Operation Manual for Elfin Software

Collaborative Robot Elfin Series

Contents

Safety Precautions

Warning Signs

There are following safety signs in different levels in this manual:

Danger:

Failure to follow instructions may result in death or serious injury.

Warning:

Failure to follow instructions may result in injury to personnel or serious damage to equipment.

Caution:

Failure to follow instructions may result in equipment damage.

Approved by

Profile

This chapter introduces the safety principles and specifications to be followed when operating a robot or robot system. Integrators and users must read this manual carefully, and especially grasp and strictly follow the contents with warning signs. Due to the complexity and high risk of the robot system, users should fully understand the risks of operation and strictly observe and implement the specifications and requirements in this manual. Users and integrators need to have high safety awareness and comply with the Safety Specification for Industrial Robots (ISO 10218).

Elfin robot is equipped with a variety of built-in safety devices and safety I/O, digital and analog control signals in and out of electrical interfaces for connecting other machines and additional protective devices. Each safety device and I/O is built according to EN ISO13849-1:2008 and adopts D performance rating (Pld) of a 3-category structure to understand the configuration of safety functions, inputs, and outputs in the user interface. As well as relevant connection methods between safety devices and I/O.

Notes:

1. The use and configuration of safety devices and interfaces must follow the risk assessment procedures for each robot application program.

2. If the robot finds fault or violation in the safety systems (for example, a line in the emergency stop circuit is disconnected or safety limit violation occurs), the robot will immediately cut off the power supply and stop it.

3. The stop time should be considered as a part of the application risk assessment.

Danger:

1. Differences in the safety configuration parameters used and those identified by the risk assessment may lead to risks that cannot be reasonably eliminated or adequately mitigated.

2. Make sure that the tools and gripping jaws should be properly connected to avoid danger in case of power interruption.

3. The end-effector is not protected by the safety system of Shenzhen Han's Robot Co., Ltd. End effectors and /or cable jumpers are not monitored.

Safety precautions

Outline

This manual contains safety measures to protect users and prevent machine from damage. The user needs to read all relevant descriptions in the manual and be fully aware of safety precautions. In this manual, we try to describe all situations, but because there are so many possibilities, it is impossible to write down all what can't do and don't allow.

Notice for Use

The following basic information needs to be understood and followed by the user when a robot or robot system is started for the first time. Other safety-related information is described in other parts of the manual. However, it is not possible to cover all aspects. In practical application, specific problems need to be analyzed.

Installation and operation:

1. Be sure to install the robot and all electrical equipment in accordance with the requirements and specifications in this manual.

2. Preliminary test and inspection of the robot and its protection system shall be conducted before the robot is used for the first time and put into production.

3. Before starting the system and the equipment for the first time, it is necessary to check whether the equipment and the system are complete, whether the operation is safe and any damage. In this test, it is necessary to observe whether it conforms to the national or regional effective safety production rules and regulations, and all safety functions must be tested.

4. The user must check and ensure that all safety parameters and user programs are correct and that all safety functions work properly. A person qualified to operate the robot is required to check each safety function. Only after thorough and careful safety test and the required safety level is reached can the robot be started.

5. Professional personnel are required for installation and commissioning of the robot according to the installation standards.

6. After the robot is installed and built, a comprehensive risk assessment shall be conducted again and documented.

7. Safety parameters shall be set and changed by the authorized personnel, and passwords or quarantine measures shall be used to prevent unauthorized personnel from changing or setting safety parameters. After the safety factors are modified, the relevant safety functions need to be analyzed.

8. In case of accident or abnormal operation of the robot, please press the emergency stop switch to stop the robot.

9. Brake is installed in the joint module of Elfin series robots. When the power supply is cut off, the robot attitude shall be maintained. Do not cut off the power supply system frequently. It is recommended that the time interval of each on/off operation should be more than 10s.

10. Elfin series robots have collision detection function. When the external force on the robot exceeds the set normal force range of the user safety, the robot

will stop automatically to prevent collision injury of the robot or operator. This function is specially designed for the safety of man-machine collaborative work of Elfin robots, but the robot system must be within normal operating range and the Elfin control cabinet shall be used. If the users use the controllers developed by themself, the robot does not have the above functions. The users shall be responsible for the dangerous consequences.

Operation and Heating

1. The robot body and the control cabinet will generate heat in operation.
Do not operate or touch the robot when it is working or just stops working.
2. Cut off the power supply and wait for an hour before the robot cools down.
3. Do not stick your finger into the heating area of the control cabinet.

Use

1. Ensure that the robot's arms and tools are properly and securely installed in place.
2. Make sure that the robot arms have enough space to move.
3. Do not use the damaged robot, please contact the technical support staff of Shenzhen Han's Robot Co., Ltd. in time.
4. Do not connect safety devices to normal I/O interfaces, only use safety I/O interfaces.
5. Ensure correct installation settings (such as installation angle of robot body, weight in TCP, TCP tool coordinates, safety configuration). Save the installation file and load it into the program.
6. Tools and obstacles should not have sharp corners or twisting points. Make sure that all people's heads and faces are out of reach of the robot.
7. Pay attention to the movement of the robot when using the demonstrator.
8. Any impact will release a significant amount of kinetic energy which is much higher than that under high speeds and high payloads.
9. Connecting different machines may increase hazards or create new hazards. A comprehensive risk assessment is conducted throughout the installation at all times. Always select the highest performance level when different security and emergency stop performance levels are required. Always read and understand the manuals for all equipment used in the installation.
10. Do not change the robot. Changes to the robot may pose risks that the integrator cannot predict. Robot authorization reconfiguration is subject to the latest version of all relevant service manuals. If the robot is altered or modified in any way, Shenzhen Han's Robot Co., Ltd. refuses to assume all responsibilities.

Application Environment

1. When a robot is connected to or work together with a machine that can cause damage to the robot, it is highly recommended to independently check all functions of robots and the robot program. It is recommended to use temporary way-points beyond workspaces of other machines to test the robot program.

2. Shenzhen Han's Robot Co., Ltd. shall not be liable for any damage to the robot or personal injury due to script error or improper operation of the robot.

3. Don't expose the robot to a permanent magnetic field all the time. A strong magnetic field can do damage to the robot.

Personnel Security

During the operation of the robot system, the safety of operators must be ensured at first. The following is a list of general precautions. Please take appropriate measures to ensure the safety of operators.

All operators using the robot system shall pass through the training sponsored by Shenzhen Han's Robot Co., Ltd. Users should ensure that they fully grasp the safety and standard operation procedure and have the qualification of robot operation. For training details, please contact our company according to the contact information shown on our official website

http://www.hansrobot.com

2. Operators using the robot system should not wear loose clothes or jewelry. Make sure that long hair is tied back when operating the robot.

3. In the operation of the equipment, even if the robot appears to have stopped, it may be in the state of imminent action because the robot is waiting for the start signal. Even in this state, the robot should be considered in motion.

4. Lines should be drawn on the floor to mark the movement range of the robot so that the operator can understand the movement range of the robot including holding tools (manipulator, tools, etc.).

5. Ensure that safety measures (e.g. guardrails, ropes or protective screens) are in place near the operation area of the robot to protect the operators and the surrounding crowd. Locks should be provided as required to prevent access of non-operators to the robot's power supply.

6. When using the operation panel and the demonstrator, be sure to carry out the operation after taking off the gloves, for the operation error may occur if wearing gloves.

7. In emergency and abnormal situations, for example, a person is clamped or surrounded by a robot, use a force(at least 500 N) to push or pull the robot arm, forcing the joint to move. To manually move the robot's arms without electric drive is limited to emergencies and can do damage to joints.

Responsibility and Specifications

Elfin series robots can be combined with other equipment to form a complete machine and the robot itself is not complete. Therefore, the information in this manual does not cover how to design, install and operate a complete robot, nor cover all the possibilities that may affect the safety of peripheral equipment of such a complete system. The safety of a full robot installation depends on how the robot is integrated. The integrator shall conduct risk assessment for the design and installation of the complete system in accordance with the national laws, regulations and safety specifications and standards. Risk assessment is one of the most important tasks that the integrator must complete. The integrator can refer to the following standards to carry out the risk assessment process. Risk assessment is one of the most important tasks that the integrator must complete. The integrator can refer to the following standards to carry out the risk assessment process.

Safety of Machinery-General Principles for Design-Risk Assessment and Risk Reduction (ISO 12100:2010).

Robots and Robotic Devices - Safety Requirements for Industrial Robots - Part 2: Robot Systems and Integration (ISO 10218-2:2011).

Technical Report of Industrial Robots and Robotic System - Safety Requirements and Task-Type Risk Assessment Methods (RIA TR R15.306 - 2014).

Safety of Machinery-General Requirements and Risk assessment (ANSI B11.0-2010).

The integrator of the Elfin robot shall fulfill, but not be limited to, the following responsibilities:

Make a comprehensive risk assessment of the complete robot system;

Confirm the design and installation of the whole system is correct;

Provide the training to users and staffs;

Create operation specifications of the complete system, and make clear the use process instructions;

Establish appropriate safety measures;

Use appropriate methods during final installation to eliminate or minimize all hazards to an acceptable level;

Communicate residual risk to end users;

Mark the integrator's logo and contact information on the robot;

Put relevant technical documents on file.

For information on applicable standards and legal guidelines, please visit the website of www.hansrobot.com

All safety information contained in this manual shall not be deemed as a guarantee of Shenzhen Han's Robot Co., Ltd., and personal injury or equipment damage caused by the operator may occur even if all safety instructions are complied with.

Shenzhen Han's Robot Co., Ltd. is committed to continuously improving the reliability and performance of products, and therefore reserves the right to upgrade products without prior notice. Shenzhen Han's Robot Co., Ltd. strives to ensure the accuracy and reliability of the contents of this manual, but is not responsible for any error or omission of information therein.

Hazard Identification

The risk assessment should take into account all potential contact between the operator and the robot during normal use as well as foreseeable misoperations. The operator's neck, face and head should not be exposed to avoid collision. If a robot is used without peripheral safety protection devices, a risk assessment shall be carried out at first to determine whether the associated hazards pose unacceptable risk, for examples:

1. The use of a sharp end-effector or tool connector may be hazardous;
2. There may be hazards in handling toxic or other harmful substances;
3. The operator's fingers may be clamped by the base or joint of the robot;
4. Risk of collision by a robot;
5. Hazard caused by that the robot or the tool connected to the end is not fixed in place;
6. Hazard caused by impact between robot payload and solid surface.

The integrator must measure such risks and its associated risk levels through risk assessment, and determine and implement appropriate measures to reduce the risks to an acceptable level. Note that there may be other significant hazards associated with specific robotic devices.

The risks associated with the collaborative operation of the Elfin series robots can be minimized to a reasonable and feasible level by combining the inherent safety design measures applied by the Elfin robots with the safety specifications or risk assessments implemented by the integrator and end users. Any residual risk that exists before the robot is installed can be communicated to the integrator and end users through this document. If identifying that there are risks in a particular application that may pose an unacceptable risk to the user in the risk assessment, the integrator must take appropriate risk mitigation measures to eliminate or minimize these risks until they are reduced to an acceptable level. It is not safe to use it until appropriate risk mitigation measures have been taken if necessary.

If the robot is installed in a non-collaborative manner (for example, when using a hazardous tool), the risk assessment may infer that the integrator needs to connect additional safety equipment (for example, safety boot equipment) in programming to ensure safety of personnel and equipment.

Intended Use

The following use statement is applicable to the case that the Elfin robot is used only as general industrial equipment, such as operating or fixing tools   or equipment, processing or transferring parts or products. Elfin robots are only permitted to be used under specified environmental conditions. For more information about the operating environment and operating conditions, see the appendix.

Elfin robots have the special security level features, collaborative operation can be performed, that is, the robots are used under the condition of no peripheral safety protection devices, but only

limited to the case without danger after a risk assessment, i.e. the expected or unexpected contact between the staff and Elfin robot or the end effector or components of the robot will not pose unacceptable risk, and the expected or unexpected contact between the staff and other objects (tools, equipment, surface, etc.) in the working area will not constitute unacceptable risk on premise that any safety protection device and the field sensors are not used.

The robot controllers and robots are only used as general industrial equipment and shall not be used in applications contrary to the intended uses, including but not limited to the following applications:

- used in inflammable and explosive dangerous environments;
- used as the device for moving or transporting people or other animals;
- used as medical equipment and other devices involving human life;
- used as the device with a significant social and public influence;
- used in vehicle, ship and other environment with vibration;
- used as climbing tools.

Emergency treatment

Emergency stop device

Pressing the emergency stop button will stop all movement of the robot. Emergency shutdowns may not be used as a risk reduction measure, but may be used as secondary protection device. If multiple emergency stop buttons have to be connected, then these buttons shall be included in the risk assessment of robotic applications. The emergency stop buttons comply with IEC 60947-5-5.

The Elfin series robots are equipped with the emergency stop buttons on the control cabinet and the demonstrator. The buttons must be pressed in dangerous or emergency situations, as shown in the figure below. The control cabinet is equipped with an external emergency stop button port which can be used by the integrator or user according to the actual situation.

If the tools or equipment connected to the end pose a potential threat, then they must be integrated into the system's emergency stop circuit; failure to comply with this warning may result in death, serious personal injury or significant property loss.

Reset from emergency state

All button-type emergency stop devices have "locking" function. This "lock" must be opened to end the device's emergency stop state.

Turn the emergency stop button to open the "lock".

To reset emergency stop state is a simple but very important step that can only be performed after ensuring that the hazard to the robot system is completely eliminated.

Forcible emergency movement of joint

In rare cases, it may be necessary to move one or more robot joints in an emergency of the robot power failure or no use of power supply. This can be

realized by:

Forced reverse drive: Use a force(at least 500 N) to push or pull the robot arm, forcing the joint to move.

Forcibly manual movement of robot's arms is limited to emergencies and may do damage to joints.

⚠️ **Excessive force safety protection of robot body**

The robot body has the function of excessive force safety protection. In the power-on and idle state of the robot body, the robot body will move passively along the direction of the collision force when the operator or other object accidentally collides with the robot body and the collision force exceeds the safety threshold. This function can reduce the damage to people, other objects and the robot body when the operator or other object collides with the robot body.

This function can reduce collision damage and requires risk assessment for other uses. For safety emergency handling.

Transportation and Precautions

When the robot is hoisted, appropriate measures should be taken for positioning the moving parts so as to avoid hazard due to accidental movement in the process of hoisting and transportation.

⚠️ During packaging and transporting, the product should be packed in accordance with the packaging standard and the necessary marks shall be applied on the outside of the packaging box. During transporting, the robot should keep stable and be fixed in the proper position. Handle should be used to lift the control cabinet. When moving the robot from its packaging material to the mounting position, hold the robot until all the bolts of the robot frame are fastened.

After fixing, power on the robot, adjust the robot attitude to the appropriate position by using the robot's dragging teaching function. Keep original packaging after transportation. Store packaging materials in a dry place in case you need to repackage and move the robot in the future.

⚠️ Make sure that your back or other body parts are not overly loaded when lifting the device.

All regional and national guidelines should be followed. Shenzhen Han's Robot Co., Ltd. shall not be responsible for the damage generated during the transportation of the equipment.

Ensure that the installation of the robot shall be implemented in strict accordance with the installation instructions in the manual.

Other Potential Risks

Note that there may be other significant hazards associated with specific robotic devices:

The fingers are clamped between the robot's legs and the frame.

The fingers are clamped between the large arm and the forearm.

Sharp edges and sharp points on the tools or tool connectors puncture the skin.

Sharp edges and sharp points on obstacles near the robot trajectory puncture the skin.

Injured by a collision with a robot.

Sprain or fracture due to the impact between the robot payload and a hard surface.

Consequences arising from loose bolts that hold the robot arms or tools.

Items fall from tools, e.g., due to inadequate clamping or power failure.

Operation error due to different emergency stop buttons on different machines.

Notice for Maintenance

All safety instructions in this manual must be strictly followed during maintenance.

Watch out

Please read the interface instructions of the Elfin electric cabinet before installing the electric cabinet.

The location of the electric control box and distribution cables should be as far away from the noise source as possible. Too close to the noise source may result in position deviation or misoperation.

Do not apply excessive force to the connector nor over-bend the cable. Failure to do so may result in poor contact or broken cable.

Please make sure the equipment is well grounded.

After connecting the cable, fix the cable with the accessory nylon clamp.

Please do not disassemble the electric cabinet by yourself, otherwise it may cause electric leakage and electric shock.

The electric cabinet is equipped with fans at both ends. Do not block them during installation to keep good ventilation.

Danger:

Remove the main input cable from the bottom of the electric cabinet to ensure complete power failure. Disconnect other energy sources from the robot arm or electric cabinet. Take necessary protection measures to prevent others from switching on the system energy source during maintenance period.

Check the ground connection before restarting the system.

Disassemble the robot arm or electric cabinet according to the ESD code.

Avoid disassembling the power supply system in the electric cabinet. The power supply system can retain high voltage (up to 600V) for several hours after the electric cabinet is shut off.

Prevent water or dust from entering the robot arm or electric cabinet.

After maintenance, it must be checked to guarantee the safety level required for the service. Check must comply with valid national or local safety laws and regulations. At the same time, all safety functions should be tested to see if they are working properly.

Elfin Software Operation Manual

Chapter 1 Introduction to Operating System

1.1 Introduction to Operation Interface

Han's Robot Demonstrator is a operation software for manual robot operation, programming, parameter configuration and user monitoring. The user can control the robot by pressing the buttons on the interface. The robot can move according to the path specified by the user to complete the expected actions. It can be said that the demonstrator is the steering wheel to control the movement of the robot and it includes the following modules:

Run: Robot start and running programmed script and locking robot

Programming: Modify or create script file

Monitoring: Monitor IO state of robot in operation, robot body state and log record;

Configuration: Set the initialization parameters of the robot, such as effective TCP, installation angle, safety settings, user coordinate system and so on;

Management: Administrator right setting and updating system;

Inching: Adjust the robot posture and action;

Menu: Login, Message, Help Center, Language Switch, IP Settings, Shutdown, Restart.

Logo indicator light: Yellow light for disabled state, green light for enabled state, purple light for running state, blue light for zero-force demonstration and red light for uninitialized/abnormal state

1.2 User Login

After the software starts, a login box pops up. Users with permissions can enter their account and password to log in. The permissions are divided into administrator and operator. The administrator has all the permissions; the operator has part of the designated permissions;unauthorized users will enter in the form of visitors and can only use the inching operation;
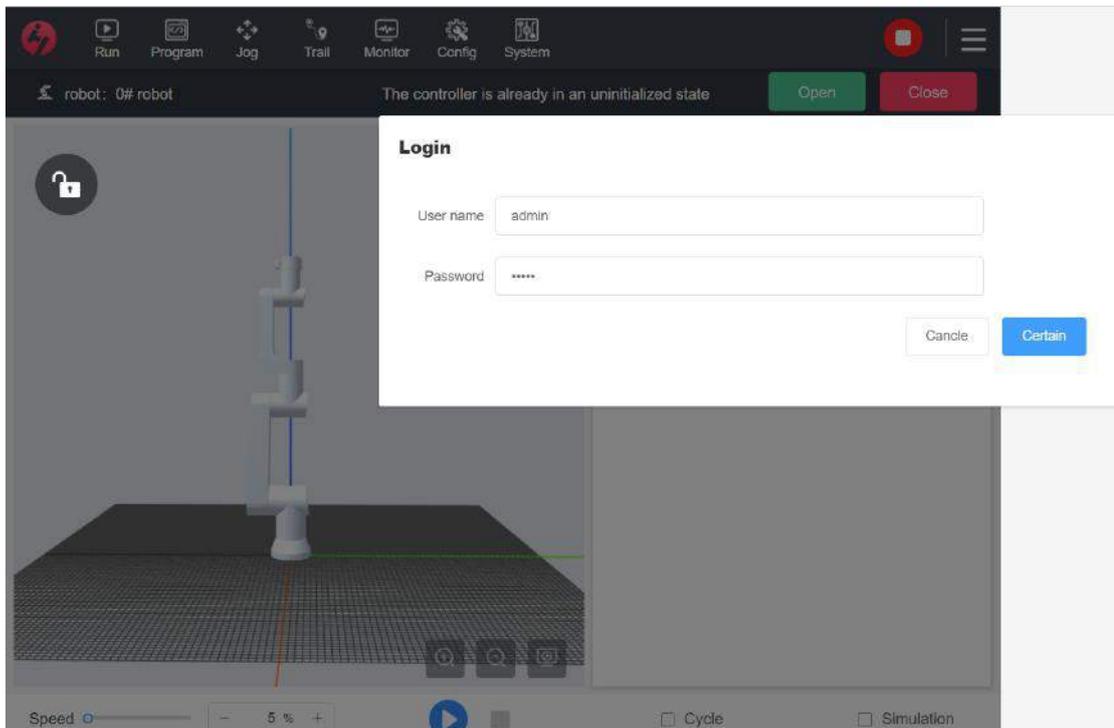
Permission type	Permission Description	Remark
Admin (administrator)	Has all permissions and is not limited;	Management of all accounts or passwords
Operator	Can use "Run, inching (except point list/save point), monitoring, menu at top and right corner"; and cannot operate other modules;	The administrator provides the login account and password
Visitor	Can use "Power on/enable, menu at top and right corner"; and cannot operate other modules	Does not need to log in

◆   After login, the user can operate the specific functions according to the permissions;

◆   If you forget account or password, please turn to the administrator for help;

◆   After login, the user can exit the account through the menu 【 ☰ 】 on the top;

◆ If you need to log in again after logout, you can log in again through the login button in the top menu;

◆ Visitors have no actual authority and refer to users failing to log in, and only can implement simple operations; visitors do not need to enter account and password to log in. They can click [Cancel] button or the blank space on the interface to cancel the login pop-up window;

◆ Administrators can configure permissions for specific users - "administrator" or "operator"

1.3 Initialization of Robot

After logging in, the robot is in the power-off state, so the robot needs to be powered on and enabled to move.
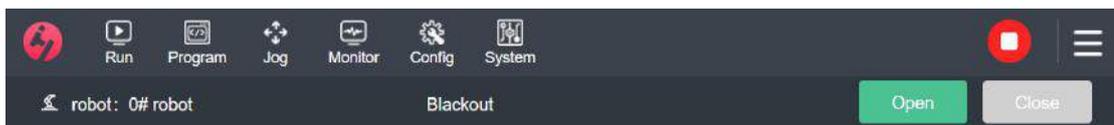
1) In consideration of safety, the robot is powered on after [Startup], but the joint braking is not released, so it cannot operate the robot to move; click [Close] to turn off the power supply of the robot;

2) After power on, the robot is in disabled state and the Logo turns yellow. Please check whether the setting information of the robot (parameters such as current payload, installation angle position, TCP and mechanical origin) is correct, and then [enable]; if incorrect, please modify it to the correct parameter. In case of setting error, the robot arm and the electric cabinet will not work properly and will pose a hazard to people or equipment.

3) Press [Enable] button to release the joint brake, at the moment, the robot starts and the

Logo turns green. With sound and slight movement, ensure that there are no obstacles around the robot to avoid damage to its body due to collision between the robot and obstacles; enabled robot can carry out normal operation; the user can now run the robot and program it;

1.4 Status Description

Robot process state description: There are corresponding descriptions of states from the start to the normal operation of the robot;

◆ After the successful start of demonstrator, the system displays a prompt of " the robot has been not powered on currently"; at this time, you can operate the "Power-on" function;

◆ In case of boot failure, there is the prompt of corresponding error and reason, you can check whether the hardware is disconnected according to the prompt;

◆ If it is still disabled after power-on, there is a description of " the robot is disabled currently" on the display, which means that the robot has been powered on but the joint braking has not been released;

◆ After it is enabled, there is a description of "the robot is enabled" on the display, which means that the robot joint braking has not been released and you can normally operate the robot to move;

◆ If there is an error in the process of execution, there is a description of "the execution error of robot", please solve the problem according to the pop-up prompt;

1.5 Top Menu
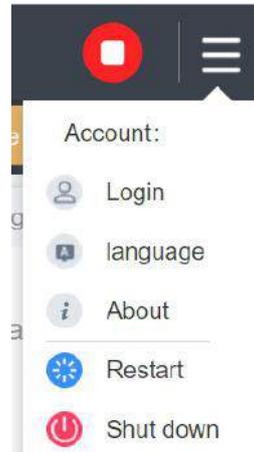
◆ **Exit:** If you have already logged in, there is a prompt of logging out the current account and the system enters the visitor mode; the user can log in again from the menu;

◆ **About:** To check the system version;

◆ **Restart System:** To restart the software; if there are files to be saved, then there is a prompt to remind the user to save the data before the system is restarted; the user can also choose to restart directly;

◆ **Close System:** To close the demonstrator; if there are files to be saved, then there is a prompt to remind the user to save the data before the system is closed; the user can also choose to close it directly;

Chapter 2 Inching

This section mainly describes how to control the robot to move to the desired position. This section can be found in the inching operation teaching interface. The inching operation teaching interface is mainly used for basic operations of robot movement and has the teaching functions for the robot to perform angular movement, spatial movement and traction.
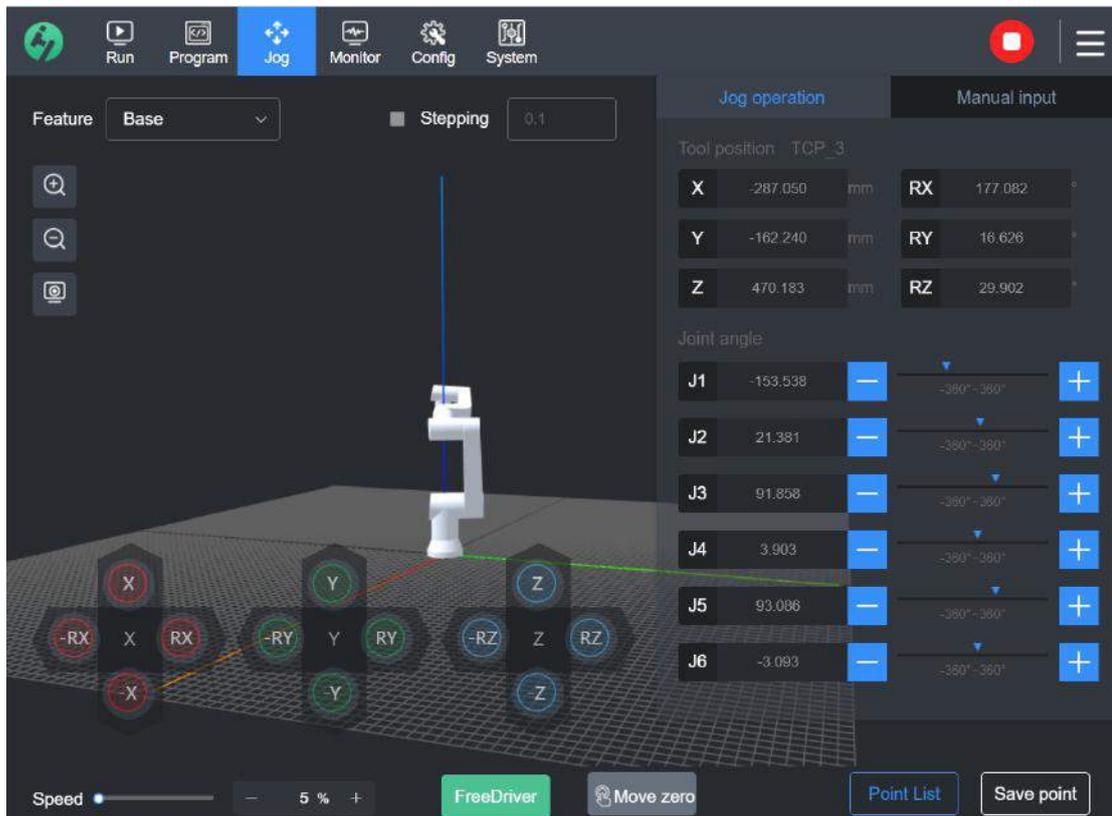
2.1 Coordinate System
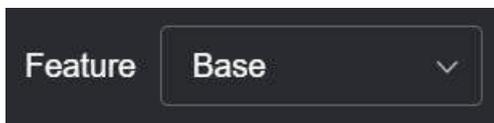
The coordinate system currently used to control the robot is defined and can be switched in the following coordinate modes: base coordinates, user coordinates and tool coordinates. If the wrong coordinate system is chosen, the robot may have unanticipated actions, thus causing unpredictable hazards.

