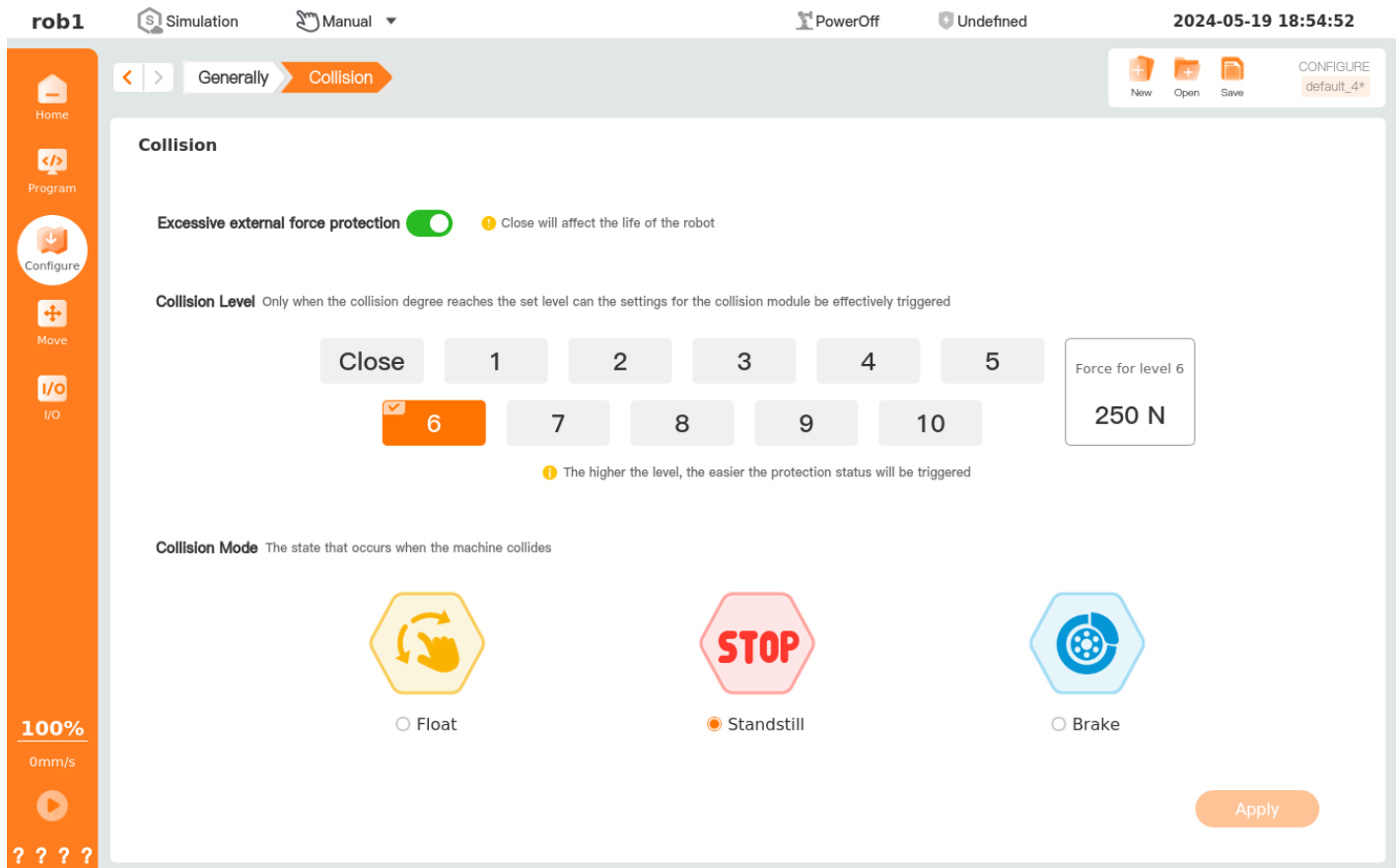
4. Run [Test] to move the gripper according to the set parameters.

Collision

The AUBO robot is equipped with protection against excessive external forces; it can detect external forces and take appropriate actions to safeguard itself and prevent damage to the surrounding environment and personnel.

The [Collision] page allows settings of collision level and collision mode, etc. The user can change the collision level and the method for handling a robot collision based on the actual situation to ensure the safety and reliability of the robot in use.

This function is specially designed for the safety of human-machine cooperation, requiring the robotic system to operate within the normal range and the use of AUBO series controllers. If users develop their own controller, the robot will not achieve the above function.



1. Protect against excessive external force

- **Open:** If the robot is in a static state and subjected to an external force that exceeds the safe level judged by the system, the robot will enter the HandGuide mode to prevent damage to critical moving parts.
- **Close:** If the robot is subjected to an excessive external force in the static state, its key actuating components may be damaged, shortening the robot's service life.

If the "collision mode" is set to "Quiesce After Collision," the robot will halt upon collision during operation. If the robot experience an additional external force, it will remain stationary.

2. Collision level

The robot features 10 safety levels that determine the sensitivity of the robot when a collision event is triggered. The lower the value, the higher the safety level, and the more force needed to trigger the collision event. The system default collision level is Level 6.

3. Collision mode

This mode involves how the robot is going to act after a collision event occurs.

- Floating after collision: After a collision, the robot stops and enters the HandGuide mode, allowing the operator to manually drag it to any safe position.
- Quiesce after collision: After a collision, the robot stops and remains stationary.
- Brake engagement after collision: After a collision, the robot stops and remains stationary with the brake engaged (this function is applicable only to robots with firmware version 1.*.*).

Safety

I/O

For information on safety I/O configuration methods and configurable functions, please refer to [Safety I/O](#).

Joint limit

Please refer to [Safety Limit - Joint Limit](#).

Robot limit

Please refer to [Safety Limit - Robot Limit](#).

Safe home

Please refer to [Safe Home](#).

ThreePosition

Please refer to [Safety Limit - ThreePosition](#).

Tool position

Please refer to [Safety Limit - Tool Position](#).

Plane

Please refer to [Safety Limit - Plane](#).

ObstacleMarker

Please refer to [Safety Limit - ObstacleMarker](#).

Fieldbus

Users can configure industrial computer network protocols for real-time distributed control that are acceptable to AuboStudio: EtherNet/IP, Modbus, or PROFINET.

Modbus

Introduction to Modbus

Modbus is an important communication protocol in industrial automation control systems. It has the characteristics of strong debugging ability, large data transmission volume, good real-time performance, etc. External devices (such as PLC) can establish connections with the robot via the Modbus protocol, and the robot can also communicate with the external devices as a Modbus slave.

The Modbus protocol includes Modbus RTU and Modbus TCP versions.

- Modbus RTU: It uses serial communication, represents data in a binary and compact way, and requires a checksum with cyclic redundancy check. Modbus RTU is mainly used in industrial and system automation applications where the communication distance between devices is long.
- Modbus TCP: It uses TCP/IP (such as Ethernet) connections and requires no checksum calculation. Compared with Modbus RTU, Modbus TCP has a faster communication speed and can connect more devices.

You can configure Modbus device information in the system and establish a connection with the Modbus device with a specified IP.

rob1 Simulation Manual PowerOff Undefined 2024-05-19 18:59:29

Fieldbus Modbus

Home Program Configure Move I/O

100% 0mm/s

Modbus client IO Setup

Units Signals Slave

+ Add Modbus Unit - Delete Modbus Unit

• Modbus RTU Device

Signal State	Name	Slave Number	Sequential Mode	Serial Port	Baud	Parity	Data Bit	Stop Bit	
1	0	DEVICE_1	1	☐		115200	None	8	0

• Modbus TCP Device

Signal State	Name	Slave Number	Sequential Mode	IP	Port	Operation	
1	0	DEVICE_0	1	☐	192 . 168 . 1 . 1	502	Test

Modbus unit management

1. Connect the Modbus device to the controller communication interface.
2. Click "Configure > Fieldbus > Modbus > Units." Click [Add Modbus Unit], select the Modbus unit mode, customize the device name and slave number, and set the relevant parameters of the Modbus device.
3. Select the device to be deleted and click [Delete Modbus Unit] to delete the device.

Modbus signals management

1. After adding a Modbus device, click "Configure > Fieldbus > Modbus > Signals." Click [Add Modbus Signal], customize the signal name, and set the device name, type, and signal address.
2. This system provides batch addition of Modbus signals. Click [Batch Add Modbus Signals], customize the signal names, and set the parameters such as device name and type.
3. The added Modbus signals can have custom functions. Please refer to "[I/O Setup](#)" for details.
4. Select the signal to be deleted and click [Delete Modbus Signal] to delete the signal.

Modbus slave management

The robot can also be used as a Modbus slave. Please refer to the ARCS Modbus User Guide for more details.

PROFINET

PROFINET is a network protocol that allows you to enable or disable the robot's link to the PROFINET controller.

EtherNet/IP

EtherNet/IP Introduction

EtherNet/IP is an industrial Ethernet protocol based on the TCP/IP protocol stack, established by ODVA (Open DeviceNet Vendor Association) and widely used for real-time data exchange between industrial automation devices. Its core advantages include:

- **Strong Compatibility:** Supports interconnection with various industrial devices such as PLCs, robot arms, and sensors, and is compatible with devices from mainstream manufacturers (Inovance, Omron, Siemens, etc.);
- **High Real-time Performance:** Adopts a "producer-consumer" communication model, enabling millisecond-level data transmission to meet real-time control requirements in industrial scenarios;
- **Flexible Configuration:** Standardizes device descriptions through EDS (Electronic Data Sheet) files, simplifying the adaptation process between different devices.

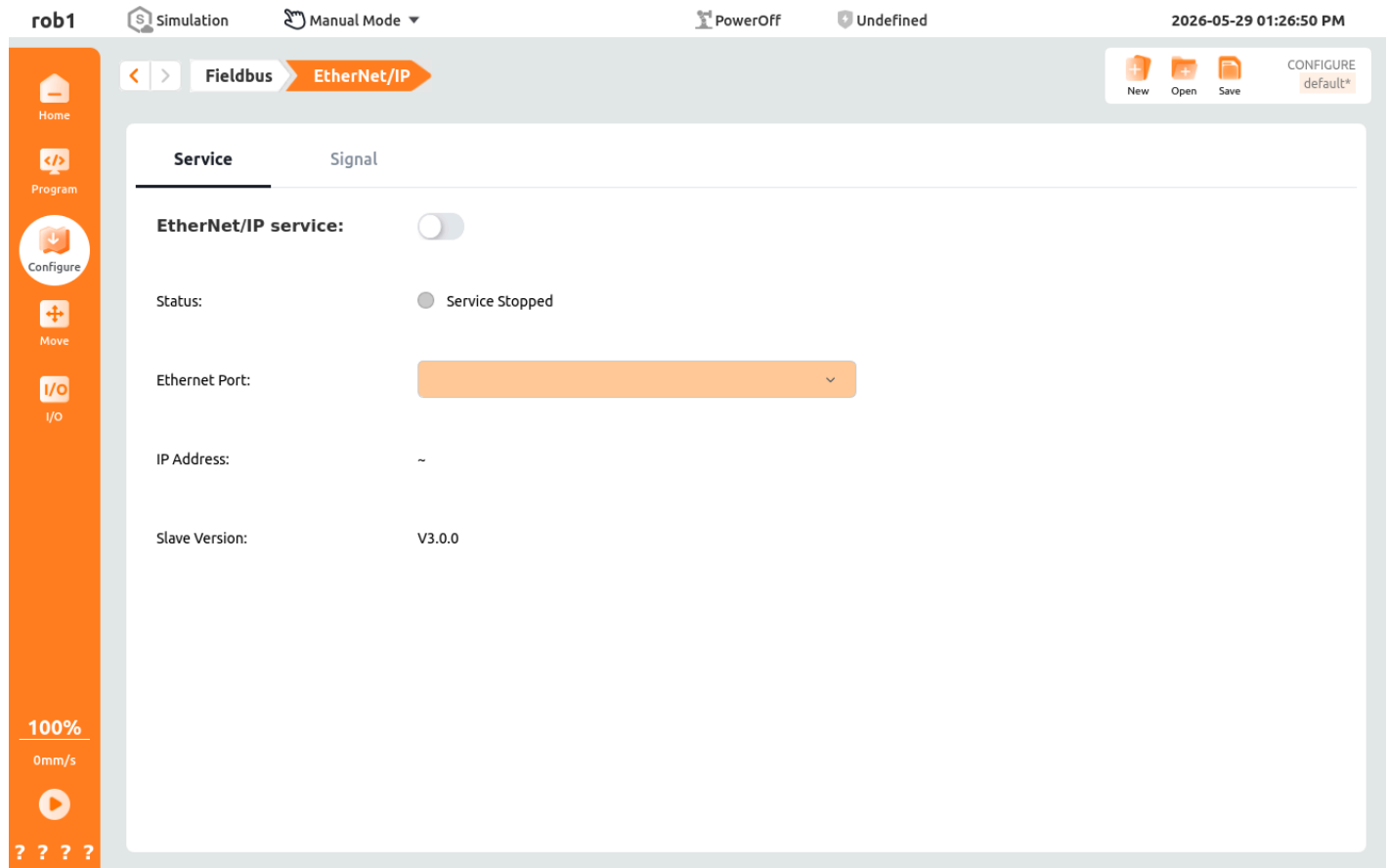
Usage

1. Hardware Connection

1. Connect the ARCS controller to the industrial network via an Ethernet cable.
2. Ensure network parameters are configured correctly.
3. Verify that the physical connection status is correct.

2. EIP Service Configuration

On the teach pendant home page, navigate to "Configure > Fieldbus > EtherNet/IP" to configure the EIP service.



EIP Service Page Functions:

- EtherNet/IP Service: Turn the EIP service on/off.
- Status: Displays service closed / master disconnected / master connected.
- Network Port: Defaults to the currently active network port; can be manually selected when the service is closed.
- IP Address: Displays the IP address of the currently active network port.
- Slave Version: Displays the EIP software version number (displayed after the master connects).

EIP Signal Page Functions:

- Signal List: EIP signal list.
- Add: Add an EIP signal.
- Delete: Delete an EIP signal.
- Name: Supports Chinese/English names.
- Signal Type:

- Input: Input signals used to receive control commands or status feedback signals sent by external devices (such as PLCs and sensors).
- Output: Output signals used to send robot status, execution results, or control command signals to external devices.
- Address Offset.Bit Offset:
 - Offset Address:
 - Input: 88-150 (even numbers only).
 - Output: 182-244 (even numbers only).
 - Bit Offset: Binary Boolean type bit offset (0~15).
- Type: Boolean, signed integer, unsigned integer, float, swapped float, binary, hexadecimal.

3. Programming Usage

Use EIP signals directly in programs without the need for indirect read/write through server registers or Modbus:

Program Node	EIP Usage
Assignment	EIP input/output signals can be used as variables
Set	EIP output signal values can be set (register and Boolean types)
Wait	Wait for EIP signal conditions to be met (e.g., EIP_1=88, EIP_2=true)
Until	EIP condition judgment; if met, skip the direction node
Loop	EIP conditions as loop conditions
If/Else If	EIP conditions as branch judgments
Switch	EIP conditions as branch expressions
Subprogram	Call EIP subprogram signals

4. IO Monitoring

Monitor EIP signal values directly on the IO interface, in the same format as the data types configured for the signals.

I/O Tab

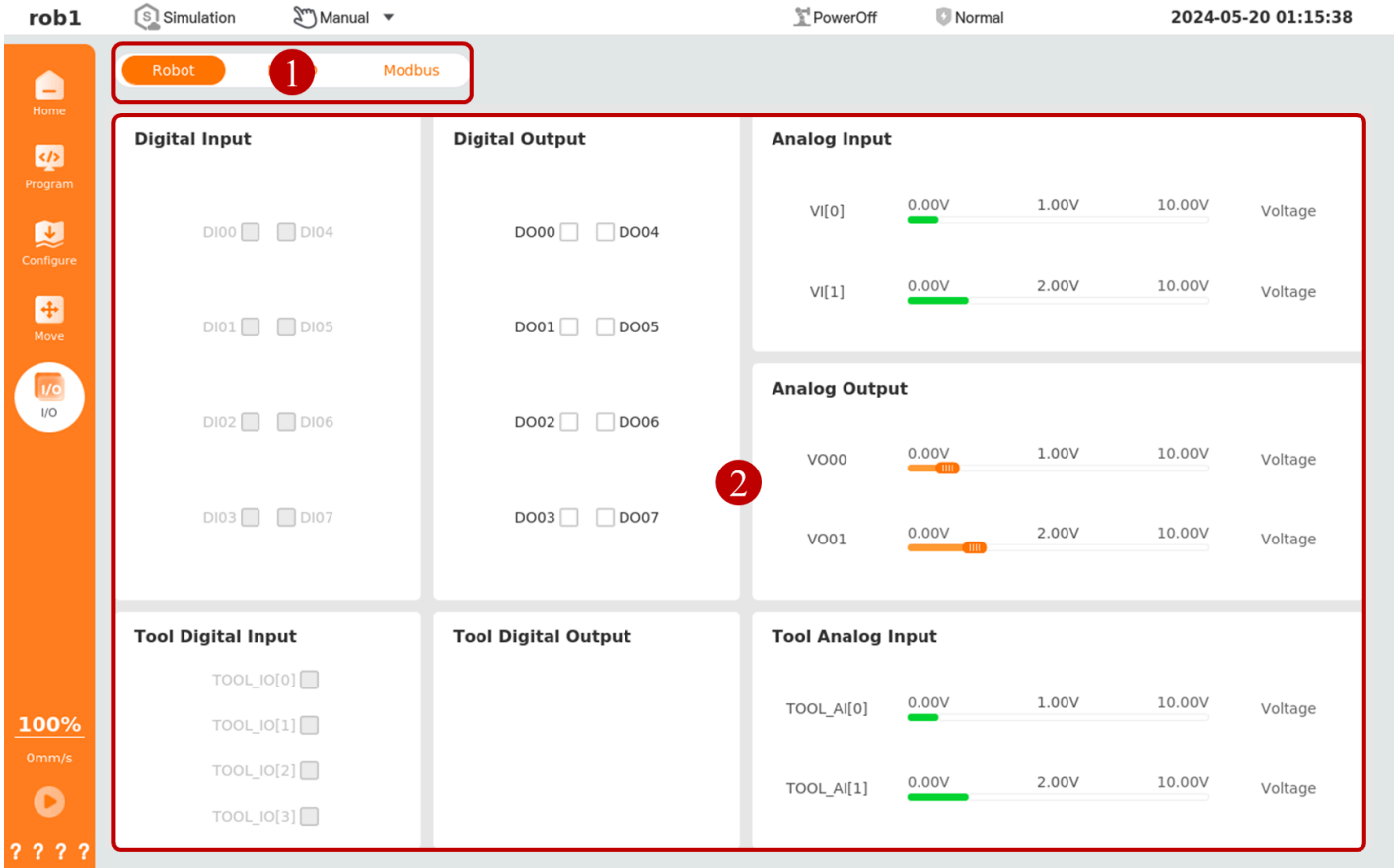
AuboStudio can monitor and set the status of all controllers, body ends, and I/O ports of external devices. In the [I/O] tab, you can monitor the status of each I/O port and can also operate the signal of the output port.

This system is classified according to the nature of I/O ports: Under the [Robot] tab, there are all user-operable I/Os, such as digital I/O, analog I/O, and tool I/O; under the [Link IO] tab, there are safety I/O and system internal I/O; under the [Modbus] tab, there are I/Os of external devices.

- General I/O: I/O with user custom function.
- Configurable I/O: It can be used as general I/O or as safety I/O. When configured for a single I/O function, it can be used as a general I/O. When configured for a configurable I/O, it can be used as safety I/O.
- Linked I/O: I/O with fixed functions, such as remote power on/off on the controller.
- Tool I/O: Specifically refers to the I/O at the end of the tool.

Info

The update frequency of the [I/O] tab is 20 Hz, so signals with frequencies above 20 Hz may not be displayed correctly.



Serial number	Name	Description
1	I/O tab	Select I/O category
2	I/O list	Display I/O under the current category

Introduction to I/O signal status

Only some models of controllers support NPN/PNP mode switching. Please refer to the Hardware User Manual for details.

Name	English	Description
Logic High	Logic High	High-level state in a digital signal. * In NPN mode, when the digital I/O port is in logic high, the output will be in the "on" state and the voltage will be close to the ground voltage (such as 0 V). * In PNP mode, when the digital I/O port is in logic high, the output will be in the "on" state and the voltage will be close to the power supply voltage (such as 24 V).

Logic Low	Logic Low	Low-level state in a digital signal. * In NPN mode, when the digital I/O port is in logic low, the output will be in the "off" state, and the voltage will have no output and will be cut off. * In PNP mode, when the digital I/O port is in logic low, the output will be in the "off" state, and the voltage will have no output and will be cut off.
Rising edge	Rising Edge	Transition edge of a signal from logic low to logic high.
Falling edge	Falling Edge	Transition edge of a signal from logic high to logic low.
Maximum voltage	Maximum Voltage	The maximum voltage is the highest voltage value that can be achieved by an analog signal. For the specific range of analog signal, please refer to the Hardware User Manual.
Minimum voltage	Minimum Voltage	The minimum voltage is the lowest voltage value that can be achieved by an analog signal. For the specific range of analog signal, please refer to the Hardware User Manual.

I/O Configuration

[Configuration > I/O Settings] allows you to configure the functions of the I/O ports available to users, including the digital I/O port, analog I/O port, and tool I/O port. For some controller models, safety I/O ports can be used as general I/O ports when they are not assigned with any value. Please refer to the [Hardware User Manual](#) for details.

The screenshot displays the AUBO Developer software interface for I/O configuration. The top bar shows the robot name 'rob1', simulation status, and various mode buttons. The left sidebar contains navigation icons for Home, Program, Configure, Move, and I/O. The main area is divided into 'Input' and 'Output' tabs. The 'Input' tab shows a table of digital input pins (DI00 to DI17) and tool I/O pins (TOOL_IO[0] to TOOL_IO[3]). The 'Output' tab is also visible. On the right, a 'Pin: DI04' configuration panel is shown, allowing the user to set the name and action for the selected pin. The 'Action in Program' dropdown is open, showing a list of actions including 'None', 'HandGuide', 'Go home', 'Start program', 'Stop program', 'Pause program', 'Popup dismiss', 'Poweron robot', 'Poweroff robot', and 'Resume program'.

Pin	Name	Action
* DI00	DI00	None
* DI01	DI01	None
* DI02	DI02	None
* DI03	DI03	None
* DI04	DI04	None
* DI05	DI05	None
* DI06	DI06	None
* DI07	DI07	None
* DI10	DI10	None
* DI11	DI11	None
* DI12	DI12	None
* DI13	DI13	None
* DI14	DI14	None
* DI15	DI15	None
* DI16	DI16	None
* DI17	DI17	None
* TOOL_IO[0]	TOOL_IO[0]	None
* TOOL_IO[1]	TOOL_IO[1]	None
* TOOL_IO[2]	TOOL_IO[2]	None
* TOOL_IO[3]	TOOL_IO[3]	None

- All: Filter the display in the I/O list by signal type. Signal types include analog, digital, tool, Modbus, etc.
- Action Preview: Display all I/O signals available to users.
- Input: Display/set all input I/O signals available to users.
- Output: Display/set all output I/O signals available to users.
- Pin: Display the interface number of the currently selected I/O signal.
- Name: Display the name of the currently selected I/O. Click to rename it.
- Restore Default: This button appears after renaming an I/O. Click to restore the default I/O name.

- I/O Tab Control: Set whether the I/O can be manually controlled on the [I/O] tab.
 - Enable: Allow manual control of this output port's signal on the [I/O] tab.
 - Manual Mode Only: Only allow manual control of this output port's signal on the [I/O] tab when in manual mode (status displays "**").
 - Disable: Prohibit manual control of this output port's signal on the [I/O] tab.
- Action in Program: Set the action triggered when the input/output port sends a signal. Click to open a dropdown box and select a system predefined action.

- Input Actions

Action	Description
HandGuide	When triggered, the robot enters HandGuide mode, allowing the operator to manually move the robot arm.
Move to Home	When this signal is triggered, the robot will automatically return to the zero position or starting point along the preset path.
Start Program	Start the currently loaded project via an external signal.
Stop Program	Completely terminate the running program via an external signal.
Pause Program	Temporarily suspend program execution via an external signal; the robot maintains its current pose.
Dismiss Popup	When a prompt popup or non-fatal warning popup appears on the teach pendant interface, use this signal to remotely simulate clicking "Confirm" or "Close."
Robot Power On	Power on (enable) the servo motor of the robot body via an external signal.
Robot Power Off	Cut power to the servo motor of the robot body via an external signal (disable/power off).
Resume Program	When the program is in a paused state, use this signal to resume execution from the breakpoint.
Start Conveyor Tracking	Start the conveyor tracking function, enabling the robot arm to move synchronously with workpieces on the conveyor.
Conveyor Trigger Queue (Low)	Typically used to stop or exit the conveyor tracking mode.

Release Protective Stop	After the protective stop condition is cleared, use this signal to restore the system to normal status.
Force Resume Program	Typically used after a non-standard pause (such as after an alarm is cleared) to force the program to resume execution from the current position or a specified logical point.
Load Program	Trigger the robot to load a preset project file via an external signal (usually requires specific parameters or file names).

- Output Actions

Action	Description
Low When Program Stopped	When the robot program stops executing, this port outputs a low-level signal.
High When Program Stopped	When the robot program stops executing, this port outputs a high-level signal.
Program Running	When the robot program is in execution state, this signal is active.
Program Paused	When program execution is temporarily suspended, this signal is active.
At Home	When the robot end flange or all joint axes reach the predefined home position, this signal is activated.
HandGuide	Indicates that the robot is currently in zero-force control mode, allowing the operator to manually drag the robot arm for teach-in.
Robot Powered On	When the main power of the controller is on and the servo drive is enabled, this signal is activated.
Internal Emergency Stop	Activated when the emergency stop condition is triggered by an internal source of the system.
External Emergency Stop	Activated when the emergency stop condition is triggered by external hardware connected to the safety input.
System Error	Activated when the controller detects a critical hardware or software fault that prevents normal operation.
Non-System Error	Activated when a non-critical, typically recoverable error or warning condition is detected.
Robot Operable	Activated when the robot is fault-free, powered on, and in a state where it can accept control commands.

Operation Mode	Indicates the current operating mode of the robot controller.
Safety Stop	Activated when a controlled stop triggered by a configured safety function occurs. Distinct from emergency stop.
Protective Stop	Activated when an external protective device connected to the system triggers the stop mechanism.
Program Unexpected Stop	When a running program is aborted due to an internal logic error, runtime exception, or external interference, this signal is activated.

I/O Function

System predefined status

Table: Predefined digital input port functions

Action	Description
None	No action triggered
HandGuide	Level triggering. When a Logic High-level signal is connected, the robot will enter the HandGuide mode. * Ignore input if there are other conditions that do not allow HandGuide. * It is valid in manual mode.
move to home	Level triggering. When a Logic High-level signal is connected, the robot starts MoveJ to the home. * Keep the Logic High during the robot movement; otherwise, the robot will stop moving. * It is valid in link mode.
Start program	Edge triggering. Start the currently loaded program when rising edge. * It is valid in link mode.
Stop program	Edge triggering. Stop the currently running program when rising edge. * It is valid in link mode.
Pause program	Edge triggering. Suspend a currently running program when rising edge. * It is valid in link mode.
Recover program	Edge triggering. Resume running a paused program when rising edge. * It is valid in link mode.
Disable popup	Edge triggering. Disable the current popup when rising edge. * It is valid in link mode.
Robot power-on	Edge triggering. * Power on the robot when rising edge while the robot is in the power-off state. The robot will then switch from power-off to idle state. * Release the brake when the rising edge occurs while the robot is in an idle state. The robot will then switch from idle state to operating state. * It is valid in link mode.
Robot poweroff	Edge triggering. Power off the robot on the rising edge while the robot is in operation or idle state. The robot will then switch from running or in an idle state to power-off state. * It is valid in link mode.

Load program	Edge triggering. Load the selected program when rising edge. * It is valid in link mode.
--------------	---

Table: Predefined digital output port functions

actions in the program	signal status
None	No actions
Logic low after program stop	When the program execution is completed or terminated, logic low will be set.
Logic high after program stop	When the program execution is completed or terminated, logic high will be set.
Program running	When the program is executing, logic high will be set. When the program is stopped or paused, logic low will be set.
Program paused	When the program is paused, logic high will be set.
Arrival at home	When the robot arm joint's spatial position reaches the home, logic high will be set.
HandGuide	When the robot is in the HandGuide mode, logic high will be set.
Robot power-on	When the robot arm is powered on, logic high will be set.
Internal emergency stop	When the robot's emergency stop device triggers an emergency stop, logic high will be set. * The functions of the robot's emergency stop device cover the teach pendant, control cabinet and fixed emergency stop input. * When connected to the emergency stop input of another robot, shared robot emergency stop can be achieved.
External emergency stop	When the external emergency stop device triggers an emergency stop, logic high will be set. * The external emergency stop device can be configured via the "emergency stop" input into the security I/O port.
System error	In case of failure, overrun, emergency stop, safe stop, or protective stop, logic high will be set.
Non-system error	When the system is in normal mode, reduced mode, or recovery mode, logic high will be set.
Robot operable	When the brake is released, the robot is powered on and in operation, and logic high will be set.

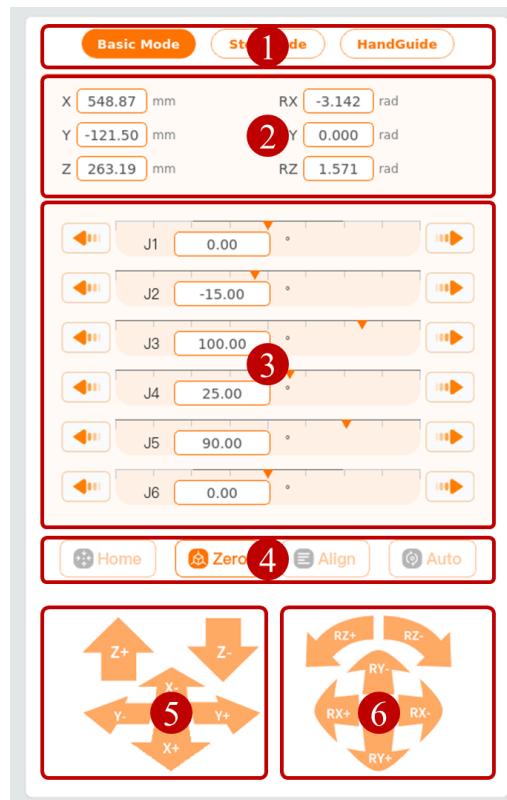
Table: Predefined analog output functions

actions in the program	signal status
None	No actions
Minimum after program stop	The minimum voltage will be output when the program execution is completed or terminated.
Maximum after program stop	The maximum voltage will be output when the program execution is completed or terminated.
Maximum during program runtime	The maximum voltage will be output when the program is executing. The minimum voltage will be output when the program is stopped or paused.
Maximum after program pause	The maximum voltage will be output when the program is paused.
Maximum at home	The maximum voltage will be output when the robot arm joint's spatial position reaches the home.
Maximum in HandGuide mode	The maximum voltage will be output when the robot is in the HandGuide mode.
Maximum when the robot is powered on	The maximum voltage will be output when the robot arm is powered on.
Maximum when the internal emergency stop is triggered	The maximum voltage will be output when the robot's integral emergency stop device triggers an emergency stop. * The functions of the robot's emergency stop device cover the teach pendant, control cabinet, and fixed emergency stop input. * When connected to the emergency stop input of another robot, shared robot emergency stop can be achieved.
Maximum when an external emergency stop is triggered	The maximum voltage will be output when the external emergency stop device triggers an emergency stop. * The external emergency stop device can be configured via the "emergency stop" input into the security I/O port.
Maximum when a system error occurs	The maximum voltage will be output in case of failure, overrun, emergency stop, safe stop, or protective stop.
Maximum when there is no system error	The maximum voltage will be output when the system is in normal mode, reduced mode, and recovery mode.
Maximum when the robot is operable	The maximum voltage will be output when the brake is released and the robot is powered on and in operation.