2.2 Fine Adjustment
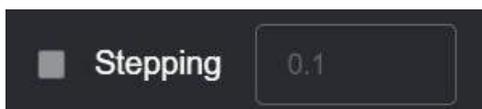
This function only takes effect when "Motion offset" option is checked. After this fine-adjustment option is checked, attitude adjustment can only be carried out in one step.

For example, for fine adjustment of 0.1, the change value will be -0.100 by single click -J1; for single click -J1 twice, the value is -0.200; the fine adjustment range is 0.1~1.
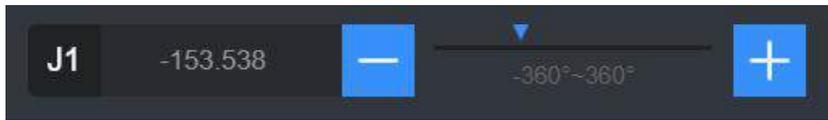
2.3 Speed Bar

The speed bar shows the current speed ratio of the robot and is limited by the safety limit level. Drag the circle on the display bar to adjust the speed ratio; click "+" or "-" button to adjust the robot speed ratio.   Adjustment range of speed ratio is 1%-100%. (The speed bar style may be adjusted according to the current interface color, but the functional data is consistent for the software as a whole)

2.4 Cartesian Workspace Motion
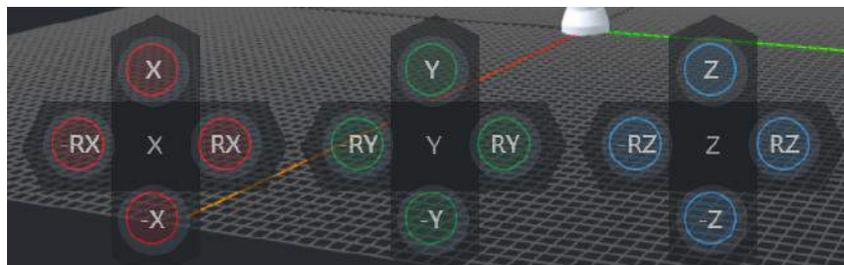
◆   Definition of movement in X, Y, Z -axis direction:

Within the safe space range, the robot end TCP moves **forward** in the axis direction, and it will **move forward by 2mm in the corresponding axis direction** for each click of the button. The robot end TCP moves forward in the corresponding axis direction when you press and hold the button; and it will stop moving when you release the button. If fine adjustment is checked, single click represents that the robot end moves forward by one fine-adjustment unit in the corresponding direction.

◆   Definition of movement in -X, -Y, -Z -axis negative direction:

Within the safe space range, the robot end TCP moves **reversely** in the axis direction, and it will **move reversely by 2mm** in the corresponding axis direction for each click of the button. The

robot end TCP moves reversely in the corresponding axis direction when you press and hold the button; and it will stop moving when you release the button. If fine adjustment is checked, single click represents that the robot end moves reversely by one fine-adjustment unit in the corresponding direction.

◆ Definition of forward rotation in RX, RY, RZ -axis direction:

RX represents the forward rotation of the robot end TCP around the X-axis; RY represents the forward rotation of the robot end TCP around the Y-axis; RZ represents the forward rotation of the robot end TCP around the Z axis. Within the safe space range, the robot end TCP rotates by 2° in the positive direction of the corresponding axis for each click of the button. The robot end TCP rotates constantly in the positive direction of the corresponding axis when you press and hold the button; and it will stop rotation when you release the button. If fine adjustment is checked, single click represents that the robot end rotates by one fine-adjustment unit in the positive direction of the corresponding axis.

◆ Definition of reverse rotation in -RX, -RY, -RZ -axis direction:

-RX represents the reverse rotation of the robot end TCP around the X-axis; -RY represents the reverse rotation of the robot end TCP around the Y-axis; -RZ represents the reverse rotation of the robot end TCP around the Z axis. Within the safe space range, the robot end TCP rotates by 2° in the negative direction of the corresponding axis for each click of the button. The robot end TCP rotates constantly in the negative direction of the corresponding axis when you press and hold the button; and it will stop rotation when you release the button. If fine adjustment is checked, single click represents that the robot end rotates by one fine-adjustment unit in the negative direction of the corresponding axis.

◆ Tool Position
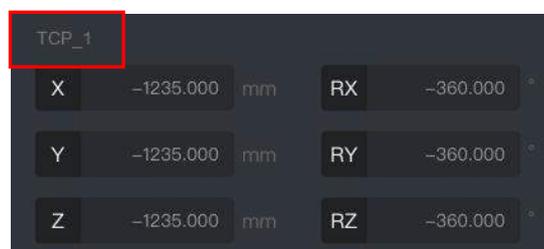
TCP: The interface displays the name of the tool coordinates currently in use. As shown in the figure above, the tool coordinates currently in use are TCP_1;

X, Y, Z, RX, RY and RZ respectively represent the spatial position vectors of the robot based

on the current tool coordinates.

As shown in the figure above, the values displayed in the text box represent the space coordinate values of the current robot terminal TCP relative to the currently selected feature coordinate system, in mm and ° for X, Y and Z- axis and RX, RY and RZ-axis, respectively.

Note: The spatial coordinate movement of the inching interface corresponds to the MoveL instruction and the PCS movement mode in the script programming.

2.5 Joint Movement

J1-J6 represent the axes from the first to the sixth of the robot. The text box displays the angle coordinate value of each corresponding axis position of the current robot in ° (angle).
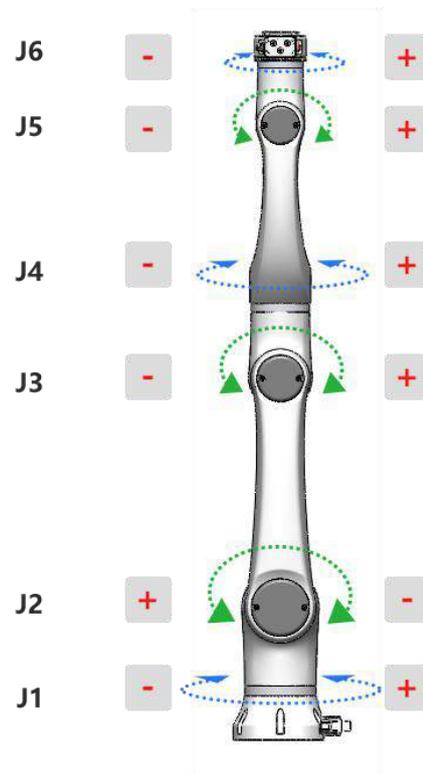
◆ **Joint Negative Direction** ▬

Within the safe space, the robot moves in the negative direction of the corresponding axis, and the text box displays the current position of the corresponding axis. The robot moves by 2° in the negative direction of the corresponding axis for each click of the button. The robot moves constantly in the negative direction of the corresponding axis when you press and hold the button; and it will stop moving when you release the button.If fine adjustment is checked, the robot moves by fine-adjustment distance input the user in the negative direction of the corresponding axis.

◆ **Joint Forward Direction** ＋

Within the safe space, the robot moves in the forward direction of the corresponding axis, and the text box displays the current position of the corresponding axis. The robot moves by 2° in the forward direction of the corresponding axis for each click of the button. The robot moves forward in the corresponding axis direction when you press and hold the button; and it will stop moving when you release the button. If fine adjustment is checked, the robot moves by fine-adjustment

distance input the user in the forward direction of the corresponding axis.

Note: The joint angular movement here corresponds to the MoveJ instruction and the ACS movement mode in the script programming.

2.6 Manual Input
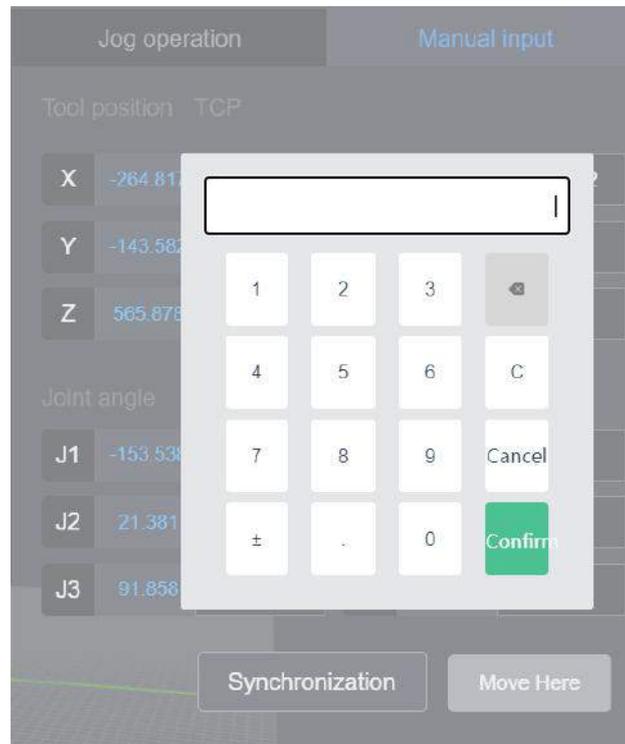
Click "Manual Input" button to manually enter the position coordinate value of the known spatial coordinate point or joint angle value. After entering the value, press and hold the "Press and Hold to Point" button so that the robot moves directly to the target point; it will stop moving if you release the button:

◆ **Joint Angle:** Input the desired value in the joint text box and then press and hold the "Press and Hold to Point" button;

◆ **Tool Position:** Input the desired value in the tool position text box and then press and hold the "Press and Hold to Point" button;

◆ **Press and Hold to Point:** When the button is pressed and held, the robot moves to the target point, at the moment, the button is disabled and there is a corresponding prompt of "The robot has arrived at this point"

◆ **Back to Initial Value:** The data in the text box will synchronize with the actual location value of the current robot;

◆ **If the current input value that the robot cannot reach after being calculated, then there is an error prompt of " The robot cannot reach, please input again"; at the moment, the user needs to input a value again;**

◆ **If the result of the current input value is normal, it is beyond the safety limit in the**

process of【 Press and Hold to Point】, there is a prompt of "Beyond the safety limit" and a reset prompt (Please see Section 8.3 Beyond Safe Space for details).

2.7 Zero-Force Demonstration

"Zero-Force Demonstration" button is green and will be red when enabled;
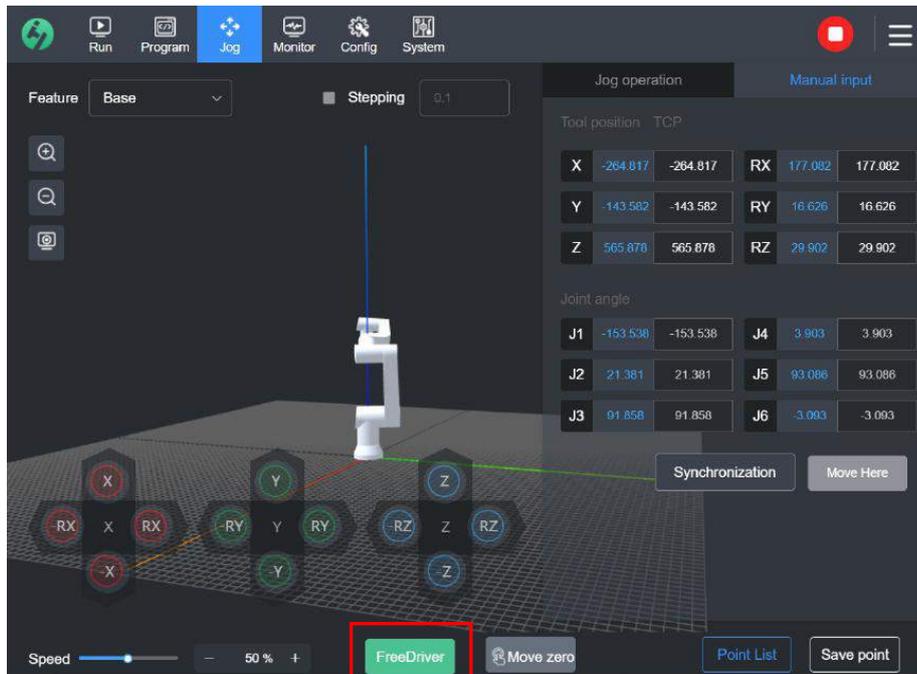
Make sure that the origin, load, center of gravity and installation angle of the machine are correct before enabling zero-force demonstration. If a parameter is wrong, the zero-force

demonstration will be automatically closed immediately after enabled and there is an error prompt.

After the zero-force demonstration is enabled, the button turns red [End Teaching], and the robot can be dragged freely. When the robot approaches the edge of the safe space or the safe limit, it will feel a force in the opposite direction, and if it continues to move towards the safe limit, it will cause the robot error. Therefore, please drag it within safe limits.

Click "End Teaching" button to exit the zero-force demonstration state.

2.8 Point List
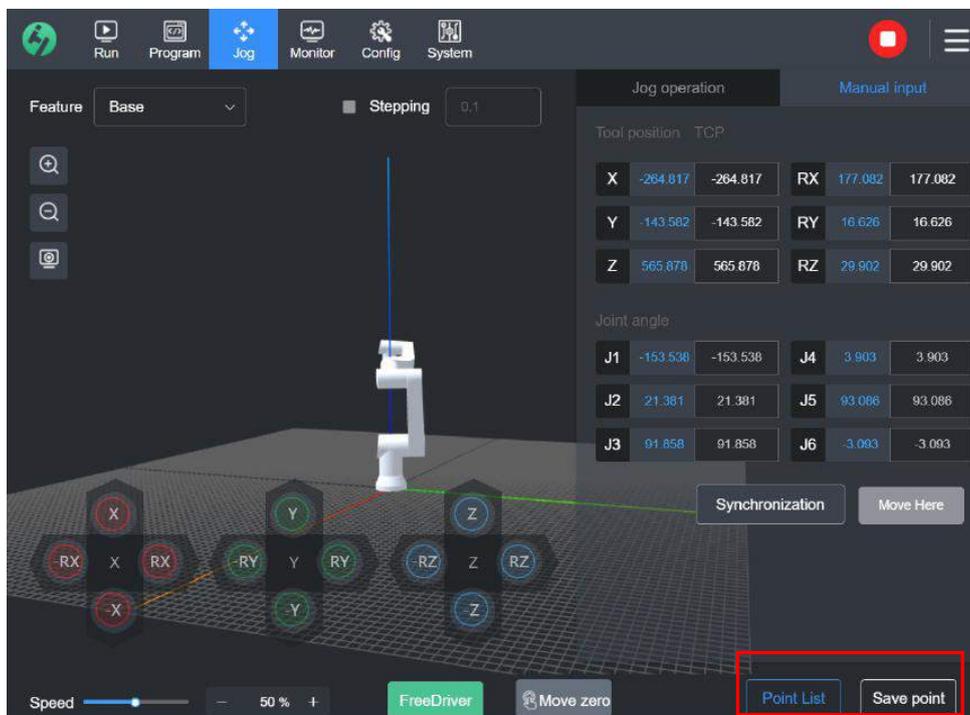
2.8.1 Point Application

In the script, there may be multiple waypoints using the same point position, so the point position can be saved in advance for direct invocation. If the position needs to be modified, the user only needs to modify the corresponding point in the point list so that all the points called in the script can be synchronously modified, which greatly increases the efficiency of point position reuse.

2.8.2 Save Point Position

Click "Save Point" button to save the current position of the robot in the point list. The point name will be saved according to the preset of the system. If it needs to be modified, it can be edited in the point list;

2.8.3 Point List
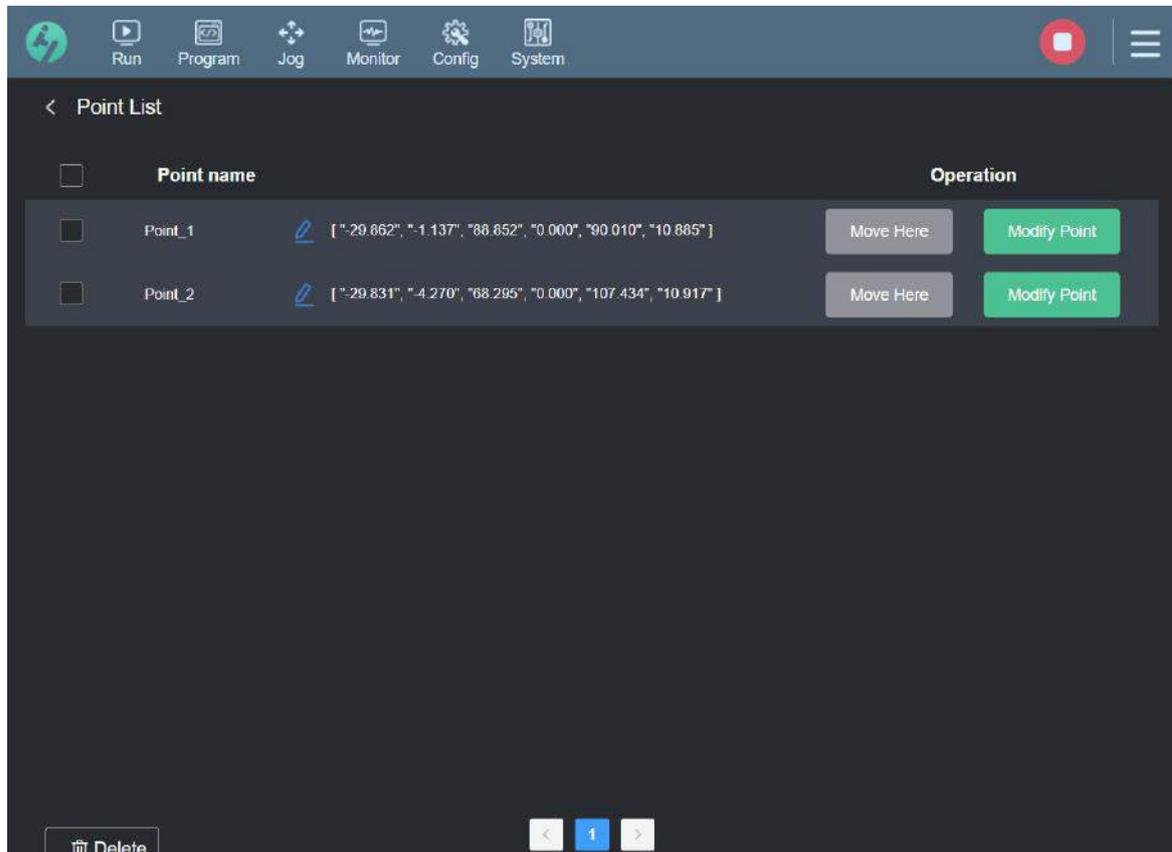
The saved points are recorded in the point list and synchronized with the script when used.

◆ **Rename:** Click the icon to modify the name of point;

◆ **Press and Hold to Point:** When the button is pressed and held, the robot moves to the point and it is convenient to check the specific location of this point;

◆ **Modify Point:** After this button is pressed, the interface jumps to the inching interface where the point can be modified and it means the successful modification after saving and then the waypoint of this point shall be modified synchronously;

◆ **Delete:** Check the block in front of the point and click "Delete" button;

Note: If a point used by a waypoint is deleted from the point list, then the point list in the

waypoint shall be deleted synchronously and the point information of the waypoint shall be cleared.

2.9 Return to Zero

Move zero
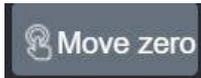
When you press and hold the "Automatic Return to Zero" button, then the robot body moves to zero position posture (vertical state) from the current posture in way of joint angle movement.

Note: Please make sure whether there are obstacles around the robot, for the robot may collide with obstacles in the path of automatic return to zero, therefore, you can let the robot to move to the safe position and then back to zero.

2.10 3D Model
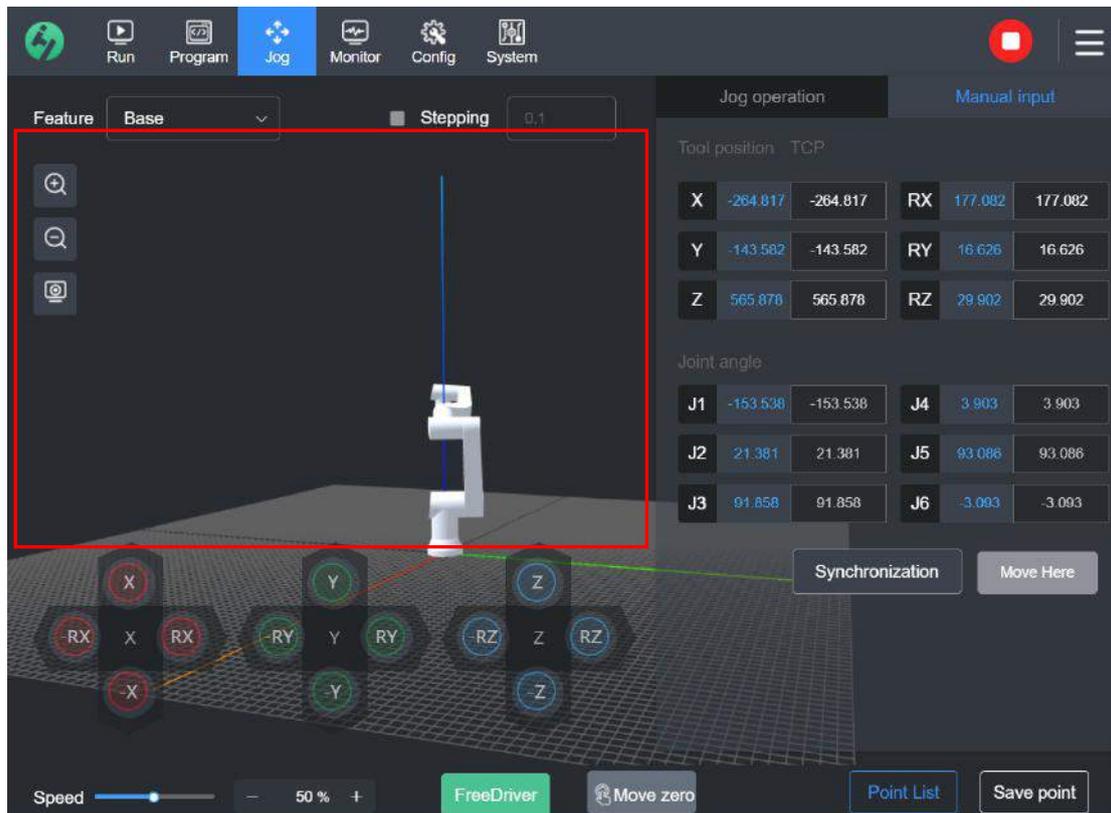
Simulate real robot posture, display it to the interface; the user can control the model display angle and size with two fingers; or by clicking the icon;

🔍	Zoom in to view the model
🔍	Zoom out to view the model

⬚	Back to the standard size of robot model
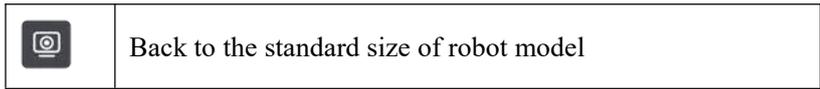
Chapter 3 Configuration

3.1 Setting TCP

The tool coordinate (TCP) is the center point of the robot tool. Each TCP contains transformation and rotation set relative to the center of the tool output flange.

The X, Y, and Z coordinates determine TCP's position, while the RX, RY, and RZ coordinates determine its direction. When all values are zero, the center point of TCP coincides with that of the tool output flange, as shown in the figure below:
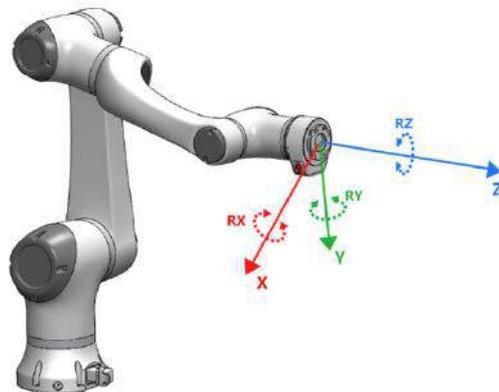
Set the tool coordinate system: The tool coordinate system can be set by directly inputting the tool coordinate parameter value or by way of manual teaching;
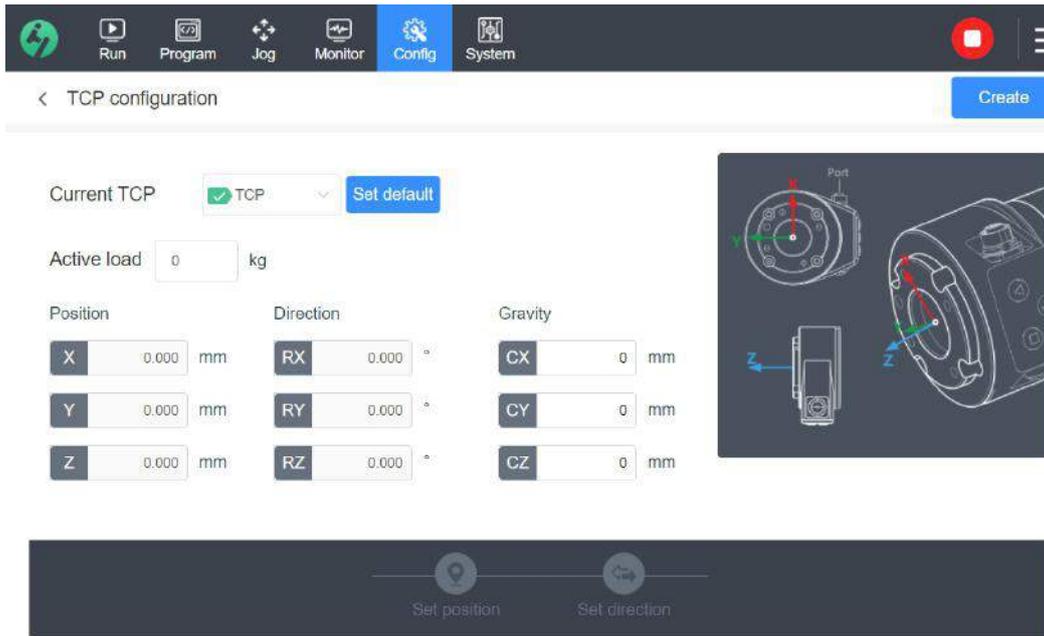
3.1.1 New TCP

When the user clicks "New TCP" button,a new TCP window pops up. The system has a preset name, and the user can also customize the TCP name. After the user clicks [OK] button and it takes effect, the newly created TCP will be displayed in the drop-down box of the current TCP. The default data in the input box is 0.