Jog Teaching

Basic mode

The robot is controlled in basic mode by default.



1. Mode tab: Select a mode to control the [Robot] or [External Axis].

Note:

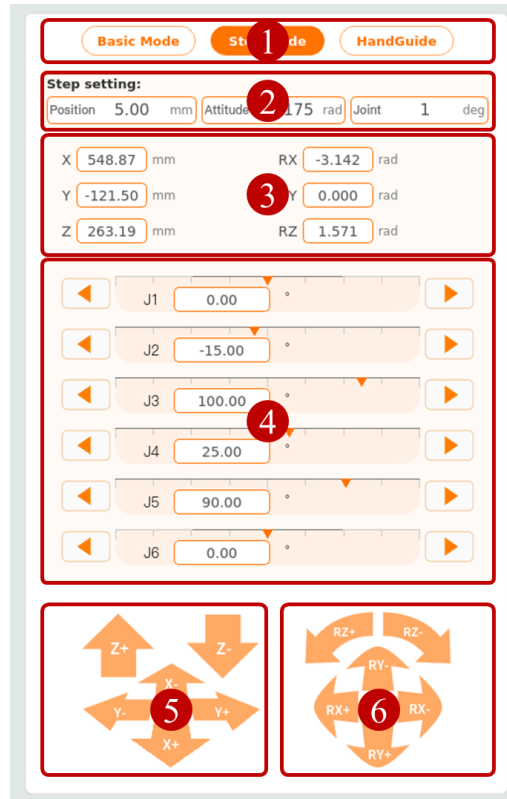
Only the wireless teach pendant supports the [External Axis] function.

2. Position and pose display/setting: Display or set the coordinates of the current position of the simulation model in the base coordinate system. Once set by the user, the orange parameter displayed on the right side of the input box is the movement amount of the robot.
3. Joint control: Joint 1 (J1) ~ Joint 6 (J6) correspond to the six joints of the robot arm from bottom to top. The user controls the robot by pressing and holding  .
4. Shortcut control button: Press and hold the button to control the robot to the specified position.

- Home: The default value set in the installation configuration. Press and hold the [Home] button to move the robot to the home position defined under the installation settings. Please refer to "[Home](#)" for the settings of home.
 - Zero: All positions where the joint is 0 degrees. Press and hold the [Zero] button to move the robot to the upright position/pose.
 - Alignment: Press and hold to adjust the end of the arm to the negative direction of the selected coordinate system's Z axis.
 - Auto: After setting in the position and pose input box, press and hold the [Auto] button to control the robot to the set position.
5. Position control button: Press and hold the button to control the robot. (X+, X-, Y+, Y-, Z+, Z-) indicate the translation movement of the robot.
6. Pose control button: Press and hold the button to control the robot. (RX+, RX-, RY+, RY-, RZ+, RZ-) change the orientation of the robot tool.
-

Step mode

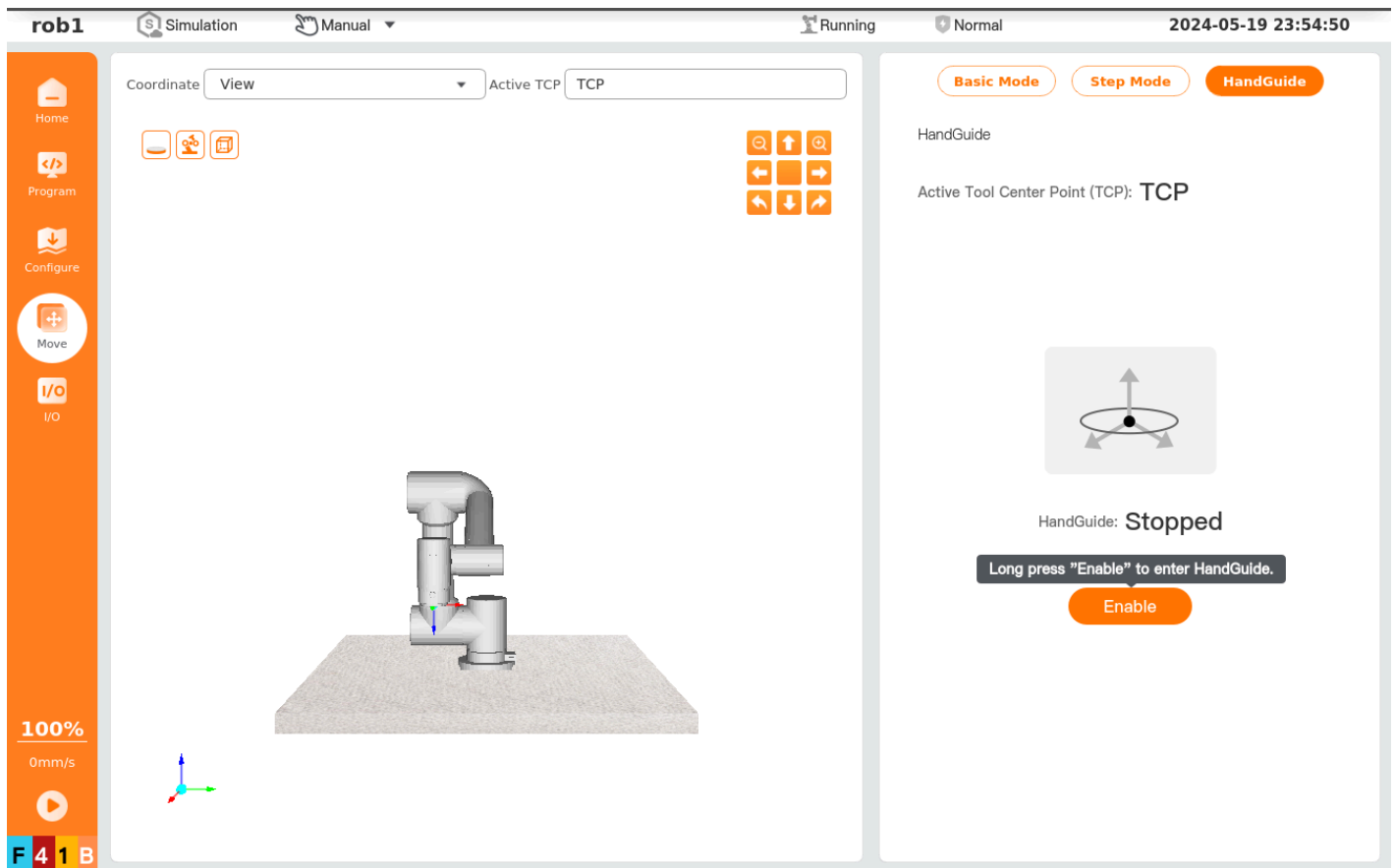
In the step mode, you can set the step value. Default step setting: position 5 mm, pose 0.0175 rad, and joint 1 deg.



1. Mode tab: Select a mode to control the robot.
2. Step setting: Set the step. The position range is 0.01 mm ~100 mm, the pose range is 0.001 rad ~ 0.2 rad, and the joint range is 0.1 deg ~ 10 deg.
3. Position/pose parameter display: Display the coordinates of the current position of the robot in the base coordinate system.
4. Joint control: Joint 1 (J1) ~ Joint 6 (J6) correspond to the six joints of the robot arm from bottom to top. The user controls the robot by clicking  buttons at both ends. Each click moves the robot once according to the step size.
5. Position control button: Click the button to control the robot. (X+, X-, Y+, Y-, Z+, Z-) indicate the translation movement of the robot.
6. Pose control button: Click the button to control the robot. (RX+, RX-, RY+, RY-, RZ+, RZ-) change the orientation of the robot tool.

HandGuide

In HandGuide mode, you can manually drag the robot.

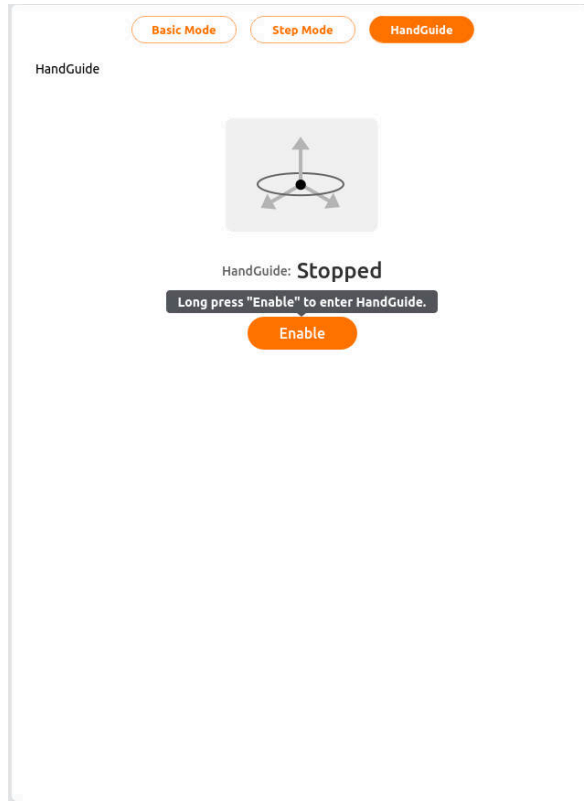


- HandGuide: Long press [Enable] to enter the HandGuide mode; the HandGuide displays "Started," and you can directly drag the robot to the target position/pose. Click [Disable] to exit the HandGuide mode; the HandGuide displays "Stopped," and then the robot keeps at the current position/pose.

Limited HandGuide

Limited HandGuide can be enabled or disabled in "[Setting > Preference > Robot](#)."

When limited HandGuide is enabled, you can lock one or more axes, controlling the TCP to move in the specified direction.



Icon	Description
	Moving is allowed along all axes.
	Moving is only allowed along the X and Y axes.
	Moving is allowed along all axes, but rotation is not allowed.
	Rotation is allowed in all axes, but moving is not allowed.

Force-Controlled Dragging

The force-controlled dragging function allows users to directly pull the robot arm by applying external force in HandGuide mode, enabling teach-in programming.

Trigger Methods

1. End button trigger: Install a force control button on the robot arm's end flange, and press the button to enter dragging mode.

2. Teach pendant button trigger: Activate dragging mode using the dedicated three-position button on the back of the teach pendant.
3. External IO trigger: Control dragging mode via external digital input signals.

Note:

DI Action needs to be configured as HandGuide.

Dragging Operation

1. After activating dragging mode, the robot arm enters free movement mode.
2. The user applies external force to pull the robot arm to the target position.
3. The robot arm responds to the dragging force in real time.

Exiting Dragging

1. Release the button.
2. Click the teach pendant button again.
3. Remove the external IO signal.

Drag damping

[Drag Damping]: Set the force required to drag each joint of the robot. The higher the value set, the more force required to drag the joint.

Info

To save the set parameters, click the [Save] button in the upper right corner. Clicking [Apply] or re-powering the robot will only make the set parameters effective but will not save them.

rob1

Simulation

Manual

PowerOff

Undefined

Home

Program

Configure

Move

I/O

100%

0mm/s

?

?

?

?

?

<>GenerallyDragDamping

NewOpenSaveCONFIGUREdefault_4*

Drag Damping

ⓘ

Six sliders correspond to drag damping of six joints respectively. The range is 0%-100%, and the higher the value, the greater the resistance

Joint1

0%

100%

<50%>

Joint2

0%

100%

<50%>

Joint3

0%

100%

<50%>

Joint4

0%

100%

<50%>

Joint5

0%

100%

<50%>

Joint6

0%

100%

<50%>

Apply

Preferences

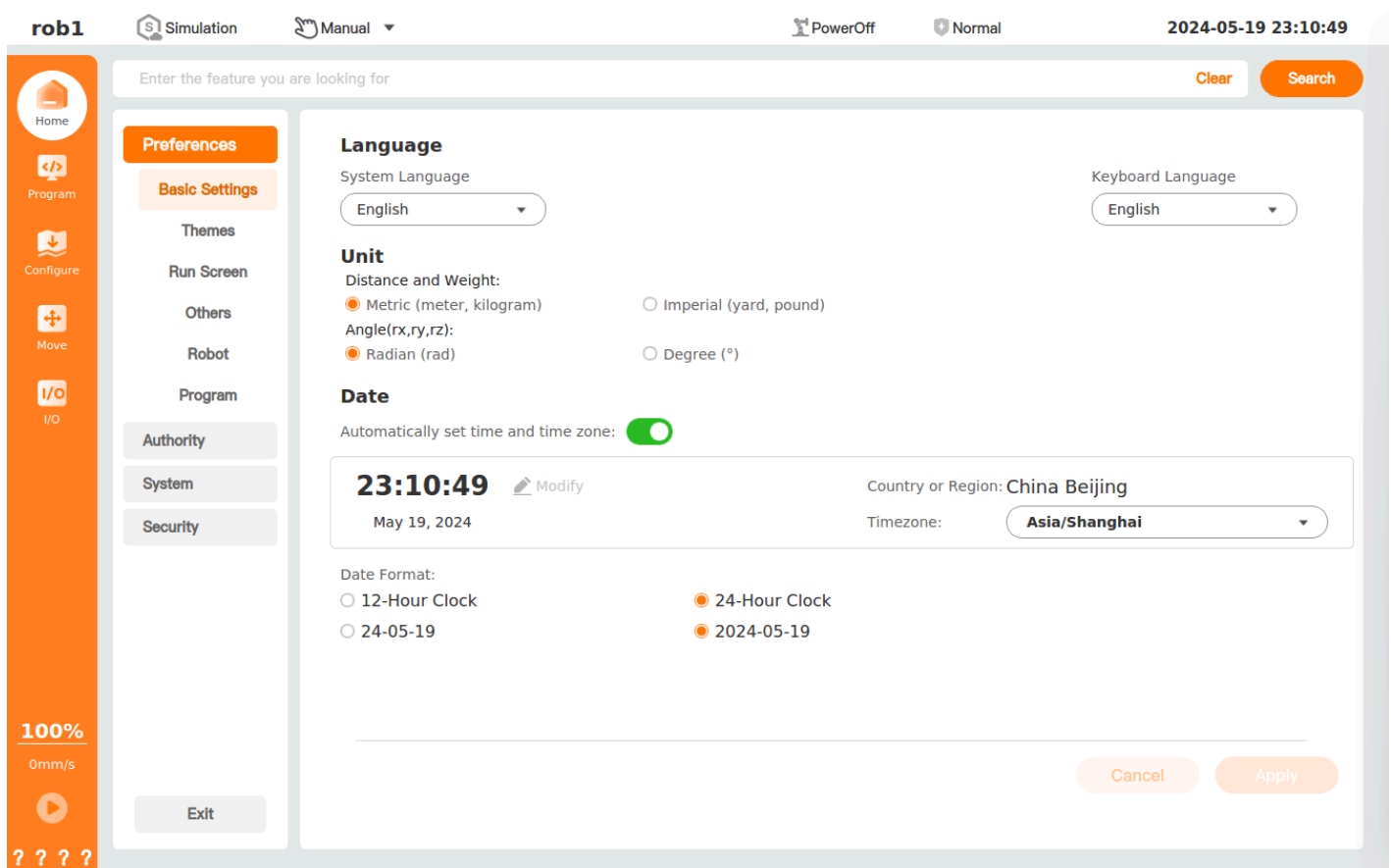
The [Preferences] settings provide users with personal system settings, which can be set according to users' own needs.

Basic settings teach pendant only

In the [Basic Settings] page, you can set the language, unit, date, etc.

Info

The settings of this interface can take effect only after the system is restarted.

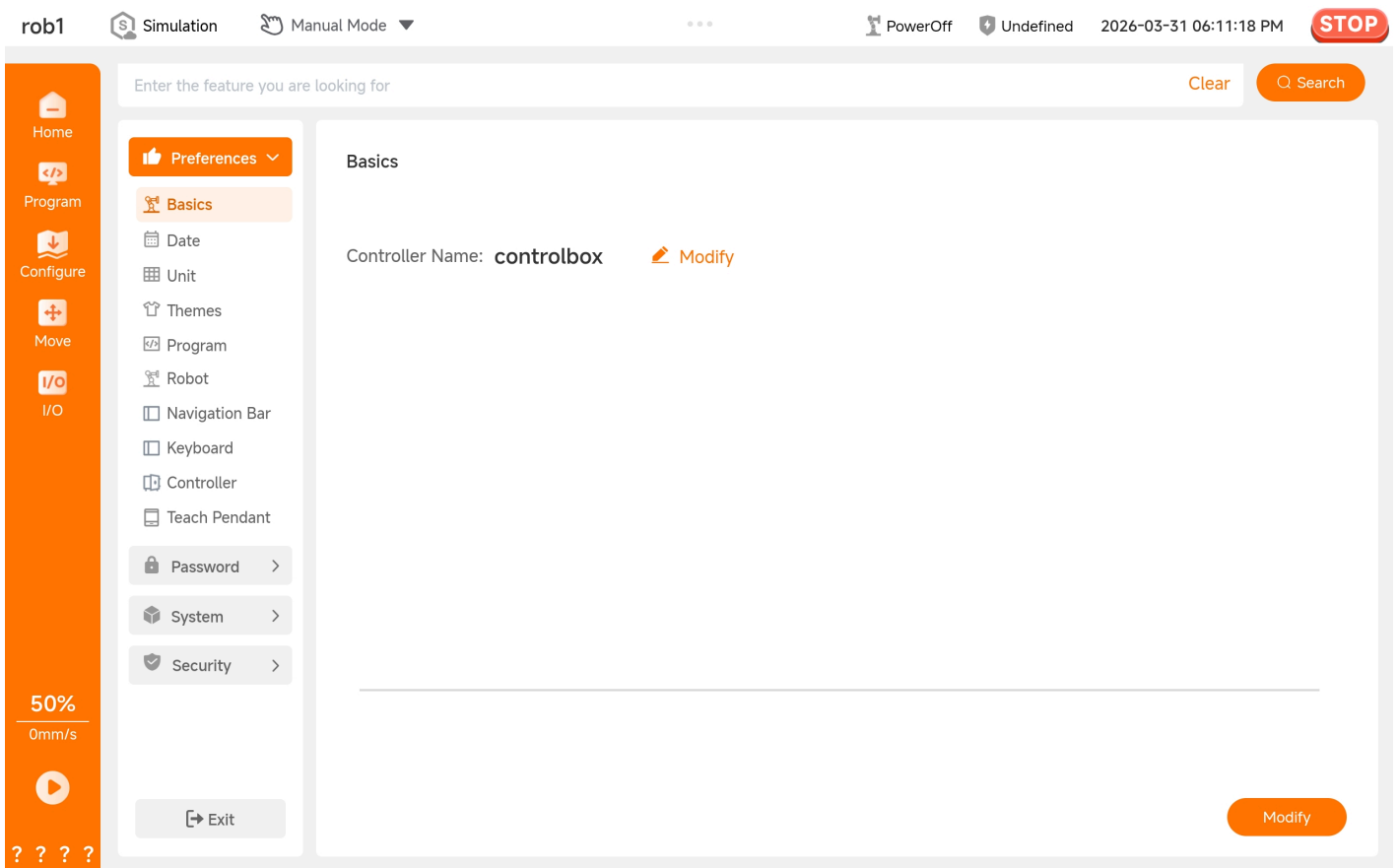


- **Language:** Users can set system language and key board language.
 - **System Language:** Allows users to set the system language. This feature supports Czech, English, Japanese, Korean, Russian, Simplified Chinese, and Traditional Chinese.
 - **Keyboard Language:** Allows users to set the keyboard language. This feature supports Chinese and English.
- **Unit:** Users can set the units of the system.

- Date: Users can set the time and date format.
- Time: AuboStudio supports two modes: System time and custom time. Select whether to follow the system time through . If you do not want to set time to follow the system time, please click [Modify]  to custom the time.
- Apply: The settings are applied.
- Cancel: The settings are canceled.

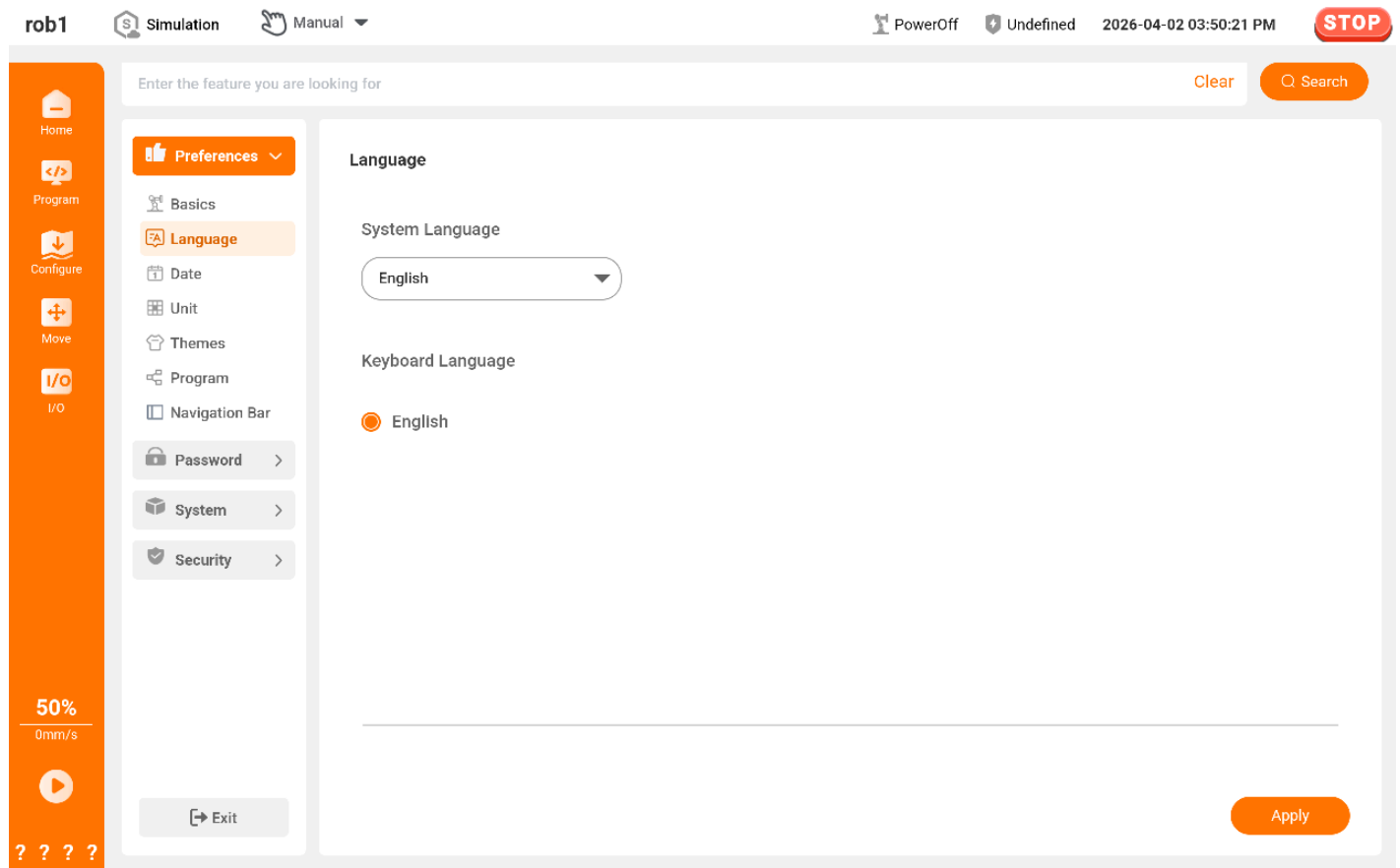
Basic Android only

The [Basic] function allows users to set the name of the currently connected controller. Click [Modify] to make it take effect.



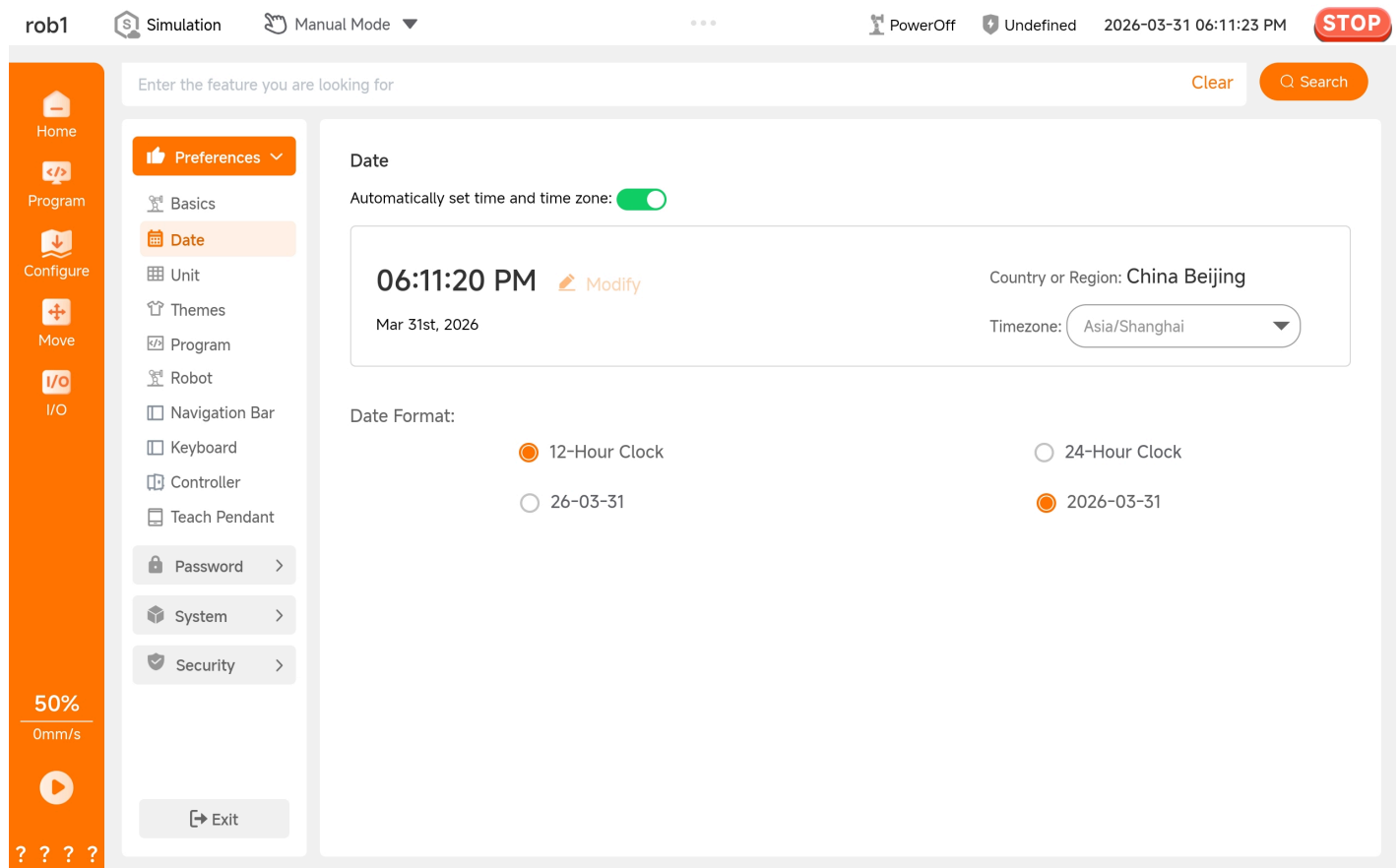
Language Android only

The [Language] function allows users to set the language of the system and keyboard.



Date Android only

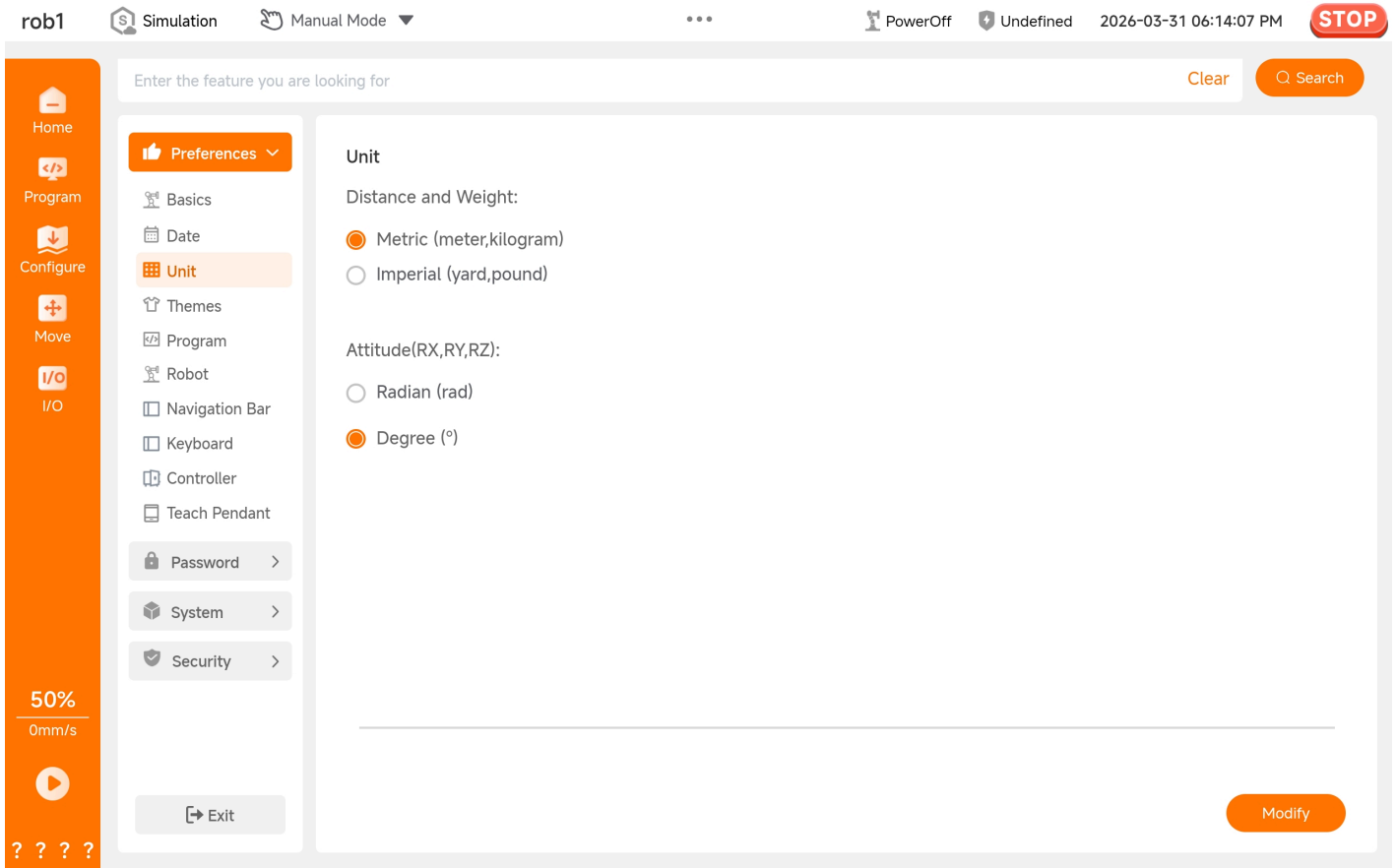
The [Date] function allows users to set the time and date format. AuboStudio supports two modes: System time and custom time.