3.1.2 Position, Direction and Center of Gravity
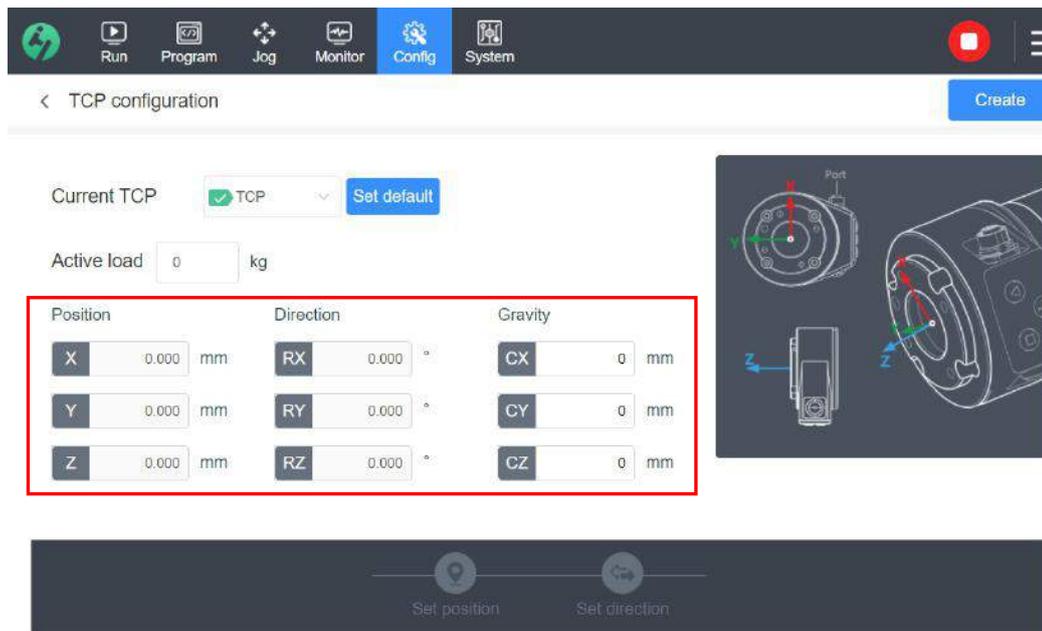
When knowing the offset of the new TCP relative to the center of the flange (measure or view

the size/ specification), the user can enter the data manually and then click [Apply] button so that it takes effect.

TCP_1=[x1,0,z1,0,0,0]
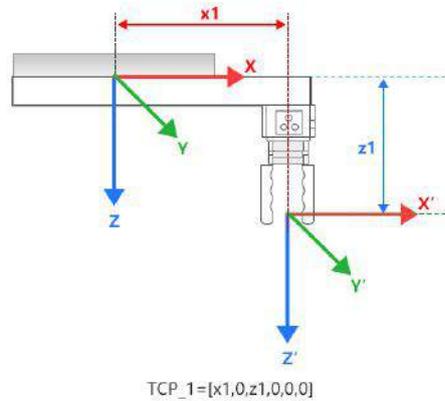
3.1.3 Set as Default

Only one of the configured TCP is the default TCP. In the TCP drop-down menu, there is a green icon to the left of the name of the TCP that is set as the default. Click "Set as Default" button to set the currently selected tool coordinates to the default TCP.

The default TCP name is displayed in the TCP position of the inching interface.
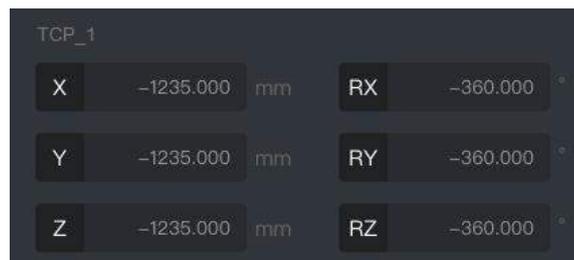
3.1.4 Set Position

The position of the origin of the new cutter coordinate system is determined by four teaching points.

Four-point method teaching principle: The end point of the cutter should be moved to the same position from four different angles for teaching.
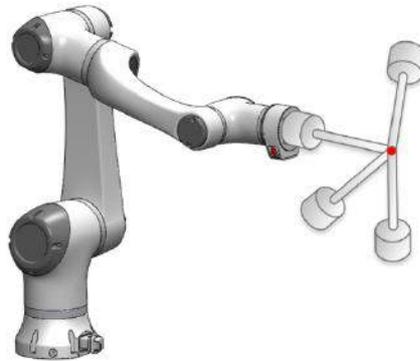
Is cutter coordinate system changed to tool coordinate system?

Is the name more regular?

Examples

1) Click "New" button to generate a new tool coordinate system TCP_1;

Current TCP   TCP_1   ∨

2) Click "Set Position" button to enter the four-point demonstration interface; move the center point of the robot cutter to the same position from different angles according to the prompts; the top right corner of the tutorial box displays the prompts during setup.
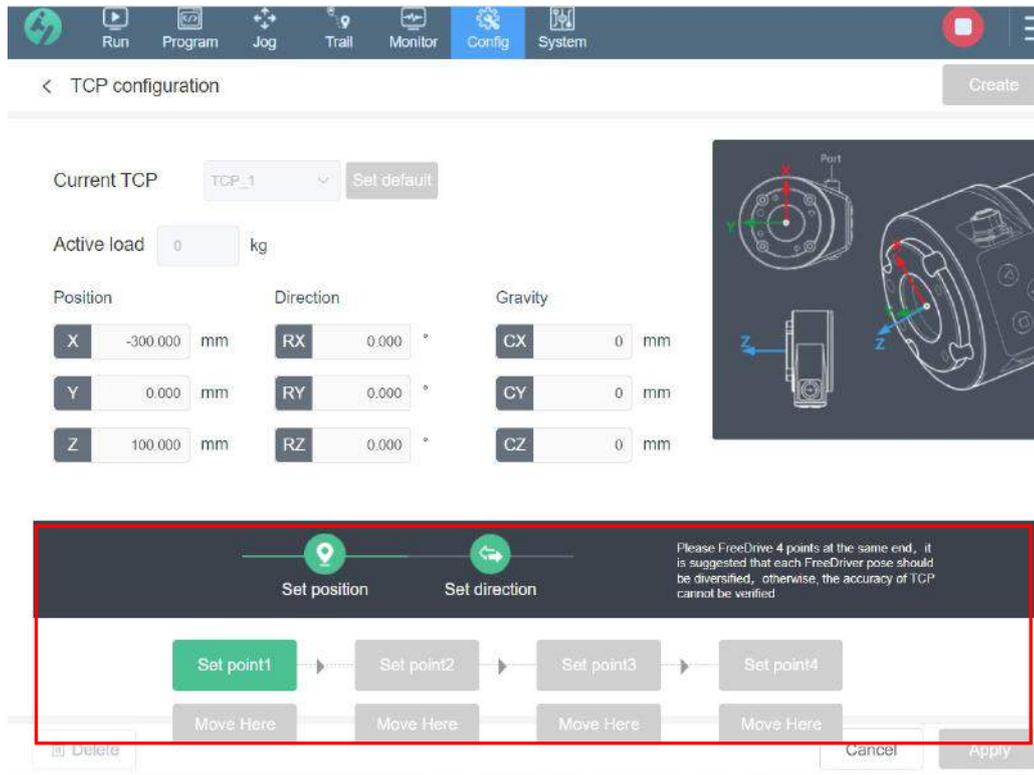
3) After the first three points have been taught, the point button turns green, indicating that the points meet the standards; at the same time, the upper right corner shows the prompt of the calculation results, indicating available;

4) After the first three points have been taught, the point button turns yellow, indicating that the point quality is general; at the same time, the upper right corner shows the prompt of the calculation results, indicating available; the user can decide whether to re-teach the point according to the actual accuracy requirement;

5) After the first three points have been taught, the point button turns red, indicating that the points are unqualified; at the same time, the upper right corner shows the prompt that the points are unqualified so the position cannot be calculated. The user needs to modify the point position again to make the button turn green or partially yellow before setting it successfully;

6) The user needs to continue teaching the fourth point to verify TCP's accuracy. When all the points are green or partially yellow, please click [Apply] button to complete the teaching position;

7) Click "Cancel" button to exit the teaching process.

Note: The teaching points must be diversified enough to ensure accuracy of calculated results.

3.1.5 Set Direction

When the direction of the tool is inconsistent with the default TCP direction, we need to define the TCP direction. TCP direction is set based on a coordinate system as a reference;
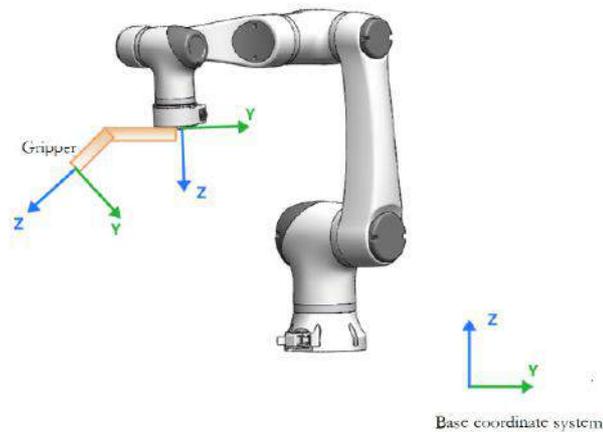
There are two ways to get the value of TCP position, one is the measurement method, the other is the teaching calculation method of the demonstrator.

3.1.5.1 Measurement Method

The Rx, Ry and Rz values of clamping jaw TCP are measured as follows.
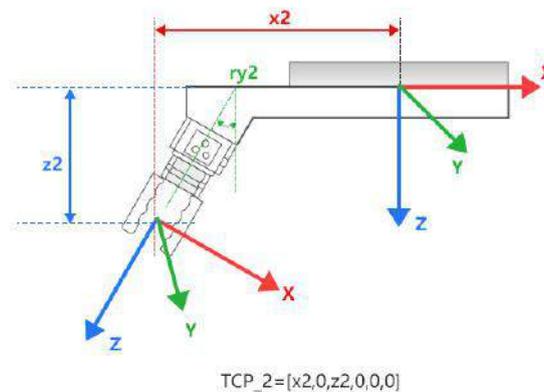
Plane XOZ of the initial TCP coordinate system rotates by an angle of "a" around the Y-axis of the initial TCP of the robot, and it can reach Plane XOZ of the clamping jaw TCP coordinate system, so Ry= a of clamping jaw TCP. and similarly, you get Rx, Rz.

3.1.5.2 Teaching Calculation Method

1) Click "Set Direction" button.
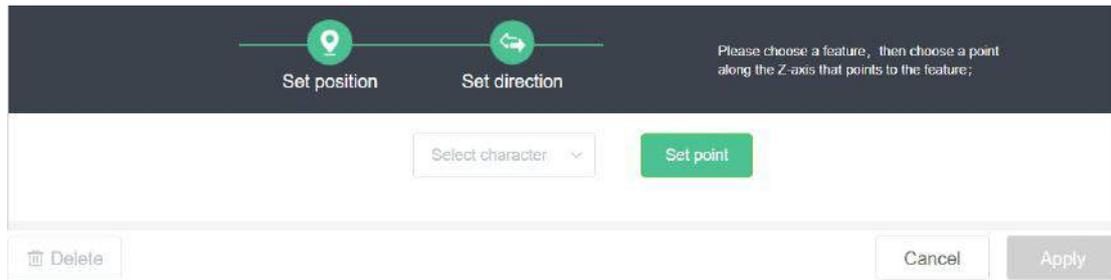
2) According to the prompt, select a coordinate system from the drop-down list. For example, the "Plane_1" coordinate system indicates that the direction of the new TCP_ is parallel to that of the "Plane_1" coordinate system;

3) Click the teaching point [Set Point] button to teach a point to make the direction of the end tool parallel to the Z-axis direction of the selected coordinate system to complete the demonstration in the TCP direction, and the upper right corner displays the prompt of calculated results.
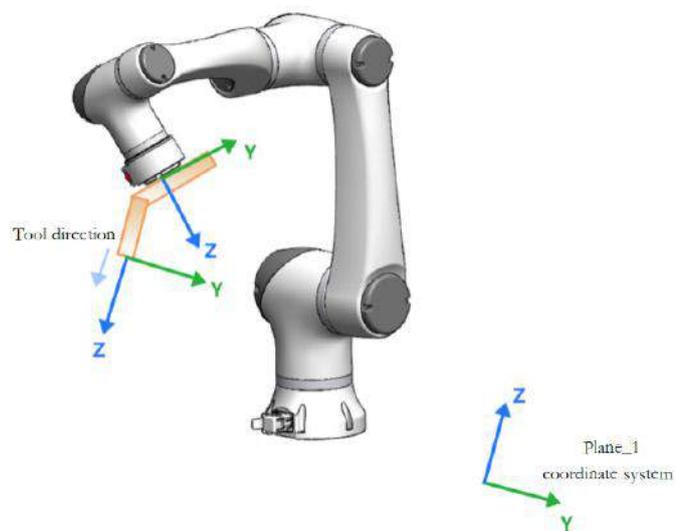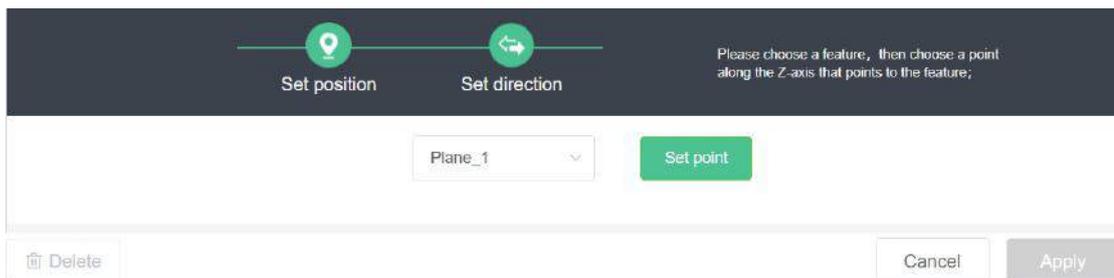
4) Click "Apply" button so that the data takes effect;

3.1.6 Payload

The user can set it according to the quality of the load installed at the end of the Elfin robot and its offset in the X, Y, and Z directions relative to the flange center point. Click the corresponding input box and input the correct value through the pop-up virtual keyboard, and then click "OK" button. For example, a 5kg robot's load is set in the range of [0~5]kg.

3.1.7 Center of Gravity

Use CX, CY, and CZ to specify the tool's center of gravity. If not specified, the center of gravity of the tool is 0. These settings are applicable to all defined TCPs.

3.1.8 Others

◆ **Delete:** Delete the tool coordinates selected in the current TCP.

◆ **Cancel:** Cancel the unsaved teaching process or the entered values.

◆ **Apply:** All edited data can take effect only after the 【Apply】button is pressed.

Note: If the TCP being used by a waypoint is deleted, the data in the waypoint is deleted synchronously; the user can re-select TCP for waypoints.

3.2 Mounting Angle of Base
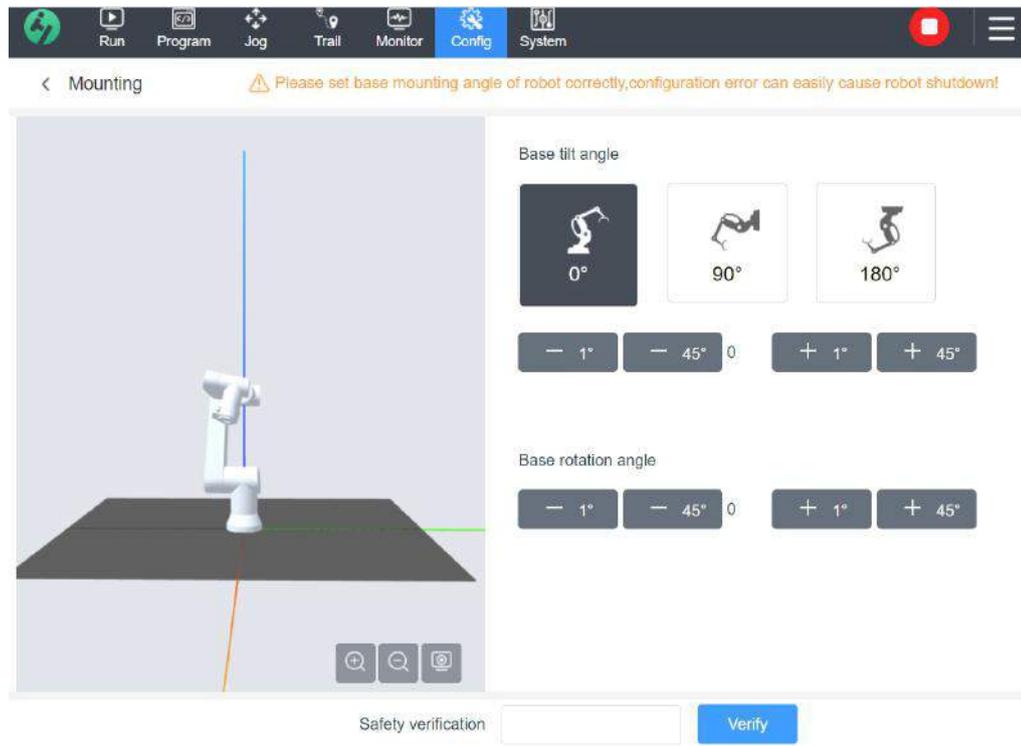
On this interface, the user can configure the mounting position and angle of the robot and should enter a security password before operation.

By default, the robot arm is mounted on the ground or the smooth worktable parallel to the ground; in such case, no change to this interface is required. If the robot arm is ceiling suspended, wall-mounted or installed at an angle, it needs to be adjusted through the button in the interface.

Warning: Be sure to use the correct installation settings. If the installation information of the robot arm is not set correctly, it will lead to frequent safe start/stop, or the robot arm may be weightless and fall down immediately after the "Zero-Force Demonstration" button is pressed.

The controller uses advanced dynamics model to ensure the robot arm moves smoothly and accurately, and supports itself when the robot arm is in zero-force demonstration mode. Therefore, it is essential that the angle and position of the robot arm should be set correctly.

3.2.1 Inclination Angle of Base

Common angles include ceiling suspended (180°), wall-mounted (90°) and ground (0°).

Other angles of inclination can be adjusted by ±1° (press and hold/single click) or ±45° (single click); ground (0°) by default.

Angle of inclination: Rotate around the Y-axis of the Base coordinate, and counterclockwise is positive;

3.2.2 Rotation Angle of Base

It can be adjusted through ±1° (press and hold/ single click), ±45°(single click). 0° by default.

Rotation angle: Rotate around the Z-axis of the Base coordinate, and counterclockwise is positive.

3.3 Set IO

Refers to the interface for configuration of input and output IO. Input IO includes universal input, configurable input, end tool input and analog input. Output IO includes universal output, configurable output, end tool output and analog output.

Among them, "Configurable Input or Output" IO can set the function and name; other IO can only remark names.

3.3.1 Configurable IO Input

CI0~CI7 are configurable input IO and can be used to set the input IO signal to trigger corresponding actions including "Enable", "Clear Error", "Script Pause", "Start Zero-Force Demonstration", "Stop", "Continue Script" and "Run Script".

Note: Each function can only be set with one IO, and multiple IO cannot be set to the same function at the same time;
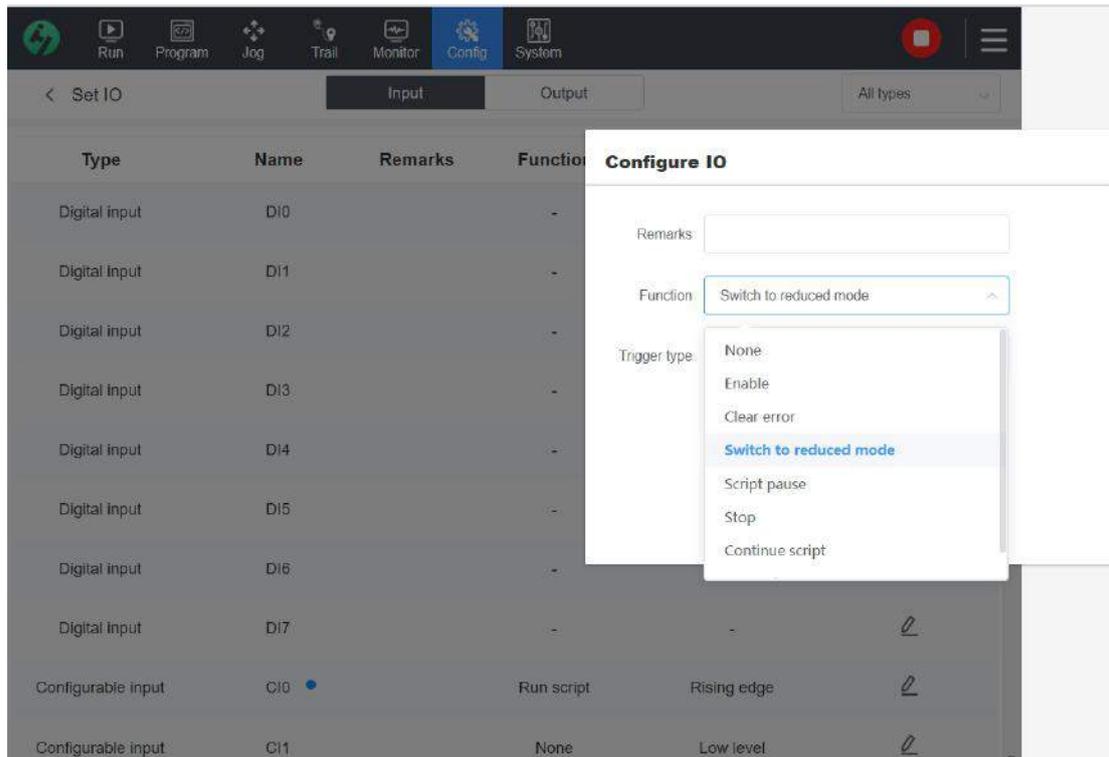
- ◆ Enable: Trigger type can be rising edge or fall edge; after this configurable function is enabled, the controller continuously monitors the IO state and issues the "Enable" command when meeting the selected trigger conditions.

- ◆ Clear Error: Trigger type can be rising edge or fall edge; after this configurable function is enabled, the controller continuously monitors the IO state and issues the "Clear Error" command when meeting the selected trigger conditions.

- ◆ Switch to Reduced Mode: Trigger type can be rising edge or fall edge; after this configurable function is enabled, the controller continuously monitors the IO state and switches to the reduced mode when meeting the selected trigger conditions.

- ◆ Script Pause: Trigger type can be rising edge or fall edge; after this configurable function is enabled, the controller continuously monitors the IO state and issues the "Scrip Pause" command when meeting the selected trigger conditions.

- ◆ Stop: Trigger type can be rising edge or fall edge; after this configurable function is enabled, the controller continuously monitors the IO state and issues the "Stop" command when meeting the selected trigger conditions.

- ◆ Continue Script: Trigger type can be rising edge or fall edge; after this configurable function is enabled, the controller continuously monitors the IO state and issues the

"Continue Script" command when meeting the selected trigger conditions.

◆ Run Script: Trigger type can be rising edge or fall edge; after this configurable function is enabled, the controller continuously monitors the IO state and issues the command of running "Func_main" function and starting the timer. To use the function to run the script, the user needs to write the function script named under Func_main, compile the script and switch to script mode before running the script through IO.

For example, CI0 is configured to enable "Zero-Force Demonstration" function and "high level". When CI0 inputs high level signal, the zero-force teaching is enabled; when CI0 inputs low level signal, the zero-force teaching is closed.

3.3.2 Configurable IO Output

CO0~CO7 is configurable output IO and can be used to set the output IO signal to reflect the current state of the robot. After these states are enabled, the robot is in output high state by default.
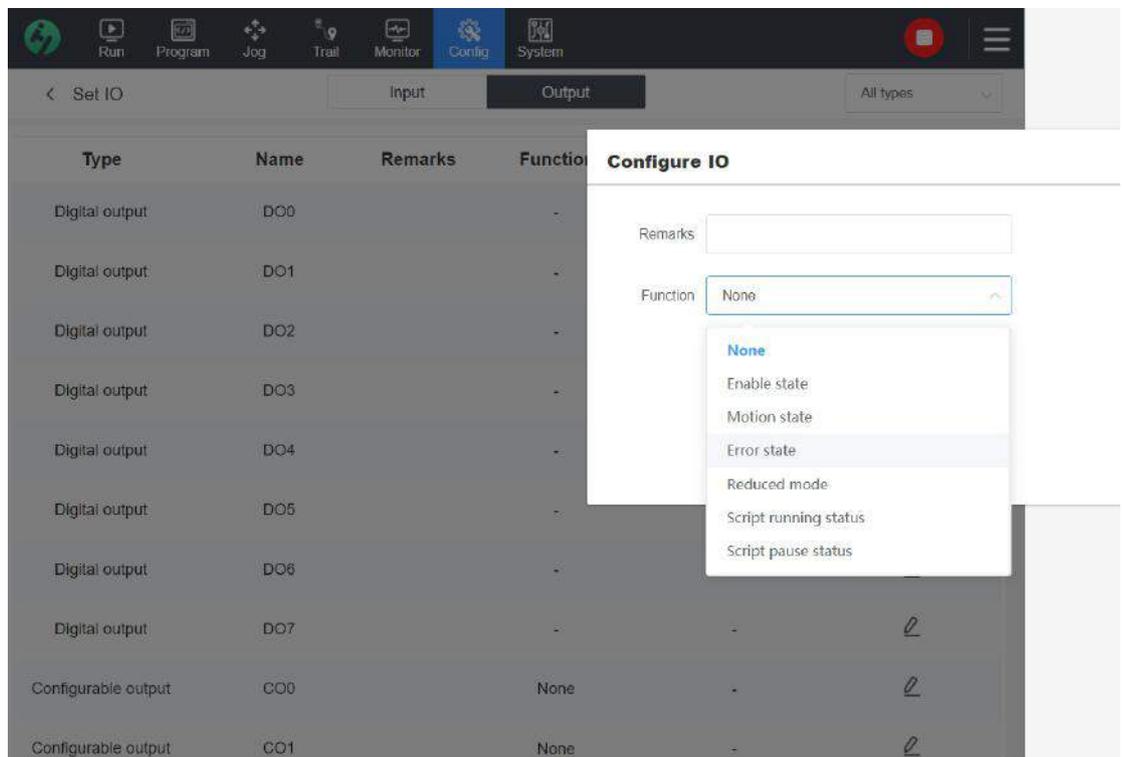
◆ Enabled State: After this configurable function is enabled, the controller continuously monitors whether the robot is in enabled state and outputs the high level signal when

meeting the conditions.

◆ Moving State: After this configurable function is enabled, the controller continuously monitors whether the robot is moving and outputs the high level signal when meeting the conditions.