-  : Select whether to follow the system time. If not, click [Modify]  to custom the time.
- Date format: Set the display format of the Date status bar.

Unit Android only

The [Unit] function allows users to set the units of the system.

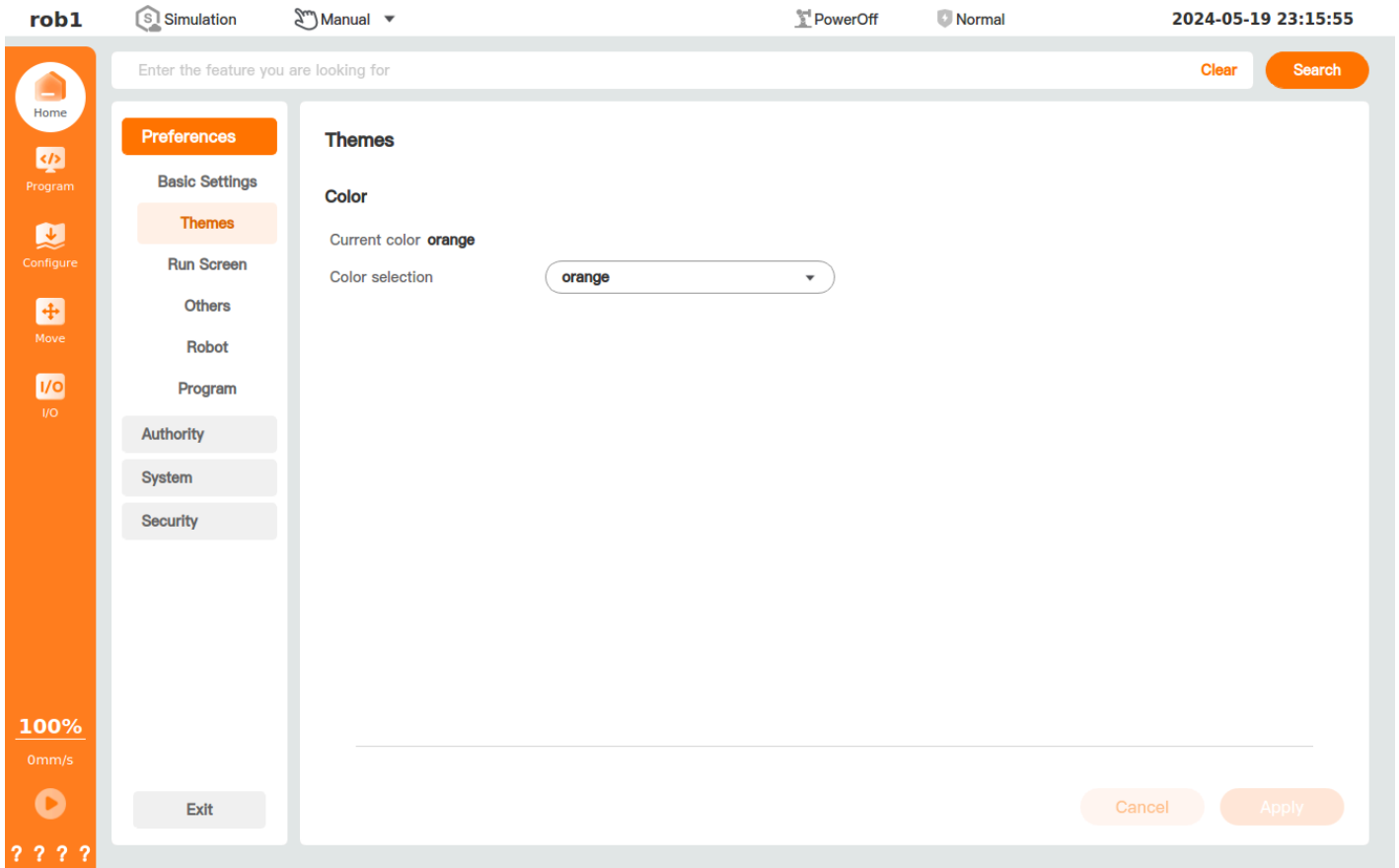


Theme

In the [Theme] page, users can set the theme of the software interface.

Info

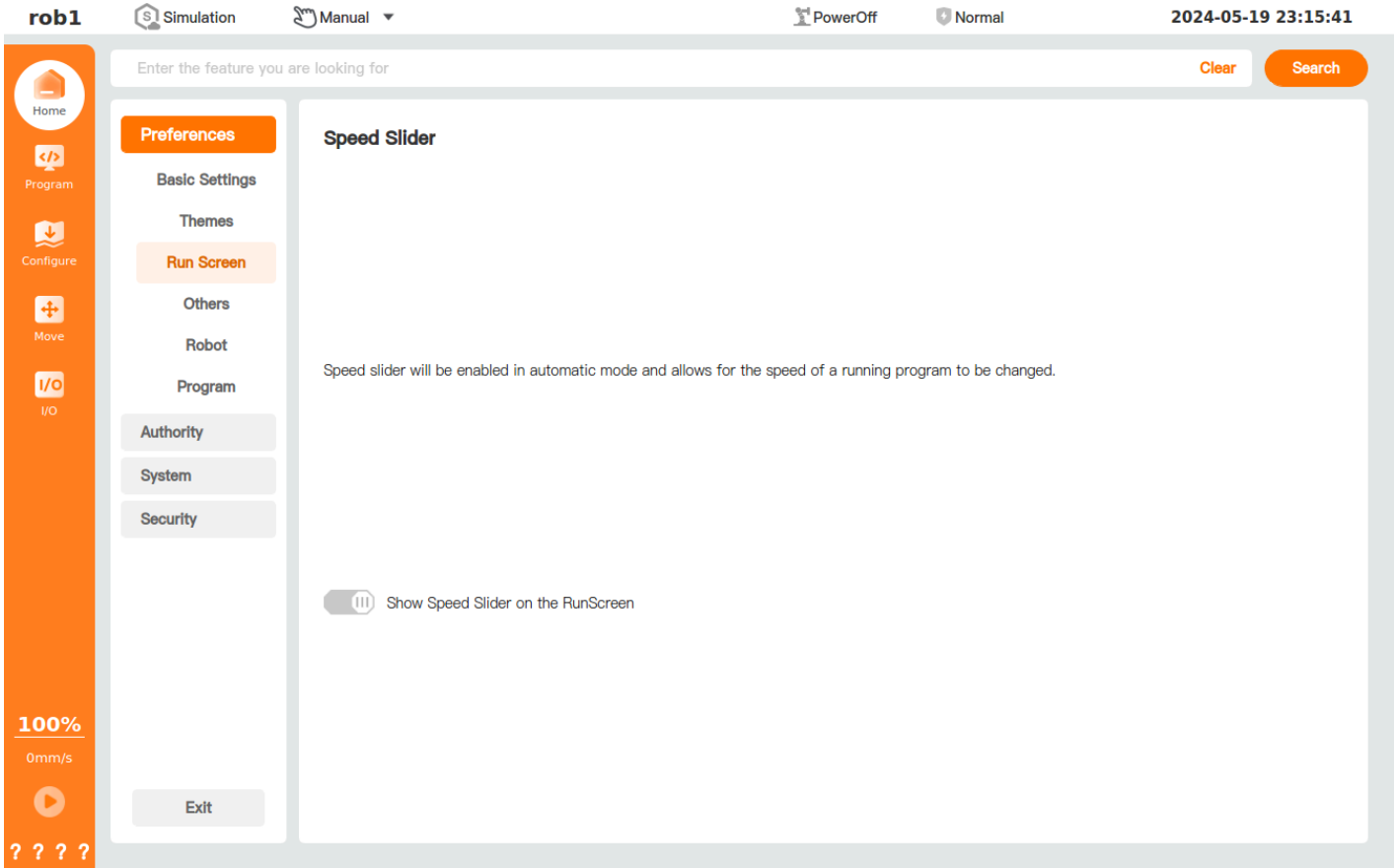
The settings of this interface can take effect only after the system is restarted.



Run screen

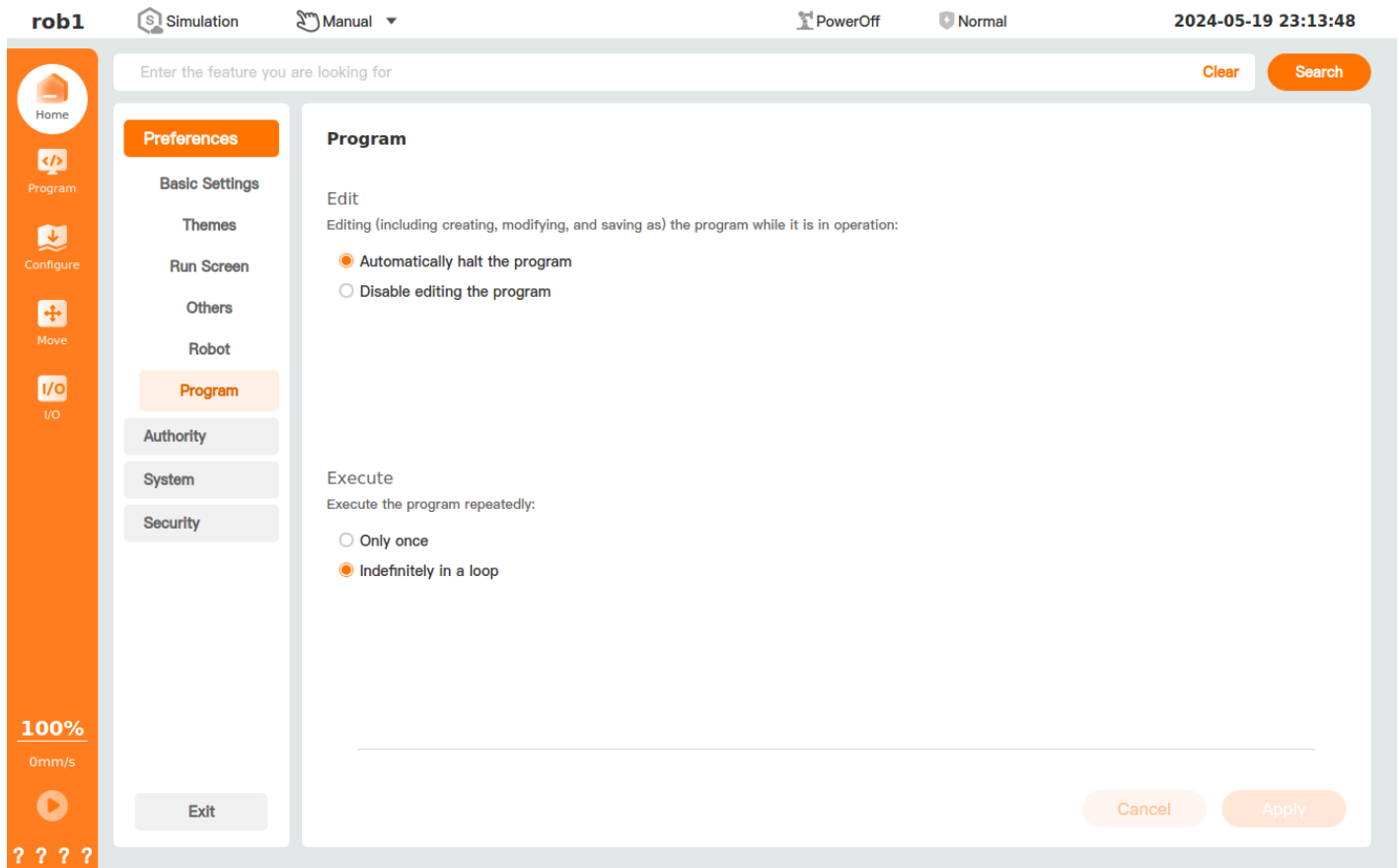
In [Run Screen] page, users can turn on or off the movement speed slider of [Speed].

In automatic mode, if this function is turned on, click [Speed] **30%** 0 mm/s at the bottom of the left menu bar; the movement speed slider will appear. If this function is turned off, the movement speed slider will not appear, i.e., the movement speed cannot be adjusted.



Program

In the [Program] page, users can set the default configuration of the editing program.



- **New Program:**

Select the run mode for the newly created program.

- **Execute once only:** The program automatically ends after executing from the first instruction line to the last instruction line, and will not repeat.
- **Loop continuously:** After executing the last instruction line, the program automatically jumps back to the first instruction line and repeats continuously until a stop command is received.

- **Edit Program:**

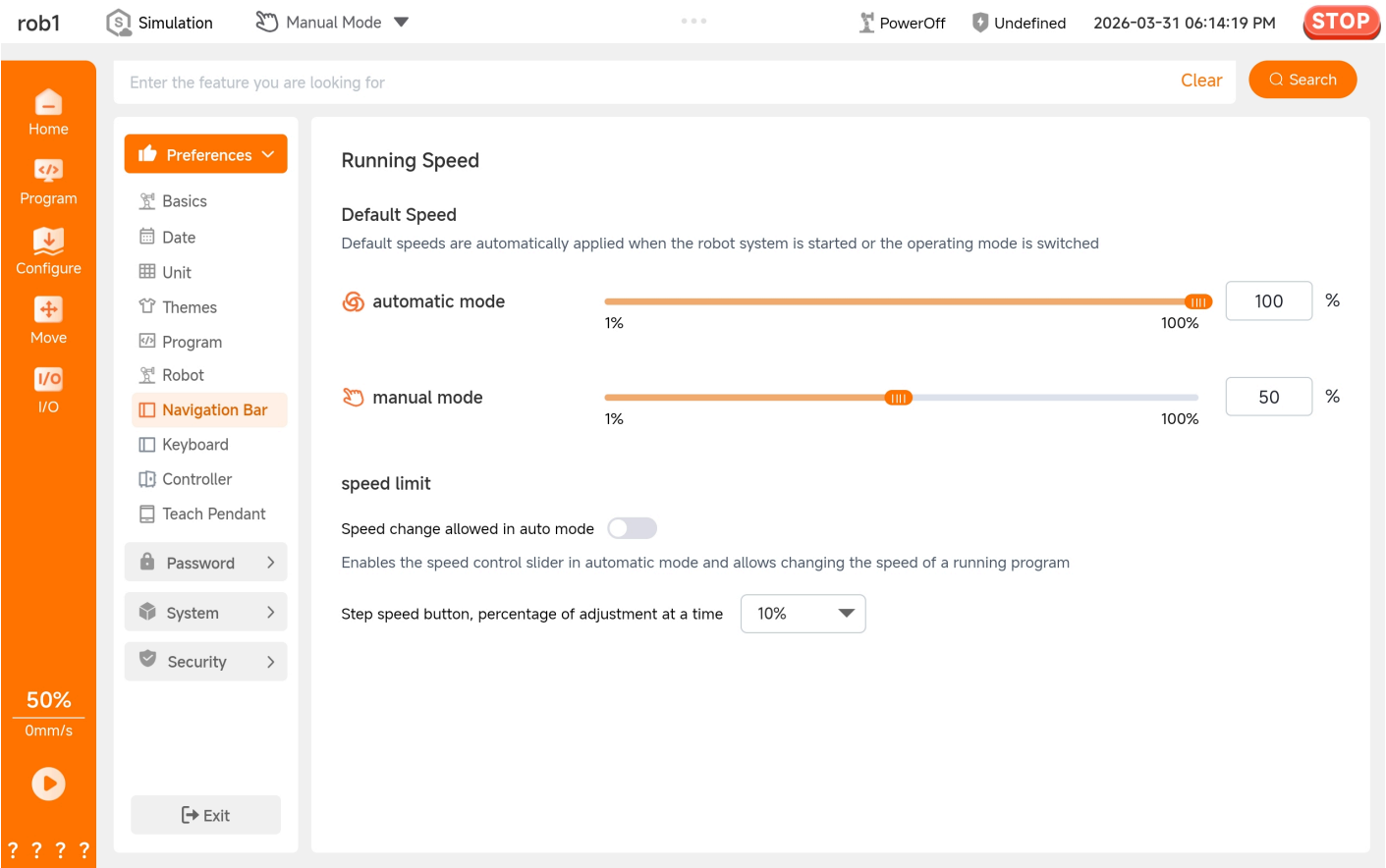
Select the action taken on a running program when it is being edited.

- **Stop the program automatically:** When an edit operation is triggered, the running program immediately stops automatically and enters editing mode.
- **Prohibit editing the program:** While the program is running, the system locks the editing function, preventing create, modify, save as and other operations.
- **Ask whether to stop the program:** When an edit operation is triggered, the system pops up a confirmation dialog asking whether to stop the currently running program. Only after the user confirms the stop can editing mode be entered.

NavBar

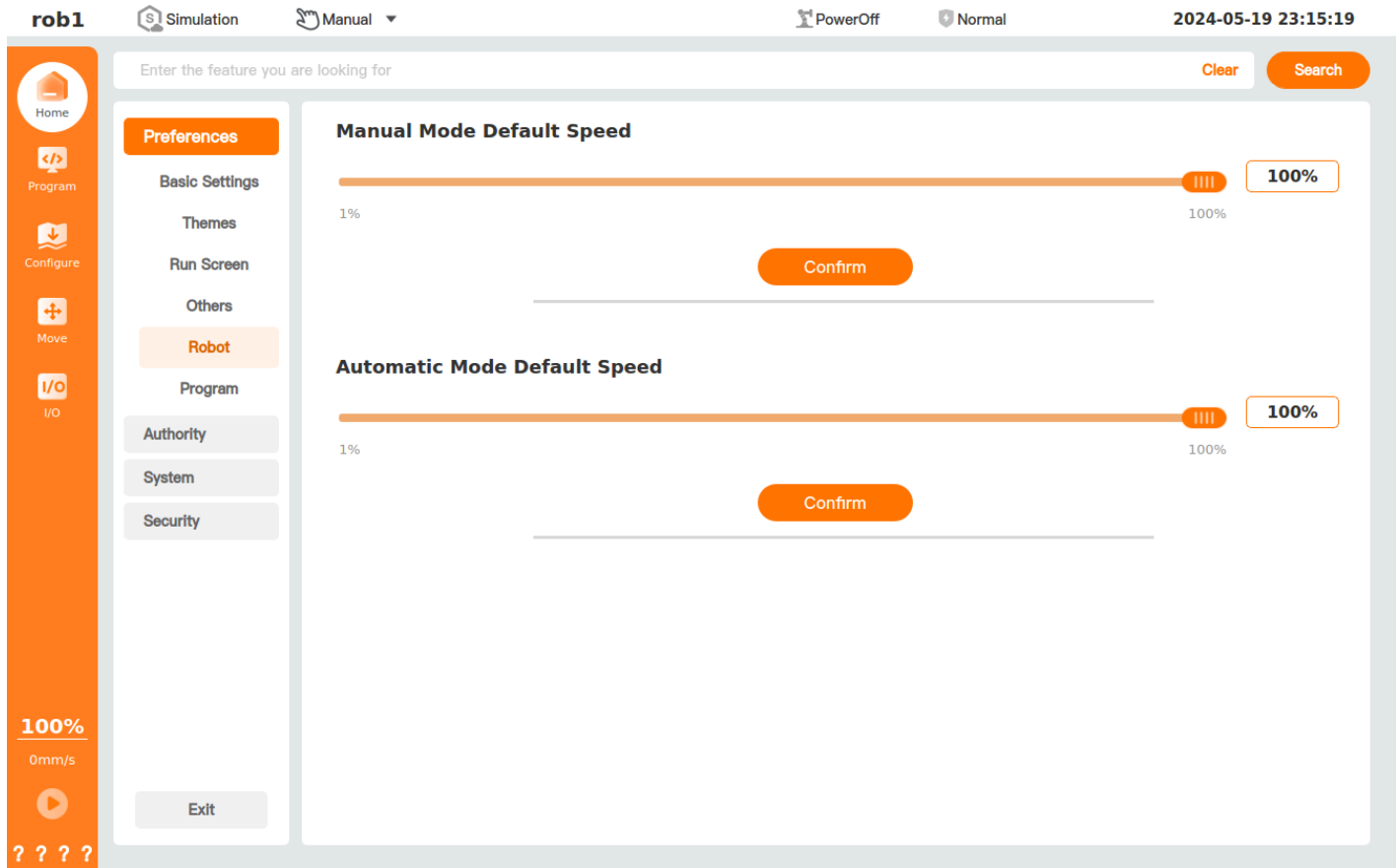
Android only

In the [NavBar], users can set the default running speed, enable/disable speed adjustment in automatic mode, and set the speed adjustment step.



Robot

The [Robot] function allows users to set a default running speed and limited HandGuide.



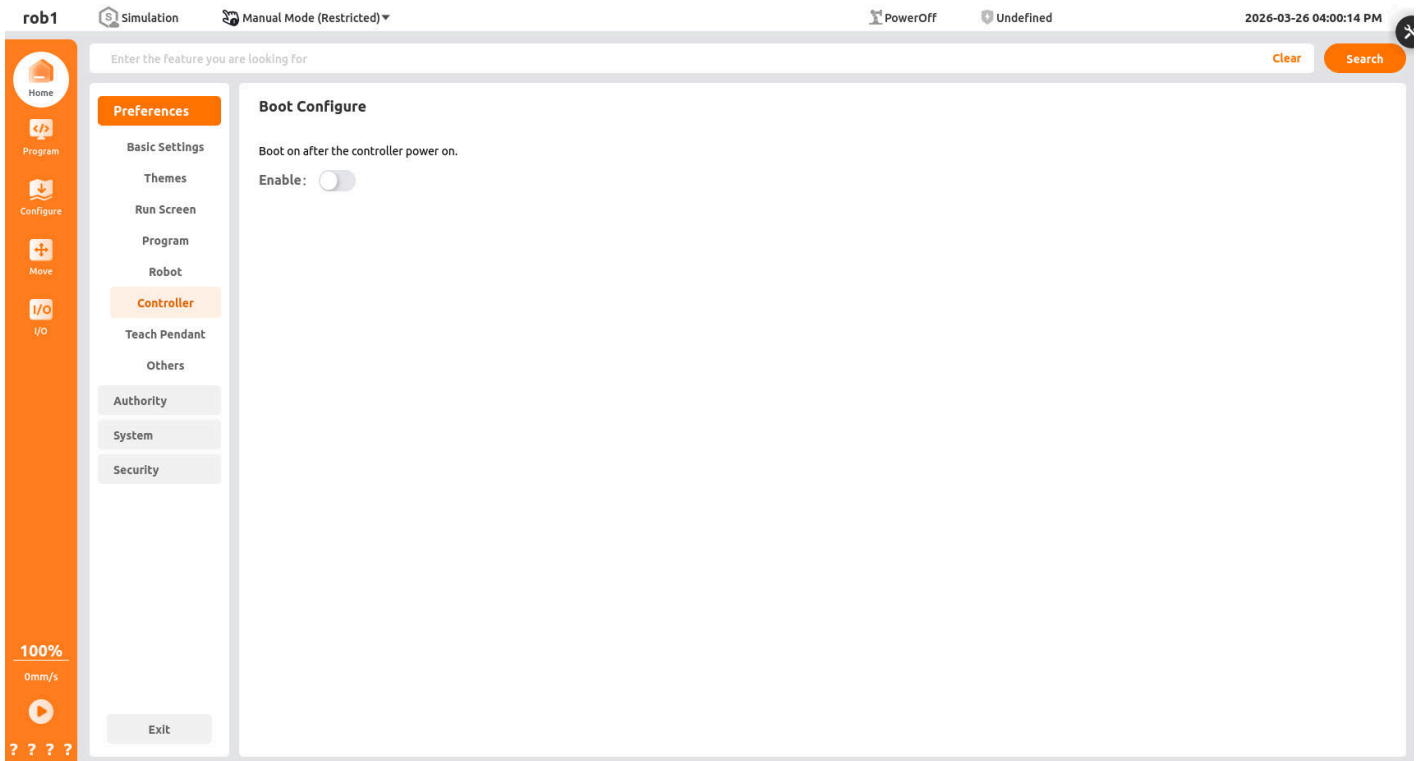
- : Drag the slider to set the robot's default speed, and click [Confirm] to apply the settings.
- After enabling "Limited HandGuide," you can adjust the limits for HandGuide on the "Nove > HandGuide" page.

Info

For the AuboStudio Android version, the function to set the default running speed is moved to the [NavBar] page.

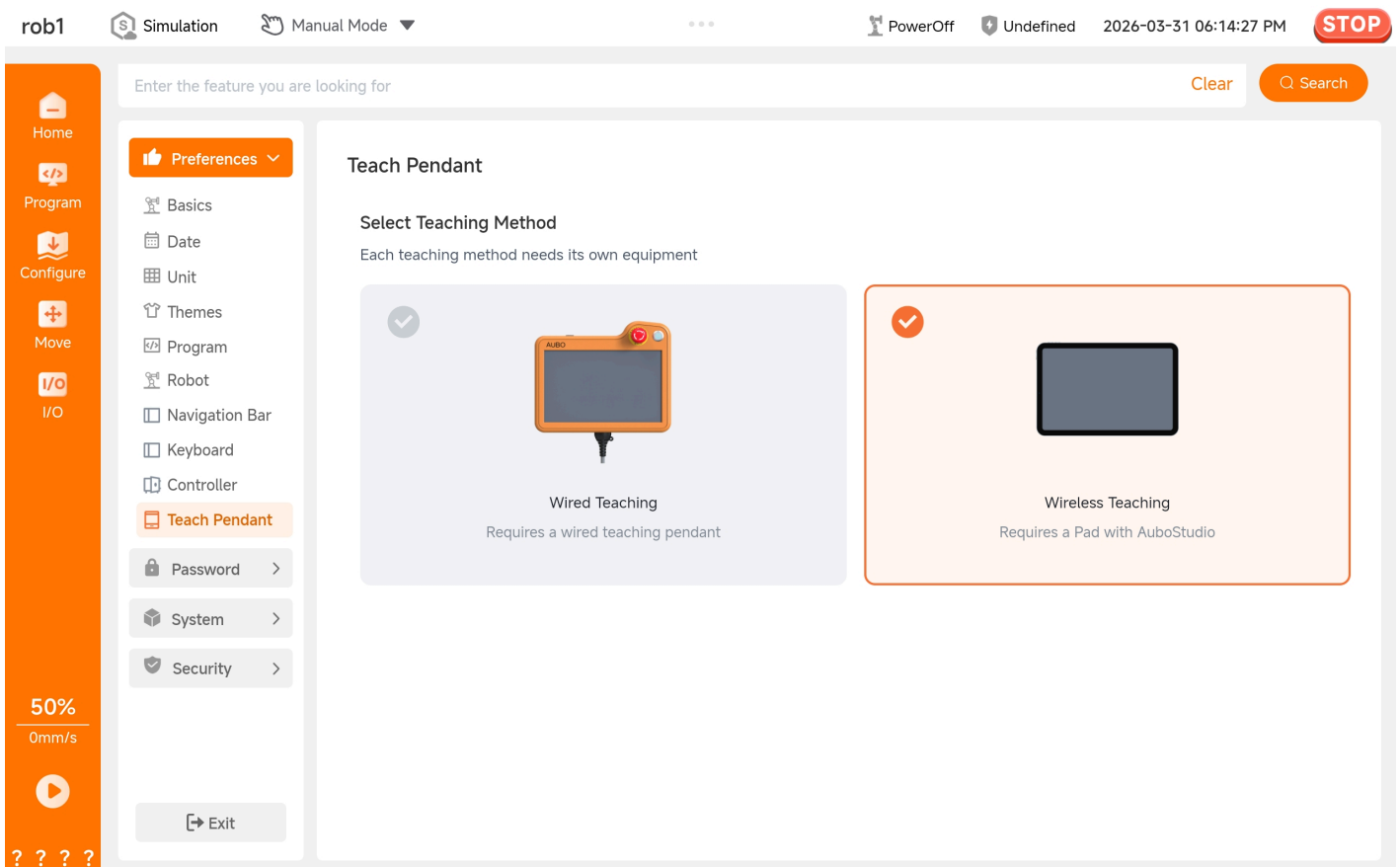
Controller teach pendant only

The [Controller] function allows users to set for automatic system startup upon power-on of the robot control box.



Teach pendant

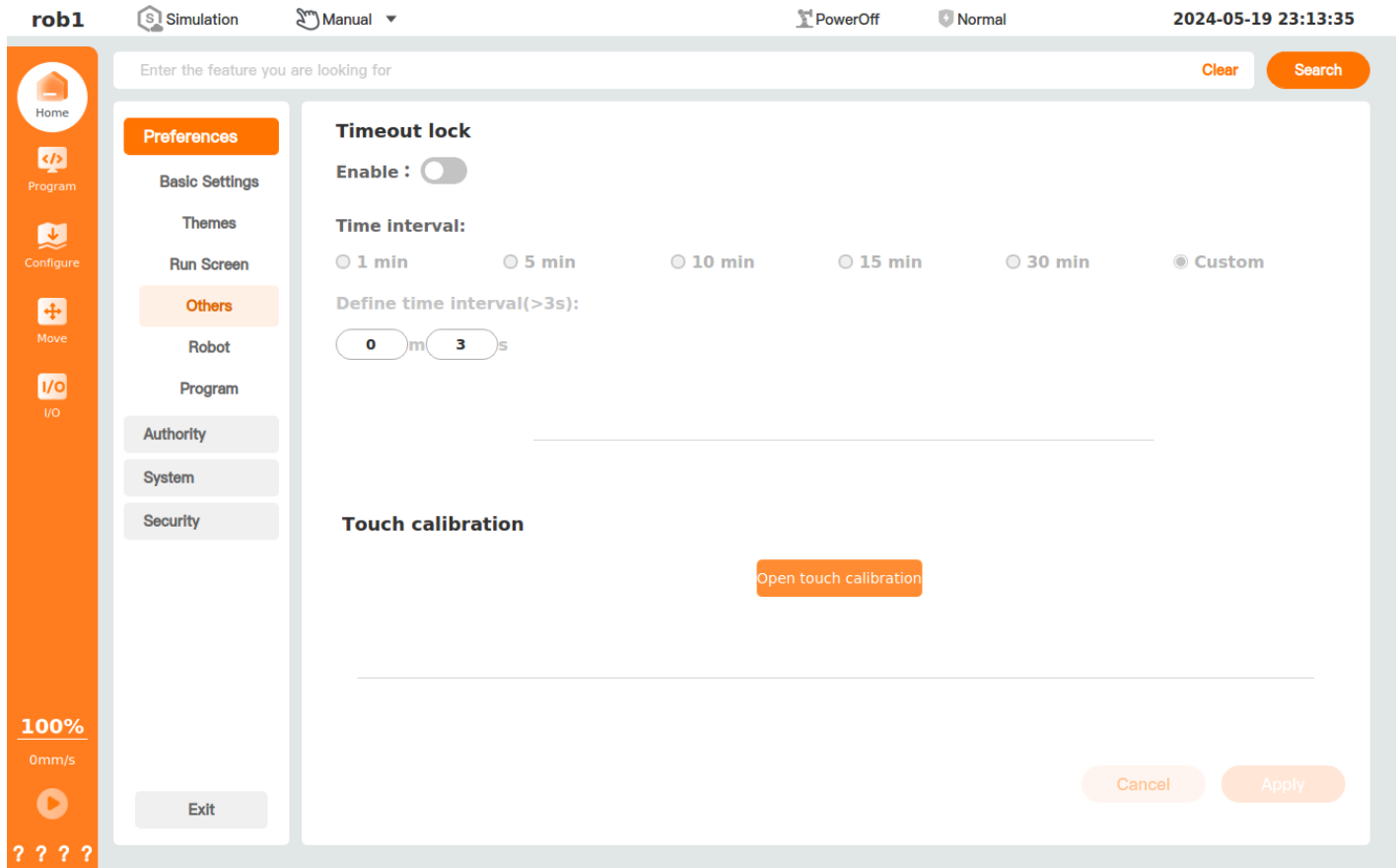
The [Teach Pendant] function allows users to set the teaching methods for the robot. Different teaching methods require different equipment.



Others

teach pendant only

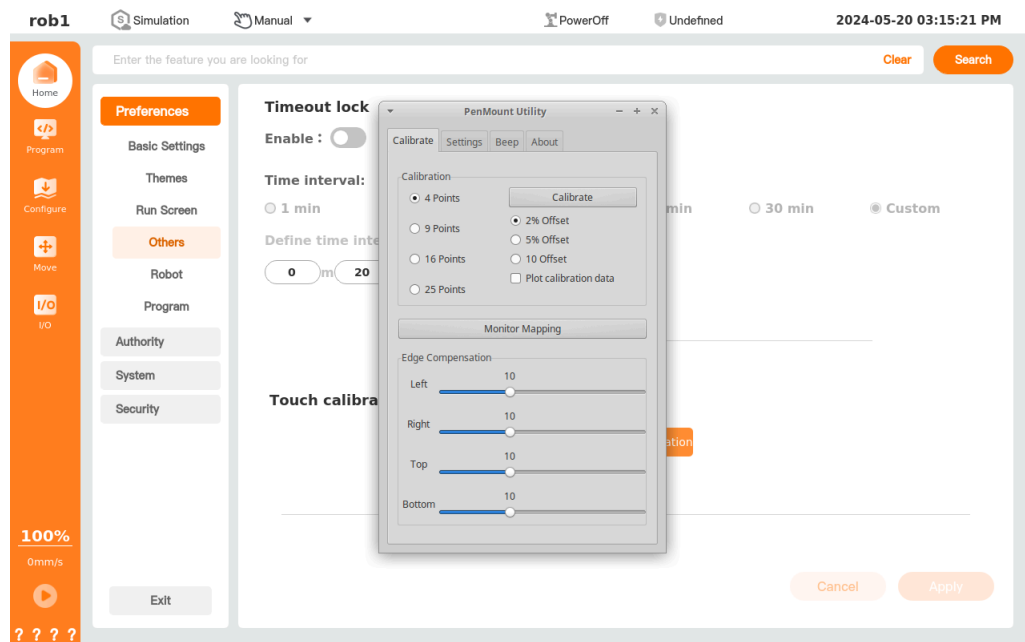
In the [Others] page, users can set the screen lock function and calibrate the touch screen.



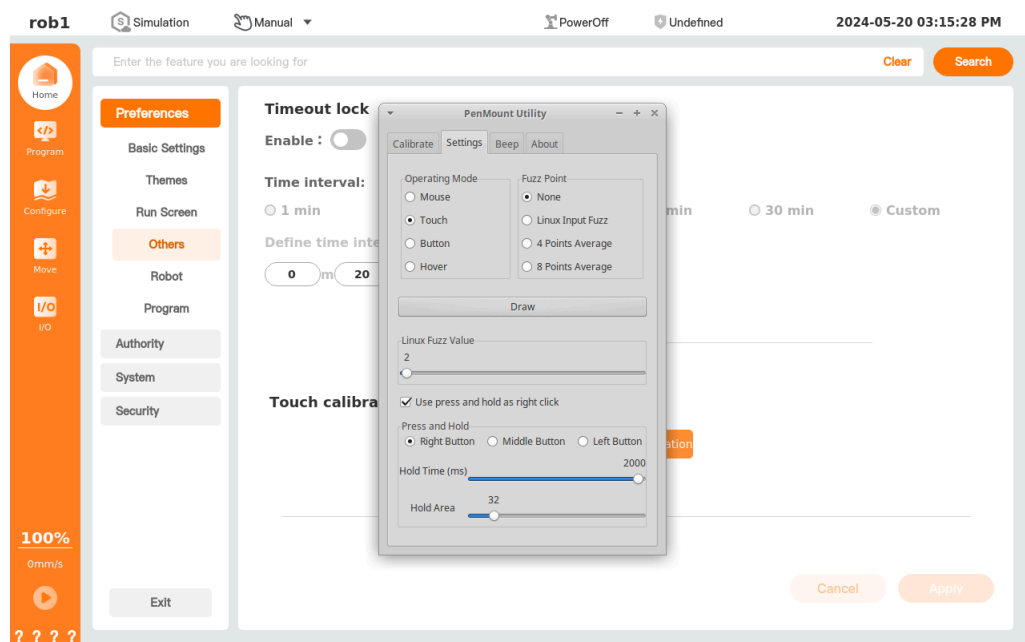
- Timeout lock:
 - Enable ☒ ☐: Enable/disable the "timeout lock" function.
 - Time interval: After turning on the "timeout lock" function, set the time interval for which the screen is not operated, and click [Apply] to save and enable the settings.
- Touch calibration: Calibrate the teach pendant screen.

Touch calibration steps

1. Click [Open touch calibration] to pop up the touch calibration software window. Click the "Settings" tab and select "Touch" in "Operation Mode."



2. Click "Calibrate" and select the number of test points in "Calibration." At present, there are four options: 4, 9, 16, and 25. The more points, the more accurate the touch calibration. It is recommended to use 4 or 9 points.



3. Click [Calibrate] to enter the calibration page, and follow the prompts to complete the touch calibration.



4. Close the touch calibration software after the touch calibration is completed.

Authority

See [Basic Operation - User Authority](#).

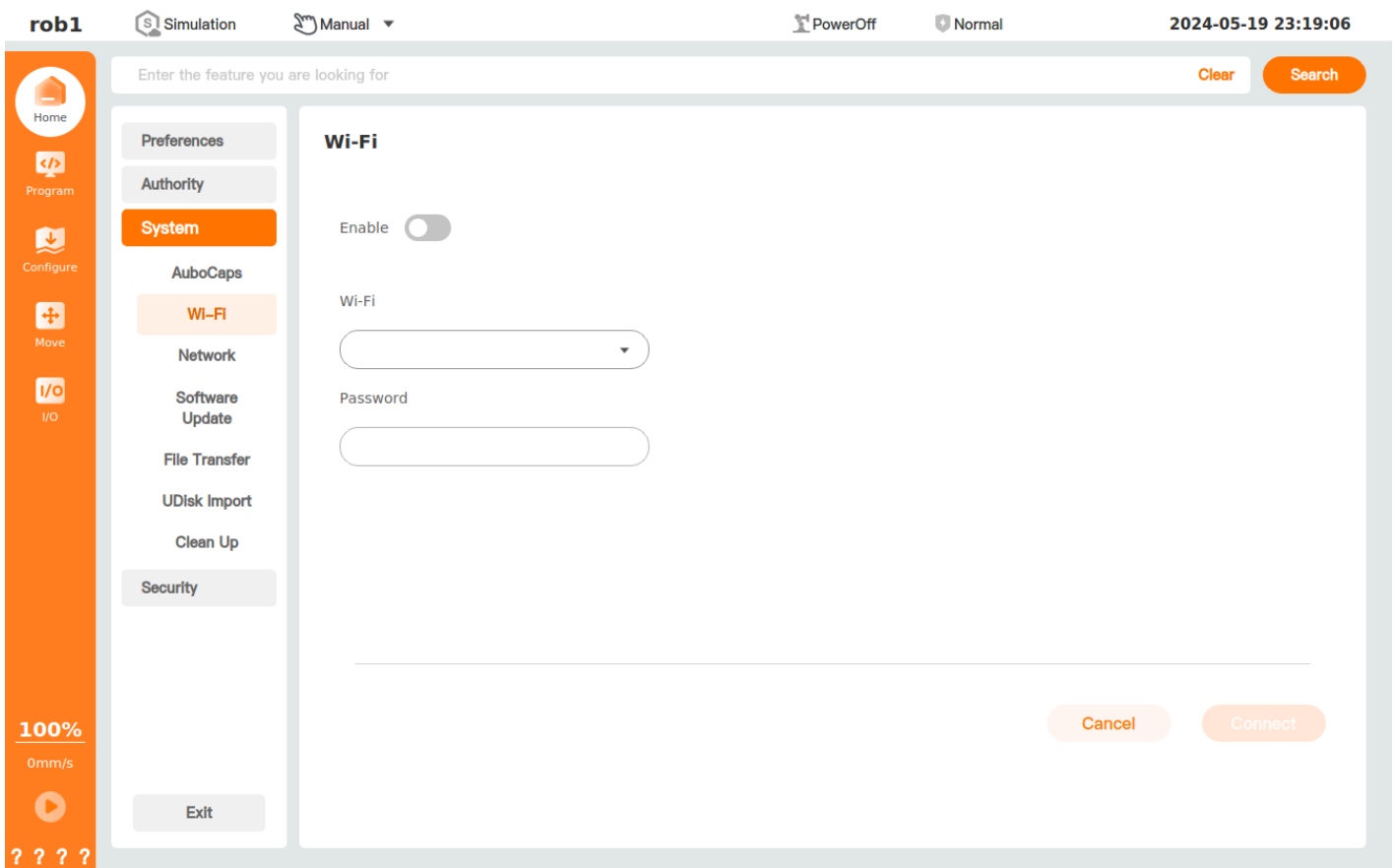
System

Wi-Fi

On the [Wi-Fi] page, users can set the name and password of the wireless network in the controller. When the controller is connected to a wired network, wireless communication can be turned on or off.

Tip

1. If you turn off the wireless hotspot, the controller must be connected to a wired network. Otherwise, you will not be able to connect the robot.
2. This function is only supported by some controller models. Please refer to the [Hardware User Manual](#) for more details.



- The default name of Wi-Fi is "(AAAA-CCCC-YYYY-)XXXX", and the default password is "12345678". Among them, AAAA is a fixed brand name; CCCC is the controller model; YYYY is the batch production date; and XXXX are the last 4 digits of the controller serial number.
- After the successful connection of the software, you can modify the Wi-Fi name and password on the page, then click [Apply] to make the settings take effect. Then the software

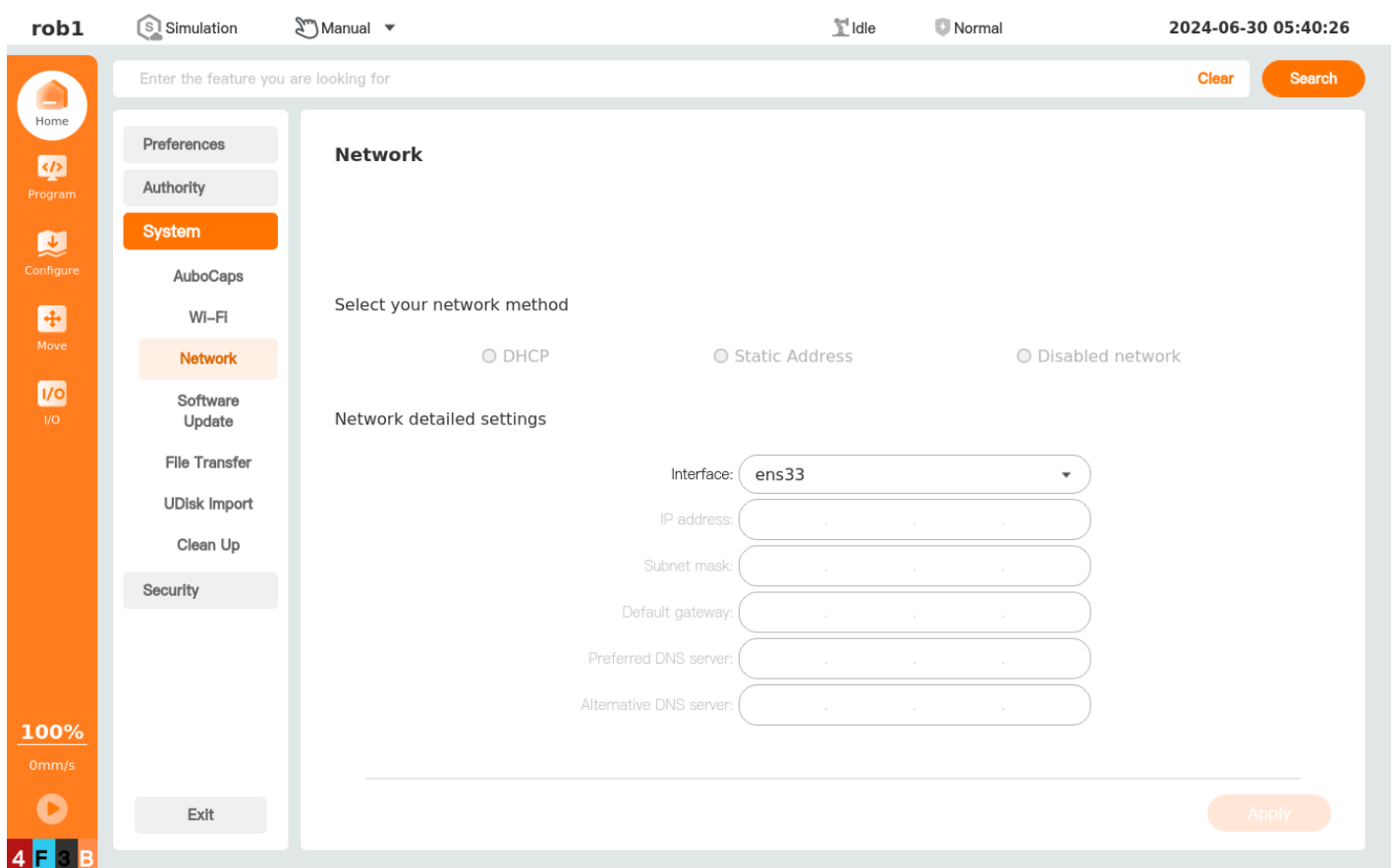
is disconnected, and just use the new Wi-Fi name and password for reconnection.

LAN interface

In the [Network] page, you can set up a network controlled by a third-party interface. After setting, the system can be connected to the external network or the LAN.

Tip

1. The network IP address of the third-party interface shall be in the same network segment as the IP address of this machine.
2. Modification to the network configuration can disconnect the network. Modify the configuration when the robot is not in use, and make sure that the program and the configuration are saved to avoid data loss.
3. Please give priority to using DHCP to connect the network. If you choose the static address, please keep the IP address in mind and make sure it is free; otherwise, you may not be able to connect to the controller.



Software update

Please refer to [Basic robot operations - Software update](#).

File transfer

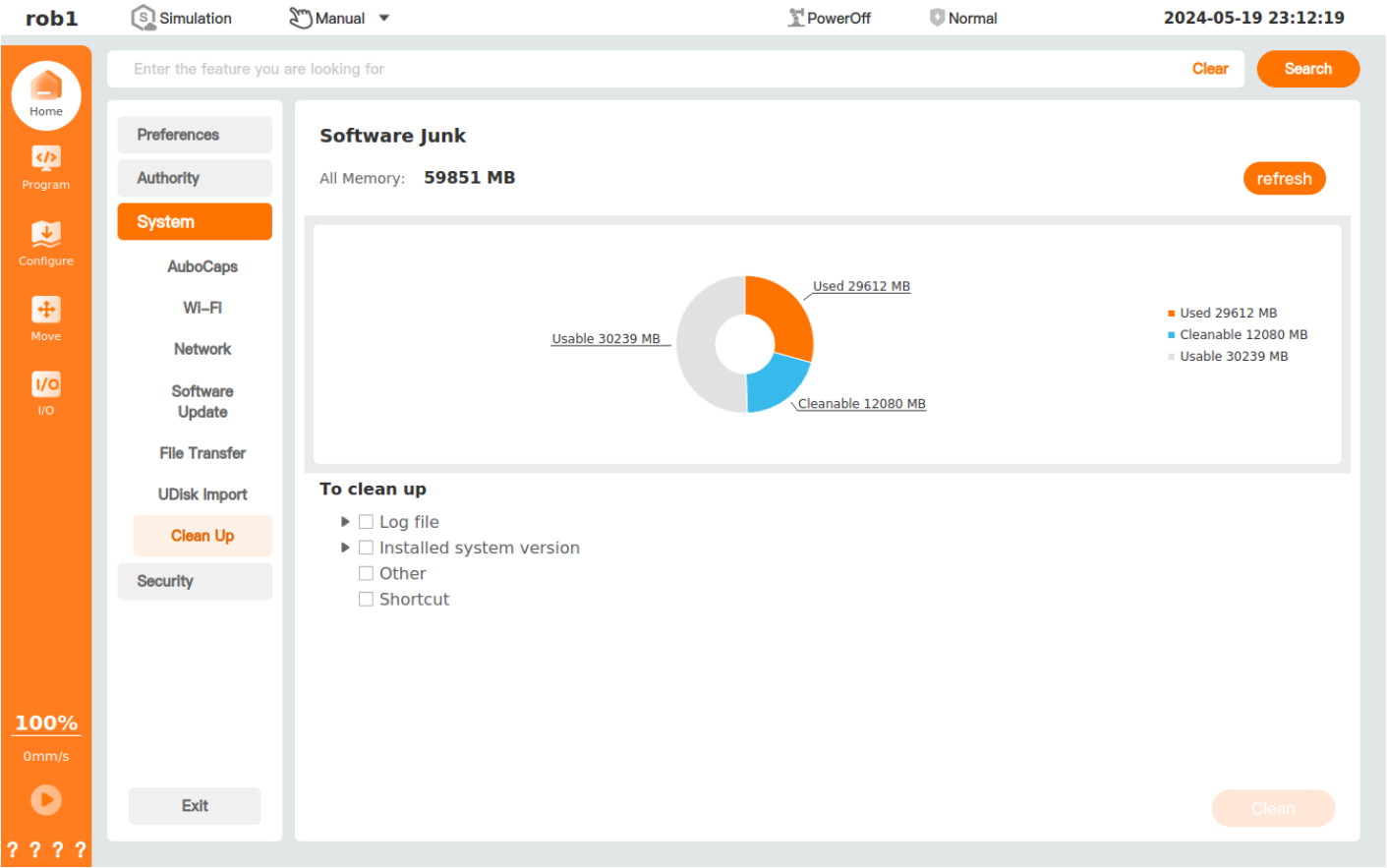
Refer to [Program and file management - File transfer](#).

UDisk import

Refer to [Program and file management - Import from USB flash drive](#).

Clean

In the [Clean Up] page, users can view the disk memory and clean up the system files.



- Refresh: Refresh the [Clean Up] page.
- Clean: Delete the checked files.

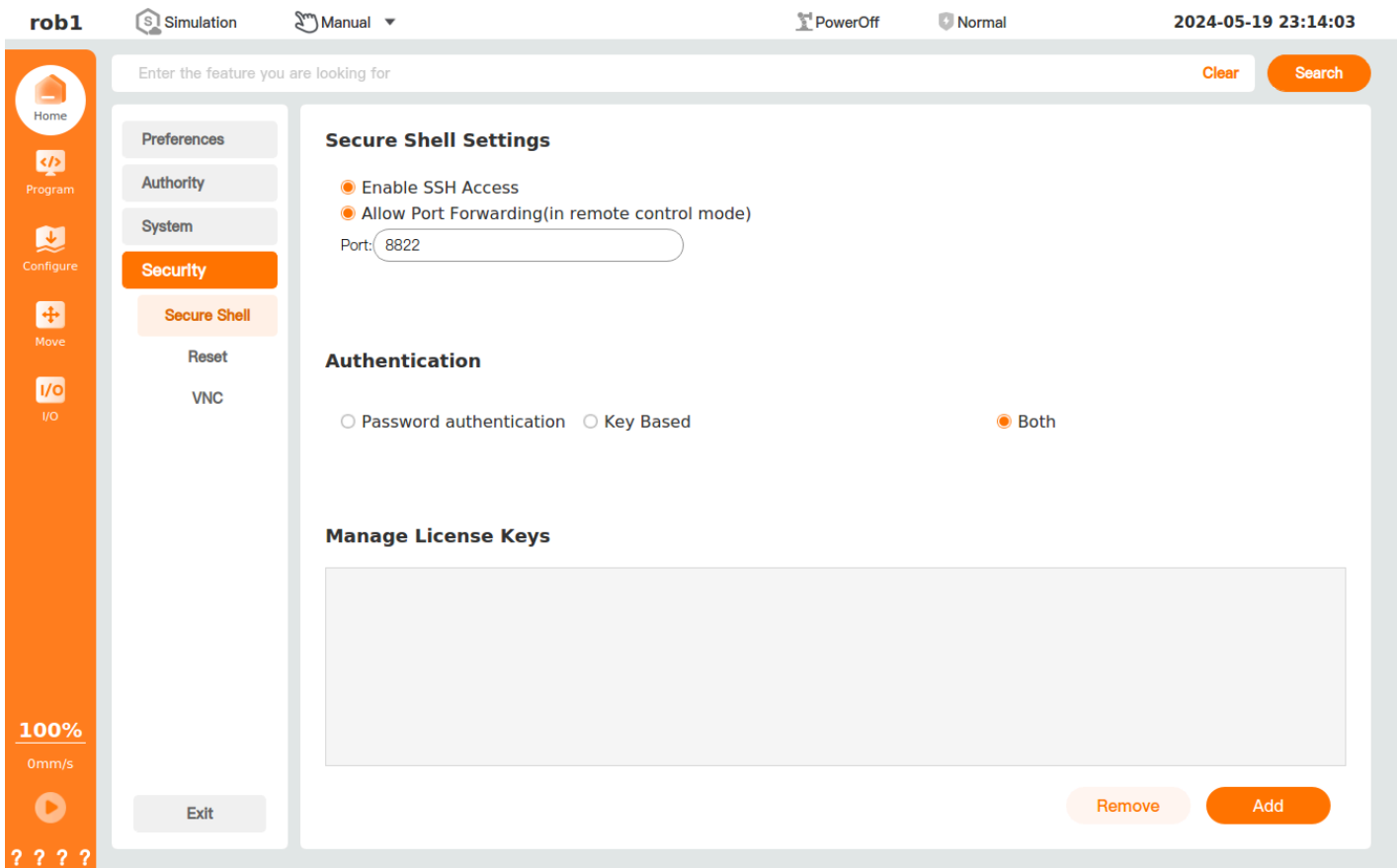
Safety

Secure Shell

In [Secure Shell] page, you can enable SSH access. After enabling it, you can remotely log in to the system as a terminal. [Secure Shell] page provides the port forwarding function, that is, the network data of the TCP port is forwarded to a certain port of the specified host, and the data is encrypted and decrypted accordingly while forwarding. . In addition, you can use key based authentication method to achieve password-free login.

Info

1. This function can only be unlocked after the admin password is set. For setting method of Admin password, please refer to [Classic mode](#).
2. Before adding a key, you need to import the key file into the folder of the system.
3. Before enabling SSH, verify that the robot is in the same LAN as the remote device (that is, ping is executed).



- Enable SSH Access: Enable/disable SSH.

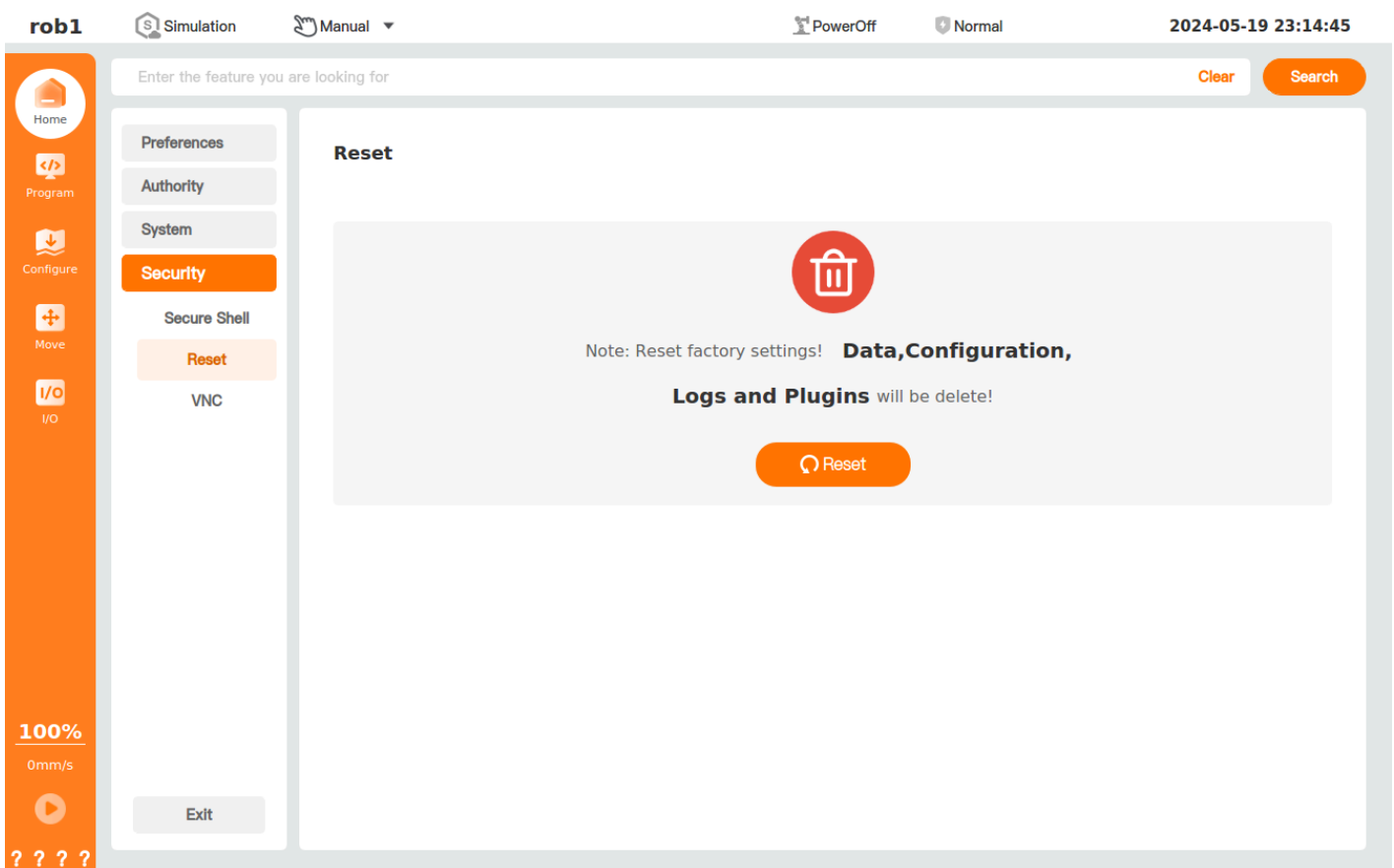
- Allow Port Forwarding (in remote control mode): When this is checked, the port number can be modified in the remote control mode.
- Port: Set the port number.
- Authentication: Select the authentication method for SSH access.
- Add: Add the key.
- Remove: Remove the key.

Reset

In [Reset] page, you can reset the factory settings of the system.

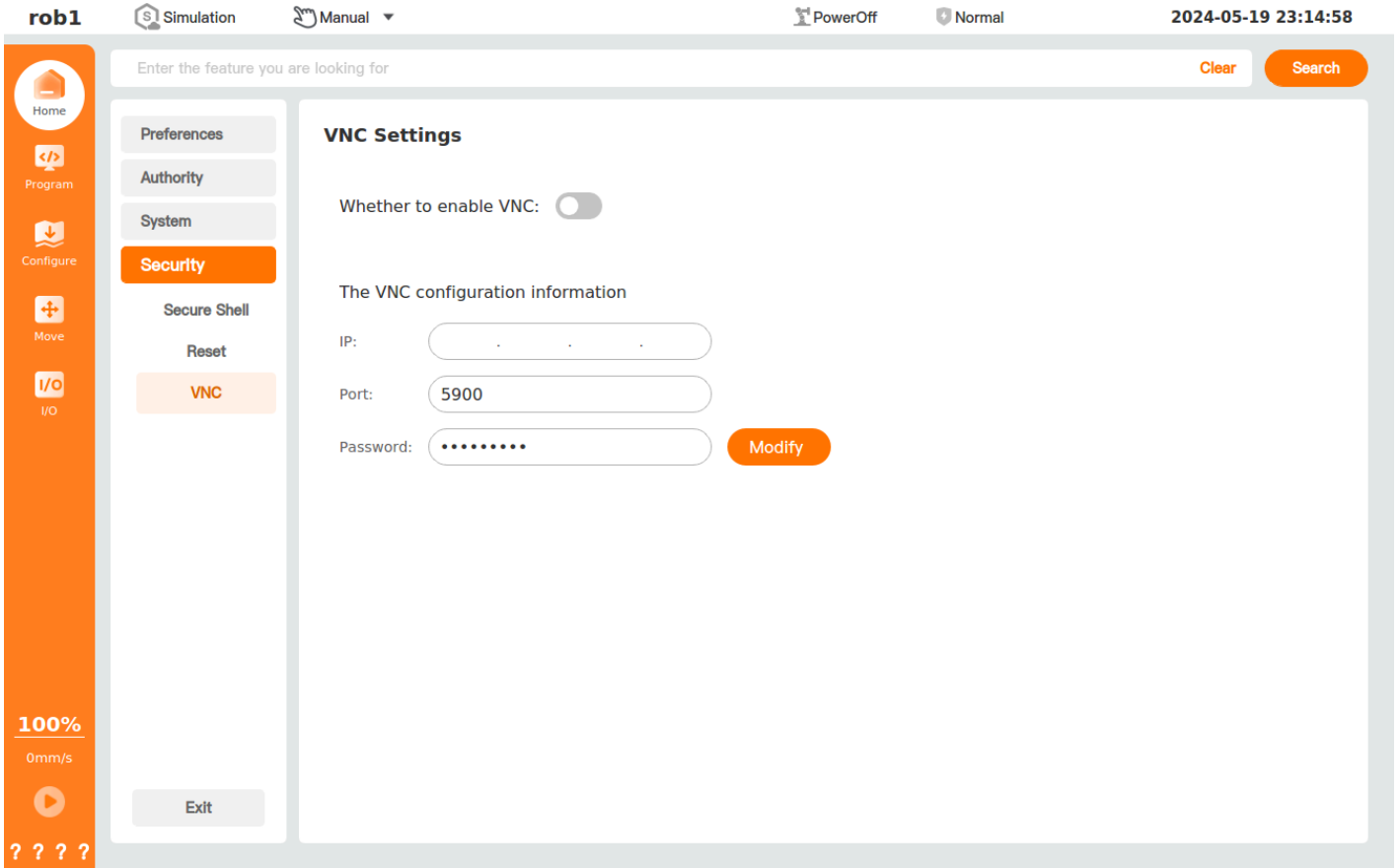
Info

You need to enter the admin password before resetting the factory settings. For setting method of Admin password, please refer to [Classic mode](#).