◆ Error State: After this configurable function is enabled, the controller continuously monitors whether the robot is in error state and outputs the high level signal when meeting the conditions.

◆ Reduced Mode: After this configurable function is enabled, the controller continuously monitors whether the robot is in reduced mode and outputs the high level signal when meeting the conditions.

◆ Script Running State: After this configurable function is enabled, the controller continuously monitors whether the robot is in script running state and outputs the high level signal when meeting the conditions.

◆ Script Pause State: After this configurable function is enabled, the controller continuously monitors whether the robot is in script pause state and outputs the high level signal when meeting the conditions.

For example, CO4 enables the function of "Moving State". When the robot is moving, CO4 outputs a high level signal. If not in motion, CO4 outputs low level signal.

3.3.3 Edit IO Parameters
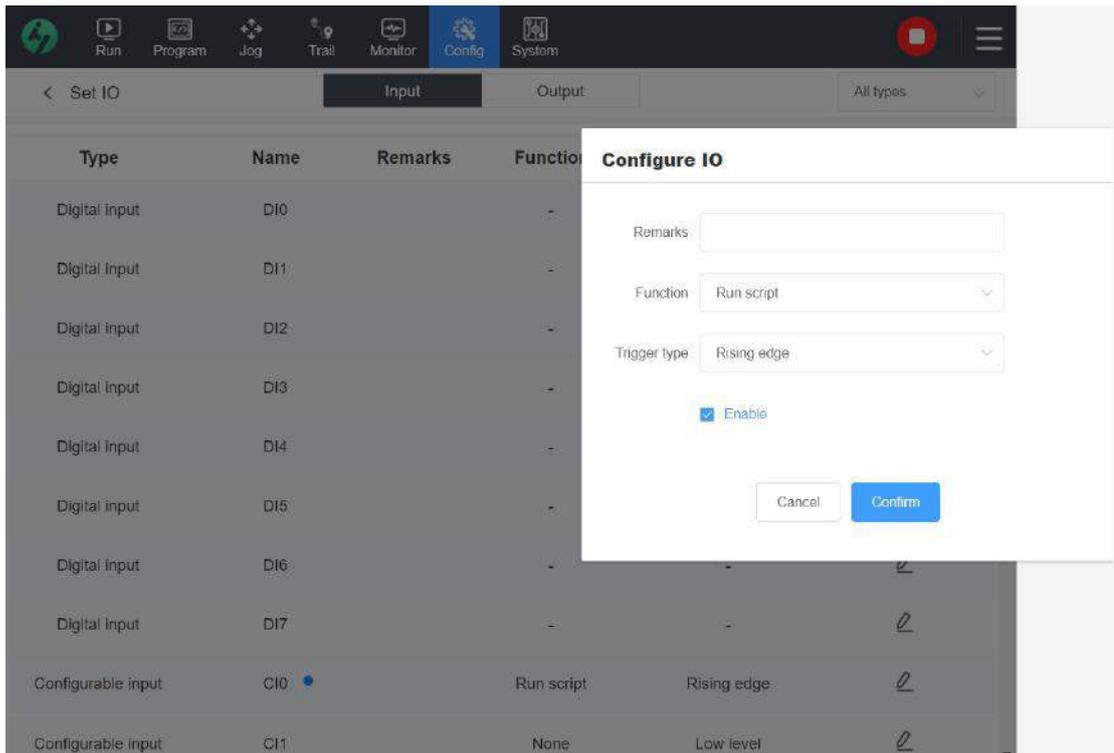
◆ **Note:** Refers to the note information on input and output signals for being easy to remember each signal. Click "Edit" button to modify the information.

◆ **Function:** Only limited to the configuration of configurable IO (input CI0 ~ CI7, output CO0 ~ CO7) to trigger operation actions;

◆ **Trigger Type:** only decided by trigger action set by configurable input IO; match "rising edge" or "falling edge", or "high level" or "low level" according to the corresponding actions;

◆ **Enable:** Refers to the function of configurable IO and is not checked by default; when the "Enable" option is checked, the corresponding function is activated to input and output IO information according to the selected function. If not checked, there is no difference between configurable IO and generic IO. There is a small blue circle superscript on the name of IO with this function enabled.

Examples

Configurable input IO: CI0 is configured to enable "Zero-Force Demonstration" function and "high level". When CI0 inputs high level signal, the zero-force teaching is enabled; when CI0

inputs low level signal, the zero-force teaching is closed. If "Enable" option is not checked, the function will not be activated.

Configurable output IO: For example, CO4 enables the function of "Moving State". When the robot is moving, the corresponding CO4 is at a high level. If not in motion, CO4 is at a low level. If "Enable" option is not checked, the function will not be activated.

3.4 Safe Limit

Configure settings such as safe collision threshold range, robot movement and speed range and security IO of the robot, and need to verify the security password before operation.

3.4.1 Normal Mode and Reduced Mode

Under normal conditions, the safety system will be operated in one of the following safety modes which have corresponding safety limit level settings:

Normal Mode: safety mode by default;

Reduced Mode: more stringent safe restriction based on the normal mode range. Tools and joints have more stringent speed and momentum limits in reduced mode than that in normal mode. Once activated, the robot arm's movement in joint space will be limited.

The user can set the safe limit level of normal and reduced modes on the safe limit interface.

How to trigger the reduced mode: It can be triggered when the position of the robot TCP beyond the **security boundary** or by using **a configurable input IO signal**.

3.4.2 General Limit

The parameters such as force, speed, power, momentum, elbow speed and elbow force of the robot in normal mode and reduced mode are different for different safety levels. Each parameter limit value for each safety level is displayed in a text box.

As shown in the figure above, after verifying the password, select the safety level, and the corresponding parameters of each safety level are displayed below, including: force, speed, power, momentum, elbow speed and elbow force. The user needs to perform risk assessment and select the appropriate safety level according to the production environment;

◆ Force: Refers to the maximum force applied by the robot tool in collision cases.

◆ Speed: Refers to the maximum speed of robot tool.

◆ Power: Refers to the maximum mechanical work done by the robot to the environment.

◆ Momentum: Refers to the maximum momentum of robot.

◆ Elbow speed: Refers to the maximum speed of the robot tools.

◆ Elbow force: Refers to the maximum force applied by the robot elbow to the environment.

"Low limit, high risk" means setting a small parameter limit; take Level 1 for example: When the robot is collided, it is relatively less sensitive and a great force is required to stop the robot. Low safety factor;

"High limit, high safety factor" means that a large parameter limit is set. Take Level 6 for example: When the robot is collided, it is relatively sensitive and only a small force can stop the robot.

There are 6 levels on the safety preset axis and the required limit is gradually increased from left to right.

Note 1: It is not necessary; do not choose the least limit level, for it cannot guarantee the safety of operators and robots.

Note 2: After setting, click "Apply" button so that it will take effect. Before exiting, please click "Lock" button.

3.4.3 Joint Limit

Joint limit is used to limit the motion of each robot joint in joint space, namely, joint motion range and joint motion speed. There are also limits in normal mode and reduced mode.

◆ Joint speed defines the maximum angular speed of each joint.

◆ Joint range defines the position range of each joint.
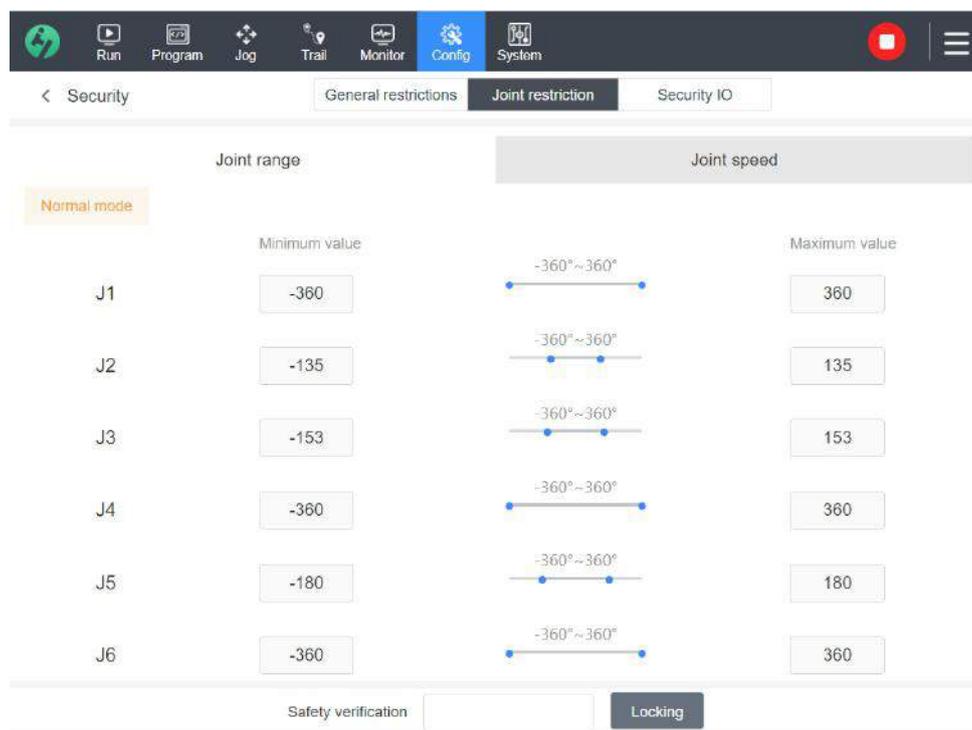
The factory setting range of joint angle of 5kg Elfin robot is: (Please refer to the attachment for configuration parameters of different models)

◆   J1 axis: ±360°;

◆   J2 axis: ±135°;

◆   J3 axis: ±153°;

◆   J4 axis: ±360°;

◆   J5 axis: ±360°;

◆   J6 axis: ±360°.

Note : After setting, click "Apply" button so that it will take effect. Before exiting, please click "Lock" button.

3.4.4 Safety Light Curtain
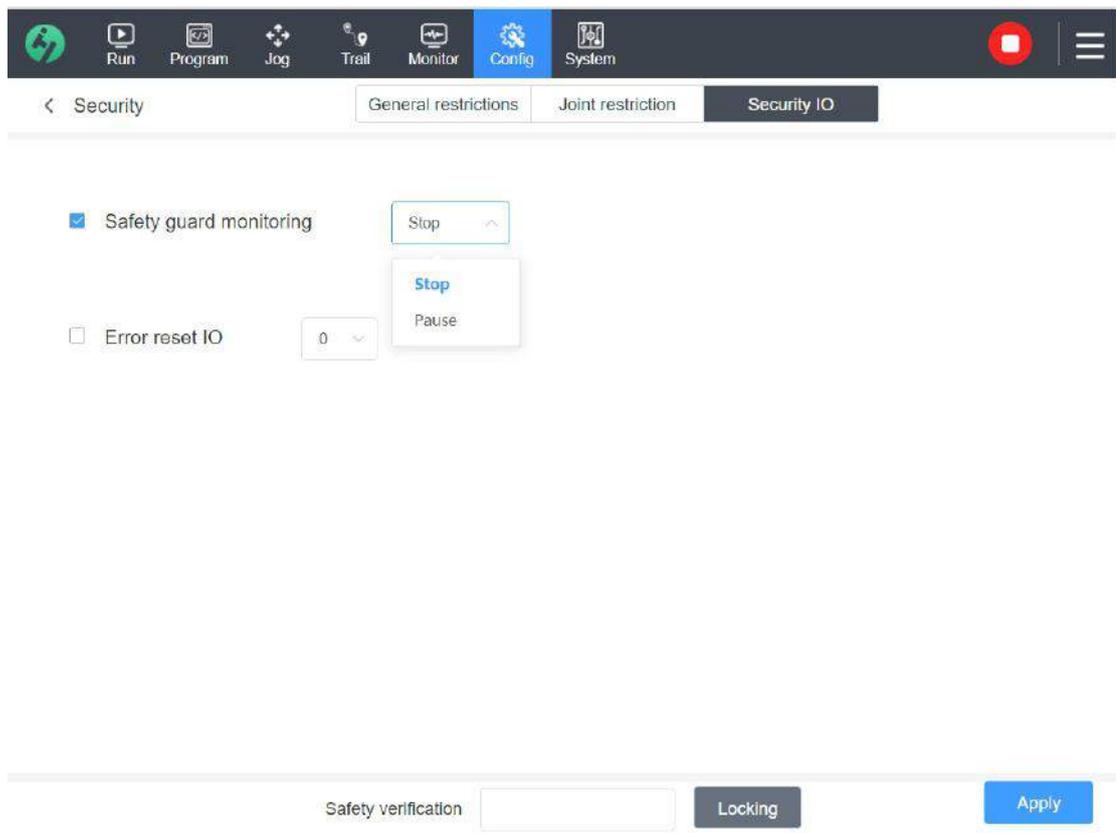
◆   **Error Reset:** After the system reports robot error, please set the output IO of the electric cabinet to high level or low level uniformly.

◆   **Light Curtain Monitoring:** Check "Enable Safety Light Curtain Monitoring" option and ,

and configure the function of stopping or pausing movement upon triggering.

[Enable Safety Light Curtain Stop]: When the safety IO signal is detected, the system stops the movement of the robot and displays pop-up window hint that the robot is in the state of safety protection and it is necessary to remove the protective switch or turn off the signal, and then click [Reset] button to clear the error before the robot can be operated again. After reset, the power supply of the robot is not cut off.

[Enable Safety Light Curtain Pause]: When the safety IO signal is detected, the script will be suspended and it is necessary to remove the protective switch or turn off the signal, and then click [Reset] button to clear the error before the current script can continue to run.

Note: ① After setting, click "Apply" button so that it will take effect. Before exiting, please click "Lock" button.

② A certain IO of the fourth-generation electric cabinet can be flexibly designated as the special IO for safety light curtain monitoring;

3.5 User Coordinate System

User coordinate system can be set by point, line and plane. The user-defined coordinate system enables the robot to work with different worktables or clamps.

Application Significance:

In practical applications, the robot does not always work in the same place, sometimes it will work in a new place and then continue to work; however, the new scene may be tilted, the robot may turn over. Due to the tilt rotation, the original movement trajectory is no longer applicable to the new scene;

In the new scenario, if it is necessary to continue to use the preset script for work, that is, on the premise that the waypoints in the entire script do not change, it is only necessary to change the corresponding user coordinate system to achieve this goal. The influence of user coordinate system on the trajectory is shown in the following figure:

1)   Original work trajectory of robot for demonstration;
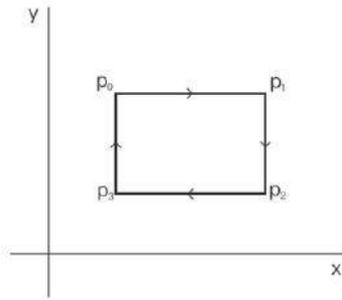
2)   After the work scenario of the robot changes, if the current coordinate system is still used, the motion trajectory of the script is shown in the figure below, and it is not the desired one of the user;
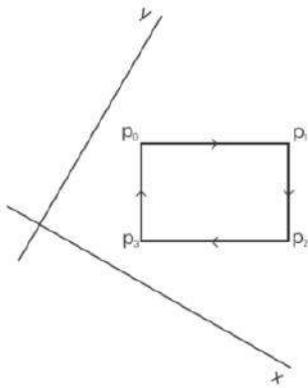
3)   The significance of self-defined coordinate system is that the robot will make the overall relative deviation as the coordinate system changes and continues to work according to the original trajectory;
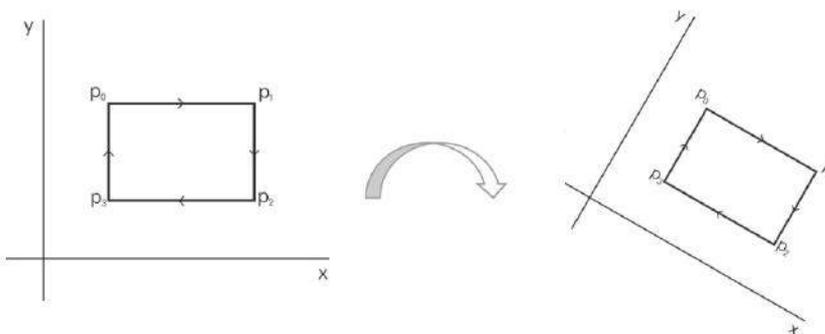
3.5.1 Point Coordinate System

Click " ➕ Add Point" button and edit "Point_N" point coordinate system in the right area; for example: "Point_1" shown in the figure below: Click "Set Point" button to set the coordinate system of the point.

The point determines the origin of the coordinate system, and the position and direction of the point coordinate system are the same as the tool coordinate of the current point (TCP's position and direction at the demonstration point are the same).
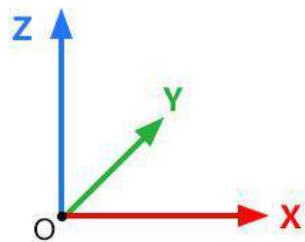
Original coordinate system
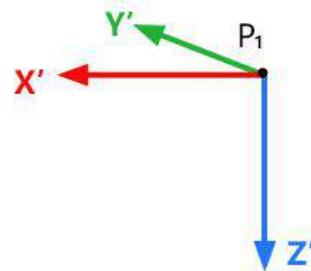
User coordinate system of point
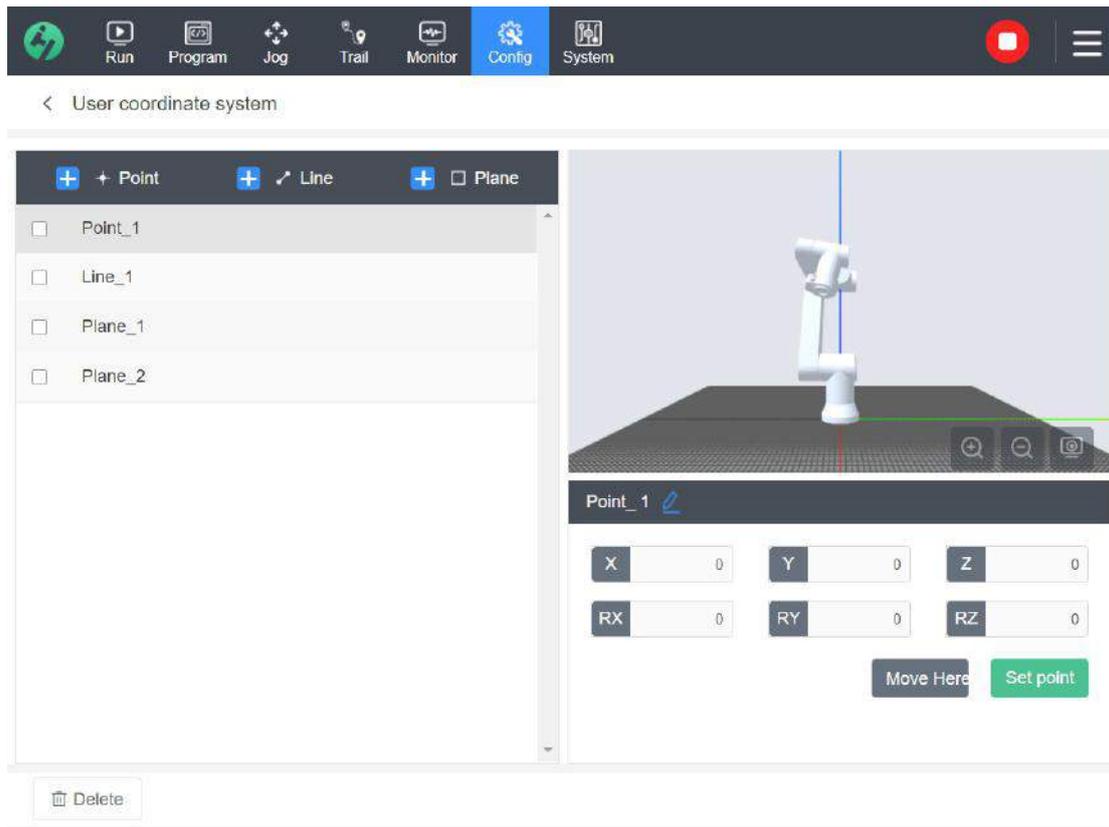
◆ Rename: Click the edit icon ✎ next to Point_N, enter a new name and confirm it;

◆ Set Point: Click "Set Point" button to jump to the inching interface to set point;

◆ Press and Hold to Point: Press this button to move the robot arm to the selected character teaching point. After the end of the movement, the coordinate system of the feature will coincide with that of TCP;

◆ Delete: Check the coordinate system in the list to delete it; after the coordinate system is deleted, if the coordinate system is used in the script, the relevant parameters in the script will be deleted synchronously;

Note: If the user coordinate system being used by a waypoint is deleted, the data in the waypoint is deleted synchronously; the user can re-select user coordinate system for waypoints.

3.5.2 Rectilinear Coordinate System

Click [➕Add Line] button, as shown in the figure below, to add the Line_1 feature.The two points can be taught to calculate the linear features: The first point P_1 points to the direction of the

second point P₂ which is the positive direction of the Y-axis of the rectilinear coordinate system. The origin of the rectilinear coordinate system is at the same position as the first demonstration point.

Original coordinate system   User coordinate system of line
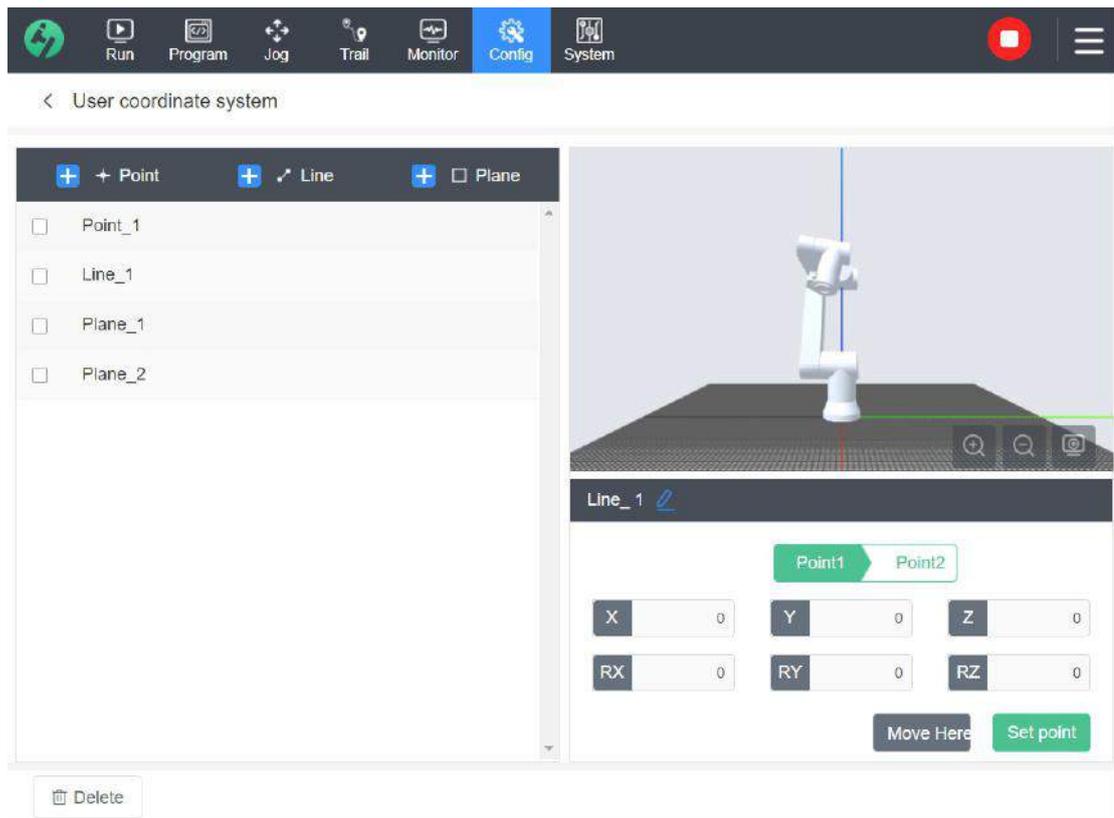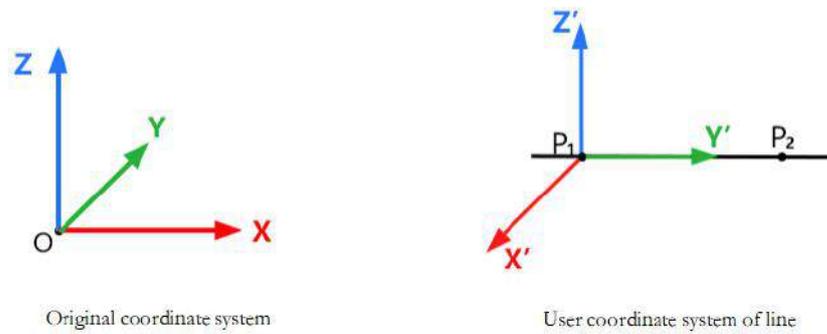
3.5.3 Plane Coordinate System

Click [➕ Add Line] button, as shown in the figure below, to add the Plane_1 feature. Plane is defined by features of three demonstration points. The origin of the plane coordinate system is at the same position as the first demonstration point.

1) Set user coordinate system and move the robot end to Point P_1;

2) When the end of the robot reaches Point P_2, Point P_1 points to the direction of Point P_2 which is Y-axis direction of the user coordinate system;

3) Continue to teach Point P_3; at the moment, three points forms a plane, and Point P_3 determines the direction of +X -axis (the direction perpendicular to P_1P_2 and pointing to P_3 is +X-axis direction);

4) The Z-axis is perpendicular to the three-point plane, and the + Z-axis direction can be determined according to the Right-Hand Rule.

Original coordinate system   User coordinate system of plane
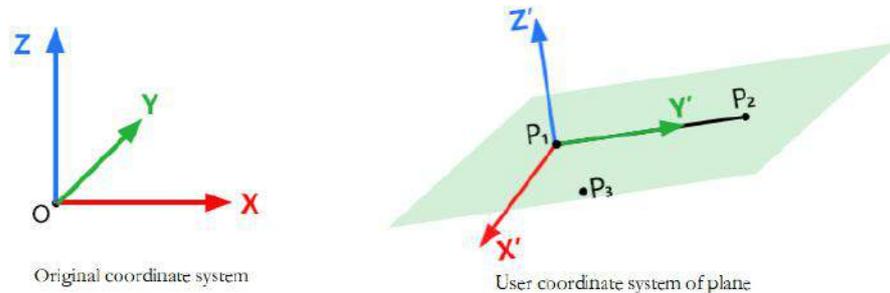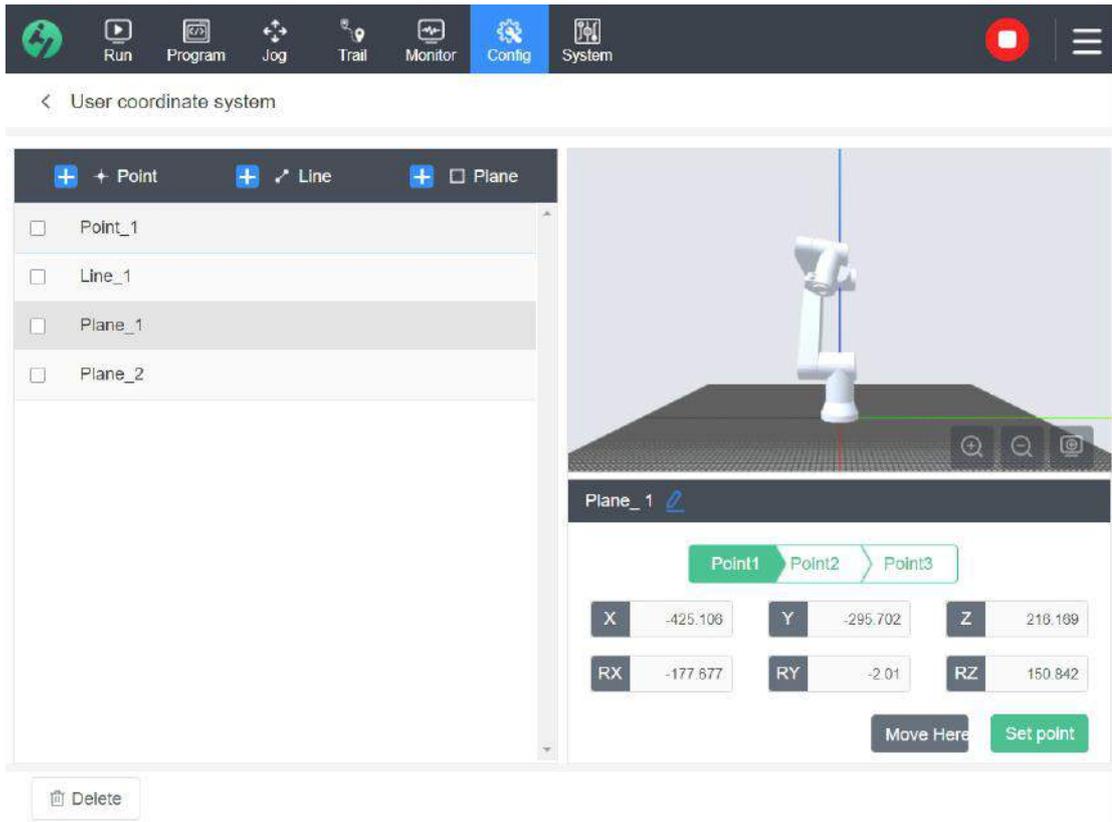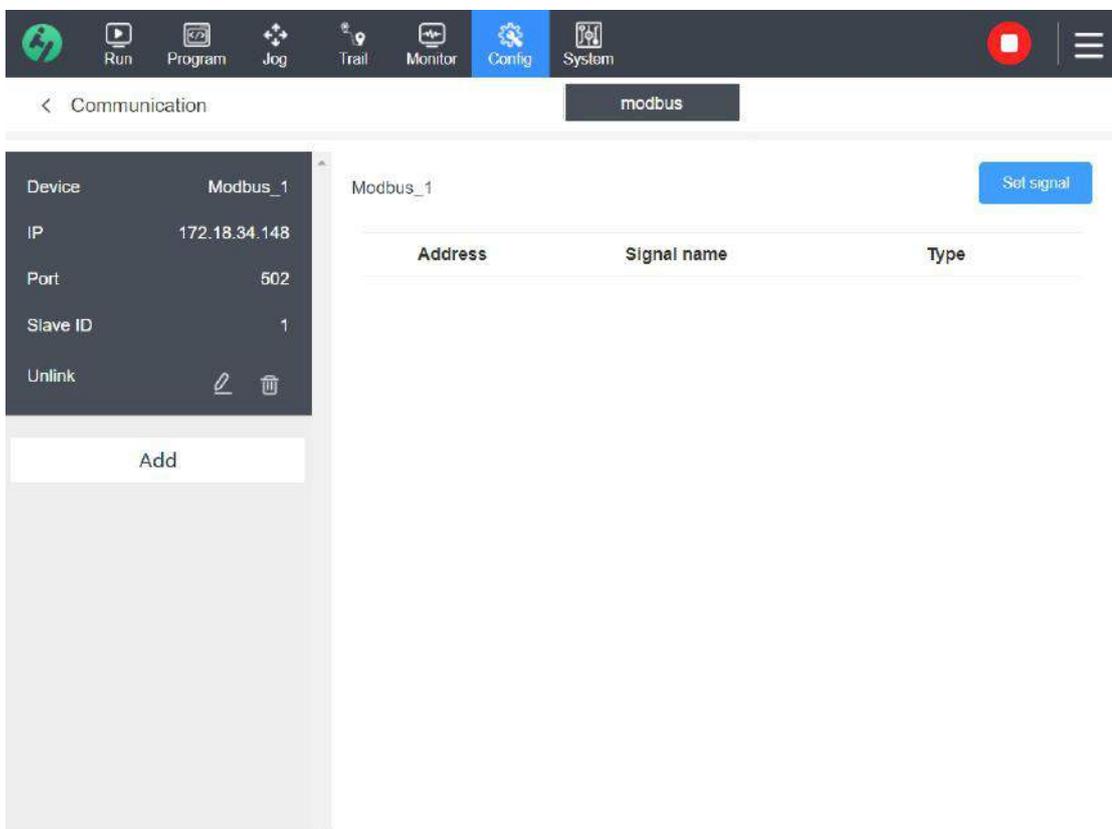
3.6   Modbus

MODBUS client (host) signals can be set. The input/output signal (register or number) can create a connection to the MODBUS server (or slave device) with the specified IP address. The address of each signal is unique.

3.6.1 Device List
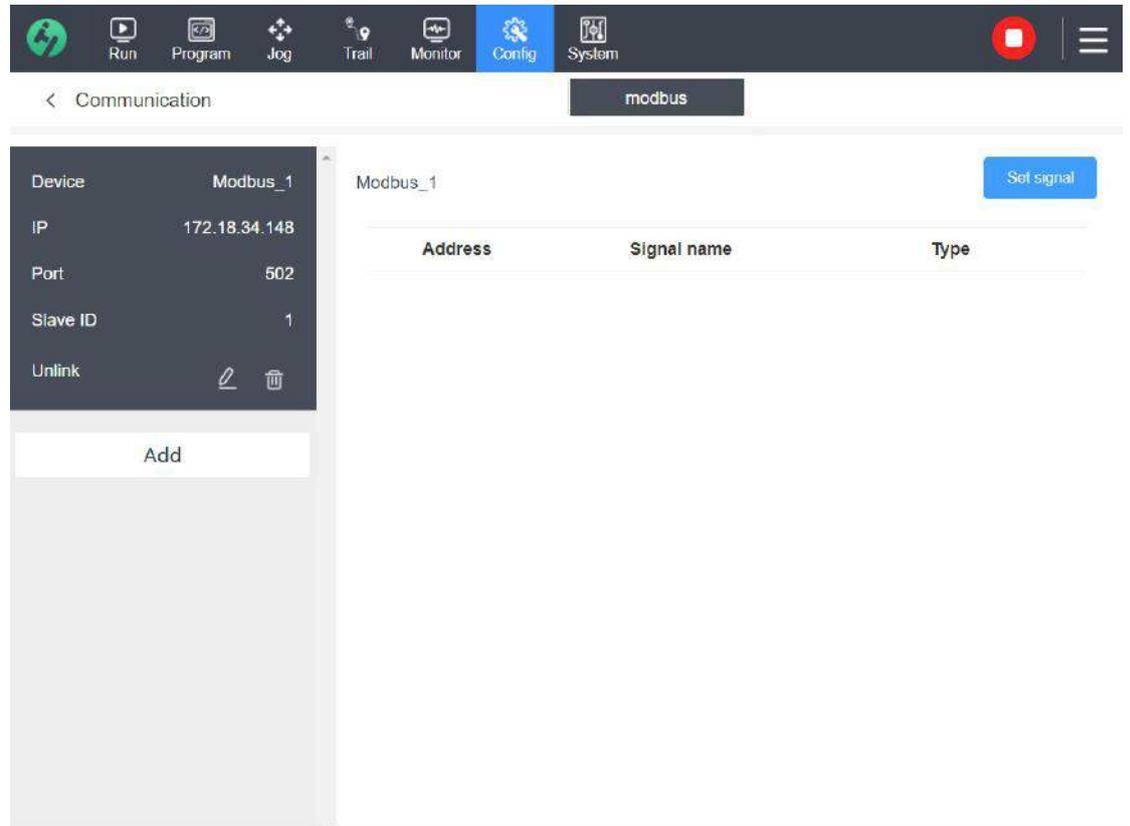
◆ Add: Click "Add Device" button and input the relevant information in the pop-up window, including IP address, port, slave station ID; Modbus name cannot be modified;

◆ Edit: To modify the Modbus device parameters;

◆ Delete: To delete the Modbus device;

◆ Selected: dark for selected, the signal list of the device is display on the right side;

3.6.2 Signal List

The signals in the list are arranged according to the added signal group interval;
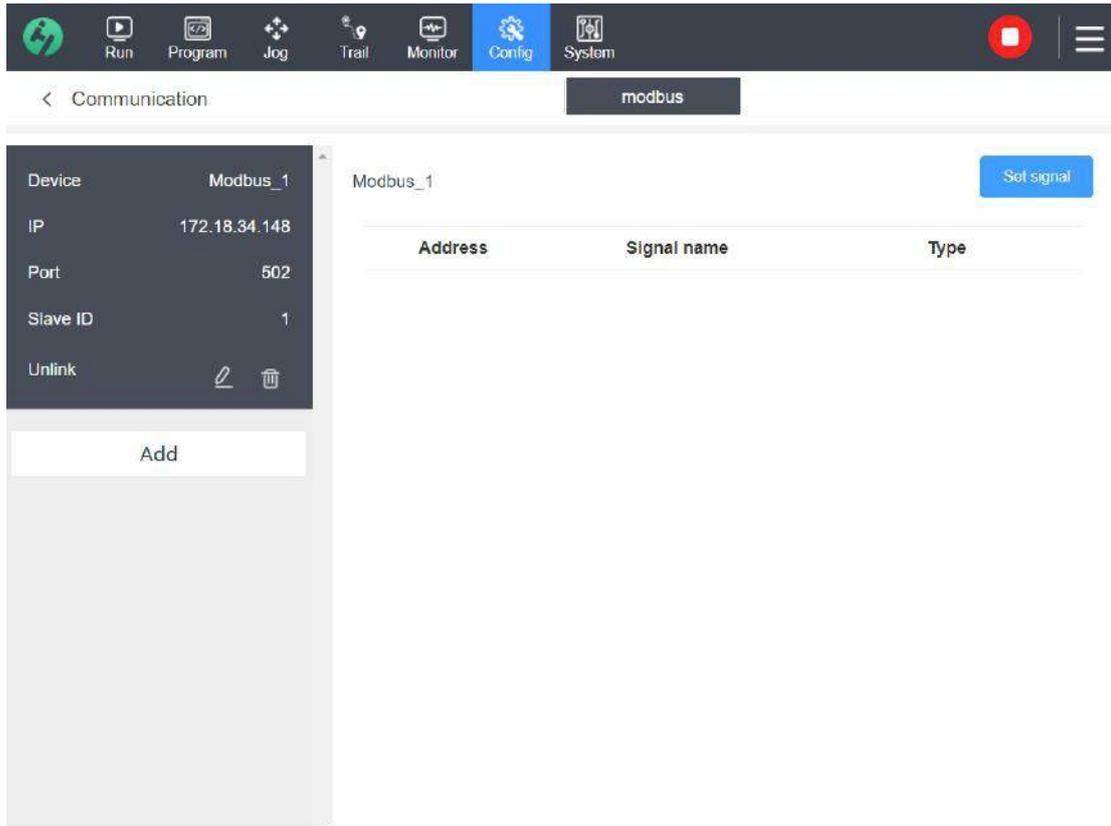
Set Signal: Click [Set Signal] button to edit signal group parameters, including signal group, signal type, address, read quantity and signal name;
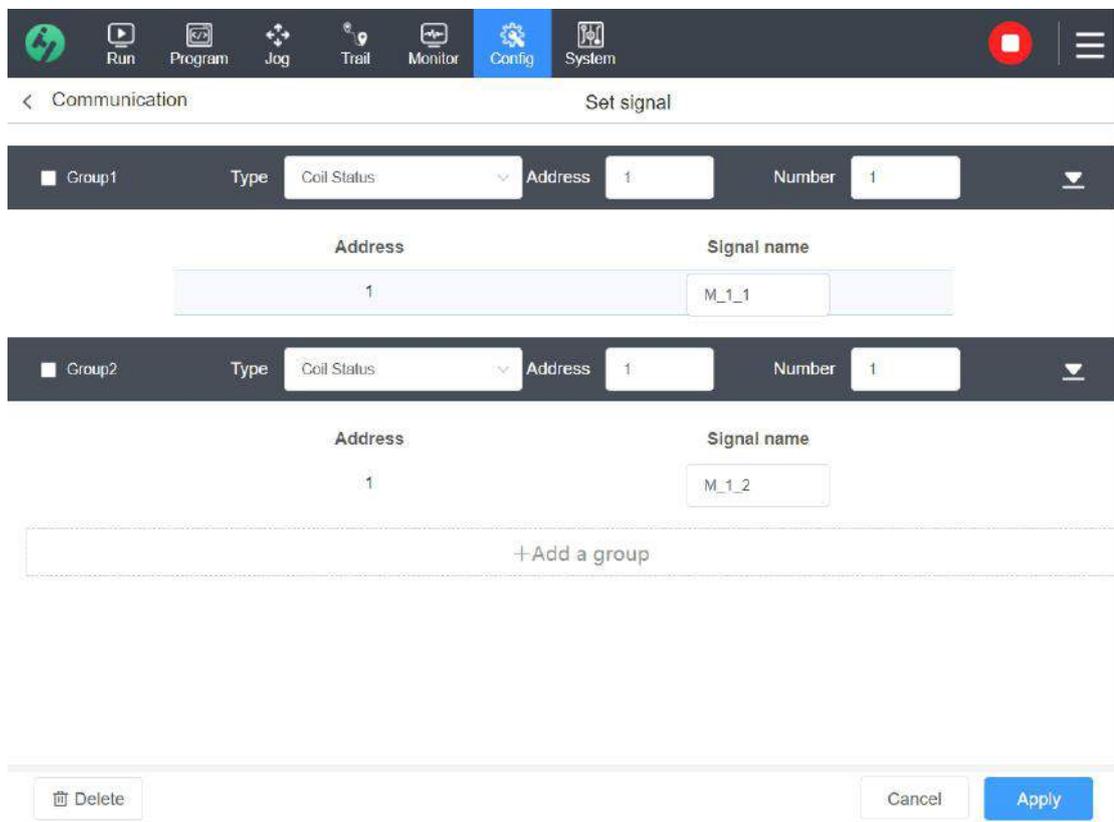
◆ **Add Group:** Add a group of same-type signals;

◆ **Type:** Use this drop-down menu to choose signal type.

Signal Type	Object	Access Type	Functions
Coil Status (Coil register)	Single bit	Read and write	For control/alarm switch status
Input Status (status register)	Single bit	Read Only	Used to input diagnostic function
Holding Register	16 bits	Read and write	Used to transmit user setting parameters
Input Register	16 bits	Read Only	Used to save input signal value

◆ **Signal Address:** Input the signal address to be read which is the starting value of the group of signal;

◆ **Read Quantity:** Input the quantity of signals to be read from the starting address to the end address; for example, the signal address is 00100 and if the quantity of signals to be read is 3, then the signals with the signal value between 00100 and 00102 will be read;

◆ **Signal Name:** Making it easy for users to find, and it cannot be repeated;

◆ **Expand:** When there are many signals in a group, these signals can be folded and hidden; and then can be expanded when the user clicks "Expand" button;

◆ **Delete:** Check the block in front of the signal group name to delete the whole group of signals;

◆ **Apply, Cancel:** It takes effect after "Apply" button and the unsaved data will be deleted when "Cancel" button is pressed;

Note: After the Modbus signal is added, the SetIO instruction in the script can be directly used; in addition, Modbus monitoring module will be added to the monitoring interface accordingly.

Chapter 4 Programming
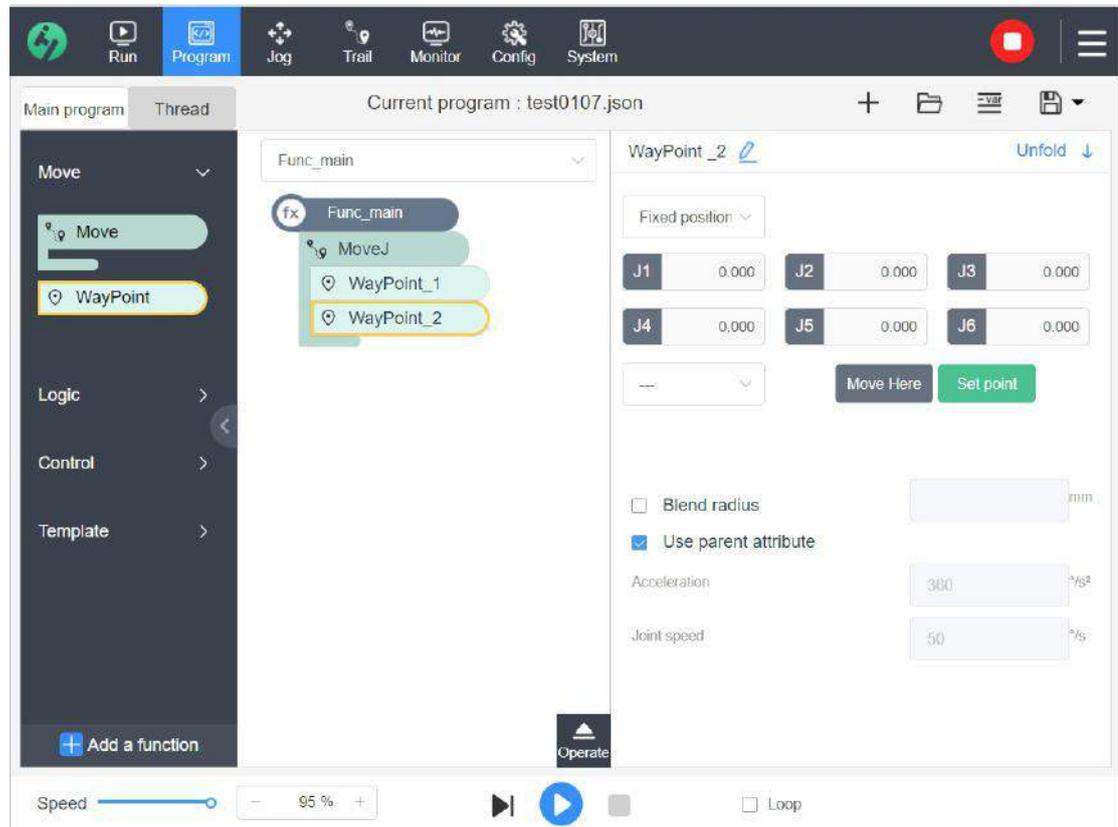
4.1 Introductory Program

1、Add a program to the programming interface under enabled condition. [New → Naming Program → Confirm New]

2、Under the Func_main function, drag a Move command from the command menu to the function node. At this time, a　waypoint will be automatically added under the Move node and named as WayPoint_1; the Move mode is MoveJ by default; then drag another waypoint to WayPoint_1 and name it as WayPoint_2.

3、In WayPoint_1, click [Set Point] button to jump to the inching interface to teach the position of the point, and click [OK] button to jump back; switch to WayPoint_2 again and teach the position of WayPoint 2 in the same way. WayPoint 1 serves as the starting point of the robot's movement, and WayPoint 2 serves as the end point of the robot's movement.

4、Click "Run" button to complete the movement.

4.2 Interface Introduction
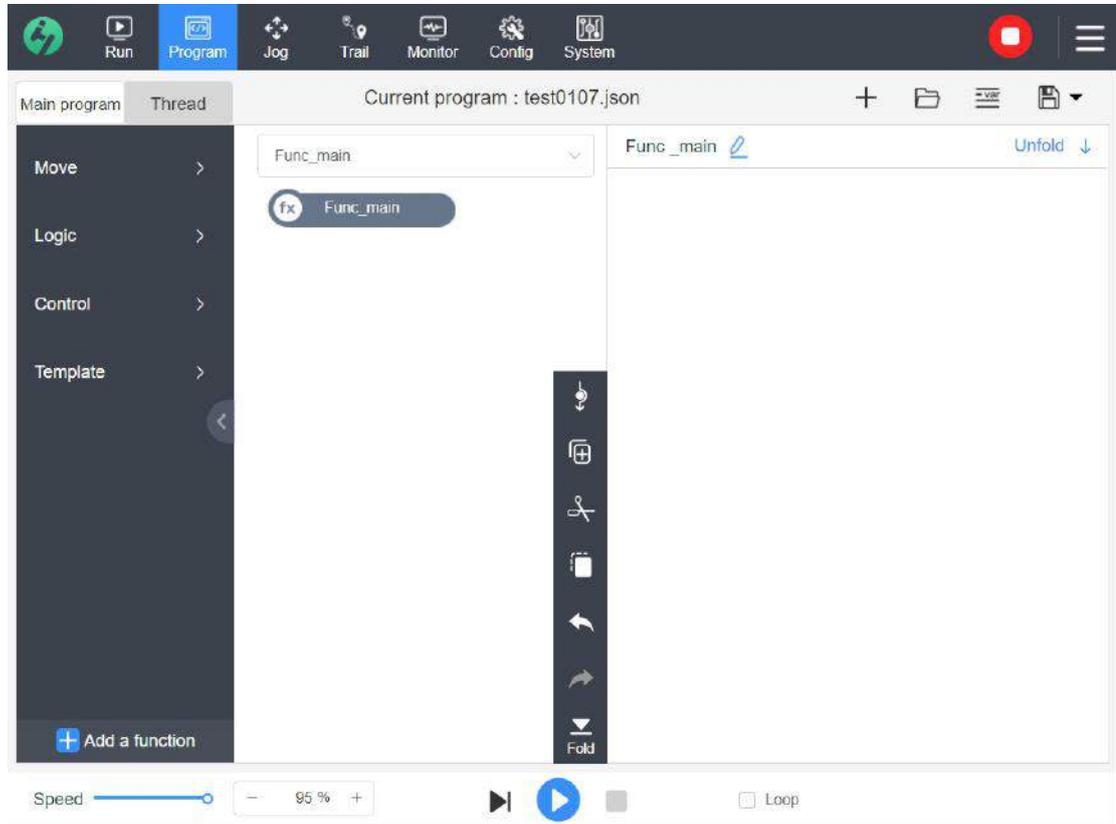
＋	New	Create a new program;	
🗁	Opening	Load the program, import an export the file, delete the program	
⁼var	Variable	Variable management, variable value view;	
💾▼	Save	After editing the program, click the "Save" button and select Save/Save as;	
✎	Remark	Remarks function or command, easy to explain the function;	
↓ ↑	Fold Expand	When the command parameter is edited, more areas can be expanded upward while the 3D model is hidden. Click it again to hide them;	
⊞	Copy	Click [Copy] button and select one or more commands to be copied from the program list, and click [OK] button to work with "Paste" function;	

	Cut	The steps are similar to "Copy" function;
	Paste	Select a command and click "Paste" button, then the copied command will be copied to the bottom of the selected command;
	Skip	Select a command, click [Skip] button, then the command turns gray and when the program runs, the command will be skipped;
	Cancel	Click "Cancel" button to cancel the last modification;
	Restore	Click "Restore" button to restore the last modification;
	Delete	When you drag the command back to the menu area, the red "Delete" button will appear, then you can loosen your finger to delete the command;
	Search	Click the "Search" button and then input the command in the search input box under the function drop-down box to search it.To exit the search,

4.2.1 New

Click [New] button and enter the file name in the pop-up box, and then click [OK] button to create a new program file, as shown in the figure below.

If there are other files not saved at present, a pop-up window will prompt whether to save the edited files when creating;
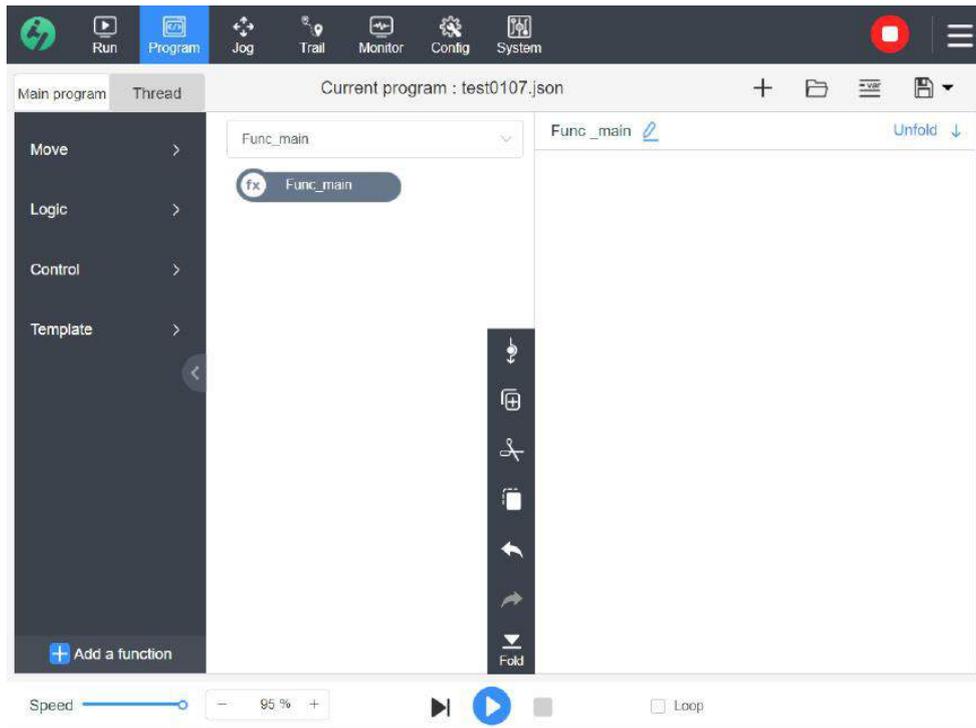
4.2.2 Program List
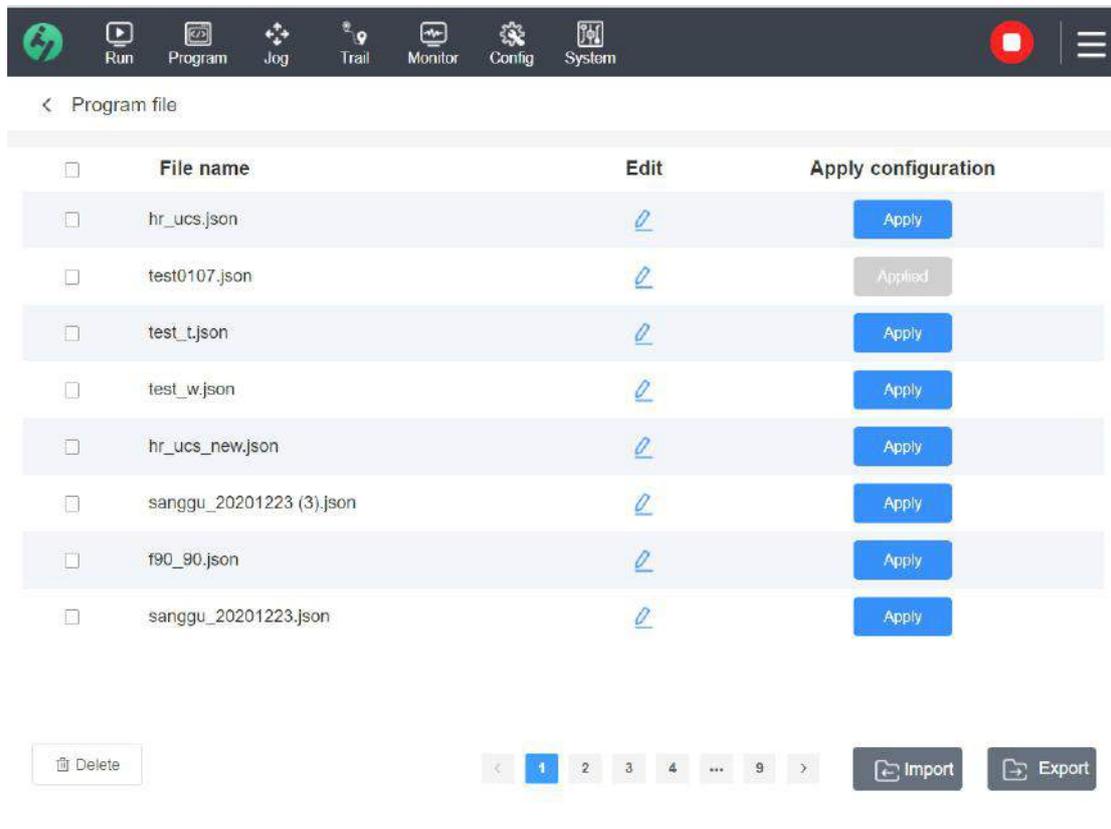
The saved files are recorded in the file list, and the user can rename, delete, import, export

and apply the files;

◆ **Rename:** Click "Edit" button and set the file name in the pop-up window and click "OK" button so that the file takes effect;

◆ **Delete:** Check the block in front of the file, click "Delete" button; if the current program is deleted, there is a prompt: The robot is currently using the program file, to delete it may affect your normal production, are you sure to delete it?