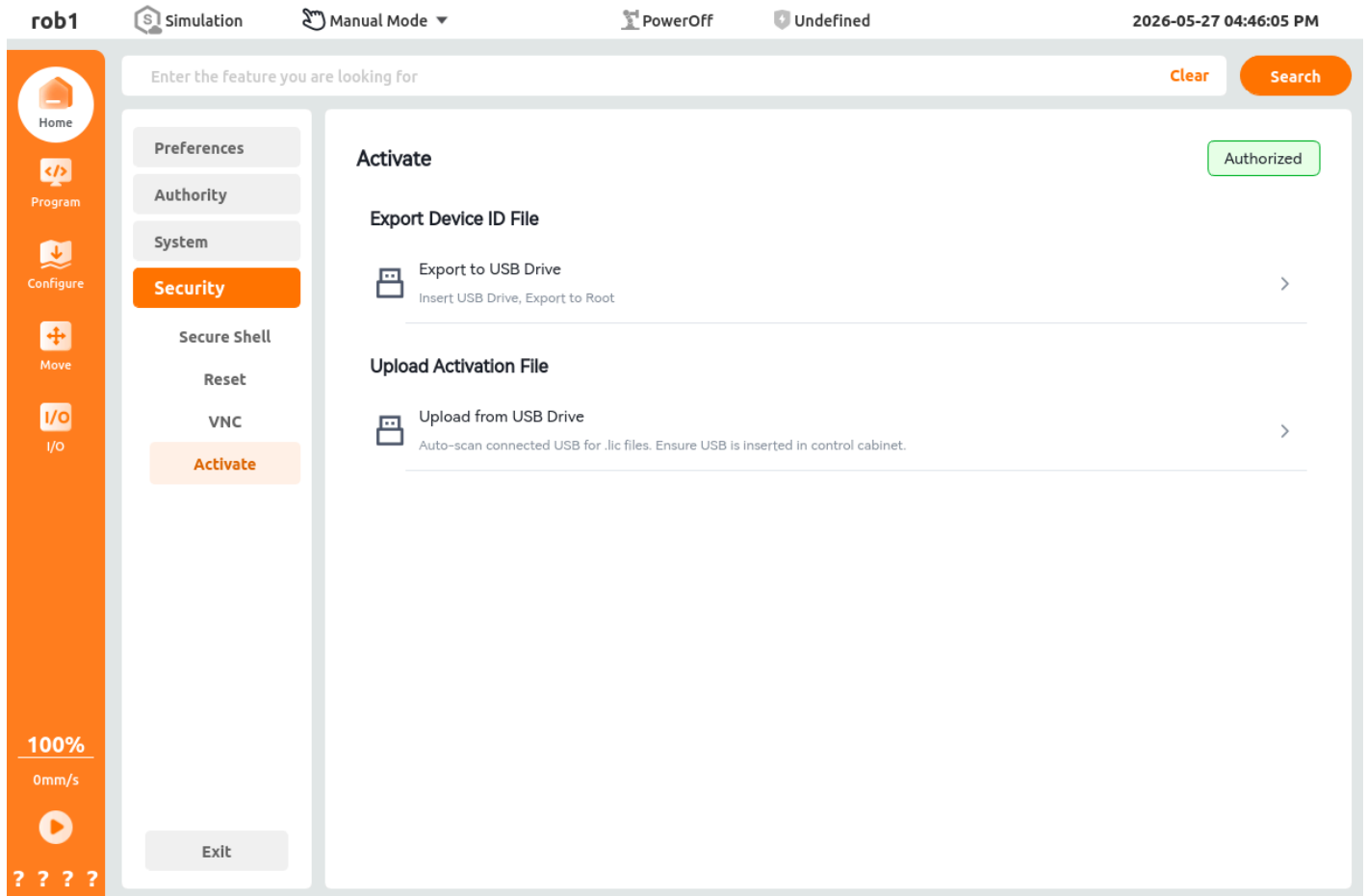
VNC

In [VNC] page, you can enable or disable VNC, and set the relevant VNC configuration information.



Activate

The [Activate] function provides ARCS firmware authorization management capabilities. Through an offline activation scheme, it ensures the legality and compliance of sample machine usage, while production machines have permanent authorization.

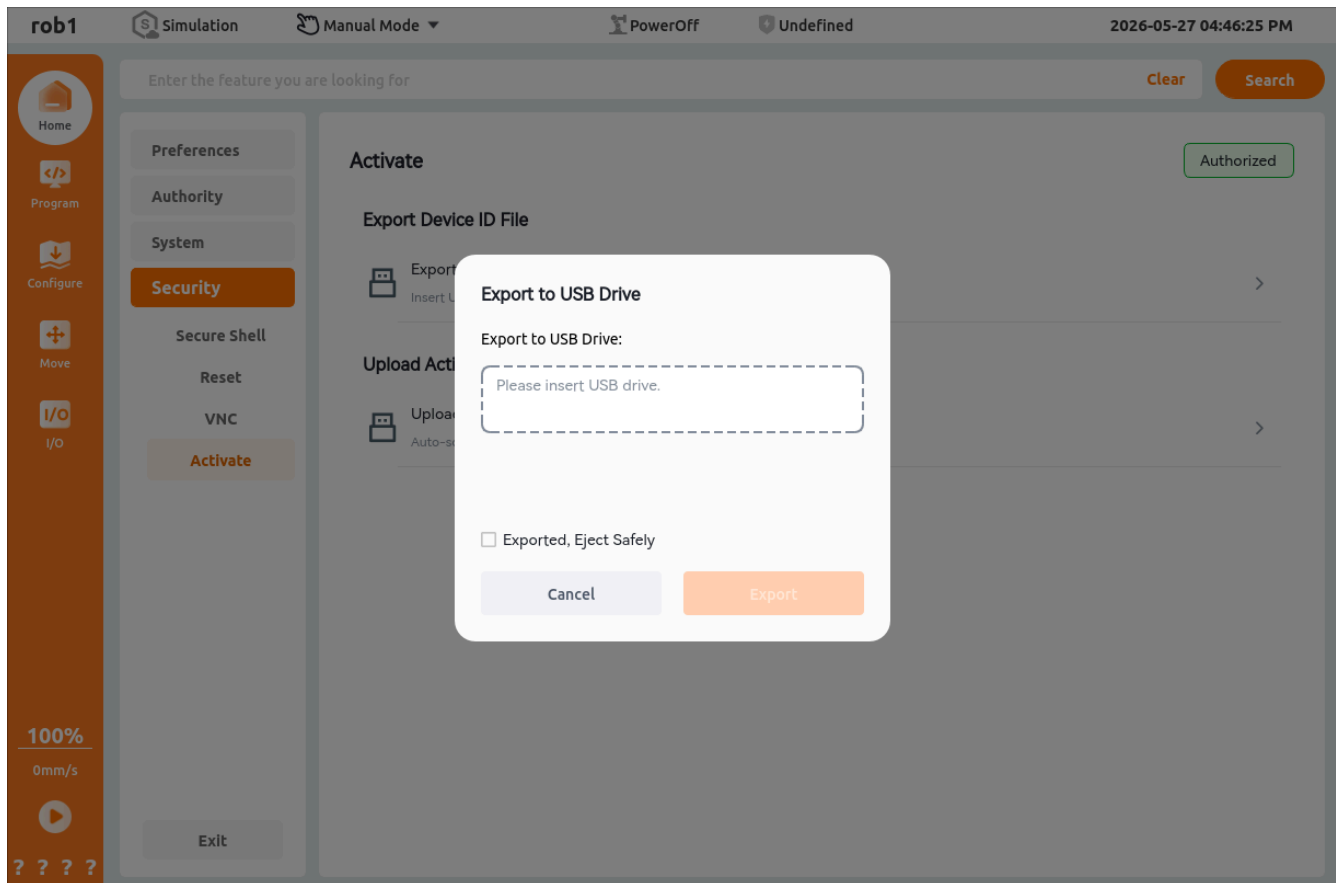


- **Export Device ID File**

Before activating a sample machine, the device ID file must be exported and sent to the AUBO technical team to generate a dedicated activation license `.lic` file.

- [Export to USB Drive]

After inserting a USB drive, click [Export to USB Drive], select the target device, and click [Export] to save the device ID file to the root directory of the USB drive.

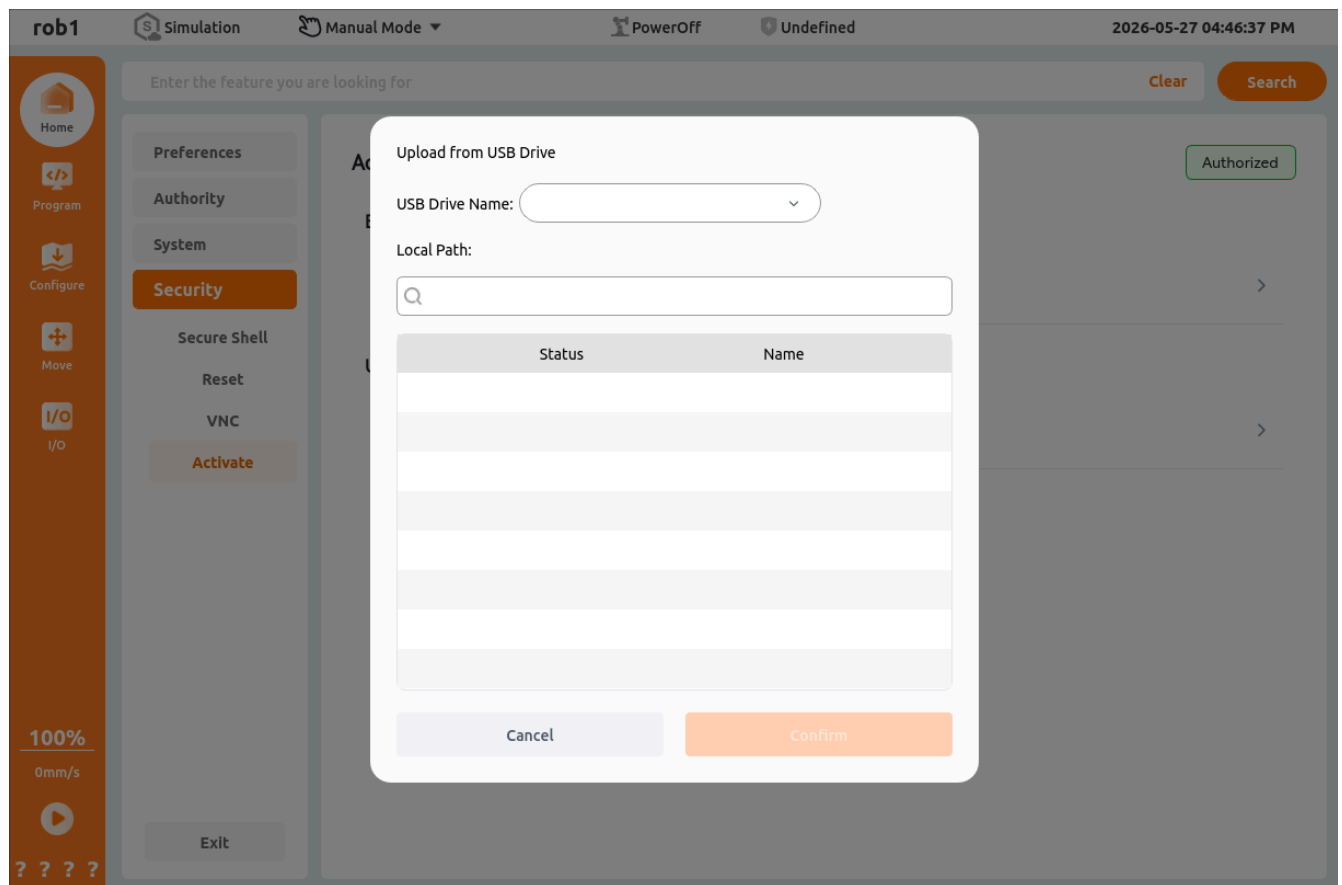


- **Upload Activation File**

After the AUBO technical team generates the dedicated activation license `.lic` file using the device ID file, the file needs to be uploaded to activate the sample machine.

- [Upload from USB Drive]

After inserting a USB drive, the system will automatically search for `.lic` license files on the connected drive (the `.lic` file must be copied to the root directory of the USB drive in advance). Select the target file and click [Confirm] to complete sample machine activation.



Warning

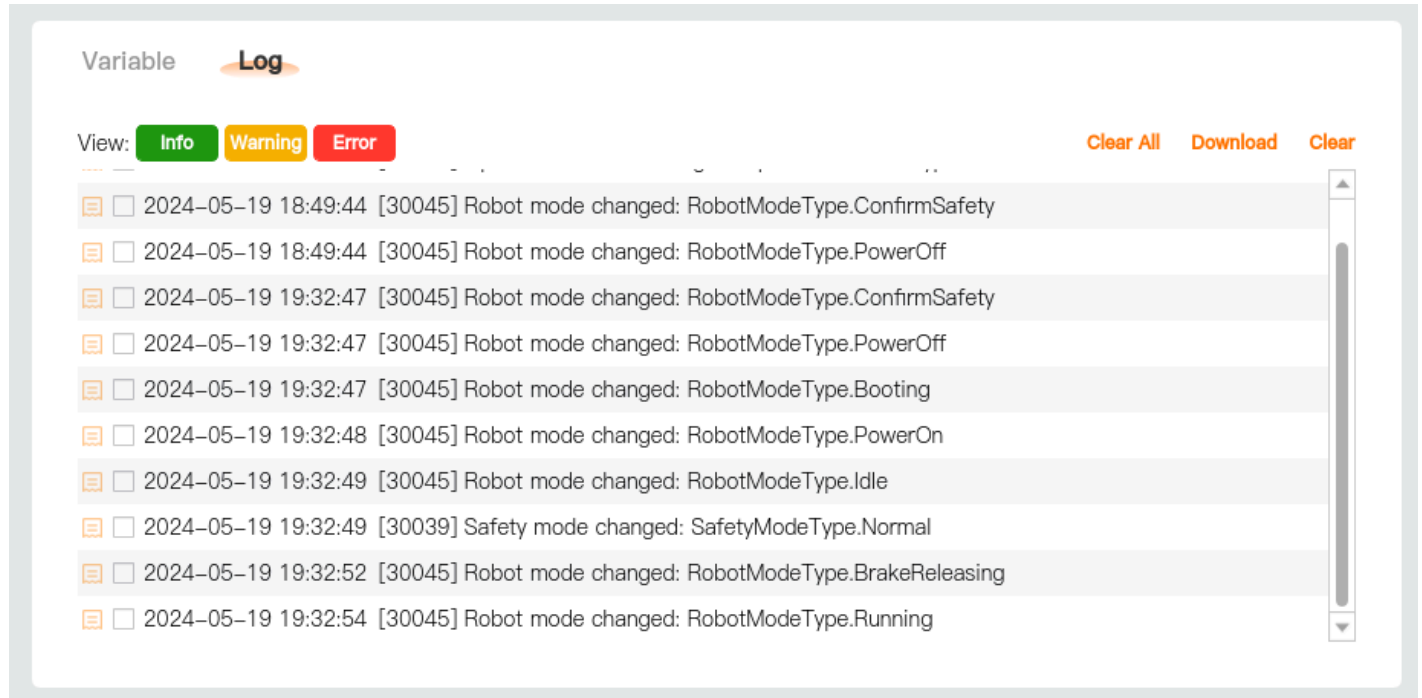
To ensure continuous normal operation of the device, please contact AUBO technical personnel for activation before the trial authorization expires. Authorization expiration will result in the following:

1. **Authorization period ≤ 7 days:** The system will display pop-ups and watermarks indicating that authorization is about to expire. At this stage, teach pendant functionality remains unaffected.
2. **After authorization expires:** The system will automatically disable some key functions of the sample machine, and will continue to display authorization status through screen pop-ups and watermarks.

View Log and Runtime Records

Controller system log

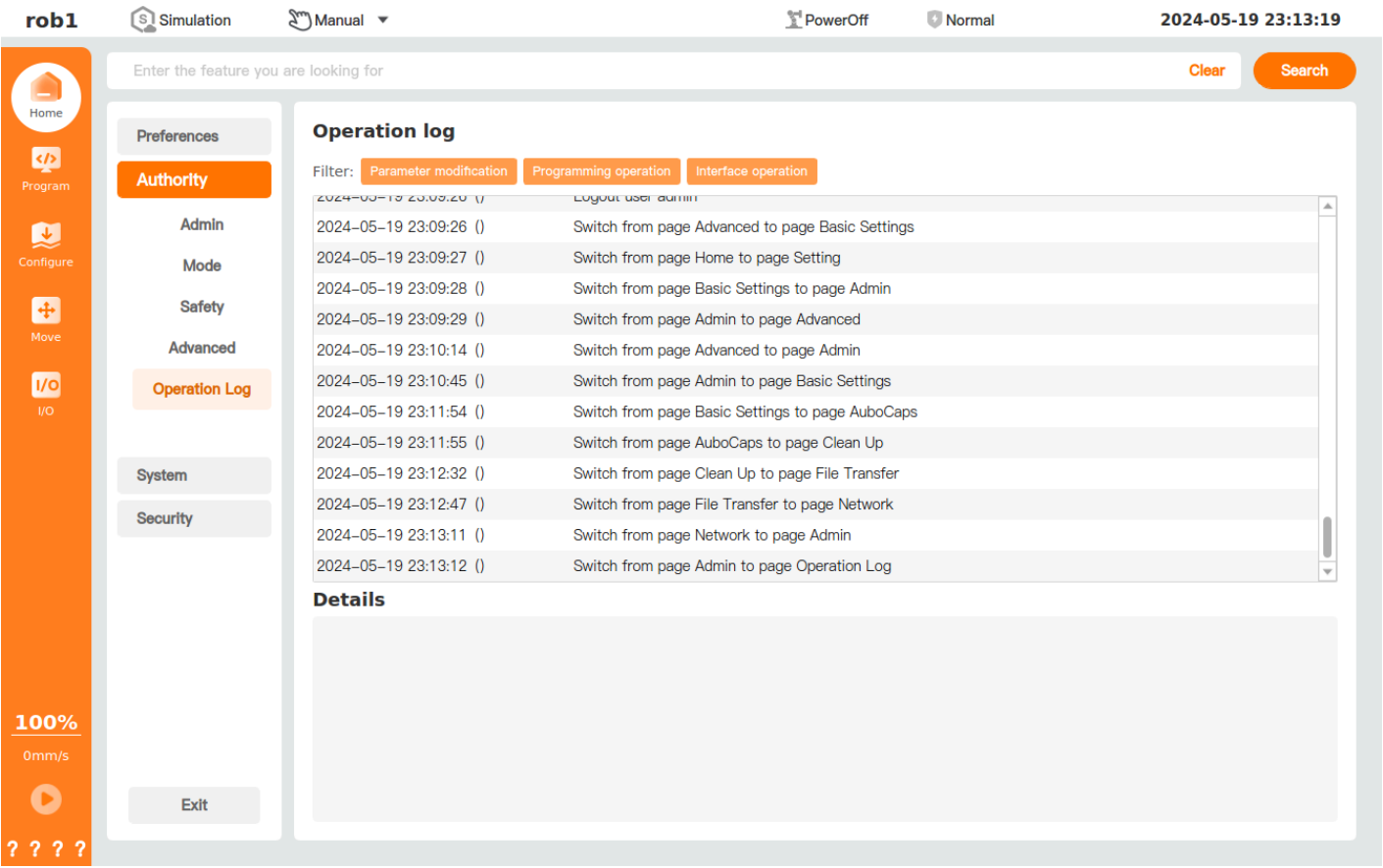
Users can view the controller system log in the "Home > Log" tab.



- View: Logs are classified based on severity into Info, Warning, and Error.
 - Info: Provide general information such as program status, controller changes, and controller version.
 - Warning: Problems that may have occurred but are recoverable by the system.
 - Error: Errors that are unrecoverable by the system and will cause a fault and safe stop of the robot.
- Clear All: Cleaning up the [Log] tab.
- Download: Export log files. For details, please refer to [Log export - Direct log export](#).
- Clear: Clear the checked logs. The cleared logs will not be displayed in the interface any more. For instructions on deleting log source files, please refer to [System settings - System - Clean Up](#).

Teach pendant operation log

Users can view all their operation records in teach pendant software in "Settings > Permissions > Operation log."



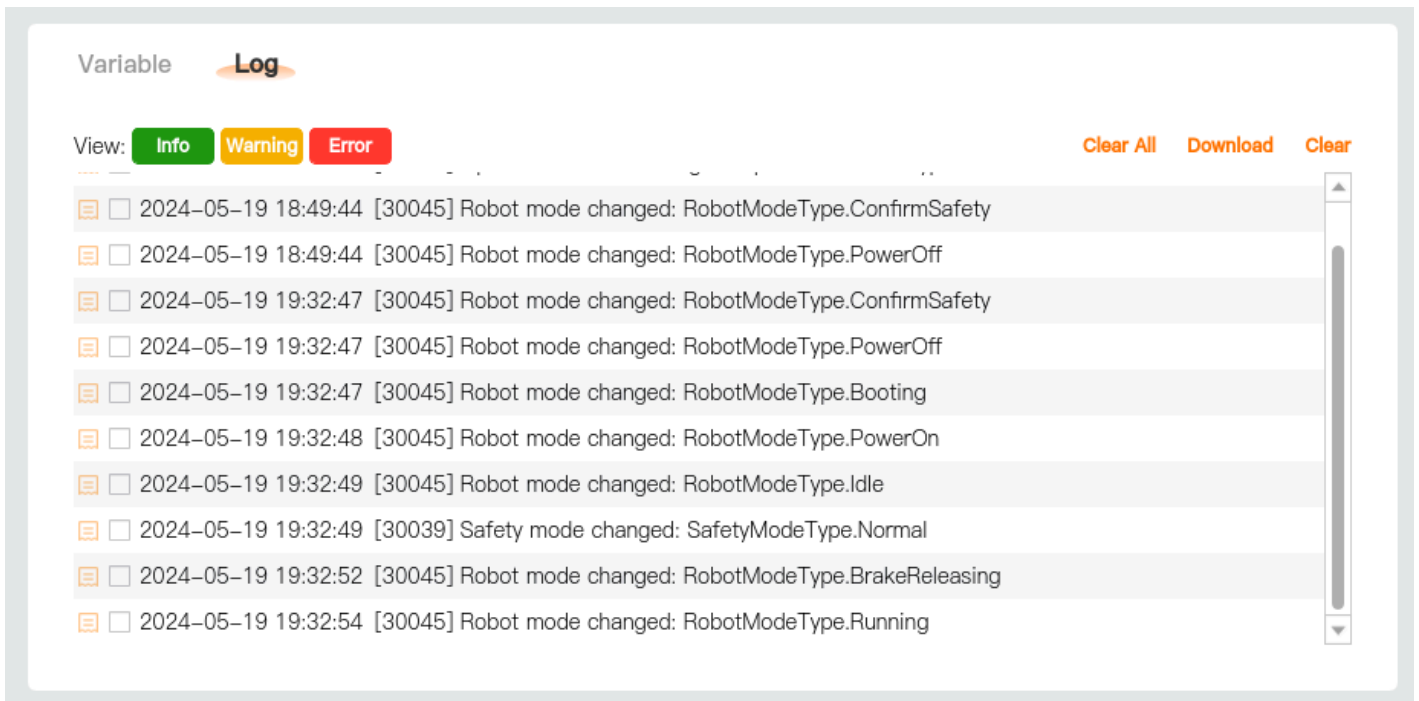
Runtime records

Users can view the program's runtime records in "Programming > Program Information Card > Trace." For details, please refer to [Program Information Card - Trace](#).

Log Export

Directly export system log

Users can click the [Download] button in the "Home > Log" tab to export the log.



- Local: Save to local.
- UDisk: Save to an external USB flash drive. For this operation, it is required to insert a USB flash drive into the controller.
- LANDrop: Transfer files through third-party software. For this operation, it is required to establish a connection with the third-party software. For details, please refer to [Program and File Management - File Transfer](#).

Tip

1. Please download the log after the program stops; otherwise, CPU overload may occur.
2. If it is unable to select the storage path or click [Select] for downloading after [UDisk] is selected, please try to insert the U disk again.
3. Keep the app running in the foreground when exporting the log through the client. Do not switch to background or lock screen or enable power-saving mode, as this may cause export failure.

Export log via log center

Please refer to [Log Center - Download Event Report](#).

Log Center

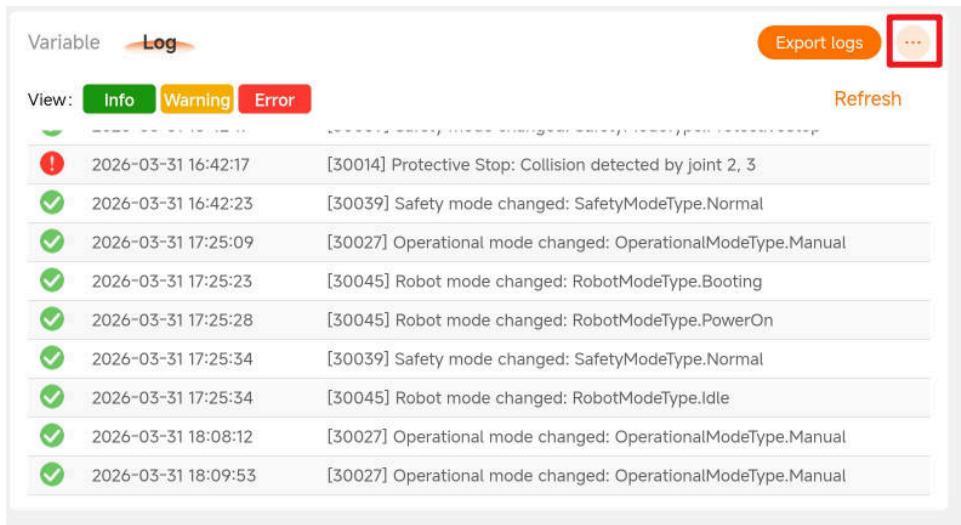
In the ARCS system, the ARCS system logs are stored within the controller system, while the Android client logs are stored on the tablet. To facilitate users' viewing, analysis, and export of logs, AUBO has launched the "Log Center" function.

In the log center, a collection of log files for a certain period of time is compressed into a compressed package, which is called an **event report**.

Tip

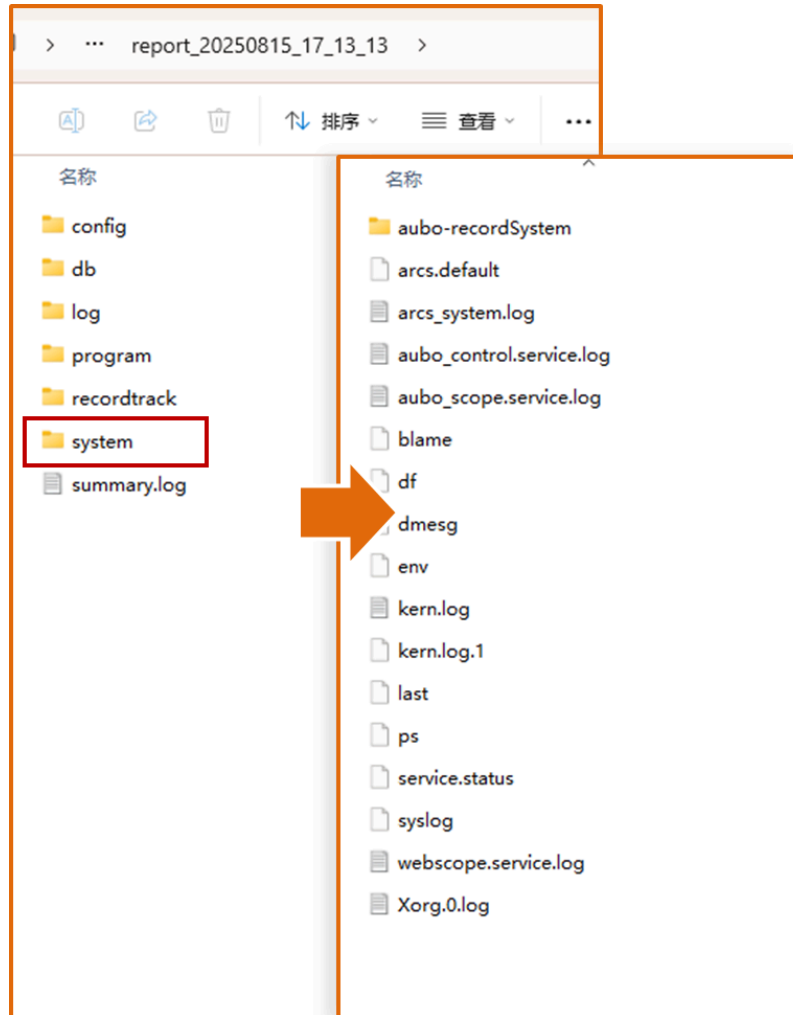
1. Supported versions:
 1. Wired teach pendant: ARCS 0.31.1 version and above
 2. Android client: ARCS 0.31.0, App version 0.9.3 and above
2. Log center is a function module of the OPS system. Please ensure that the OPS system is installed before using it.
3. For OPS system installation, please contact the relevant technical personnel.

Entry to log center



Contents of compressed package of event report

The contents of the compressed package of the event report contain a collection of Linux system logs, in addition to the existing compressed package of logs.



Operation guide

Generate event report

There are two ways to generate an event report. When the system detects certain anomalies, it will automatically generate an event report, such as collisions, triggered protective stops, software crashes, etc. At this point, users only need to wait for the event report to finish generating and then download and use it. Event reports can also be generated manually.

rob1

Simulation

Manual Mode

PowerOff

Undefined

2026-03-31 06:16:05 PM

STOP

Home

Program

Configure

Move

I/O

I/O

50%

0mm/s

???

Log Center

Log

Event Report

Issues

Log Tool

Export Log

Settings

Generate Report

Delete All

Event Time	State	Event
2026-03-31 18:15:34	Normal	User Report: Other
2026-03-31 16:43:50	Normal	ProtectiveStop: Safety mode changed, SafetyModeType.ProtectiveStop
2026-03-31 16:42:17	Normal	ProtectiveStop: Safety mode changed, SafetyModeType.ProtectiveStop
2026-03-31 16:41:01	Normal	ProtectiveStop: Safety mode changed, SafetyModeType.ProtectiveStop
2026-03-31 16:40:21	Normal	ProtectiveStop: Safety mode changed, SafetyModeType.ProtectiveStop
2026-03-31 16:39:48	Normal	ProtectiveStop: Safety mode changed, SafetyModeType.ProtectiveStop
2026-03-31 11:09:41	Normal	ProtectiveStop: Safety mode changed, SafetyModeType.ProtectiveStop
2026-03-31 11:07:39	Normal	ProtectiveStop: Safety mode changed, SafetyModeType.ProtectiveStop
2026-03-31 11:06:42	Normal	ProtectiveStop: Safety mode changed, SafetyModeType.ProtectiveStop
2026-03-31 11:06:12	Normal	ProtectiveStop: Safety mode changed, SafetyModeType.ProtectiveStop
2026-03-31 11:05:01	Normal	ProtectiveStop: Safety mode changed, SafetyModeType.ProtectiveStop
2026-03-31 11:03:11	Normal	ProtectiveStop: Safety mode changed, SafetyModeType.ProtectiveStop
2026-03-31 11:02:38	Normal	ProtectiveStop: Safety mode changed, SafetyModeType.ProtectiveStop
2026-03-31 11:02:23	Normal	ProtectiveStop: Safety mode changed, SafetyModeType.ProtectiveStop
2026-03-31 11:01:57	Normal	ProtectiveStop: Safety mode changed, SafetyModeType.ProtectiveStop
2026-03-31 11:01:19	Normal	ProtectiveStop: Safety mode changed, SafetyModeType.ProtectiveStop

rob1

Simulation

Manual Mode

PowerOff

Undefined

2026-03-31 06:15:02 PM

STOP

Home

Program

Configure

Move

I/O

I/O

50%

0mm/s

???

Log Center

Log

Event Report

Issues

Log Tool

Export Log

Settings

Generate Report

Delete All

Event Time	State	Event
2026-03-31 16:43:50	Normal	ProtectiveStop: Safety mode changed, SafetyModeType.ProtectiveStop
2026-03-31 16:42:17	Normal	ProtectiveStop: Safety mode changed, SafetyModeType.ProtectiveStop
2026-03-31 16:41:01	Normal	ProtectiveStop: Safety mode changed, SafetyModeType.ProtectiveStop
2026-03-31 16:40:21	Normal	ProtectiveStop: Safety mode changed, SafetyModeType.ProtectiveStop
2026-03-31 16:39:48	Normal	ProtectiveStop: Safety mode changed, SafetyModeType.ProtectiveStop
2026-03-31 11:09:41	Normal	ProtectiveStop: Safety mode changed, SafetyModeType.ProtectiveStop
2026-03-31 11:07:39	Normal	ProtectiveStop: Safety mode changed, SafetyModeType.ProtectiveStop
2026-03-31 11:06:42	Normal	ProtectiveStop: Safety mode changed, SafetyModeType.ProtectiveStop
2026-03-31 11:06:12	Normal	ProtectiveStop: Safety mode changed, SafetyModeType.ProtectiveStop
2026-03-31 11:05:01	Normal	ProtectiveStop: Safety mode changed, SafetyModeType.ProtectiveStop
2026-03-31 11:03:11	Normal	ProtectiveStop: Safety mode changed, SafetyModeType.ProtectiveStop
2026-03-31 11:02:38	Normal	ProtectiveStop: Safety mode changed, SafetyModeType.ProtectiveStop
2026-03-31 11:02:23	Normal	ProtectiveStop: Safety mode changed, SafetyModeType.ProtectiveStop
2026-03-31 11:01:57	Normal	ProtectiveStop: Safety mode changed, SafetyModeType.ProtectiveStop
2026-03-31 11:01:19	Normal	ProtectiveStop: Safety mode changed, SafetyModeType.ProtectiveStop

Generate Report

Event Type:

Software crash

Program error

Wrong collision detection

User experience

Other

Period:

5 minutes

1 hour

24 hours

3 days

All

Description:

Please enter text remarks.....

①

Tap [Generate] to generate an event report containing operational data and logs for the problem period.

☐ Auto upload event reports

Generate

Cancel

All event reports will be displayed in the list.

rob1

Simulation

Manual Mode

...

PowerOff

Undefined

2026-03-31 06:16:05 PM

STOP

Home

Program

Configure

Move

I/O

I/O

50%

0mm/s

???

Log Center

Log

Event Report

Issues

Log Tool

Export Log

Settings

Generate Report

Delete All

Event Time	State	Event
2026-03-31 18:15:34	Normal	User Report: Other
2026-03-31 16:43:50	Normal	ProtectiveStop: Safety mode changed, SafetyModeType.ProtectiveStop
2026-03-31 16:42:17	Normal	ProtectiveStop: Safety mode changed, SafetyModeType.ProtectiveStop
2026-03-31 16:41:01	Normal	ProtectiveStop: Safety mode changed, SafetyModeType.ProtectiveStop
2026-03-31 16:40:21	Normal	ProtectiveStop: Safety mode changed, SafetyModeType.ProtectiveStop
2026-03-31 16:39:48	Normal	ProtectiveStop: Safety mode changed, SafetyModeType.ProtectiveStop
2026-03-31 11:09:41	Normal	ProtectiveStop: Safety mode changed, SafetyModeType.ProtectiveStop
2026-03-31 11:07:39	Normal	ProtectiveStop: Safety mode changed, SafetyModeType.ProtectiveStop
2026-03-31 11:06:42	Normal	ProtectiveStop: Safety mode changed, SafetyModeType.ProtectiveStop
2026-03-31 11:06:12	Normal	ProtectiveStop: Safety mode changed, SafetyModeType.ProtectiveStop
2026-03-31 11:05:01	Normal	ProtectiveStop: Safety mode changed, SafetyModeType.ProtectiveStop
2026-03-31 11:03:11	Normal	ProtectiveStop: Safety mode changed, SafetyModeType.ProtectiveStop
2026-03-31 11:02:38	Normal	ProtectiveStop: Safety mode changed, SafetyModeType.ProtectiveStop
2026-03-31 11:02:23	Normal	ProtectiveStop: Safety mode changed, SafetyModeType.ProtectiveStop
2026-03-31 11:01:57	Normal	ProtectiveStop: Safety mode changed, SafetyModeType.ProtectiveStop
2026-03-31 11:01:19	Normal	ProtectiveStop: Safety mode changed, SafetyModeType.ProtectiveStop

View event report

Click on a message in the event report list to display details of the event report.

208 / 232

Report Information



Report ID

report_20260331_18_15_34

State

Normal

Event

User Report: (Other)

Diagnostic Information

Diagnostic Code: 140005

Event Time: 2026-03-31 18:15:34.643

Teachpendant: Wireless

Description:

Please enter text remarks.....

Save

Upload Server

Download

Save to the Download/00RLogs directory on the Pad

Download event report

Log center offers multiple download methods, including batch download, individual download, download to USB flash drive, and download to tablet.

rob1

Simulation

Manual Mode

...

PowerOff

Undefined

2026-03-31 06:16:29 PM

STOP

Home

Program

Configure

Move

I/O

I/O

50%

0mm/s

?? ??

Log Center

Log

Event Report

Issues

Log Tool

Export Log

Settings

Robot Log

Export to USB Drive

Insert USB drive and export to the root directory of the USB drive

Download to Pad

Save to the Download/00RLogs directory on the Pad

Tap Download Log (Old) to use the old version

rob1

Simulation

Manual Mode

...

PowerOff

Undefined

2026-03-31 06:16:51 PM

STOP

Home

Program

Configure

Move

I/O

I/O

50%

0mm/s

?? ??

Log Center

Log

Event Report

Issues

Log Tool

Export Log

Settings

Robot Log

Export to USB Drive

Insert USB drive and export to the root directory of the USB drive

Download to Pad

Save to the Download/00RLogs directory on the Pad

Tap Download Log (Old) to use the old version

Download Log

Report Filter:

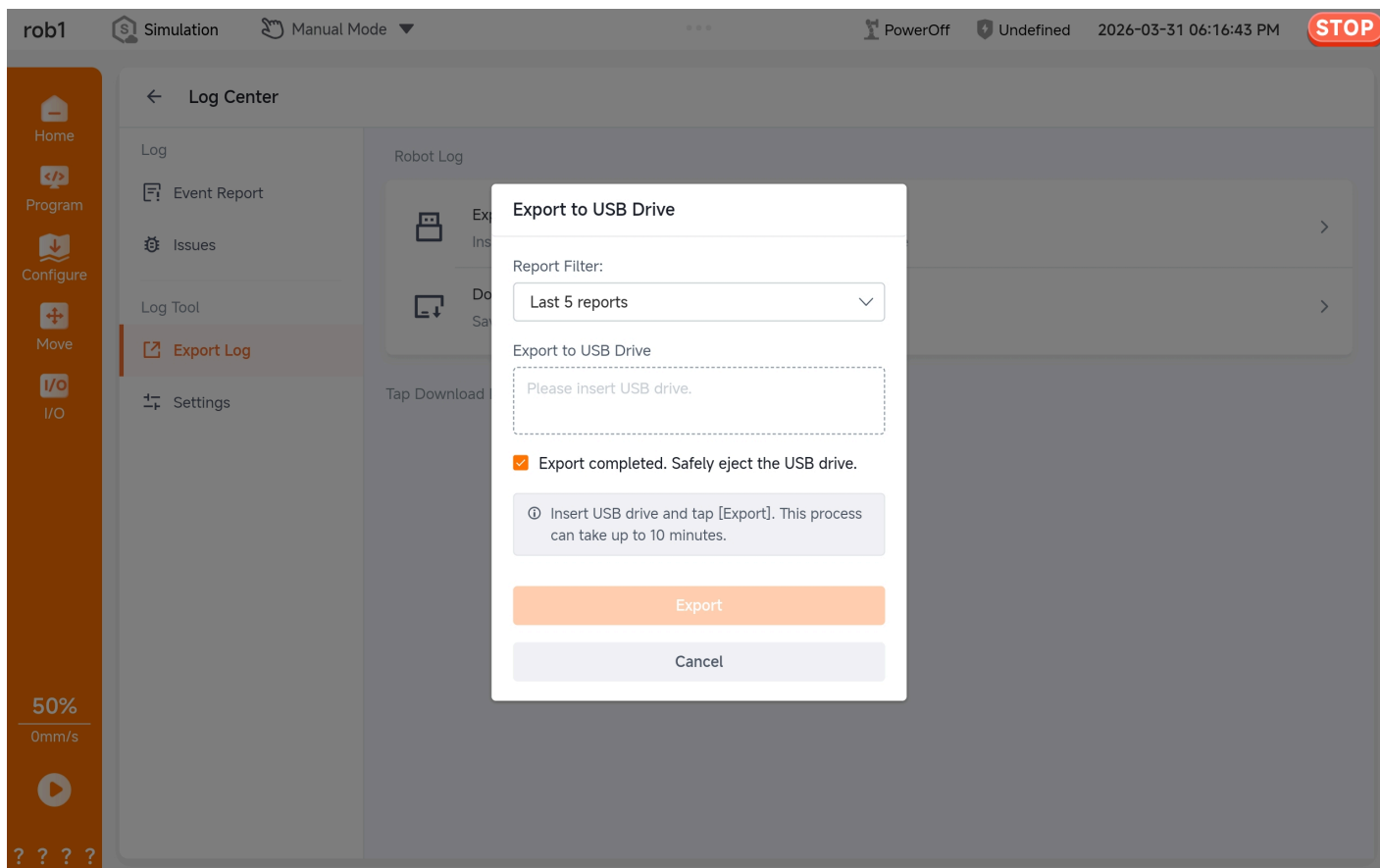
Last 5 reports

Destination Folder (Pad):

Download/00RLogs/

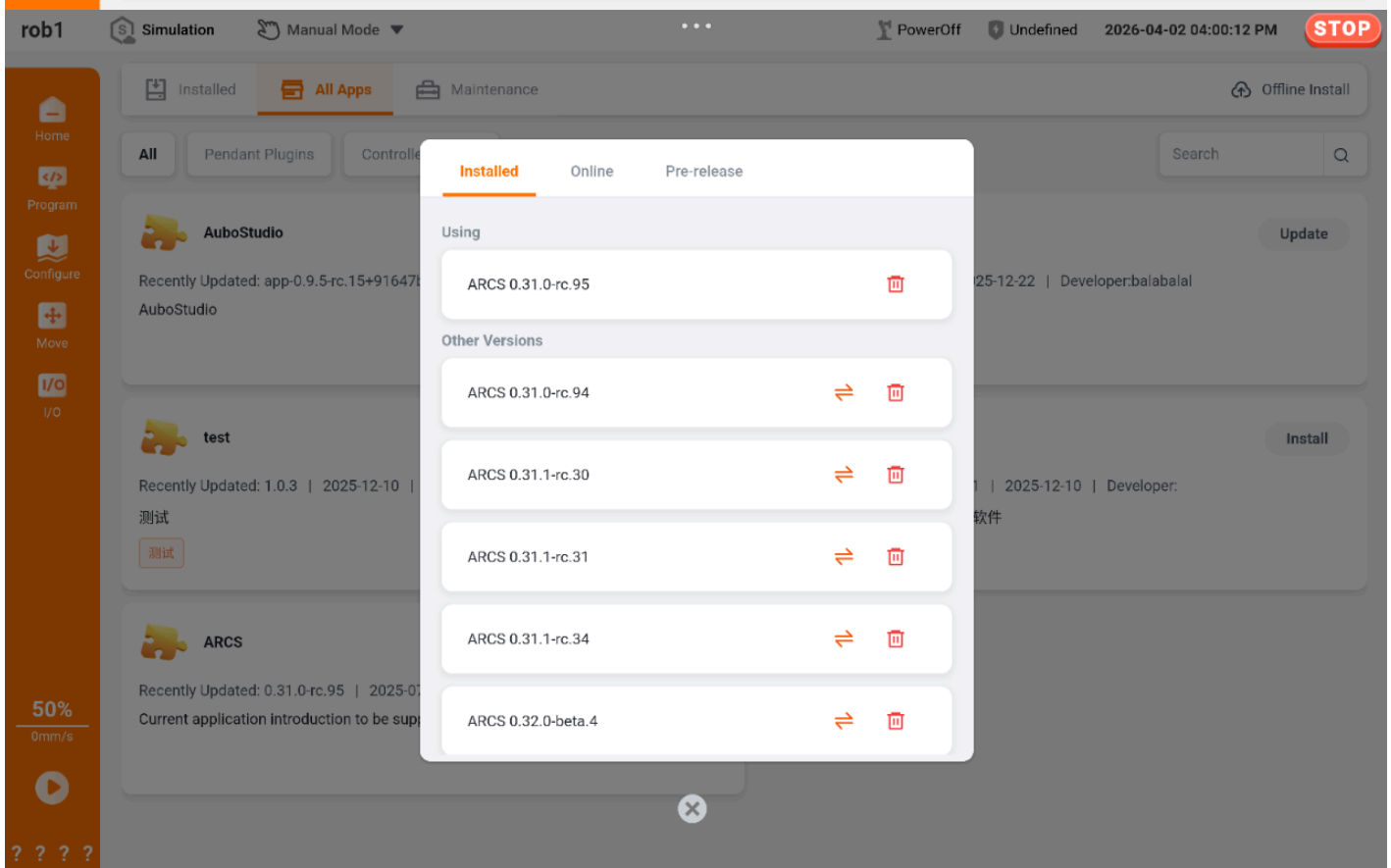
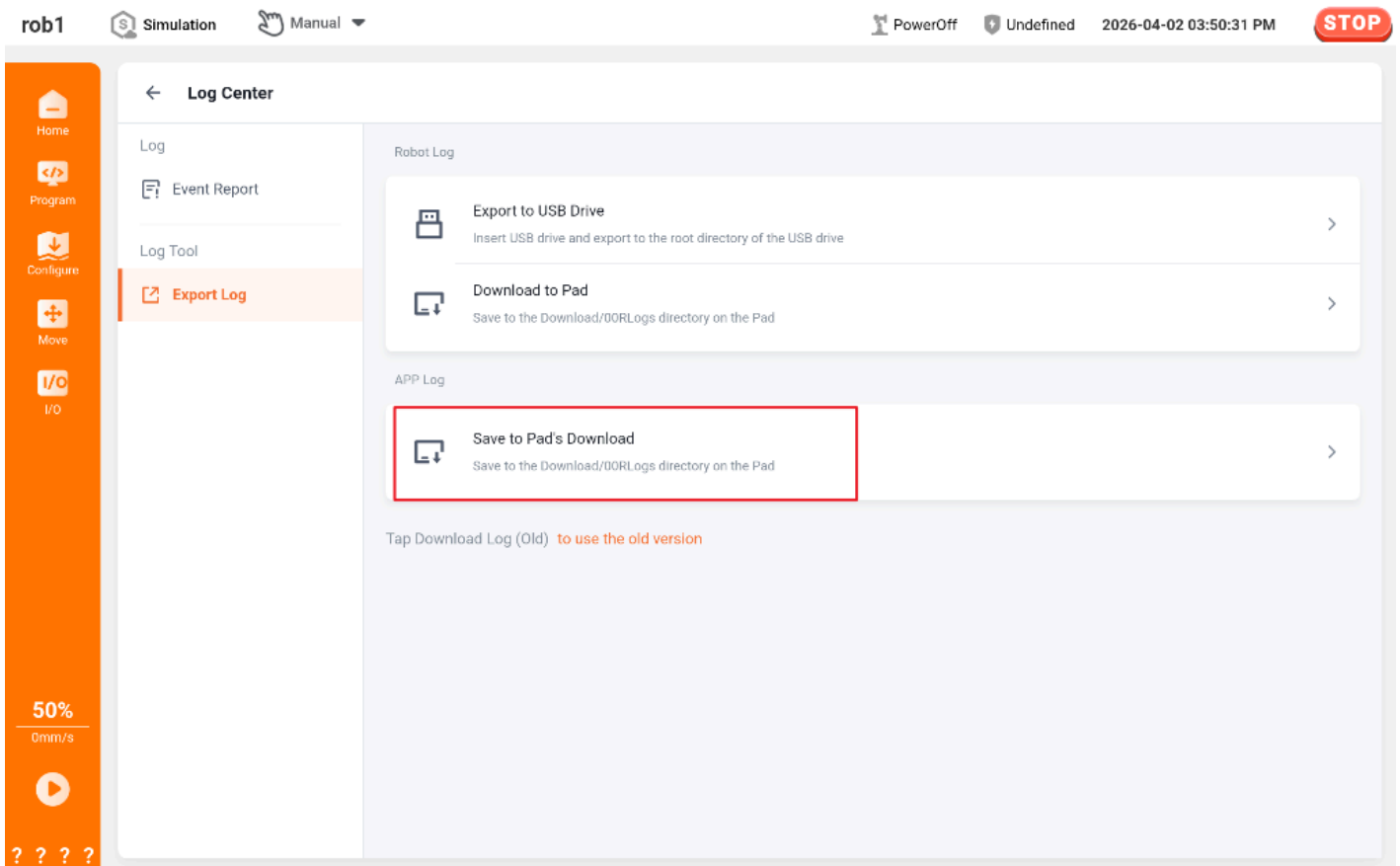
Download

Cancel



Save client logs

AuboStudio client logs are stored on the tablet. When the client encounters problems, client logs need to be exported separately.

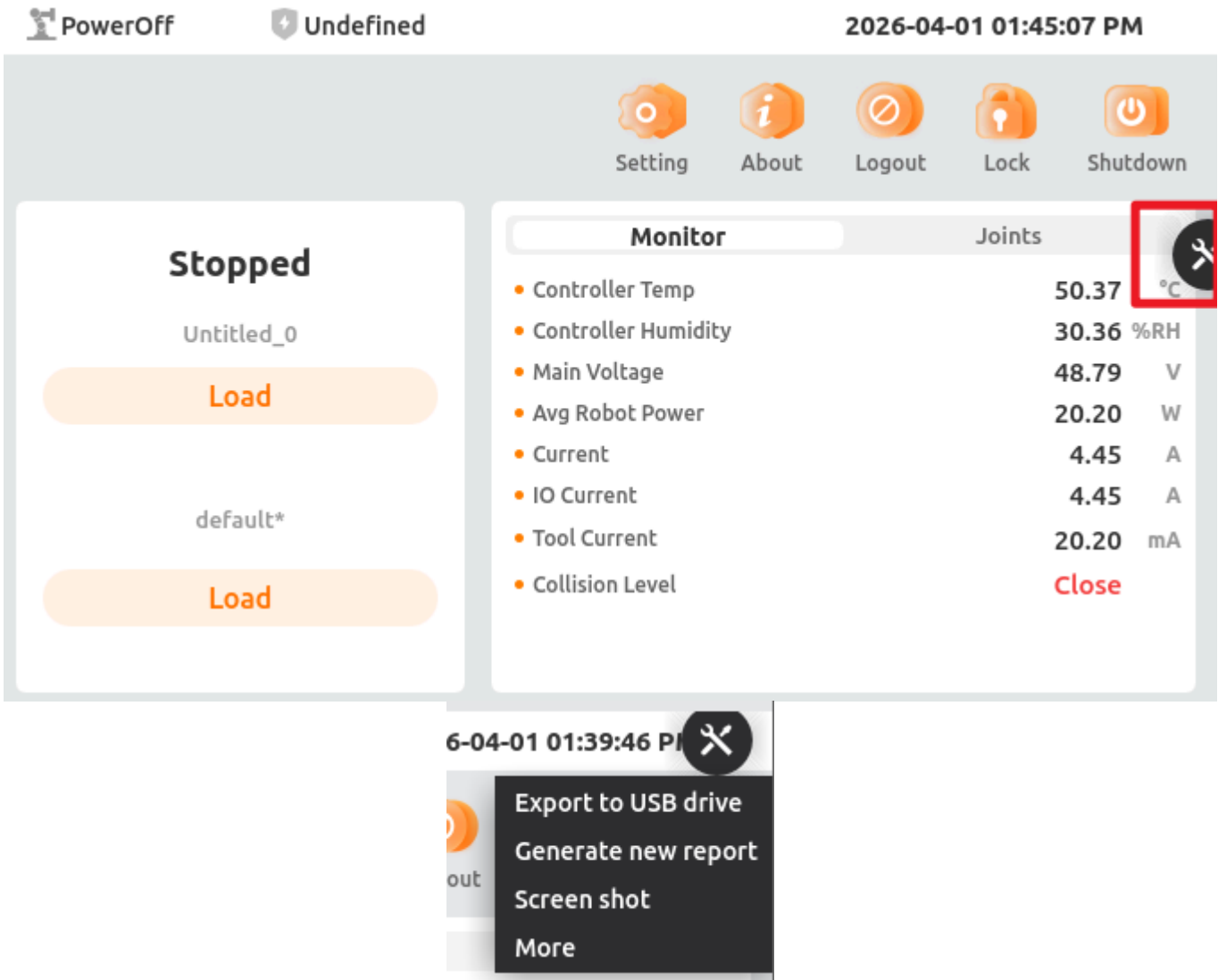


Tip

For AuboStudio APP versions 0.9.4 and above, users can directly download client logs from the </Download/00RLogs/> directory on the tablet.

Floating ball teach pendant only

The floating ball is a unique feature of the wired teach pendant. It automatically appears at the edge of the screen when the system starts up. The floating ball can quickly access the Log Center and provide rapid feedback when system anomalies occur.



The functions of the floating ball include:

1. Export to USB drive

Export to USB Drive

Report Filter:

Last 5 reports

Export to USB Drive:

Please insert USB drive.

☐ Export completed. Safely eject the USB drive.



Insert USB drive and tap [Export]. This process can take up to 10 minutes.

Export

Cancel

2. Generate a new report

Generate new report

Event type:

Software crash

Program error

Wrong collision detection

User experience

Other

Period:

5 minutes

1 hour

24 hour

3 day

All

Description:

Please enter text remarks.....

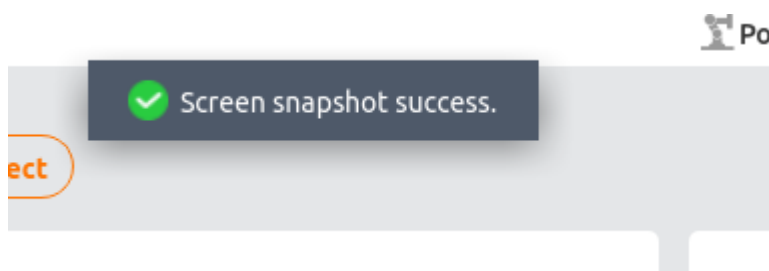
Tap [Generate] to generate an event report containing operational data and logs for the problem period.

☐ Upload event reports

Generate

Cancel

3. Screenshot



Screenshot

The "Screenshot" function can be used to capture system status and abnormal situations. Users can click "Screenshot" to take a screenshot. All screenshots are automatically saved to

the controller's `/root/Pictures/log_genie` directory. The system can save up to 50 recent screenshots.

After taking a screenshot via "Screenshot," click "Generate New Report." The system will automatically package the relevant screenshots into a report according to the time period, which will facilitate subsequent problem analysis and diagnosis.

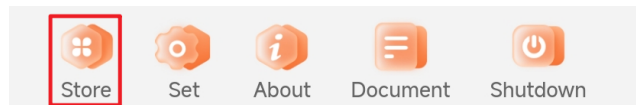
App Store

App store provide users with services for installing, updating, and managing applications. Through the App store, users can easily access various applications to expand system functionality.

Tip

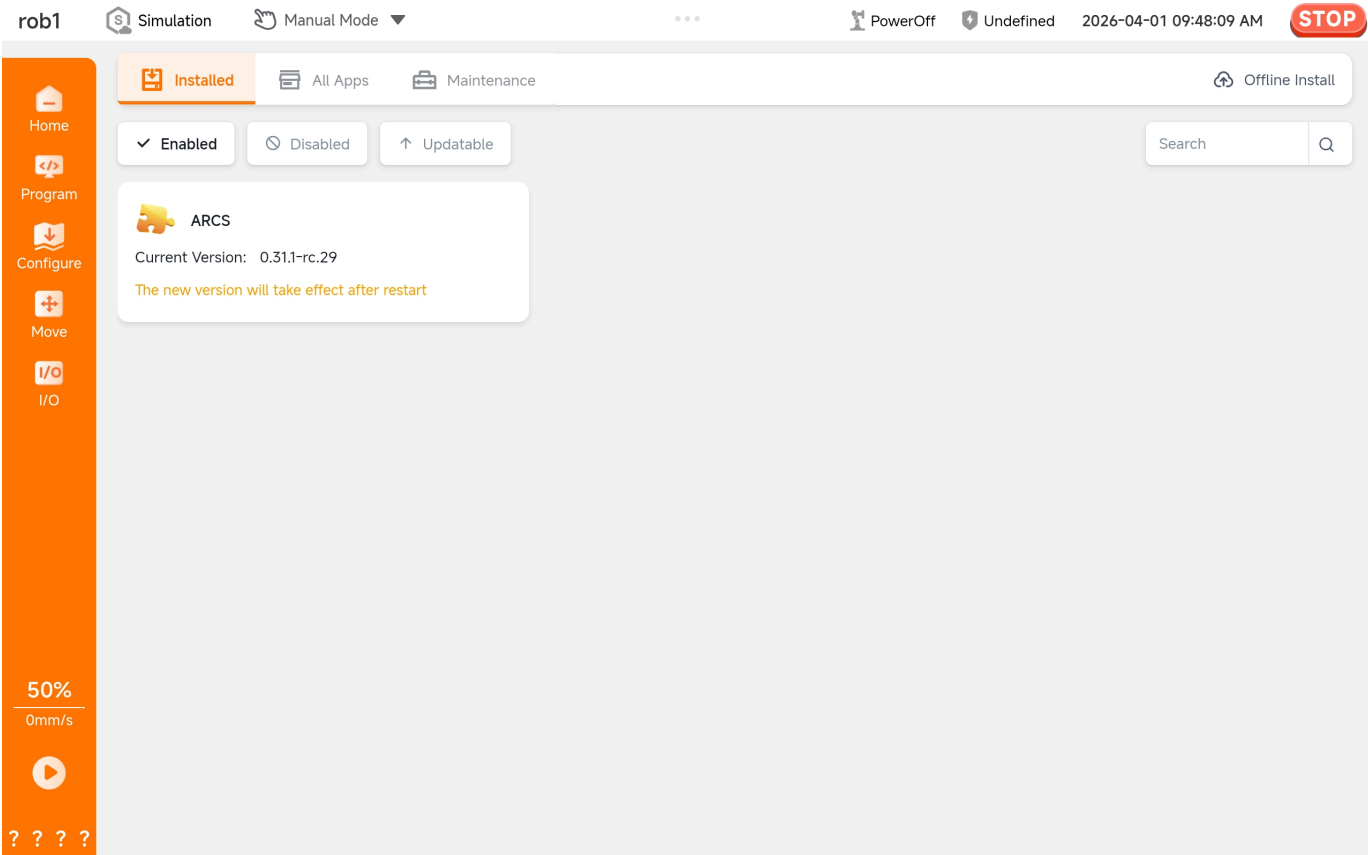
1. Supported versions:
 1. Wired teach pendant: ARCS 0.31.1 version and above
 2. Android client: ARCS 0.31.0, app version 0.9.3 and above
2. The app store is a functional module of the OPS system. Please ensure that the OPS system is installed before use.
3. For OPS system installation, please contact the relevant technical personnel.

App store entry

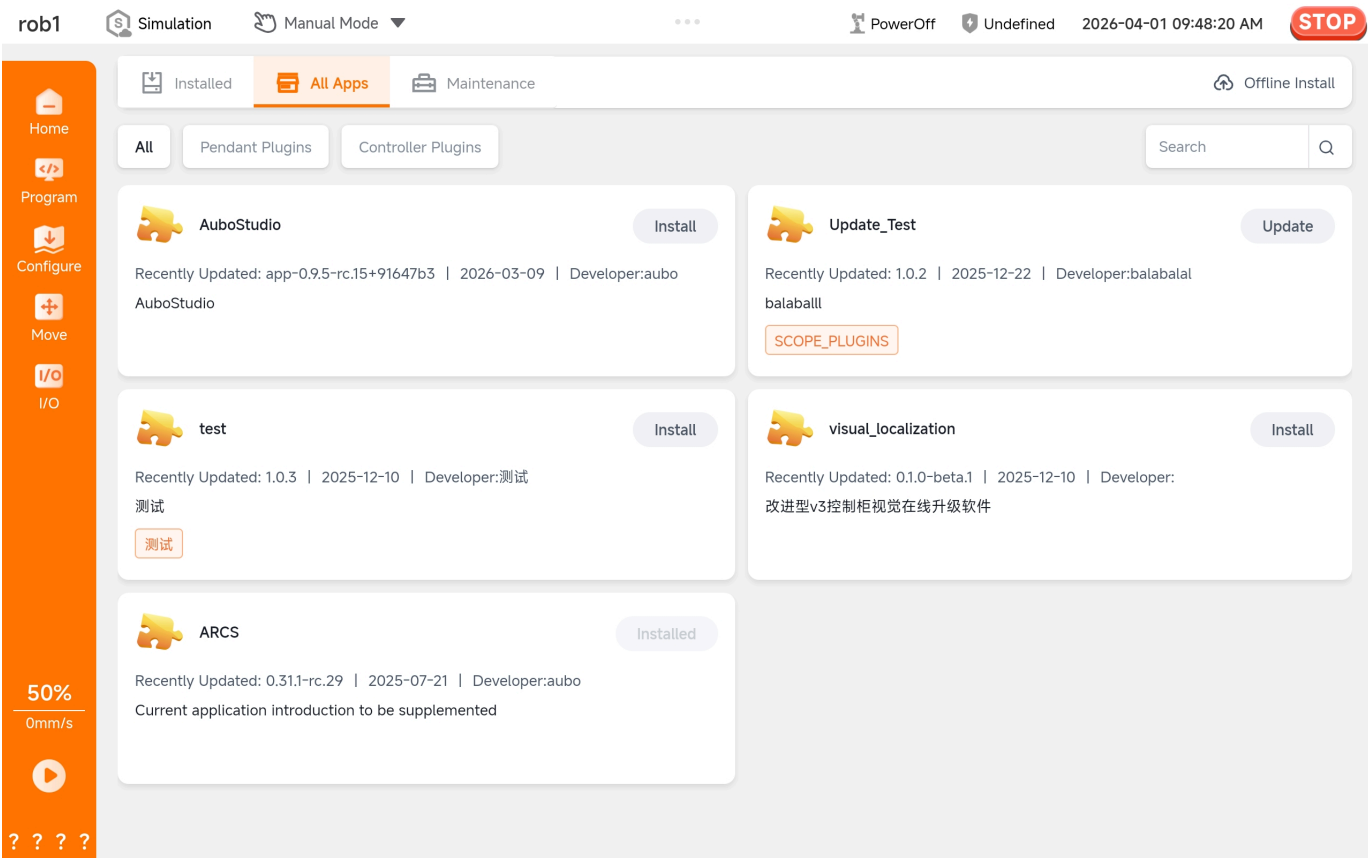


App list view

- Installed: All apps installed on the system are displayed.