◆ **Import:** The file is imported from the default path from the system or from the USB flash disk to the file list and then the imported file shall be displayed in the file list;

◆ **Export:**Check the block in front of the file, click "Export" button; you can export the file to the system's default path or USB flash disk;

◆ **Apply:** Open the program; click [Apply] button. If the program being edited is still not saved, a prompt box will pop up to remind you to save the current file. The user can choose to save the current file or directly apply the selected file. If the selected file is applied directly, the current file shall be not saved;

4.2.3 Function
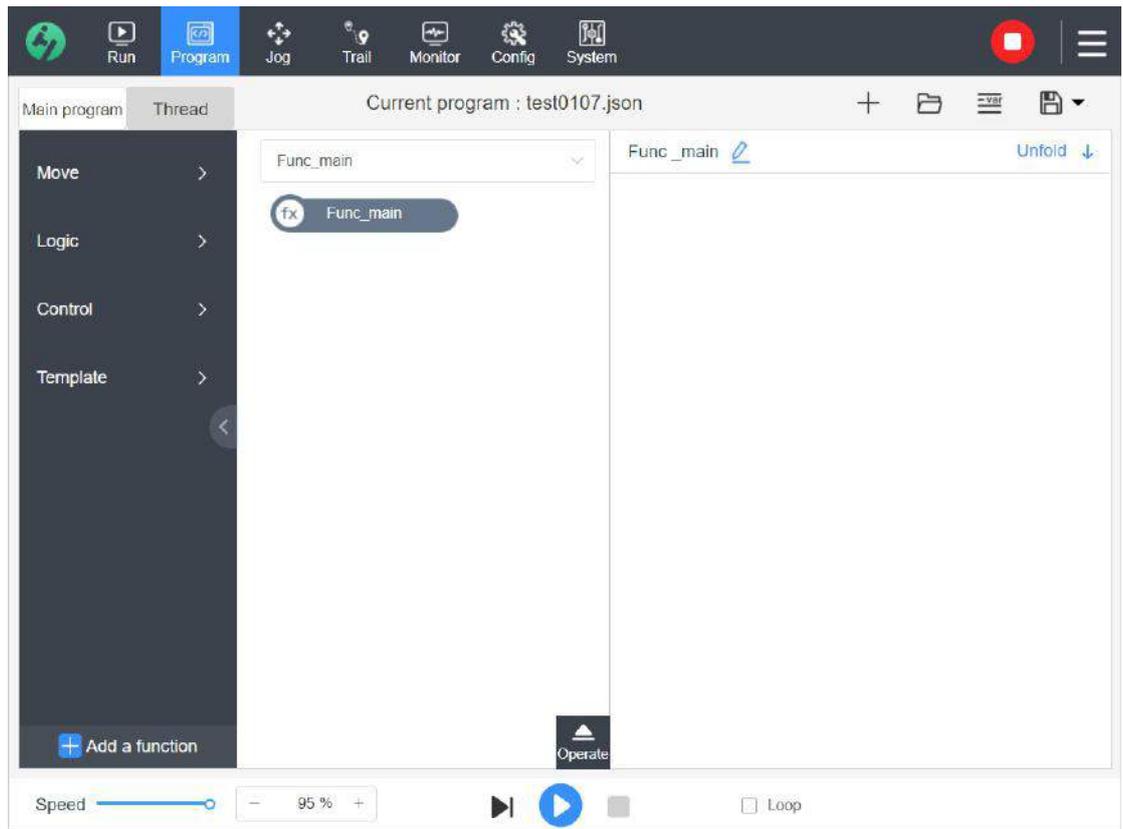
Func_main: Refers to the main function and is automatically added after a new program is created and cannot be deleted, that is, the program will only run from Func_main when the "Run" interface is running. During debugging in the programming interface, the use can directly select the function from the drop-down box to run;
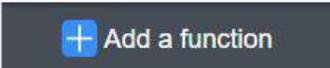
Func_N: ![Add a function] Click "Add Function" function, then a function will be added in the function drop-down box and named as Func_1,2,3...; for ordinary function, the user can drag it to the menu to delete it;

4.2.4   Command

◆ **Drag for removal:** Drag the command from the command menu to the program list node;

◆ **Command color block:** Commands are classified according to menu; motion command in green, logic command in yellow, control command in blue, template command in gray, and third party plug-in command in red.

◆ **Command Style:** The command being edited are selected with the frame in yellow and the command color block length varies with parameters and title;

◆ **Command Title:** Take "if" for example,　"if_ note" is shown after the note, there are 24 characters at most.

◆ **Parameter Box:** some commands have parameter box prompt, indicating that the command of the current parameter settings, displaying 24 characters at most, convenient to quickly browse command parameters set in the list.

4.2.5 Definition of Components
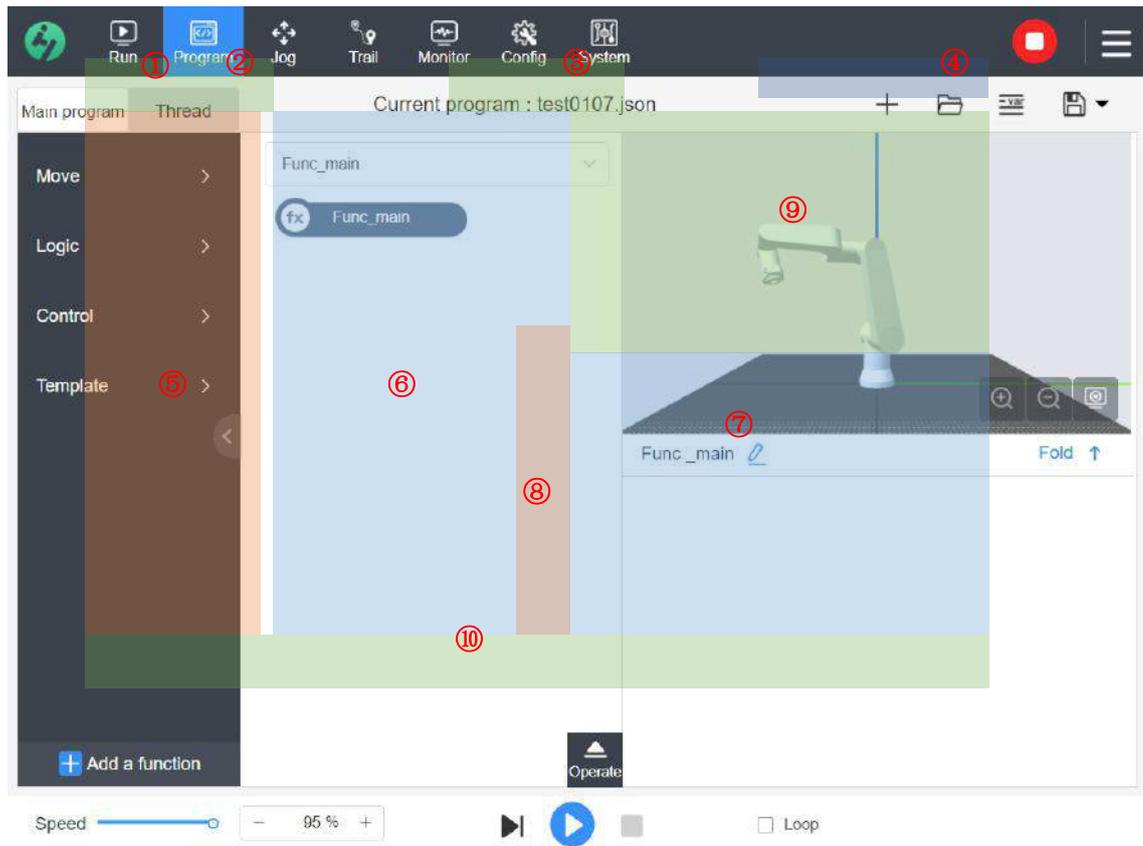
1) **Main Program:** When the robot is working, main programs, including motion processing, logic control, template, third-party plug-in reference, etc.

2) **Thread:** When the robot is working, it is parallel with the main program. Thread can be used to control external machines independent of the robot arm, generally for signal status monitoring, and can communicate with the robot program through variables and output signals.

3) **File Name:** Display the name of the current program file.

4) **New, Load, Variable and Save**

5) **Command Menu:** Expand the classified menu and drag it to the program list for use; the entire menu can be folded and hidden.

6) **Program List:** A script composed of a series of commands displayed as program nodes. The user can drag and drop a command from the menu to a node in the program list. There must be a command in the program list to run; the position of commands in the

list can be changed freely by drag and drop.

7) **Parameter Setting:** Display and set the properties related to the selected command, and can also remark the command information.

8) **Operating Space:** Select a command in the program list to copy, paste, skip and implement other operations.

9) **3D Model:** Display the current motion state of the robot. The buttons at the bottom left of the screen can manipulate the size of the perspective, view the coordinate system, the moving path,waypoints and the currently used security boundary of the program.

10) **Bottom Running Area:** Debug program, set loop debugging, adjust motion speed, and set current file as default program **file**.

4.3 Movement Command

4.3.1   Move

The movement modes include joint movement instruction (MoveJ), space linear movement instruction (MoveL) and circular arc movment instruction (MoveC).

From the command menu on the left, expand the movement classification and drag the move command to Func_main node. MoveJ is added by default. The user can switch the movement modes among MoveJ, MoveL and MoveC from the drop-down box.

Note: Switching the modes between MoveJ and MoveL does not affect the waypoint under its node. switching other modes can easily cause the waypoint data loss under the node. If MoveJ is switched to MoveC, all its waypoints will be deleted, only the passing point and the end point are left.
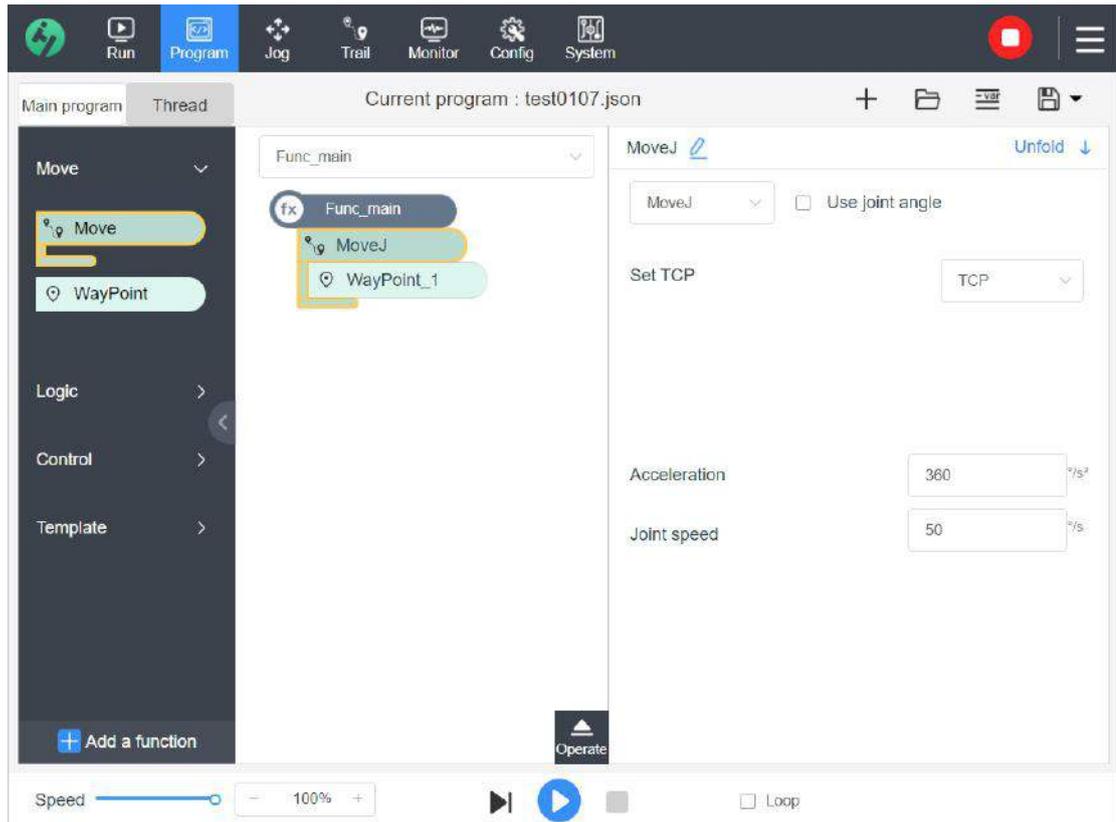
4.3.1.1 MoveJ Joint Movement

All the joint axes of the robot move rapidly and synchronously from one point to another. It is suitable for the situation where there is no requirement for the terminal trajectory of the robot. The end of the robot moves along an irregular curve. Please pay attention to the danger of collision.
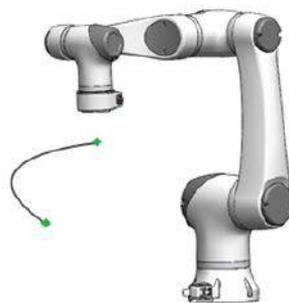
Parameters include: Whether to Enable "Use Joint Angle", Set TCP, Acceleration and Joint Speed;

◆ **Use Joint Angle**

If "Use Joint Angle" is enabled, the TCP option is not available. When a program moves between robots, the waypoints defined using joint angles are not affected by TCP.

◆ **Set TCP**

The user can switch the appropriate TCP according to the end tool. Select the TCP that matches the current end tool from the drop-down box. When a waypoint is set, the program remembers the spatial coordinates of the selected tool features of the point. The TCP option is "Factory TCP" by default.
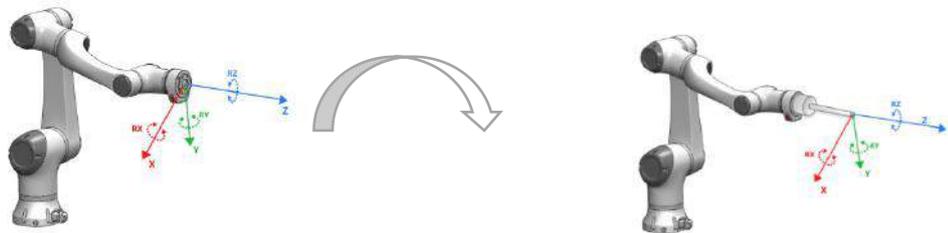
Tool Coordinate Change Comparison Digram after Switching TCP

◆ **Acceleration**

To set the joint acceleration at which the robot moves from one waypoint to next waypoint.

◆ **Joint speed**

To set the joint speed at which the robot moves from one waypoint to next waypoint.

Note: These parameters will have the parent attributes of all waypoints of the MoveJ node, and the same for other modes of motion.

4.3.1.2 MoveL Space Linear Movement

If you want the robot to move in a straight line, you can use the space linear movement command. The robot moves the tool center point TCP from one point to another in a straight line. When the starting and ending attitudes are different, the attitude will be rotated to the ending attitude in sync with the positions.
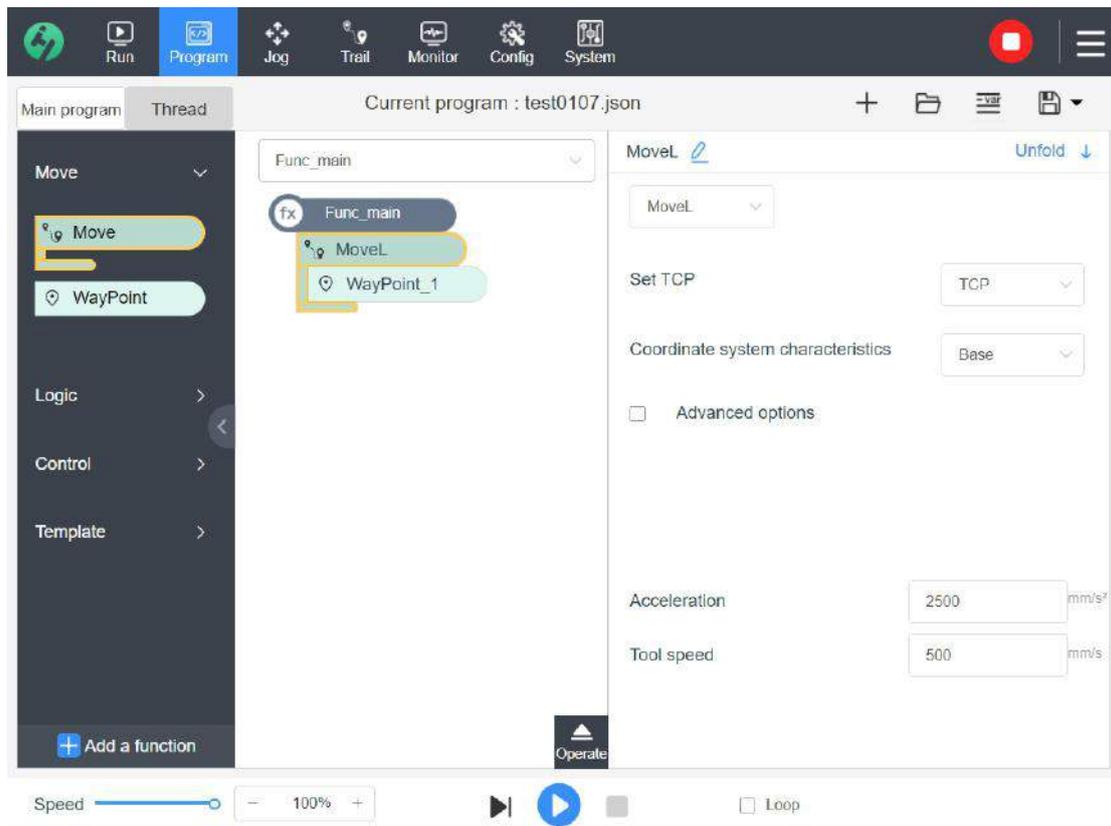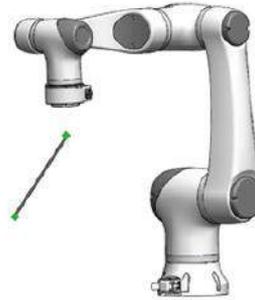
Parameters include: Set TCP, Coordinate System Feature, Acceleration and Tool Speed;

◆ **Set TCP**

HYPERLINK \l "_3.1　Set TCP" (Refer to the description of "Set TCP" in MoveJ for details)

◆ **Coordinate System Feature**

Select the coordinate system that conforms to the current working environment from the drop-down box. After the coordinate system is switched, the trajectory formed by the

waypoints will be shifted relative to the whole coordinate system. (Refer to the description of "User Coordinate System" for detailed diagram)

4.3.1.3 MoveC Circular Movement Instruction

The MoveC instruction can implement circular trajectory motion, and determine a circular trajectory (full circle or circular arc) in a three-point manner. Three points cannot be in the same position or in a straight line, otherwise these points cannot determine a circular trajectory.
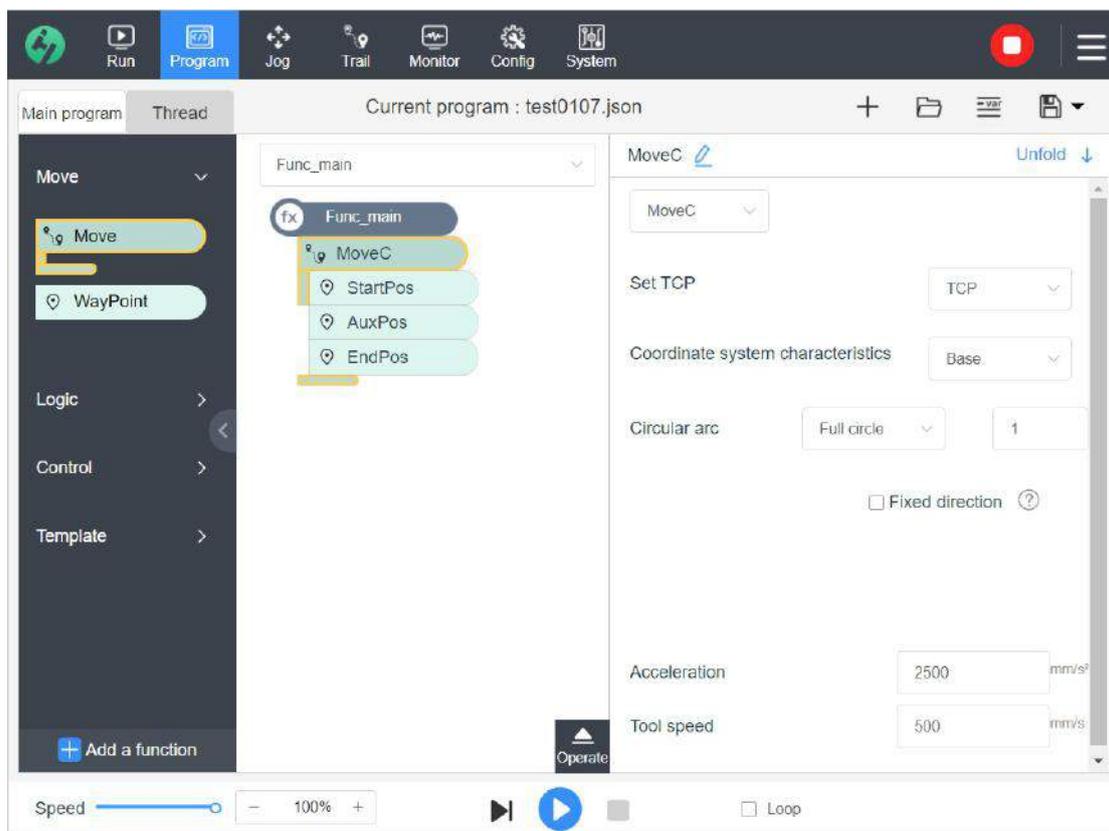
◆ **Set TCP**

HYPERLINK \l "_3.1 Set TCP" (Refer to the description of "Set TCP" in MoveJ for details)

◆ **Coordinate System Feature**

HYPERLINK \l "_3.5 User Coordinate System" (Refer to the description of "Coordinate System Feature" for details)

◆ **Blending Radius**

◆ **Function**

Function	Description
Circular Arc	There are two types: circular arc and full circle
initial position StartPos	Current position of robot, namely starting point of circular arc
Passing Position ThroughPos	Refers to a point through which a circular trajectory passes, PCS coordinate values can be "taught manually", or coordinate values can be passed through "Select Variables".
Target Position EndPos	The target point of a circular trajectory, if full circle, is also regarded as a passing point. The values can be taught as PCS coordinates or passed in through variables.
Number of Circles	Refers to the number of the trajectories repeated by the robot if the arc type chosen is a full circle
Attitude Type	There are two types: 1) When "Defined by Target Attitude" is not checked, the attitude changes from the attitude at starting point to that at the end point and is independent of the attitude at the second point. The effect is shown in the figure below: Point 1is the starting point and Point 2 and Point 3 are the passing point of the arc and the target point. 2) When "Defined by Target Attitude" is checked, Point 1is the starting point and Point 2 and Point 3 are the passing point of the arc and the target point. Based on the attitude of the tool at the starting point, it moves from the starting point to the end point. The effect is shown in the figure below:
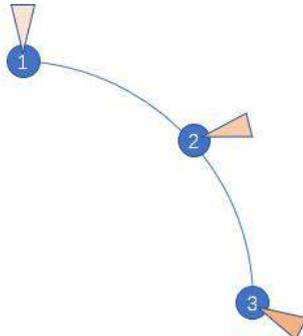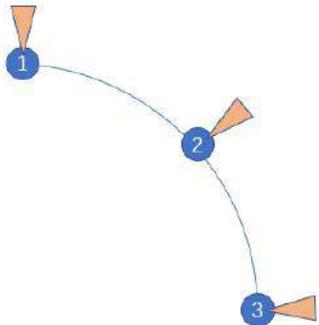
4.3.2 **WayPoint**

Refers to the point in the motion path of the robot. Waypoint is a key element of the robot's

program and indicates the position to which the robot arm is moving. Fixed position waypoint is demonstrated by actually moving the robot arm to the appropriate position.