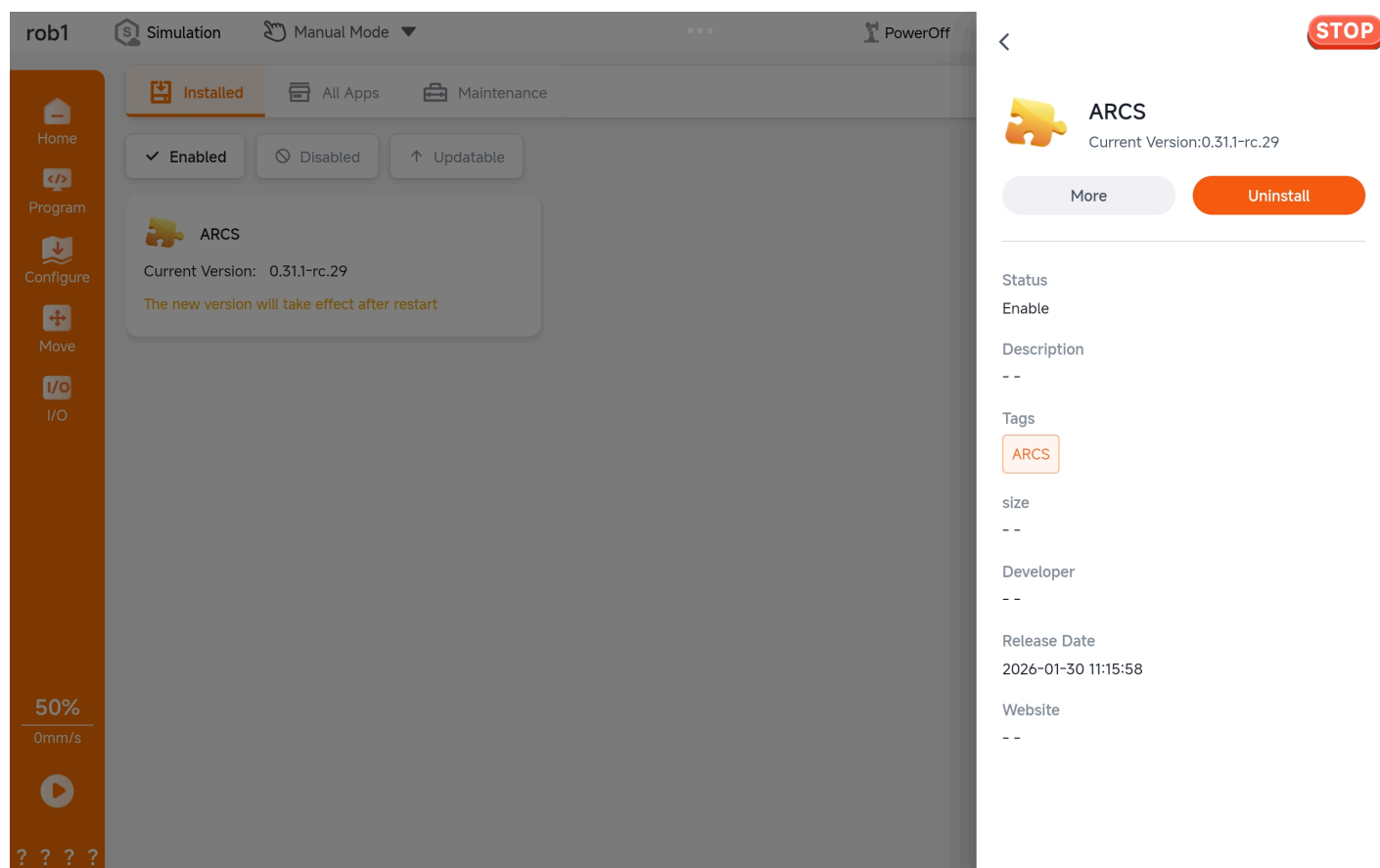


- All Apps: Apps available from the store are displayed.



App details

Clicking the app tab will display app drawer details on the right side of the page.

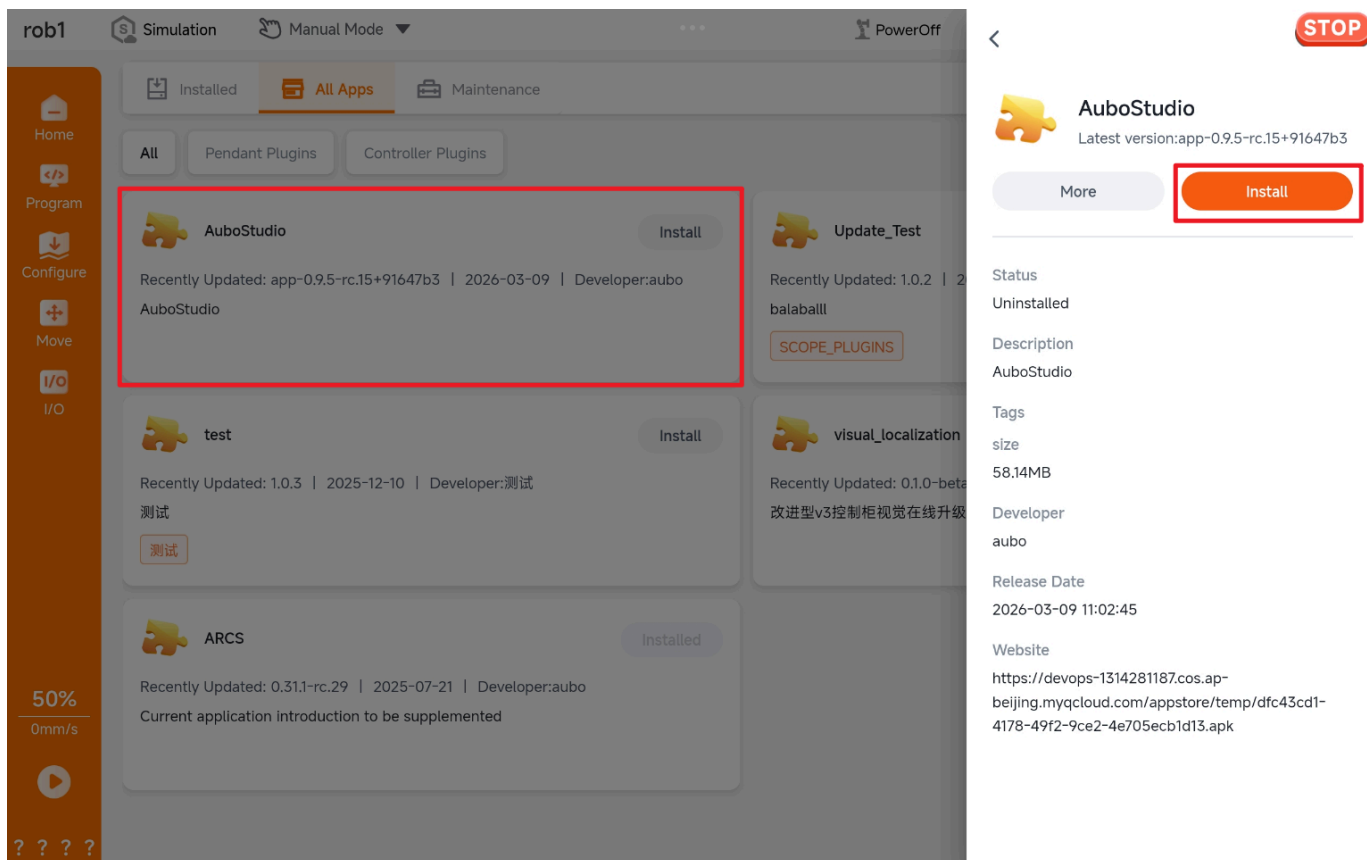


App management

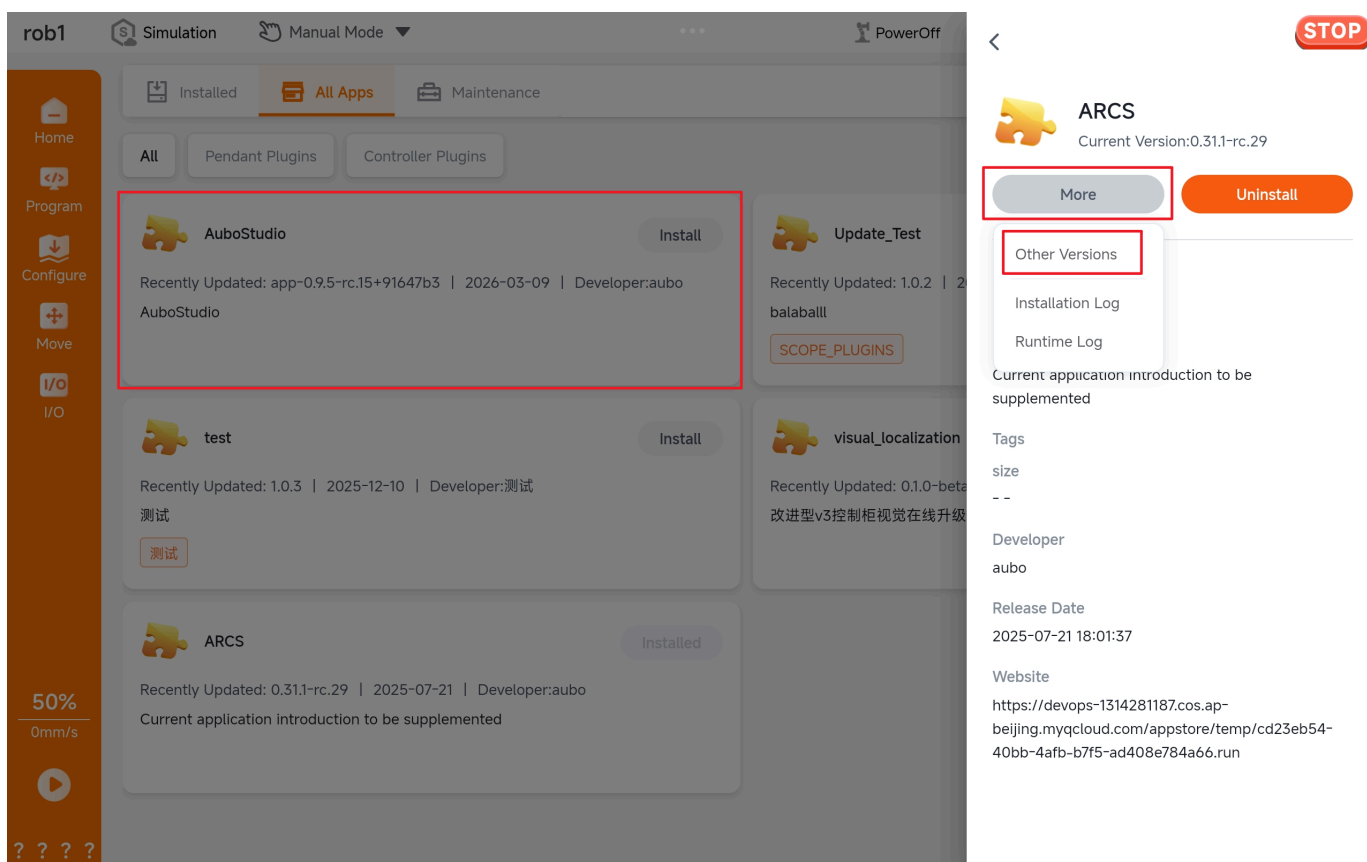
Online installation

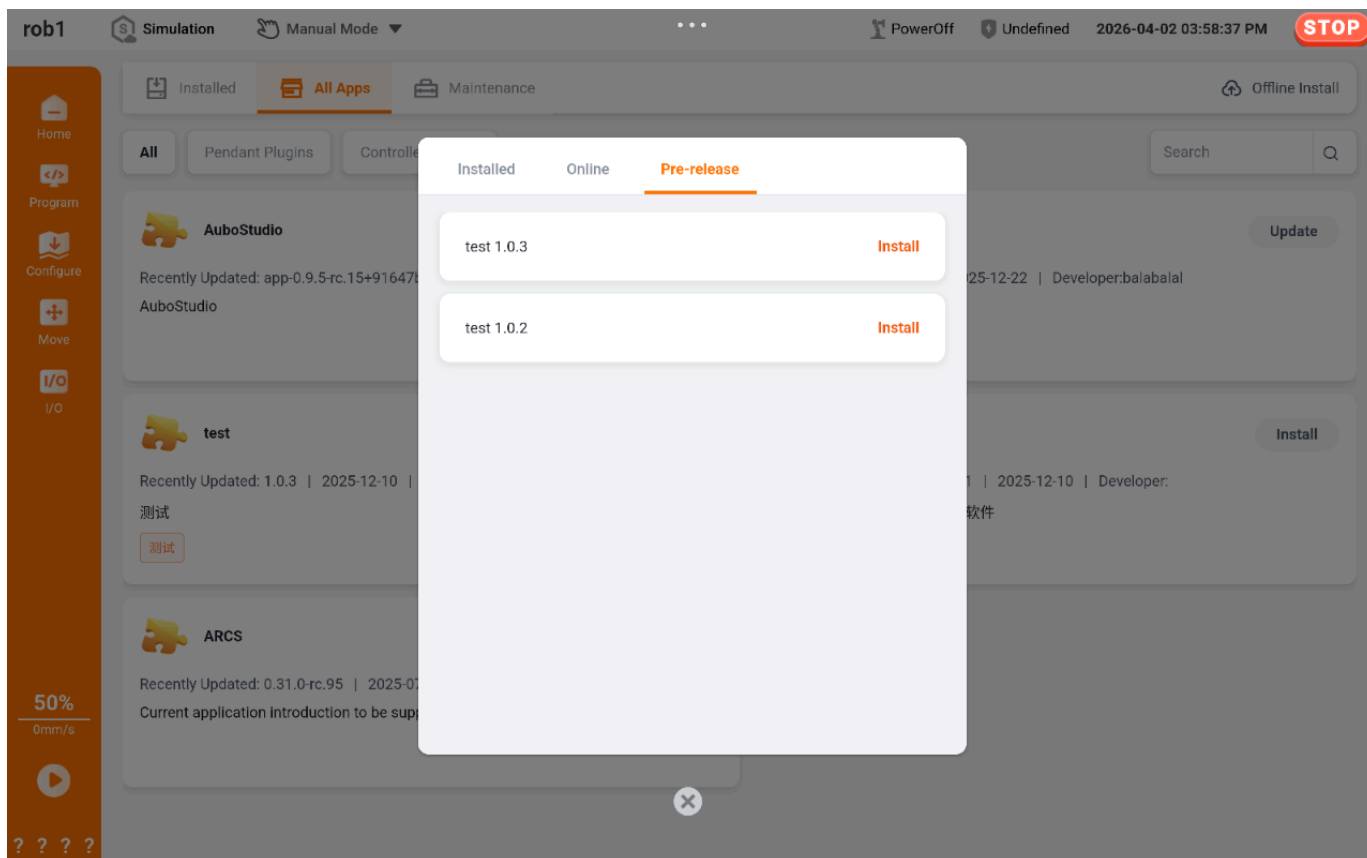
Users can directly select and install apps from the app list:

1. Standard installation: The latest version is installed by default.



2. Multiple version installation: The official version or the prerelease version can be selected.

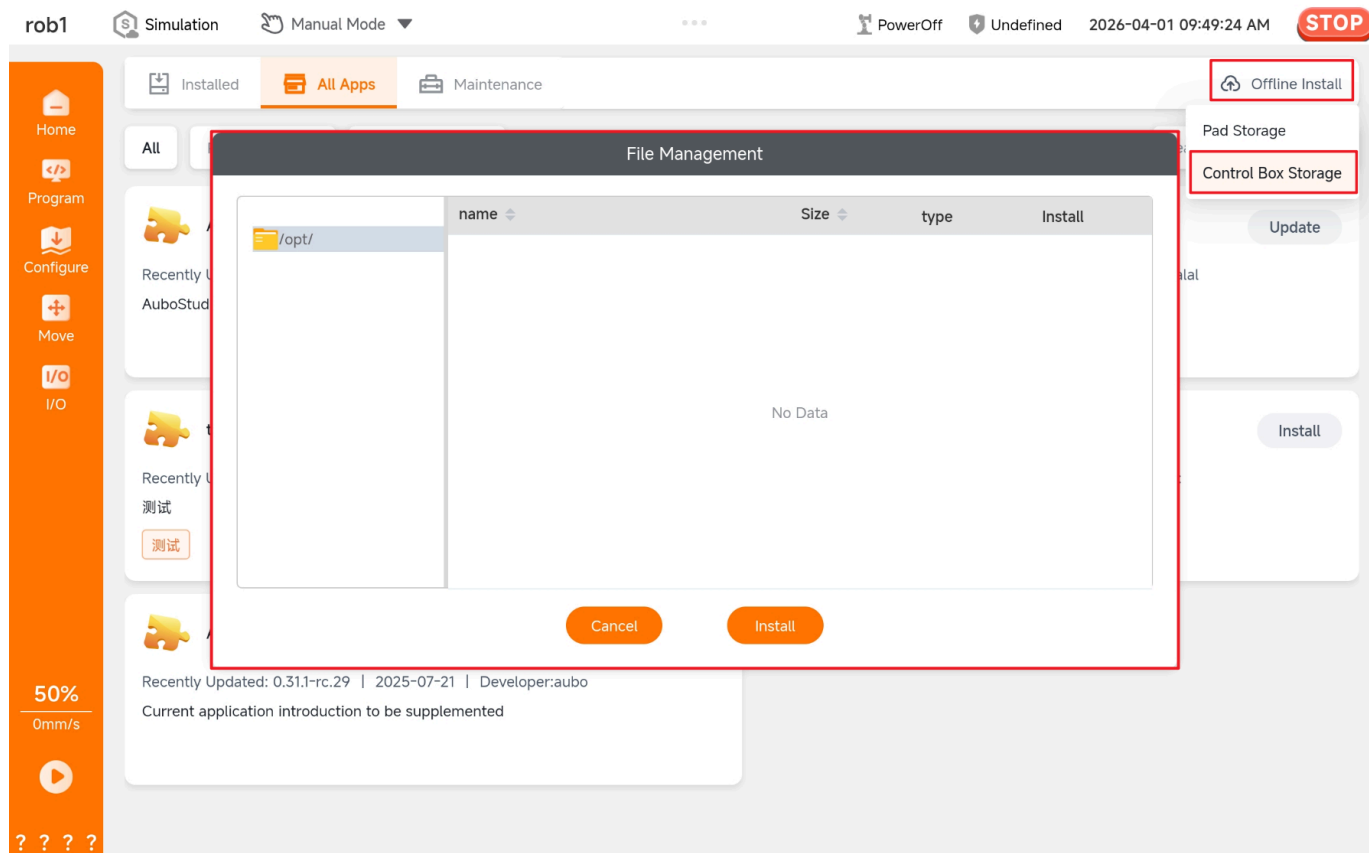




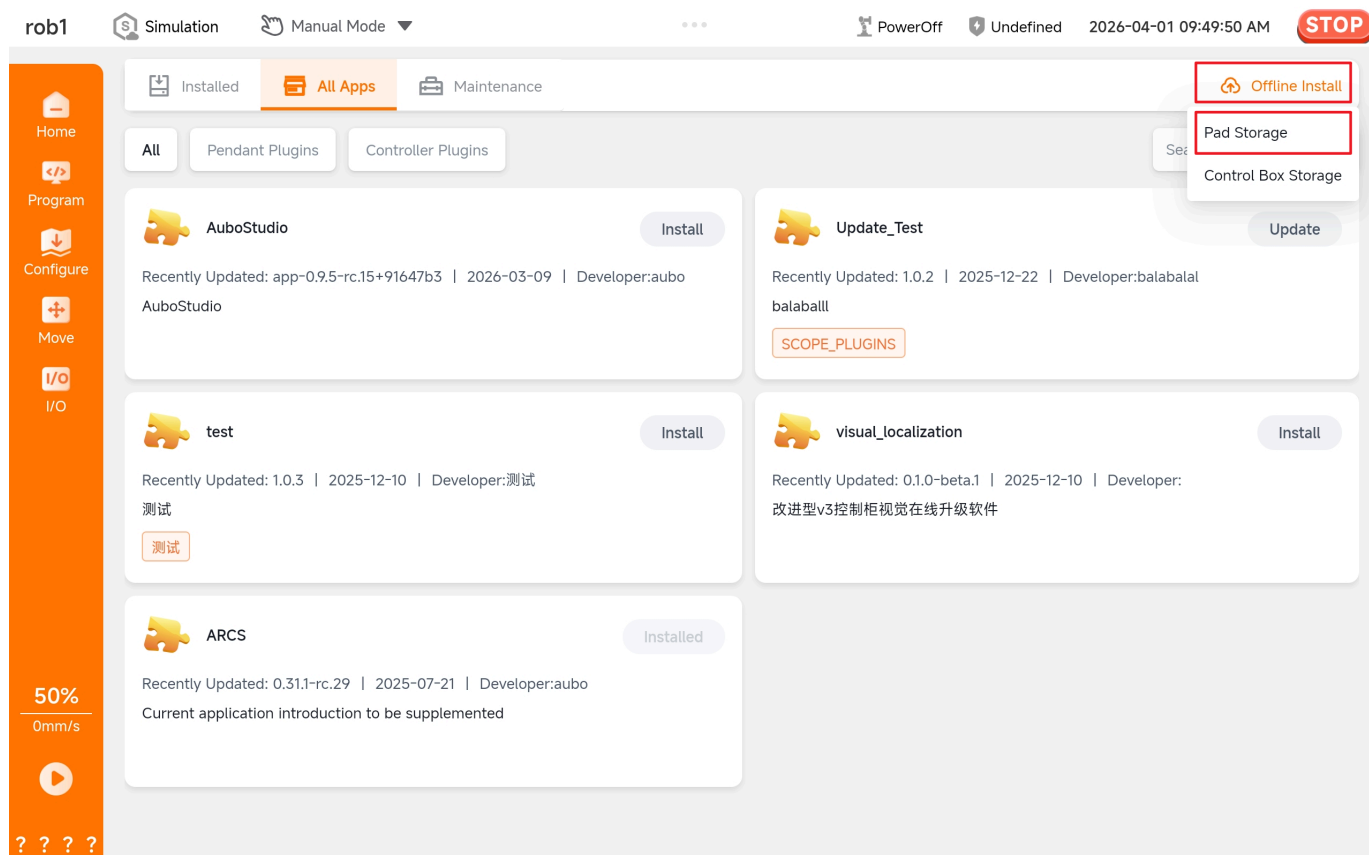
Offline installation

Users can upload the app installation package to the control box or tablet and install it by using the "Offline Installation" function on the right side of the top menu bar.

1. Local installation and USB installation for control box: Select "Open control box storage file" (RPK format supported), then select the app package from the list and click Install.

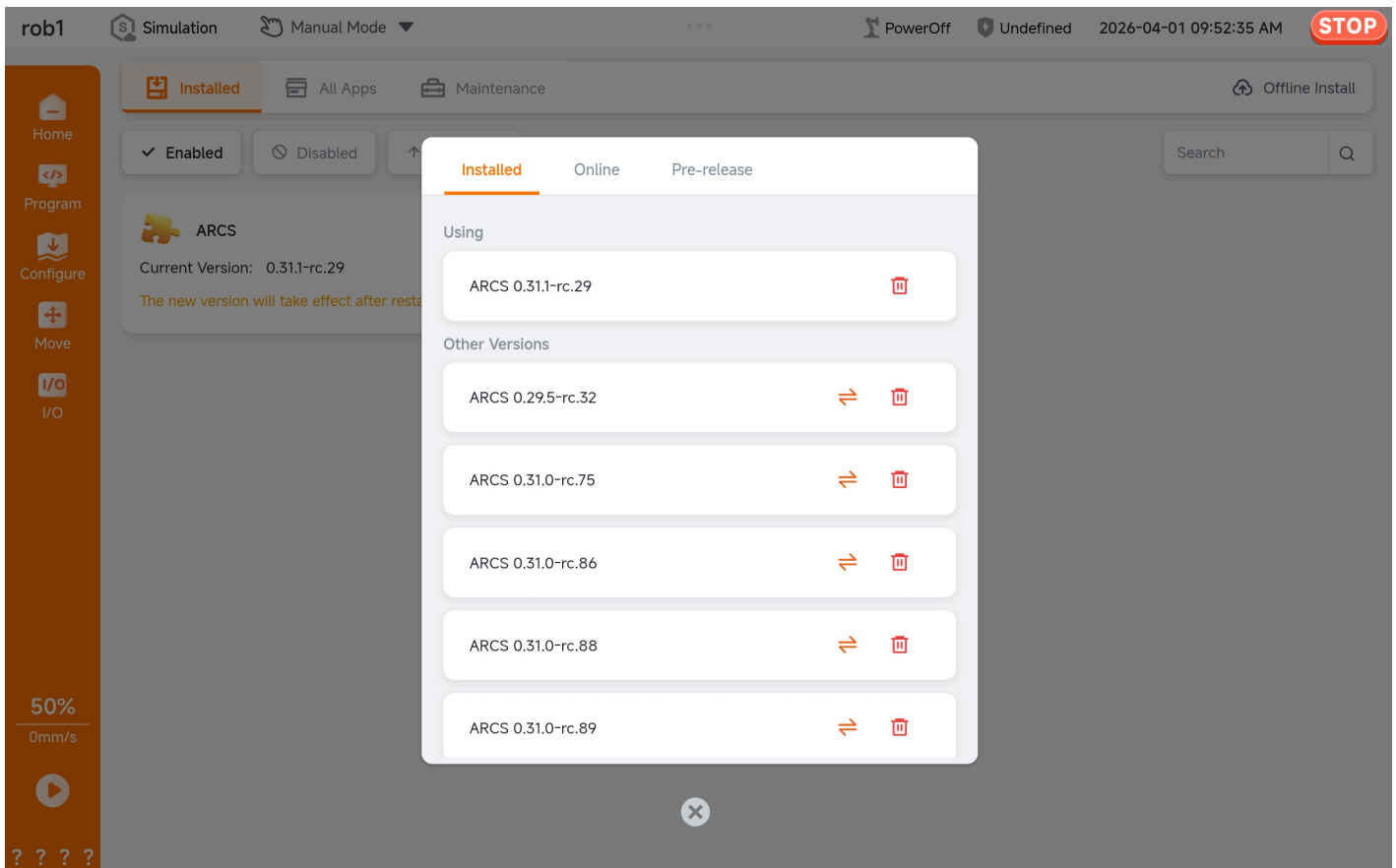


2. Upload installation package stored on tablet: Select "Open tablet storage file" (RPK format supported), then select the app package from the list and click Install.



Version switching

1. Click the tab to display the app drawer pop-up.
2. Click "More Operations > Other Versions."
3. The version list displays information on all versions of the current app.
4. Click the [Apply] button for the corresponding version to switch versions.

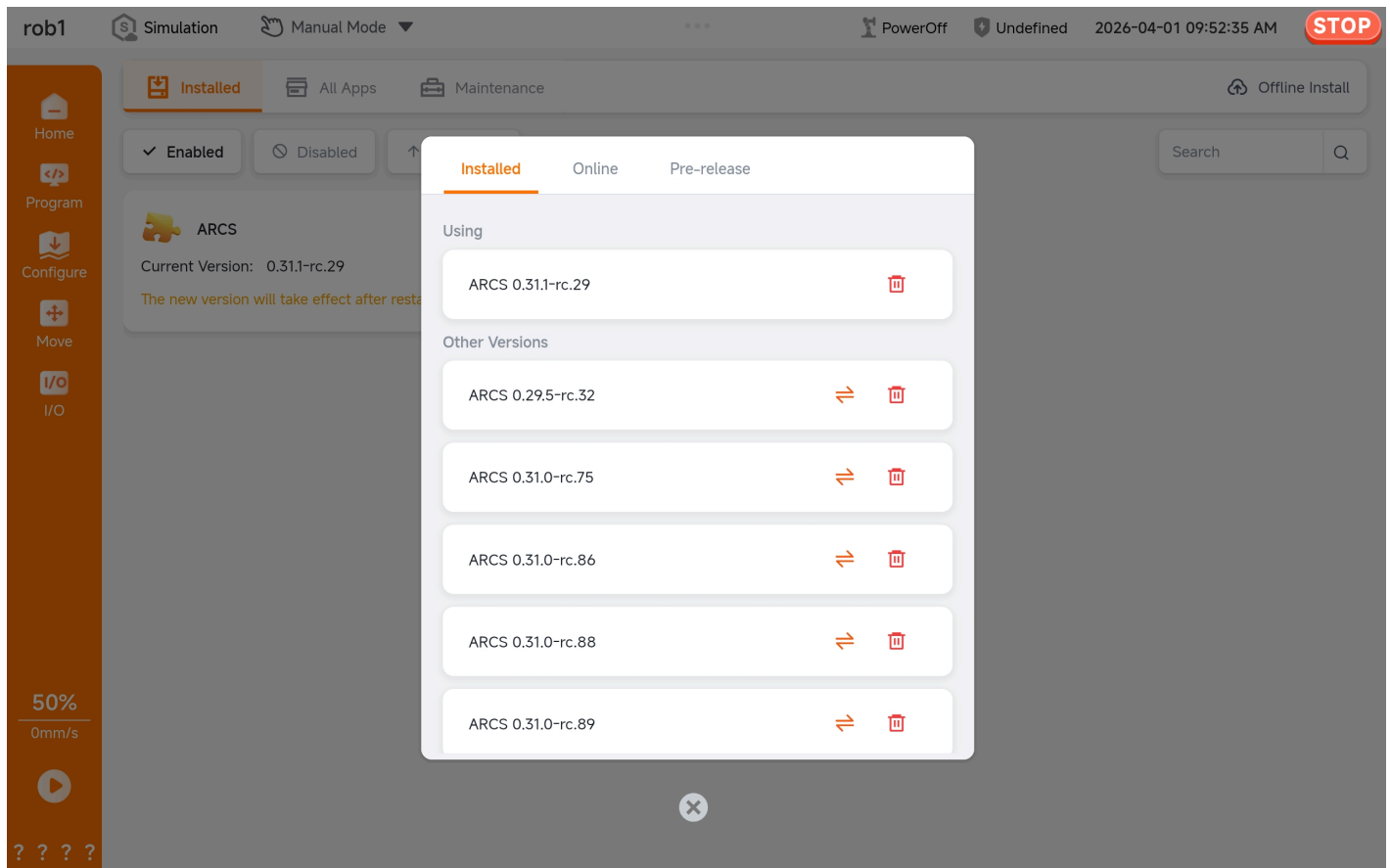


Uninstall app

Click the [Uninstall] button in the version information.