Waypoints are used only within a Move node and cannot stand alone as a first-level node (except within a template). Subject to parameters in Move, waypoints should be taught based on the relation between the characteristics applied in Move and TCP to achieve the desired TCP location.
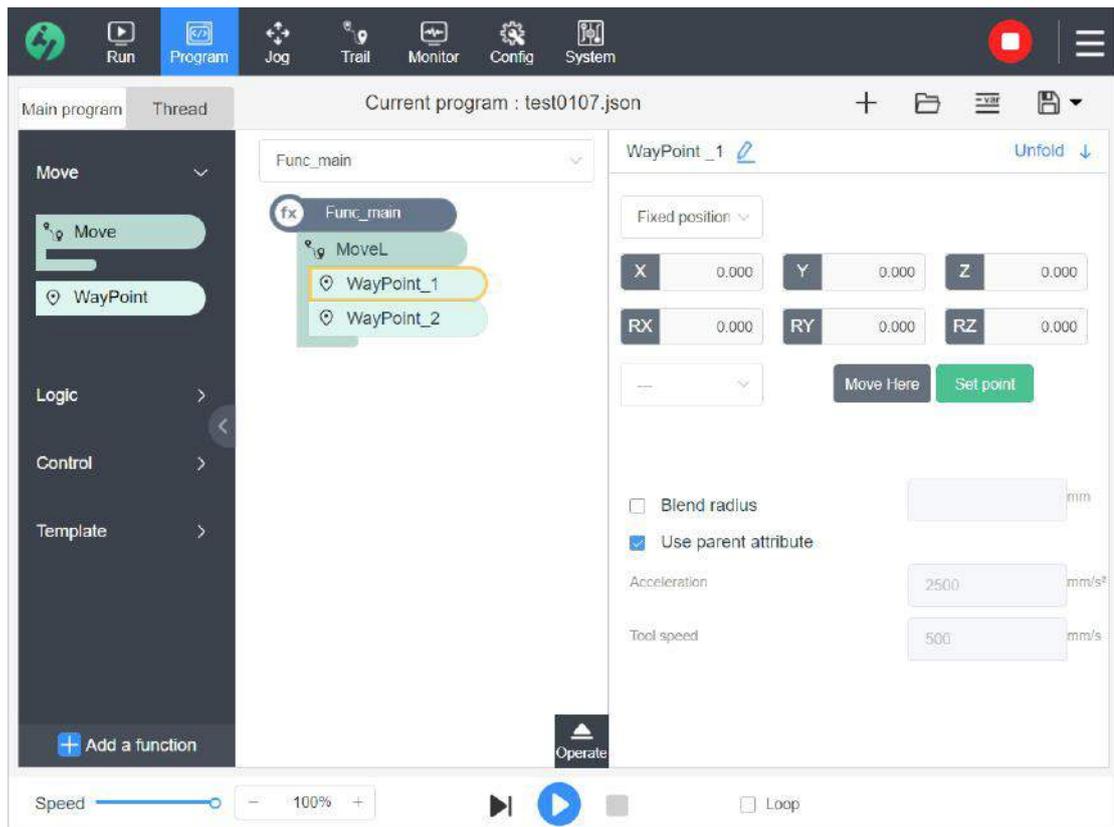
There are three alternative ways to locate waypoints: fixed position, translation and variable; the user can switch the location mode according to the actual needs and the current option is "fixed position" by default.

4.3.2.1 Fixed Position

◆   Point Information

If the waypoint is in MoveL, the position information of the current point is P[X,Y,Z,RX,RY,RZ]; if the waypoint is in MoveJ, P[J1,J2,J3,J4,J5,J6] are displayed;

◆   Set Point

Teach the point and get the robot to move to the actual position.

◆　Press and Hold to Point

Press and hold the button to make the robot move to the current point position, and there will be a prompt when it moves to the point.

◆ Point List

Click the drop-down box to select the point. The point position information of the current point will be displayed in the position information field. Click [Press and Hold to Point] button to view the specific position of the point. You can edit the point list in the inching interface.

Note: If a point used by a waypoint is deleted from the point list, then the point list in the waypoint shall be deleted synchronously and the point information of the waypoint shall be cleared.

◆ Use Parent-Level Parameters

If "Use Parent-Level Parametersr" is checked by default, acceleration and joint speed in Move (tool speed) are used; if not checked, the currently filled acceleration and joint speed (tool speed) are used.

◆ Blending Radius

The adjacent two straight lines use the set blending radius to make the arc smooth transition. The operation is characterized by continuous motion and no stop at the waypoint. The smaller the blending radius is, the larger the path corner will be; conversely, the larger the blending radius is, the smaller the path corner will be.

Blending radius is used for smooth transition of arc. The radius set should not exceed the straight-line distance, otherwise an error will be reported. If two straight lines can be connected as a line, the radius set is invalid.

Note: Check it in "MoveL" and "MoveJ" to enable blending radius; if not check it,   meaning that the robot will not stop at this point but move to the next point; if the "Blending Radius" option in MoveC is checked, then it indicates that the blending radius of the current point is used; otherwise, the blending radius parameter set in MoveC is used.

Examples

A. Use Blending Radius in MoveJ Instruction
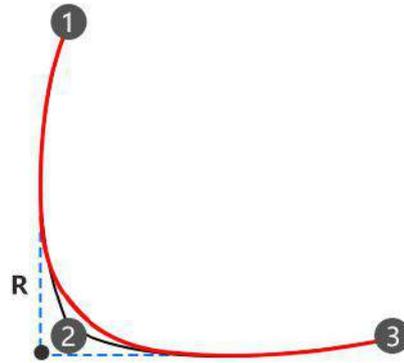
As shown in the figure, P1 is the current position of the robot, P2 is the target point of the first MoveJ instruction, and P3 is the target point of the second MoveJ instruction. The red trajectory is the calculated moving trajectory.

B.   Use Blending Radius in MoveL Instruction
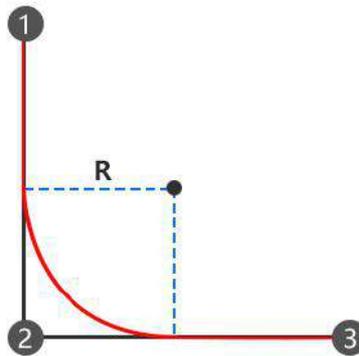
As shown in the figure, P1 is the current position of the robot, P2 is the target point of the first MoveL instruction, and P3 is the target point of the second MoveL instruction. The red trajectory is the calculated moving trajectory.

C.   Use Blending Radius in MoveC Instruction
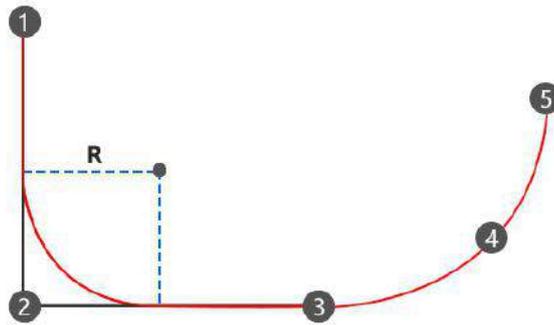
P1 is the current position of the robot, P2 is the target point of the first MoveC instruction, and P3 is the target point of the second MoveC instruction and also the starting point of the arc. P4 is the position where the arc passes through and P5 is the target position of the arc.

As shown in the figure, the red trajectory is the calculated moving trajectory. Two straight lines use the set blending radius to make the smooth transition of the arc, and MoveL+MoveC make the straight lines and the arc move continuously.

4.3.2.2 Translation
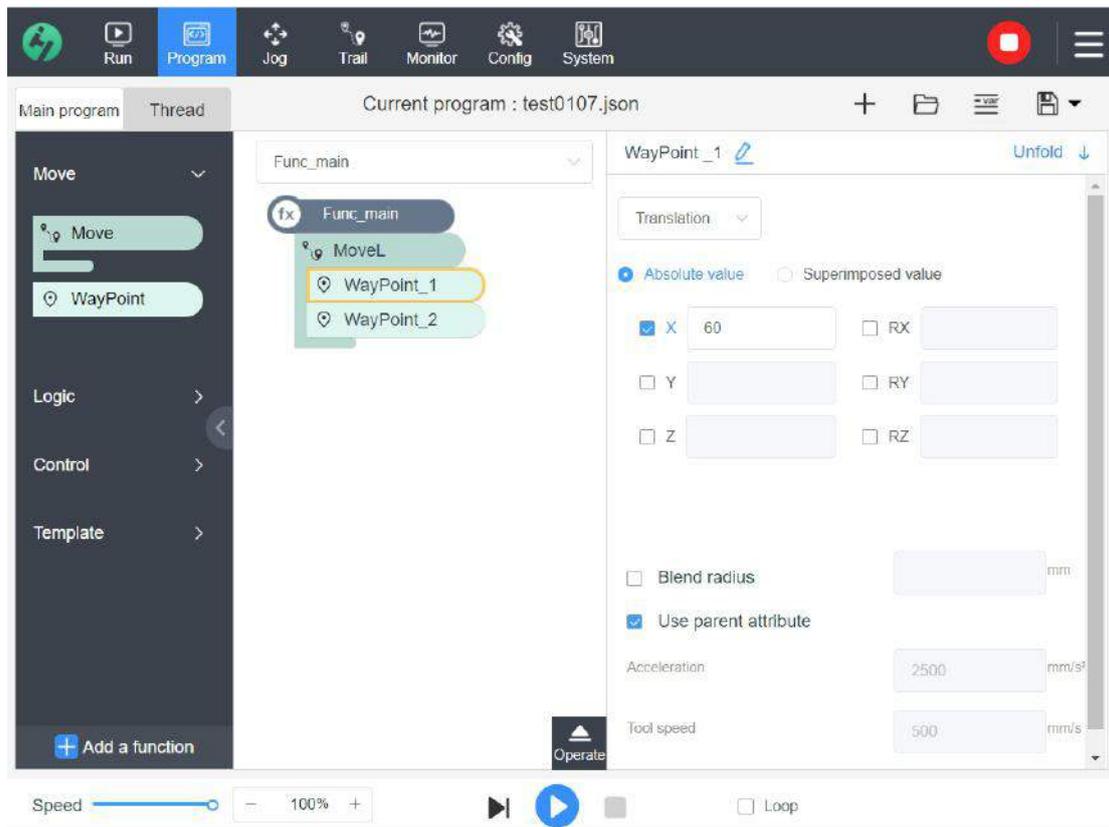
◆ Absolute Value: Move to the absolute value position.

◆ Superposition Value: the distance of moving a superposition value in the current position.

◆ Check Motion Axis: Specify the some axes of robot for absolute or superposition translation (J1,J2,J3,J4,J5,J6；X,Y,Z,RX,RY,RZ)

◆ HYPERLINK \l "_4.3.2 WayPoint" Blending radius (Refer to the description of "Blending Radius" in "Waypoint Fixed Position" for details)

◆ Use parent attribute (Refer to the description of parent attribute in "Waypoint Fixed Position" for details)

4.3.2.3 Variable

Variable can be added to the variable list, and the feedback point position information can be received through script command so as to assign the value to the variable. When the robot moves to the point, it will move according to the obtained position of the variable;
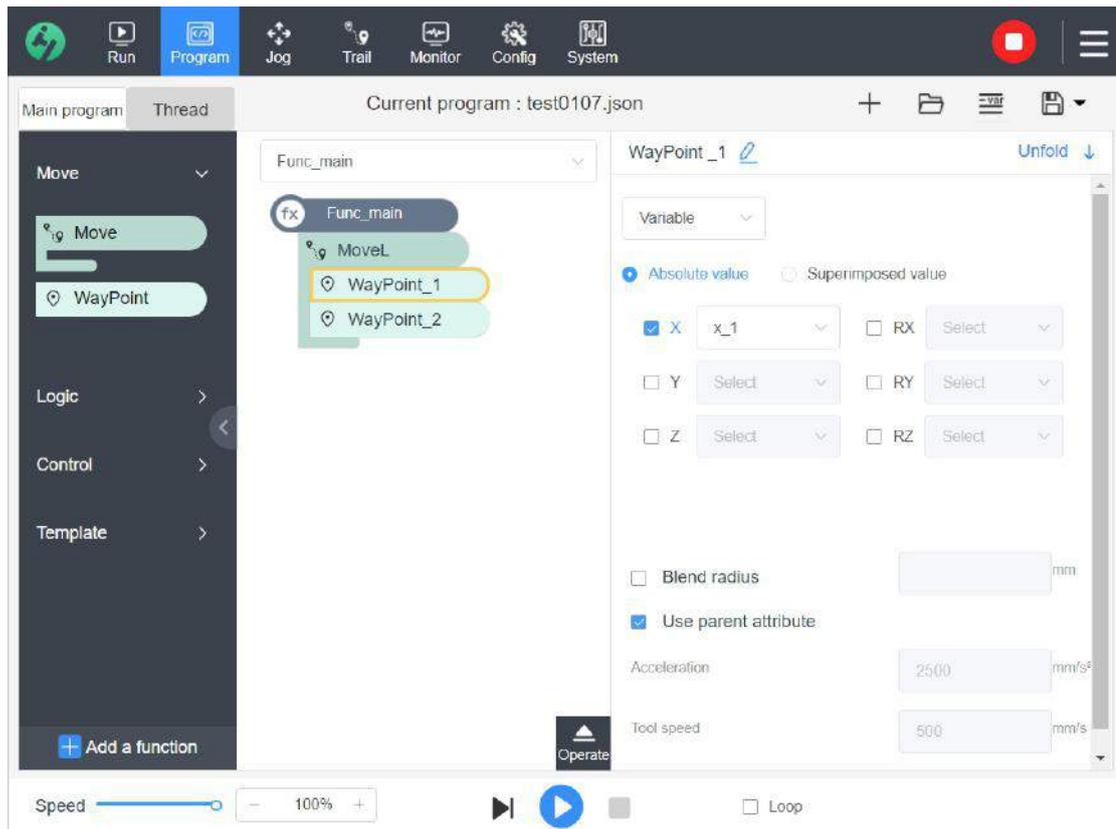
4.4 Logic Instruction

4.4.1　If

It is a judgment command and can instruct the robot to perform different actions according to different judgment conditions. When the "If" condition is satisfied, the action under the "If" node is executed. During the execution process, the command line in "If" node is still executed regardless of whether the condition is still judged to be true.

Each "If" judgment command can have more than one ElseIf commands and one Else command. The command line in the "If" node will run if the "If" condition is met, otherwise the ElseIf or Else will be judged in turn until the condition is met.
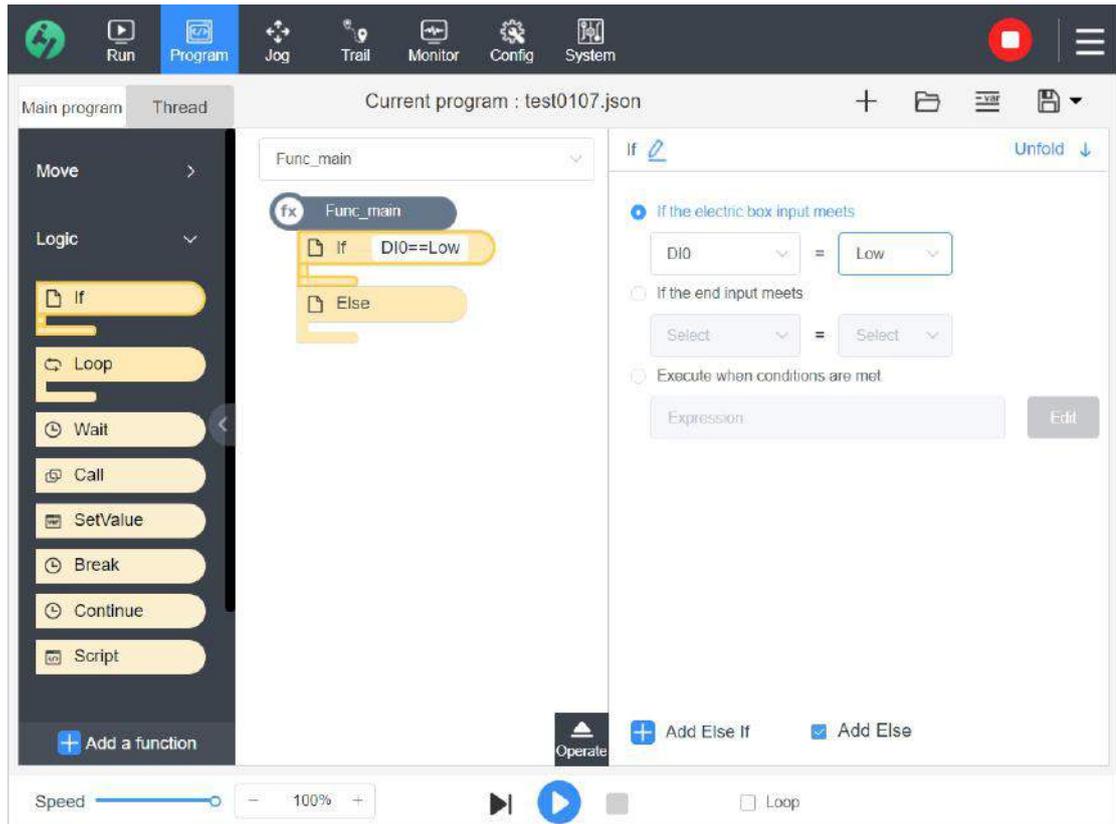
The system provides 3 judgment conditions:

◆ Judge input IO status of electric cabinet.

◆ Judge input IO status of robot end.

◆ Judge the custom conditional expression.

4.4.2   Wait

It is a wait command and used to keep the system waiting for specified time during running a program or continue to run the program after a condition is met.

Specified waiting time, I/O signal or expression. If you choose not to wait, you will do nothing.
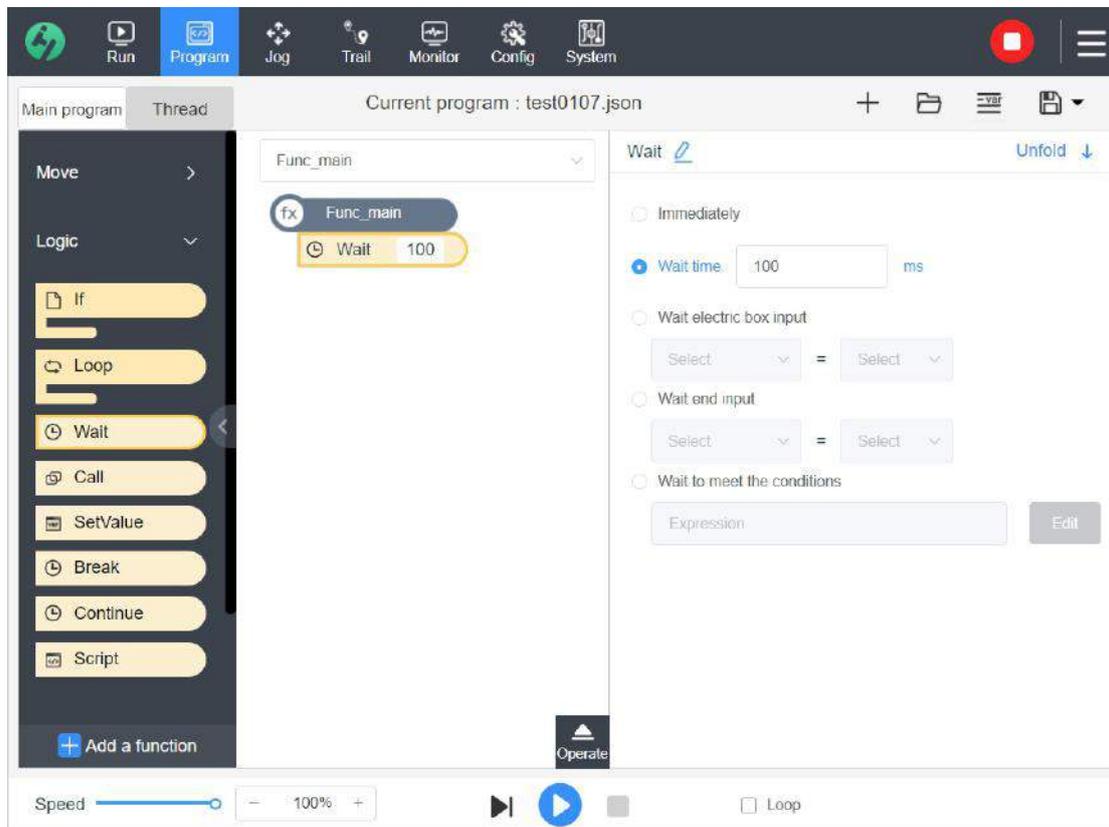
4.4.3 Loop

It is a loop command. This logic is used to loop through the program until a set number of loops is exceeded or a preset condition is not met. Loop conditions include infinite loop, number of loops and custom conditional expressions.
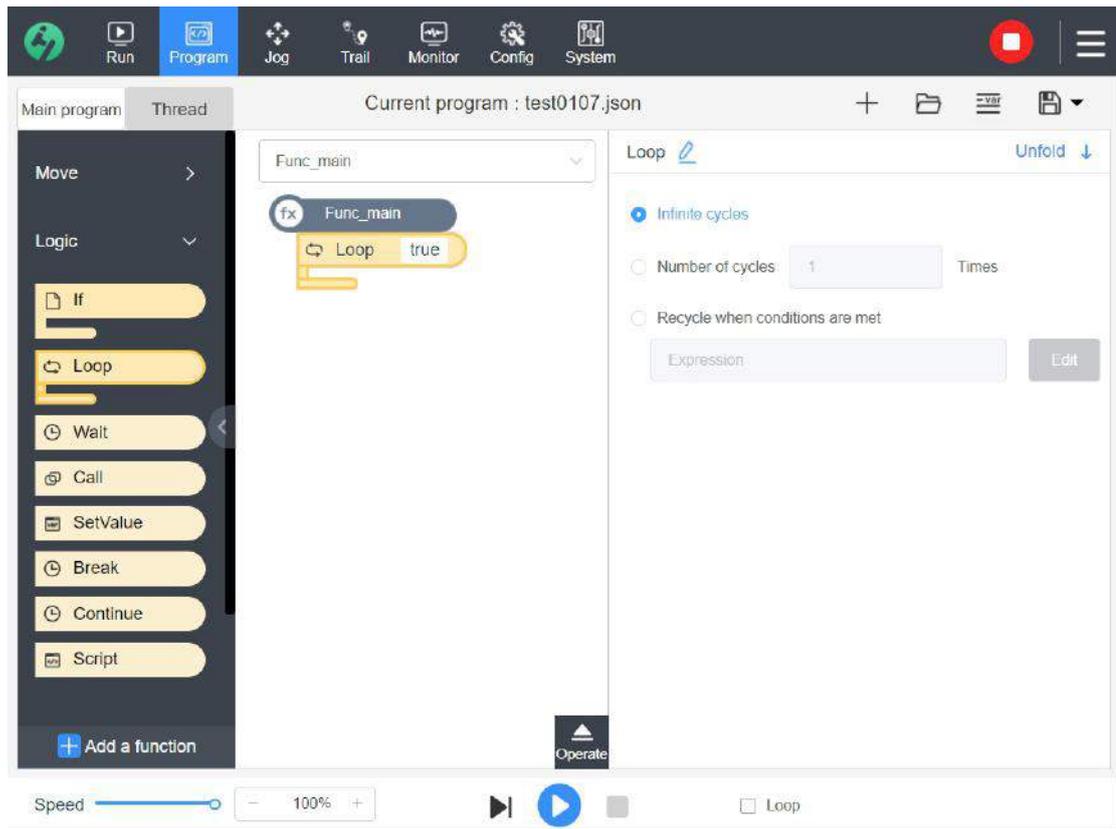
4.4.4   Break

When the loop executes a Break statement, it exits the entire loop and executes the statement outside the loop.

◆ End the current entire loop and execute the statement at the bottom outside the current loop. Ignore any other statements and loop conditions in the loop body.

◆ Only jump out of a loop; if the loop is nested, then you need to use the Break to jump out gradually according to the nested hierarchy.

◆ Break can only be used in Loop node;
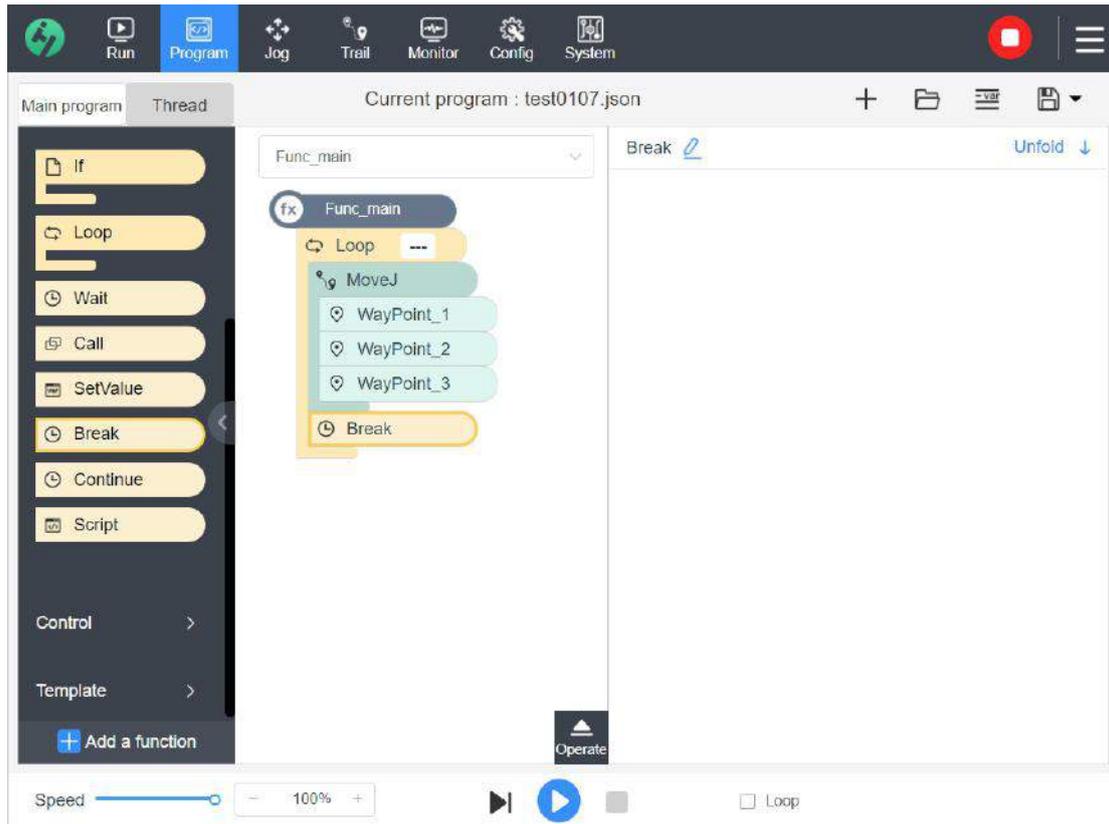
4.4.5 Continue

When the loop statement is executed to "Continue" statement, the current loop ends and the next loop starts. If it is the last loop, the "Continue" statement will be the same as the "Break" statement.

◆ Terminate the execution of this loop, that is, skip the statements in the current loop have not been performed after the continue statement, and then implement the judgment of next loop conditions.

◆ Terminate the current loop, but not jump out of the loop, but continue to judge loop condition and execute the statement. The system can only end one process in the loop but cannot terminate the loop and continue.