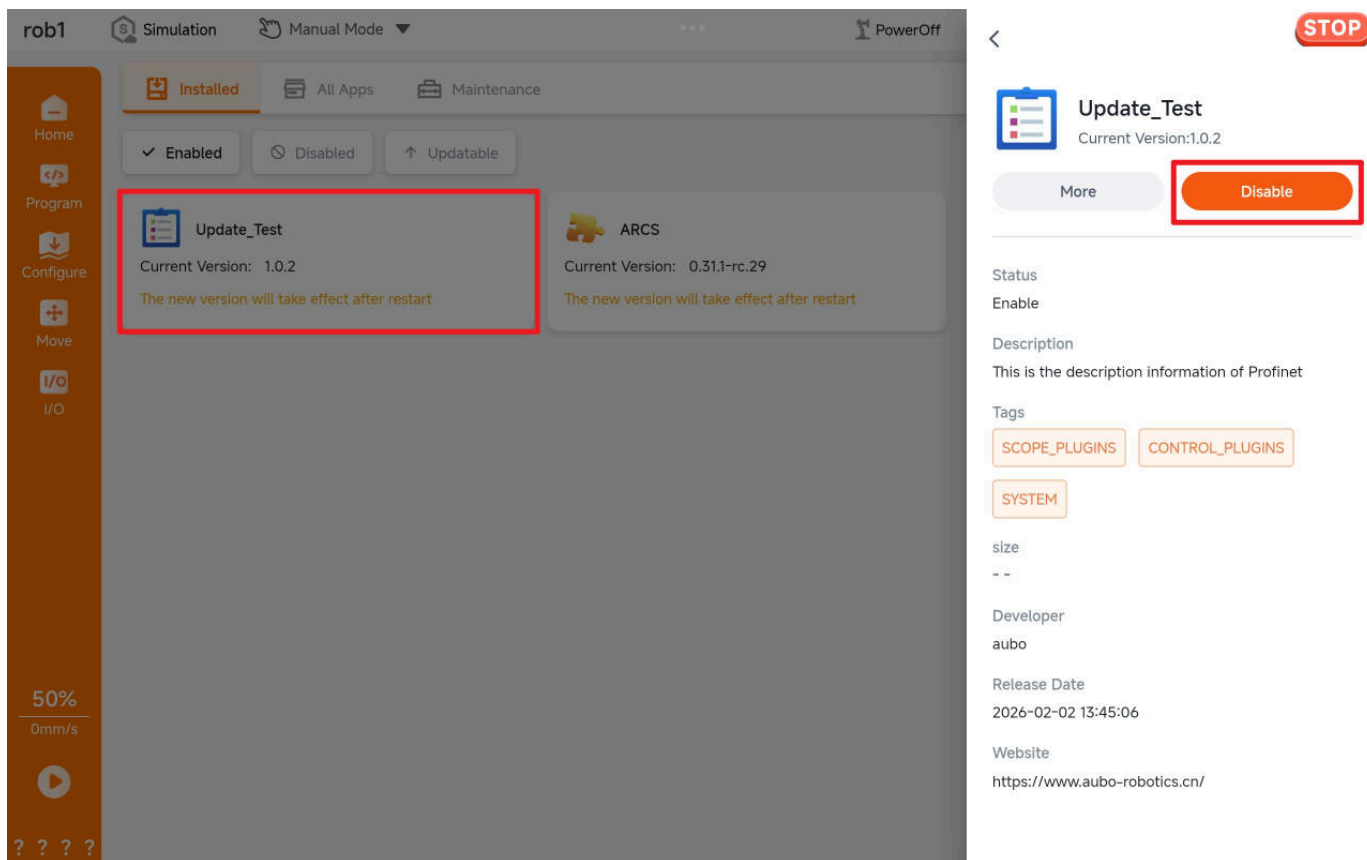
Tip

The version currently in use must be disabled before it can be uninstalled.

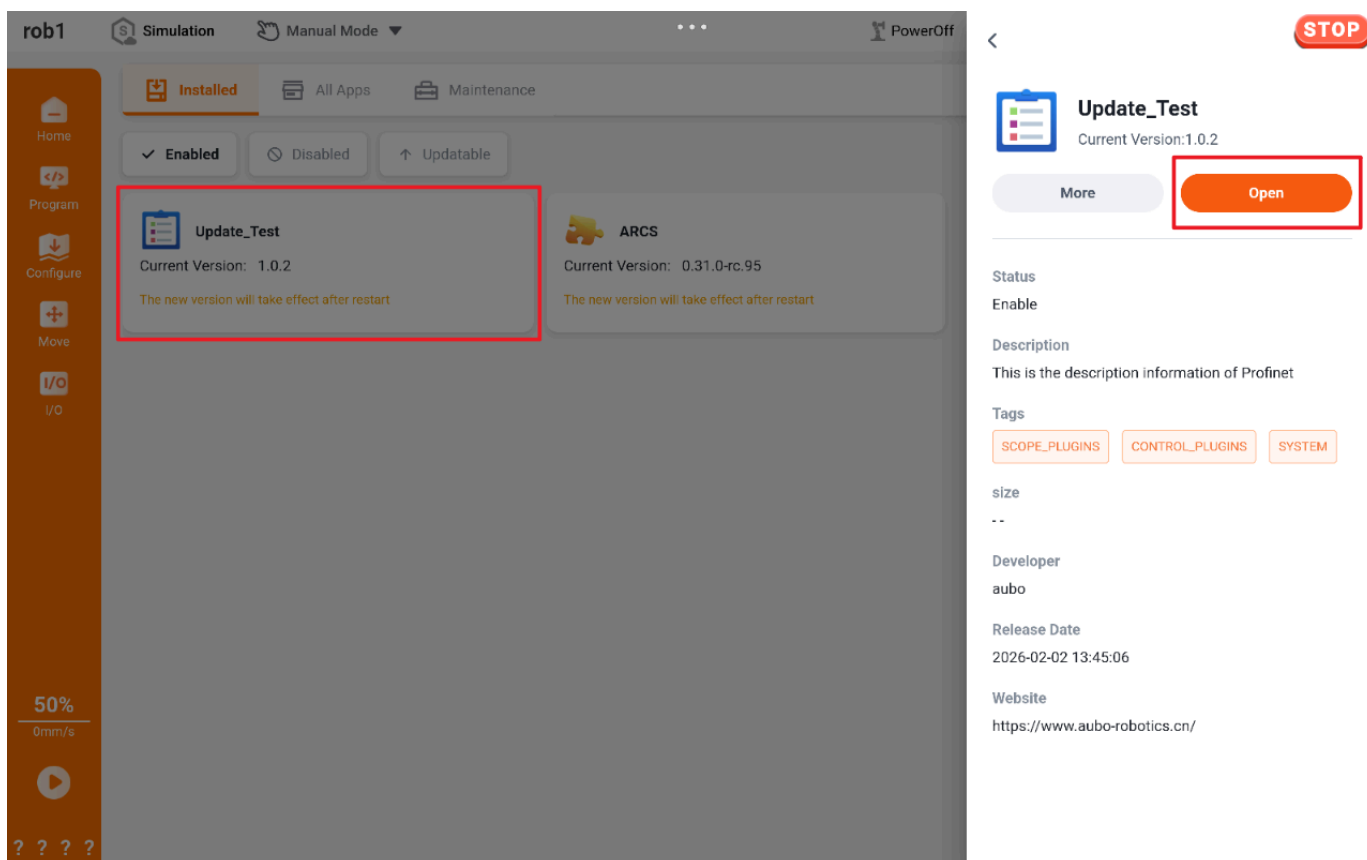


Operation guide

1. Click the [Enable/Disable] button on the app drawer page and restart the system for the changes to take effect.



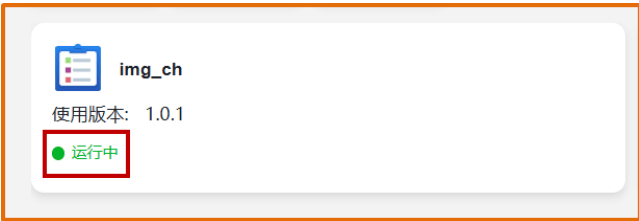
2. After the app is enabled, if the app is linked to the [Configure] page, clicking [Open] will access the app's configuration page.



Monitoring and log

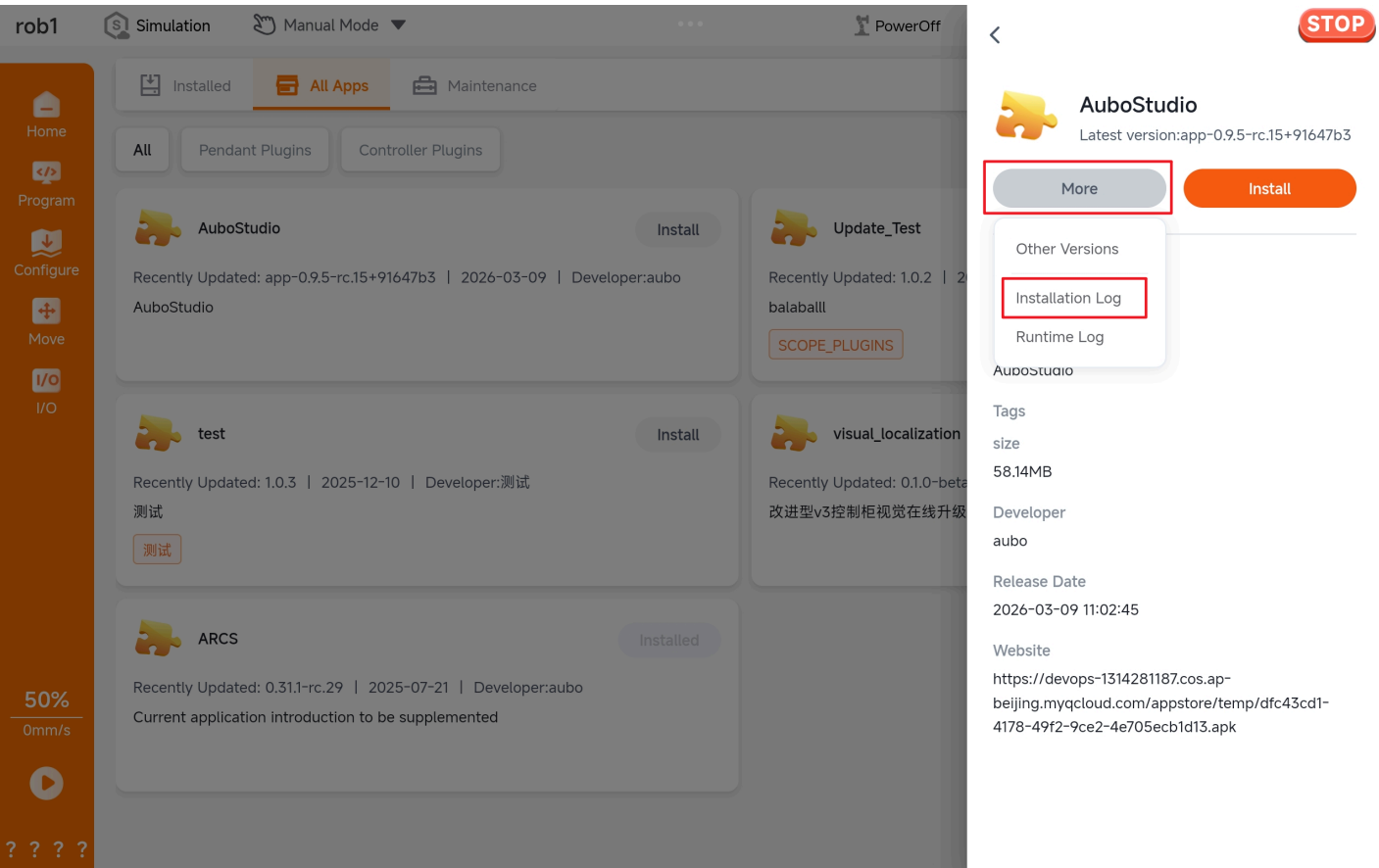
Status monitoring

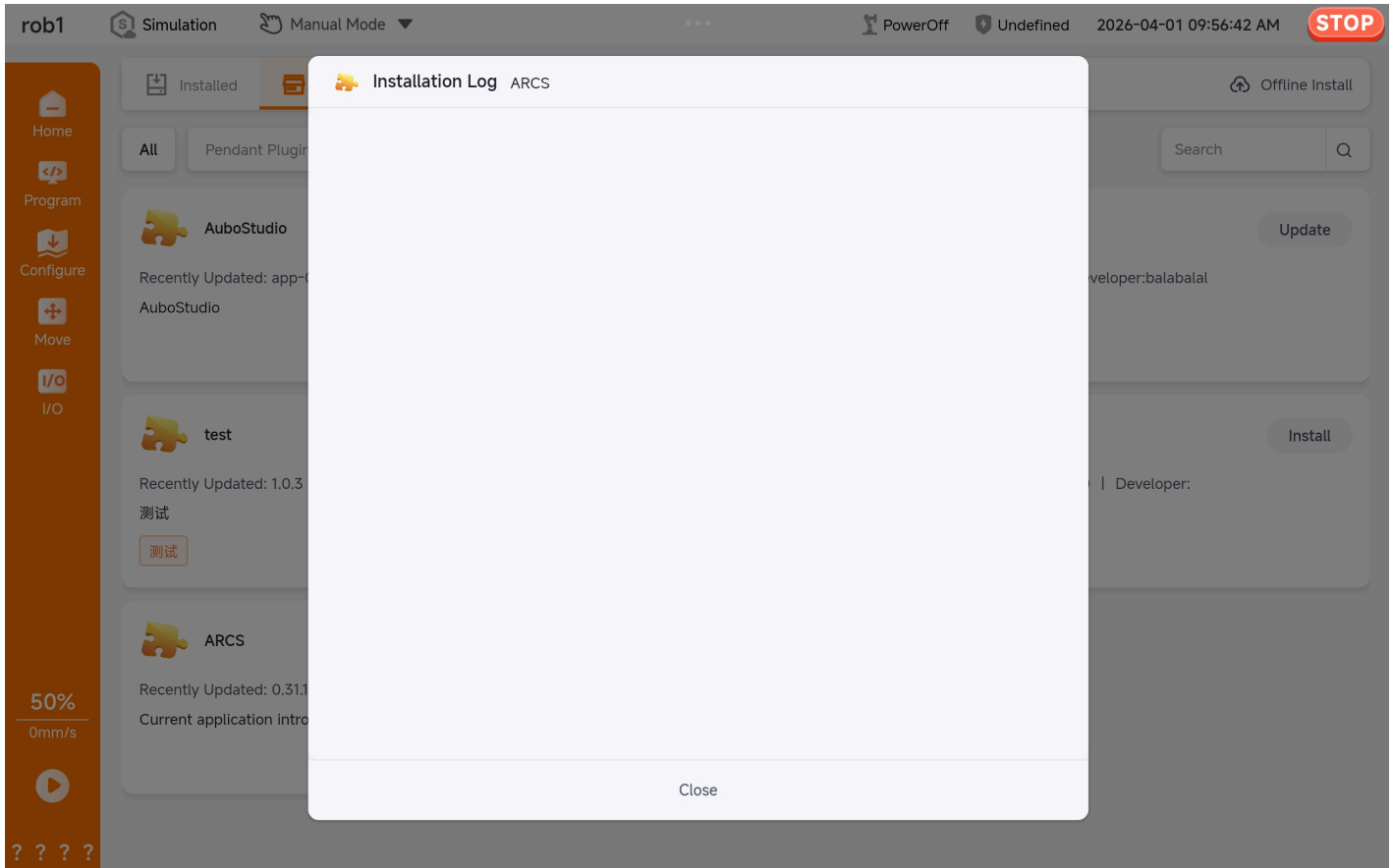
The system automatically monitors the operating status of all enabled apps and displays it in real-time on the app tab.



Installation log

An installation log is generated in real time during app installation. After installation, you can view the log by clicking on the app tab drawer page and selecting "More Operations > Installation Log."





Running log

All operations are recorded in the running log, and filtering by log level is supported. Click on the app tab to display the app drawer pop-up, then select "More Operations > Running Log" to view it.

rob1

Simulation

Manual Mode

PowerOff

Installed

All Apps

Maintenance

All

Pendant Plugins

Controller Plugins

AuboStudio

Install

Recently Updated: app-0.9.5-rc.15+91647b3 | 2026-03-09 | Developer:aubo

AuboStudio

test

Install

Recently Updated: 1.0.3 | 2025-12-10 | Developer:测试

测试

ARCS

Installed

Recently Updated: 0.31.1-rc.29 | 2025-07-21 | Developer:aubo

Current application introduction to be supplemented

Update_Test

SCOPE_PLUGINS

Recently Updated: 1.0.2 | 2026-03-09 | Developer:balabalal

visual_localization

Install

Recently Updated: 0.1.0-beta | 2025-12-10 | Developer:改进型v3控制柜视觉在线升级

Home

Program

Configure

Move

I/O

I/O

50%

0mm/s

?

?

?

?

STOP

ARCS

Current Version:0.31.1-rc.29

More

Uninstall

Other Versions

Installation Log

Runtime Log

Current application introduction to be supplemented

Tags

size

--

Developer

aubo

Release Date

2025-07-21 18:01:37

Website

https://devops-1314281187.cos.ap-beijing.myqcloud.com/appstore/temp/cd23eb54-40bb-4afb-b7f5-ad408e784a66.run

rob1

Simulation

Manual Mode

PowerOff

Undefined

2026-04-01 09:56:58 AM

STOP

Installed

All Apps

Maintenance

All

Pendant Plugins

Controller Plugins

AuboStudio

Update

Recently Updated: app-0.9.5-rc.15+91647b3 | 2026-03-09 | Developer:aubo

AuboStudio

test

Install

Recently Updated: 1.0.3 | 2025-12-10 | Developer:测试

测试

ARCS

Install

Recently Updated: 0.31.1-rc.29 | 2025-07-21 | Developer:aubo

Current application introduction to be supplemented

Offline Install

Search

Q

Home

Program

Configure

Move

I/O

I/O

50%

0mm/s

?

?

?

?

Runtime Log

ARCS

All

Info

Warning

Error

No logs available

Close

Plugin

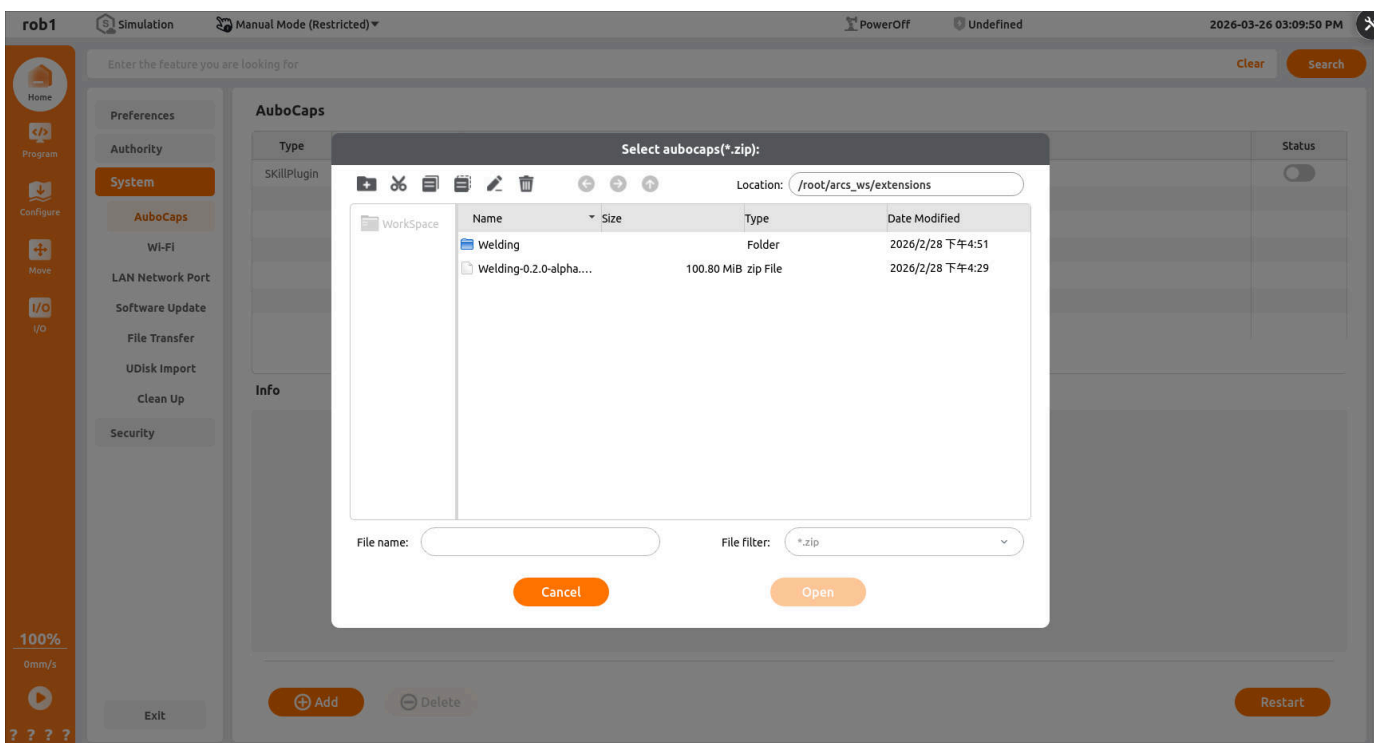
AuboStudio allows third-party developers to extend software functionality according to their own needs, giving the software unlimited scalability.

Import plugin installation package

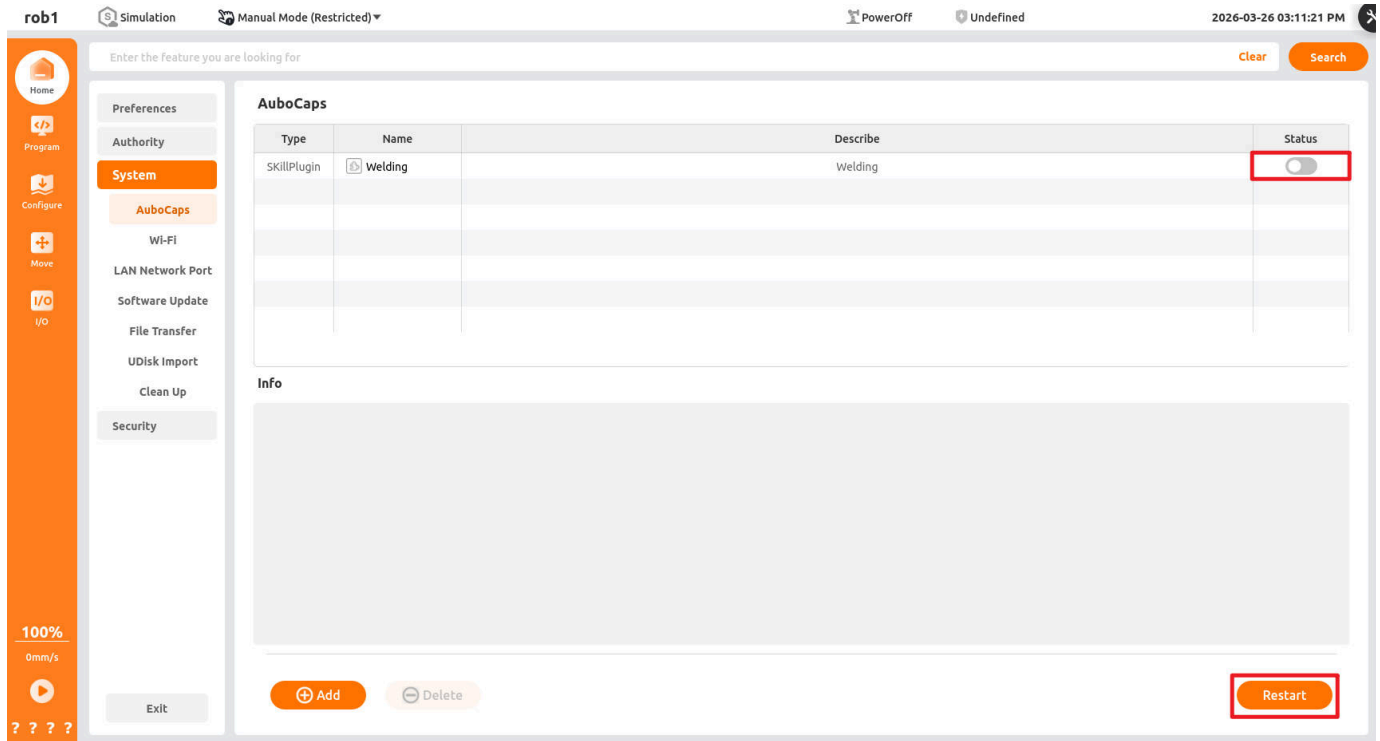
Users can import the plugin installation package into the AuboStudio workspace (path: /root/arcs_ws/extensions) using methods such as [USB flash drive import](#) and [File Transfer](#).

Load plugin

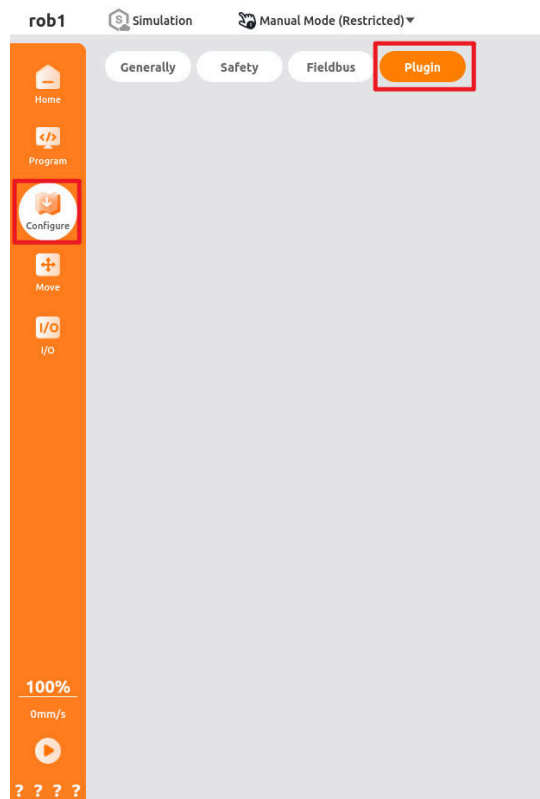
1. Open AuboStudio, click "Home > Settings > System > Plugin," click [Add], select the plugin installation package you want to install, and click [Open] to load the plugin.

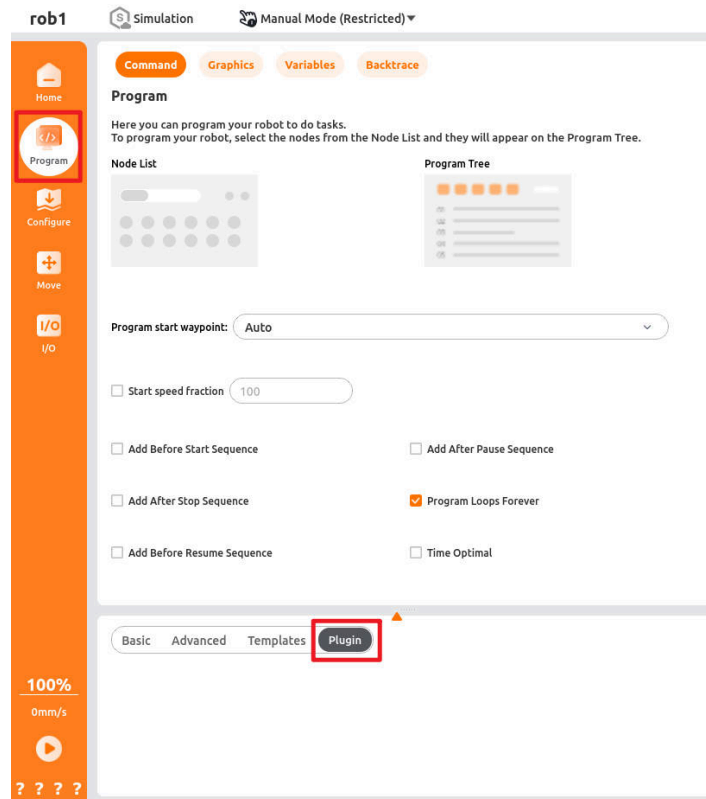


2. After loading is complete, change plugin status to , and click the [Restart] button in the lower right corner to restart system, so the changes take effect.



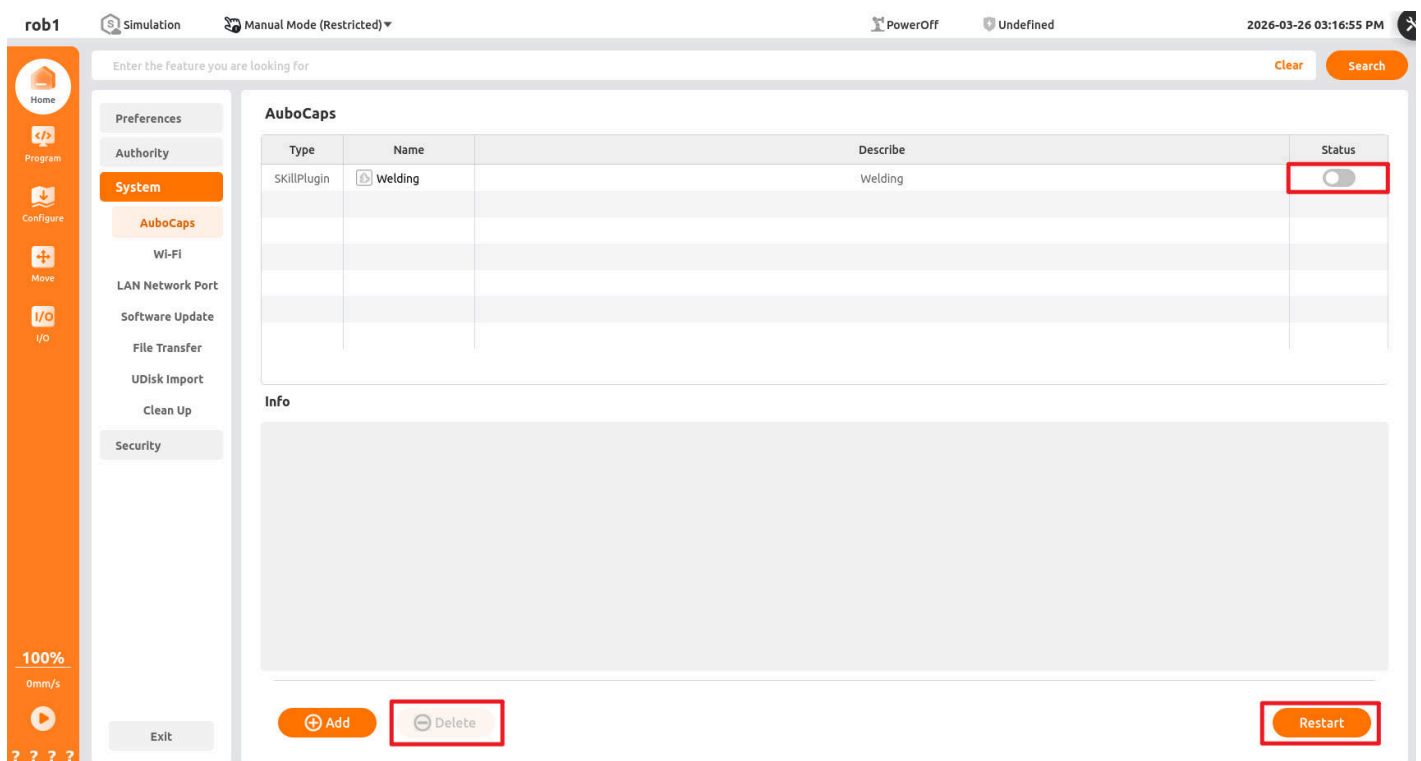
3. After the system restarts, click "Programming > Plugin" and "Configuration > Plugin" to view the program node and installation node of the plugin respectively.





Uninstall plugin

Open AUBOStudio, click "Home > Settings > System > Plugin", select the plugin installation package you want to uninstall, and change its status to  disable plugin **or** click the [Delete] button to remove the plugin, and then click [Restart] in the lower right corner to restart the system, so the changes take effect.



Appendix: Client Installation Environment Requirements

Table: AuboStudio APP Installation Environment Requirements

Category	Name	Requirements	Remarks
Hardware	CPU	armeabi-v7a 7th generation and above ARM processors (ARM 32-bit)	Most mobile phones on the market currently use this CPU type
		armeabi-v8a 8th generation, 64-bit ARM processor (ARM 64-bit)	New devices released in the past two years use this CPU type
		x86	A small portion of tablets use x86 processors. When selecting Intel x86 in the AS emulator, an x86 processor is used, and most other common third-party emulators also normally use x86
	Monitor	Minimum resolution 1280*800 px Recommended screen ratio: 8:5	--
Soft	Operating system (OS)	Android 10 and above	-