◆ "Continue" statement can only be used in Loop nodes;
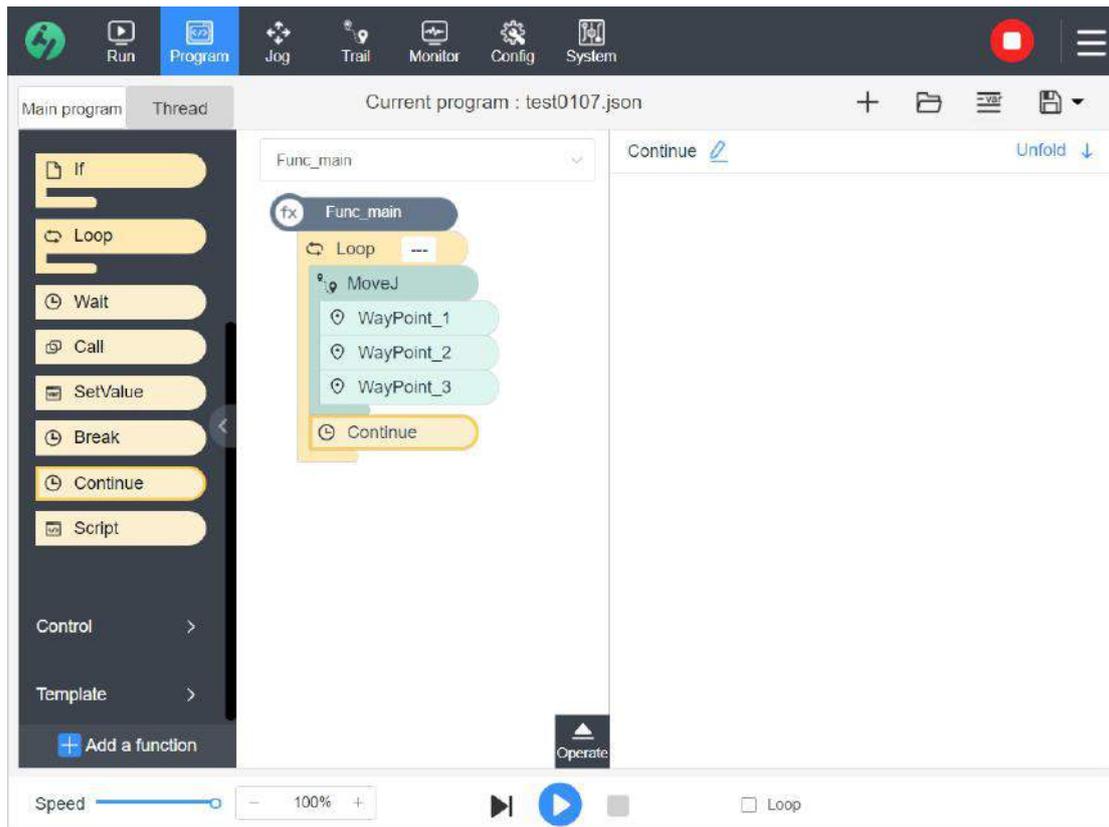
4.4.6 Call

It is a call command; the logic is to allow different functions to run interleaved by calling other functions while a function is running. After running the program line of the called function, it returns to the called function to execute the next line of the call.

A repeated operation can be written as a function for repeated call.

To make it easier to debug and read scripts, you can also divide different processes into different function modules and execute them by call.
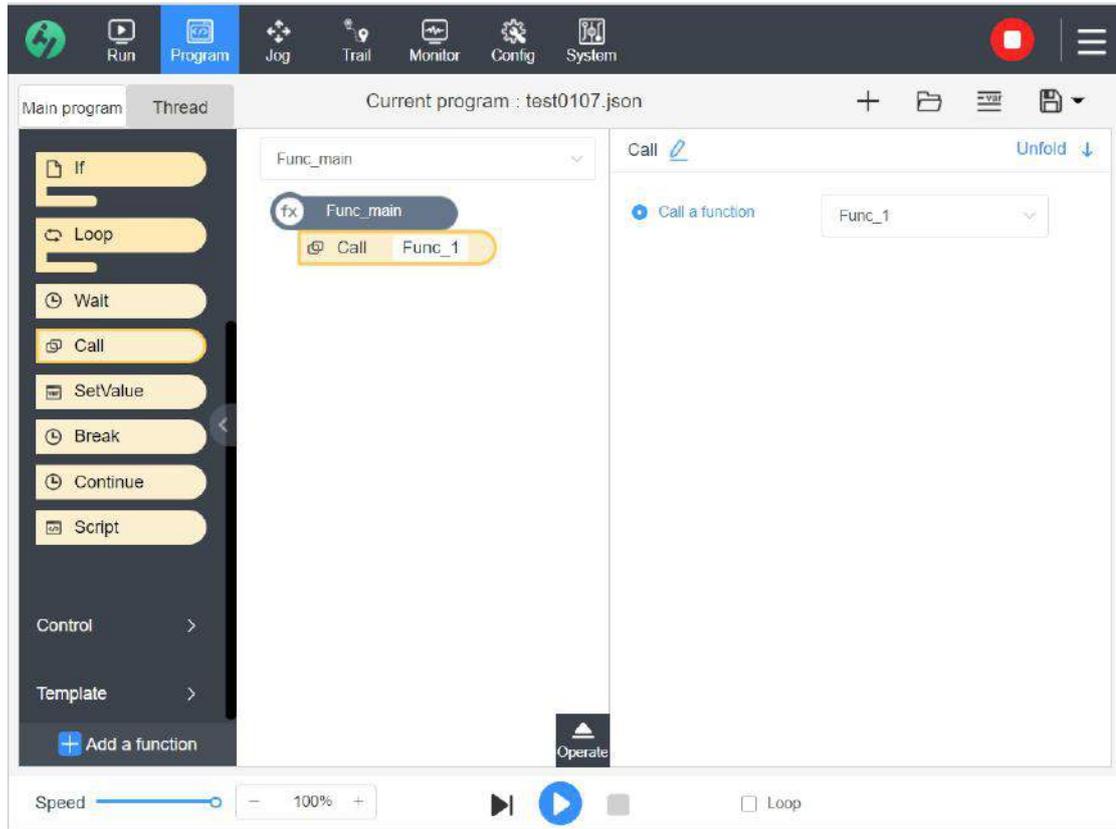
Call Type:

◆ Function List: Directly select a function from the drop-down box to call it; when the current function runs to call command, it jumps to the script of the function called to execute it;

◆ Function Variable: Commonly used to form call conditions according to the needs of single or multiple processes, used to match and perform different functions; when the value of a variable matches the name of a function, the function is called; this method can simplify the judgment conditions. The variable type is string.

4.4.7 Set Value
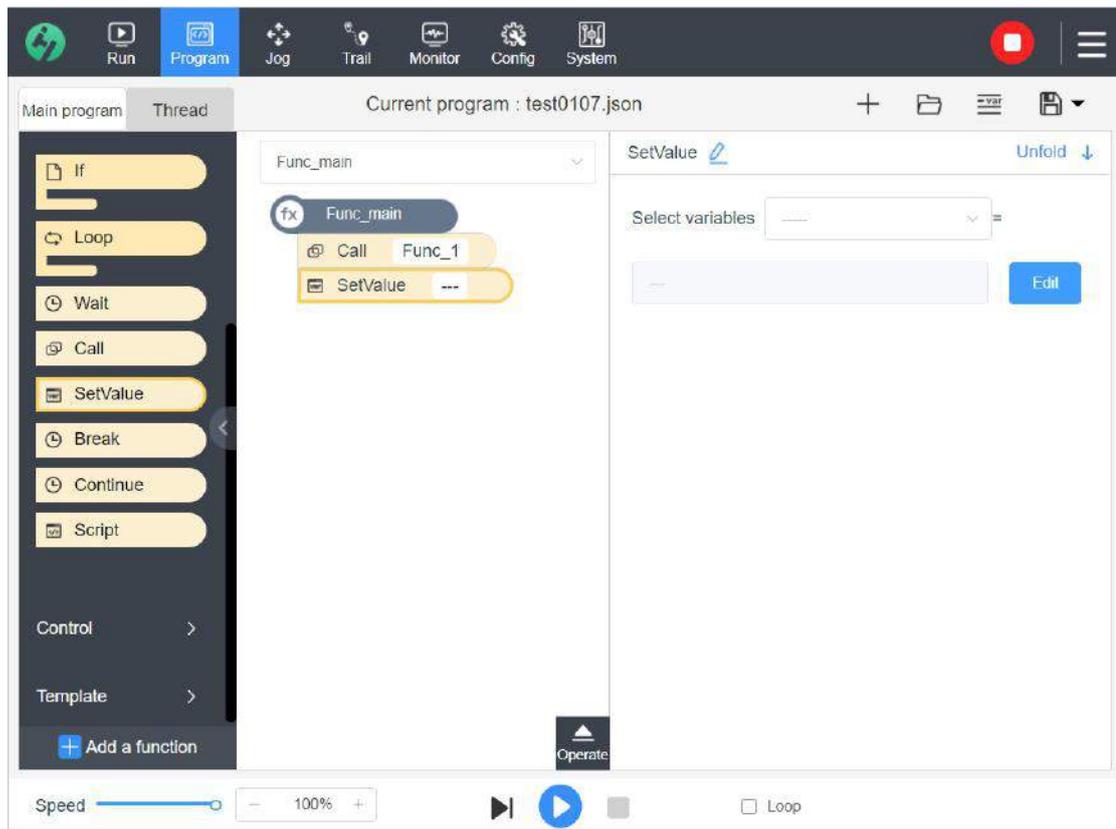
Assign a value to a variable. The new variable can be obtained by editing and calculated the expression and the currently selected variable can be assigned with a value. This command is useful in complex programs.

4.4.8 Script

Script files are used to access the underlying real-time scripting language executed by the robot controller. For example, point information acquisition, communication and exchange;
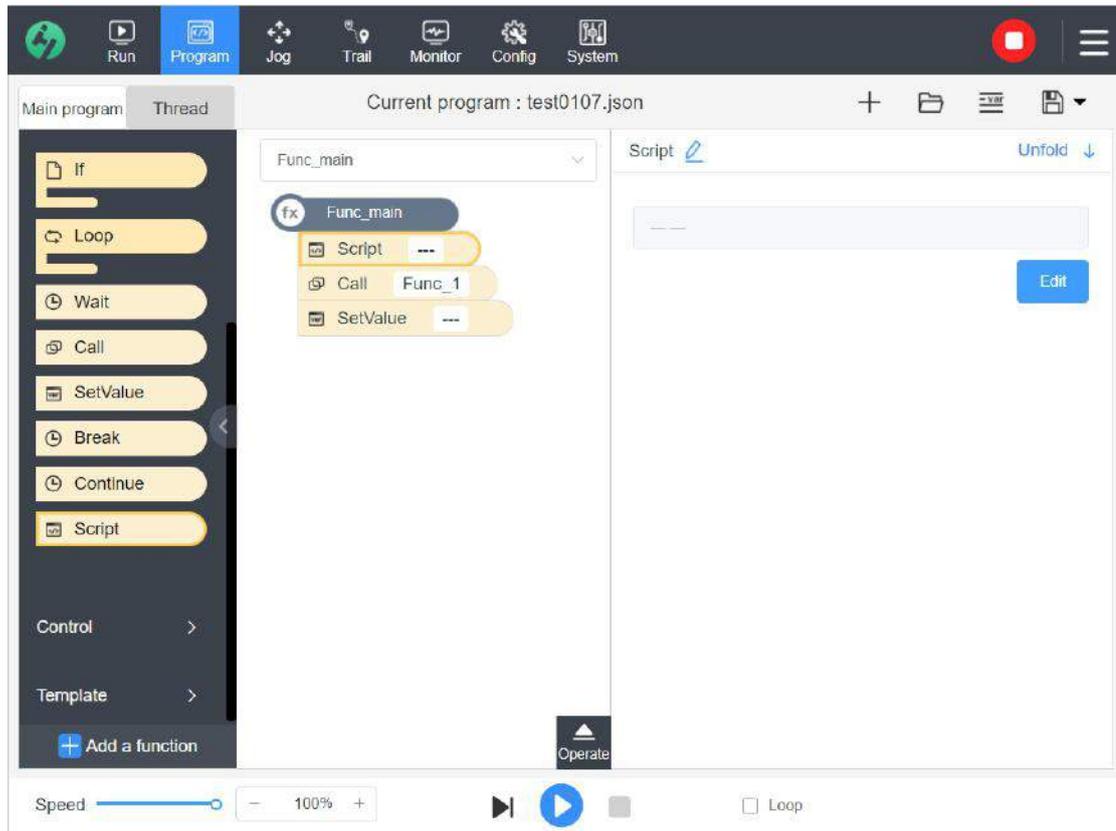
The user can import and export Python files from other sources and then edit them in the scripts, or create script files directly. The function of editing the file should be completed in the file editing interface, as shown in the following figure. The edited file needs to be saved before it can be referenced to the program. The file name will be displayed in the property information of Script.
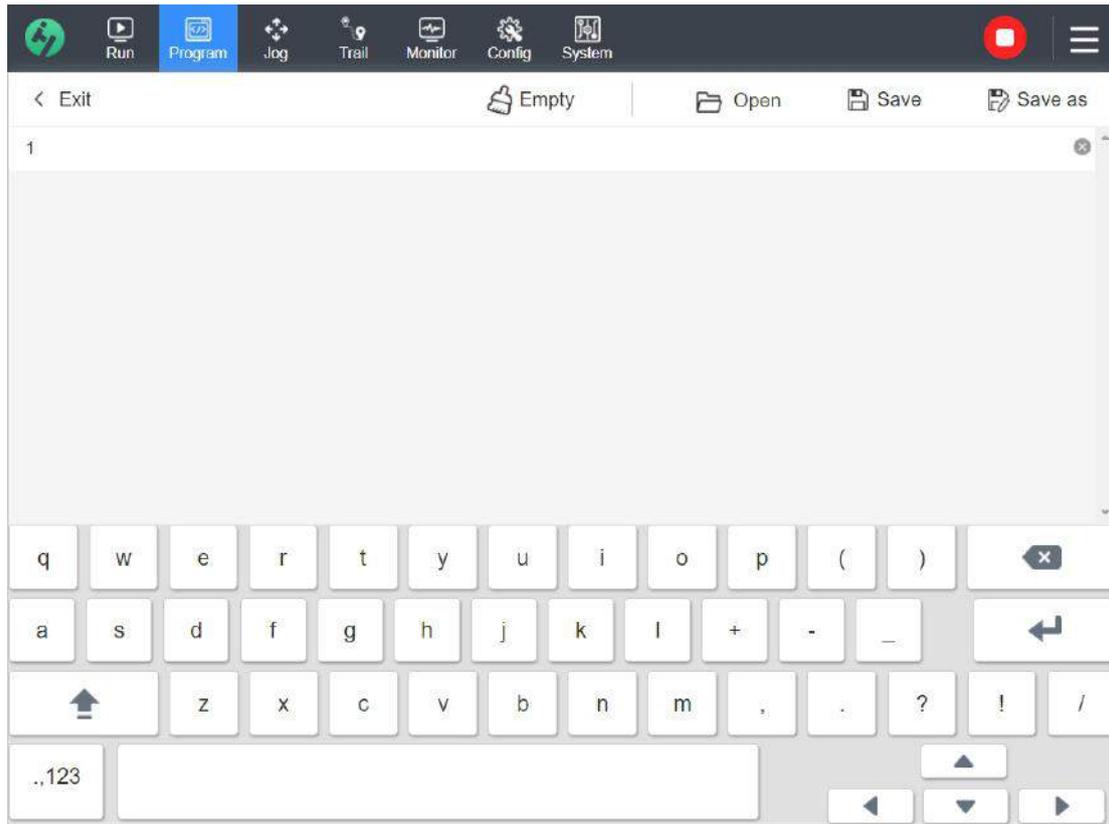
◆ Empty: Empty all the codes; there is a pop-up tip to confirm whether to empty them before emptying;

◆ Open: Open the file, import the files in the system default path or from other sources; if a script is currently being edited, there is a tip for the user whether to save the file now before importing;

◆ Save: Save the file to the default path or export the file by USB flash disk, and fill in the saved file;

◆ Save As: Save as file to the system default directory or export the file by USB flash disk, and fill in the file name;

◆ Exit: Exit the editing interface. When exiting, the system will detect whether the file has been saved. If not, it will remind the user whether to save the current file, and then exit. After exiting, the file name is displayed in the Script property, indicating that the file is successfully referenced to the program.

4.5 Control Command

4.5.1   Set IO

Set the output IO state of the electric cabinet, the end output IO state, the current or voltage value of the analog output channel of the electric cabinet as well as the current or voltage variable of the analog output channel of the electric cabinet. Modbus output state can also be set high or low level if modbus is currently configured for the system.

For example, set the output IO state of the electric cabinet, then the corresponding IO signal can be set high or low level. When the robot runs to the SETIO command, the IO state changes, and then it executes the subsequent commands.
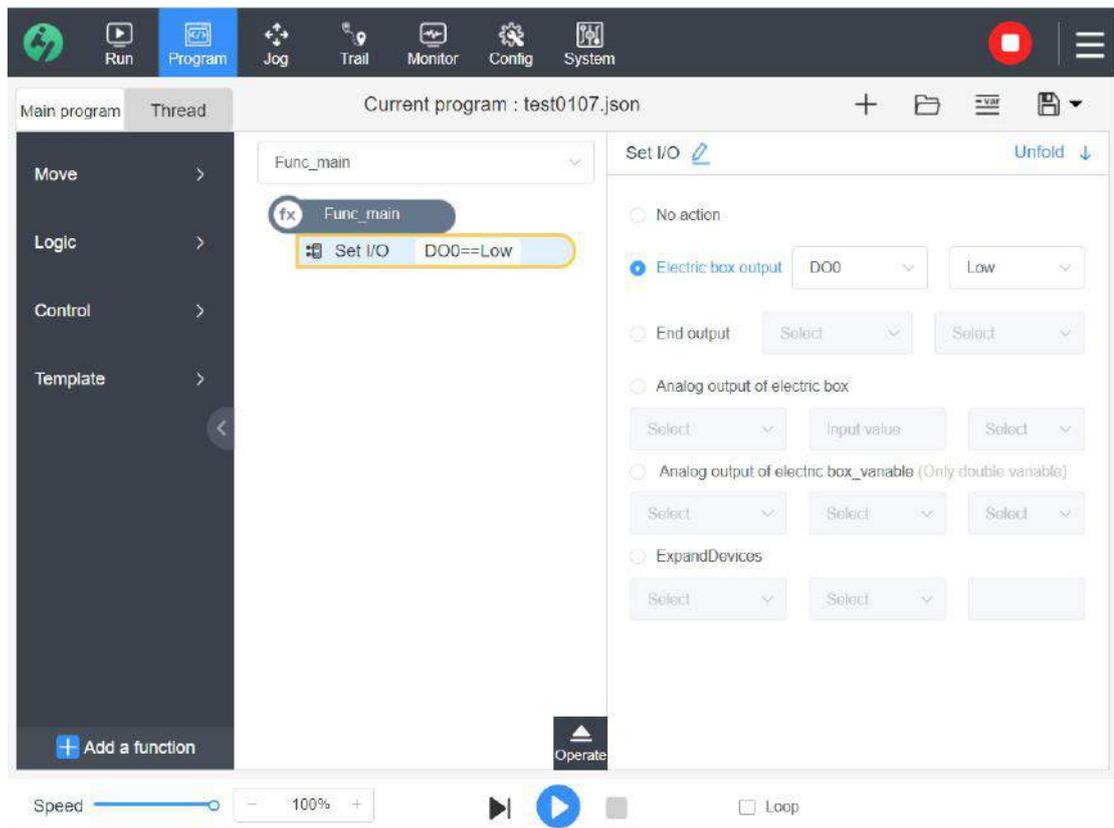
4.6 Model

4.6.1 MoveZ

The "Z"- path is used for welding path. The Z-movement plane can be determined and the Z-peak distance can be set according to the path of the starting and ending points for teaching.
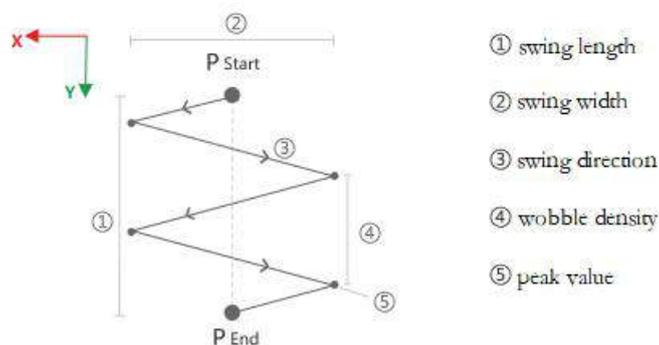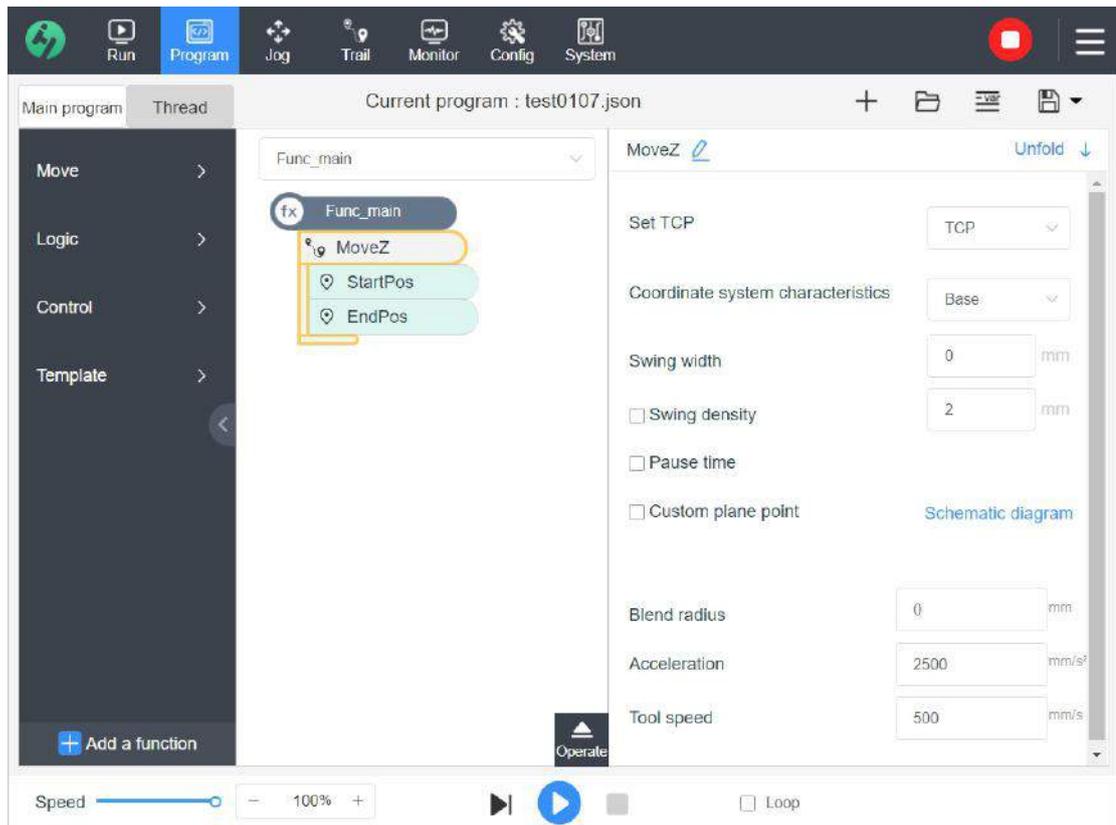
① Swinging Length: the straight-line distance that the robot moves from the starting point to the end point;

② Swing Width: the width that the robot moves in the current plane;

③ Direction of Motion: The robot moves along the direction of the Z-shaped position;

④ Density: the distance between the peaks on the same side; represents a motion period, and the density determines the motion period of the robot;

⑤ Pause Time: The robot will pause when it reaches a peak value. Different pause times can be set at the points on Side +X and -X;

Remarks

◆ The moving plane of the robot is perpendicular to the plane of Z-axis in the user coordinate system by default;

◆ Criteria for Side +X: P_{Start} and P_{End} line segments determine the + Y direction; based on the Z-axis direction of the user coordinate system, +X direction can be determined. +X direction determines the direction of movement from P_{Start} to P_2;

◆ The user can also "Custom Plane Point", namely a new point, namely, the plane made up by starting point, end points and third point serves as Z-motion plane of the robot, at this point, the direction of the third point is the +X direction;

◆ if not set Z-density, the current moving speed of the robot can determine the Z- motion period;

4.7 Thread
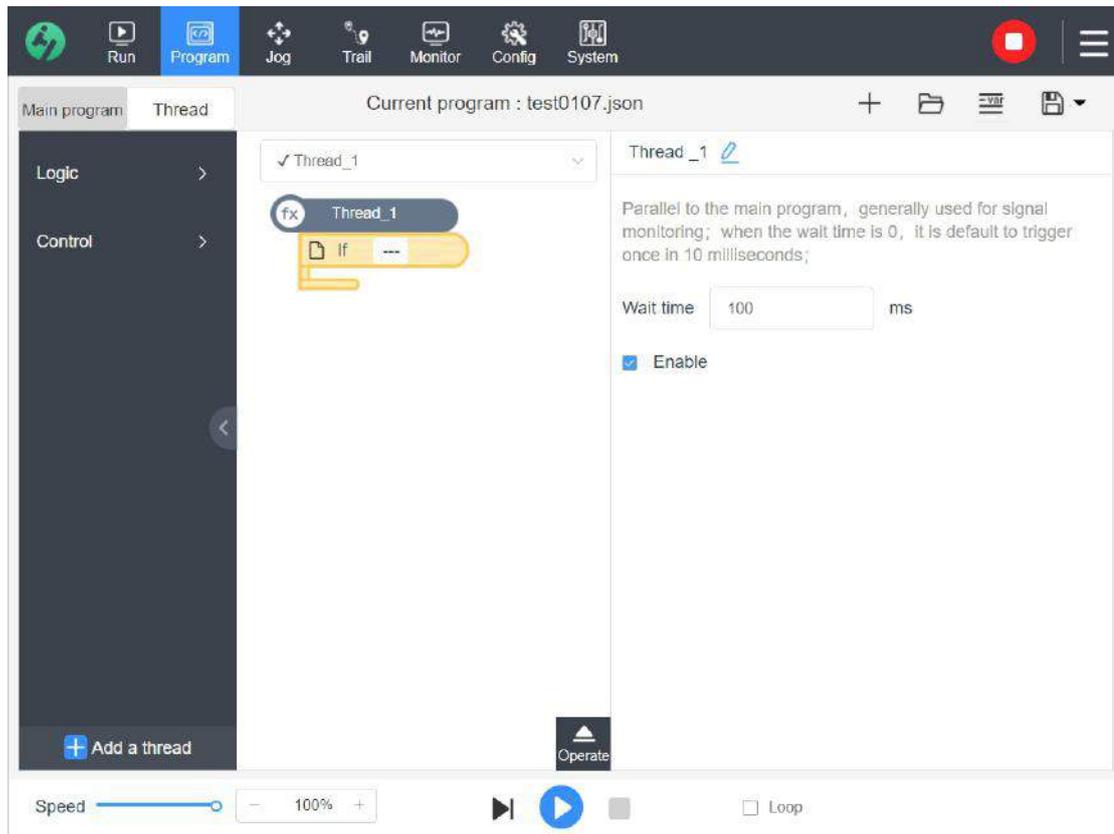
Thread is a parallel process of the robot's main program. Thread can be used to control external machines unrelated to the robot arm. Thread can communicate with the robot program through variables and output signals.　Timed execution, real-time access to IO status, etc.(Up to 5 threads can run simultaneously).

◆　Wait Time: For example, to input 0 ms means that thread has been parallel to the main program in loop execution;10ms indicates loop execution after thread timing for 10ms.

◆　Enable: If this option is checked, meaning that this thread is used; if not checked, then this thread is invalid; the checked thread shall be marked with "√" in the thread drop-down list.

4.8 Variable

4.8.1 Variable List

Edit variables to a maximum of 128.

◆ Add Variable: To create a new variable,

◆ Edit:【 ✐ 】 is used to modify the name and type of the selected variable, assign the
initial value, whether to save last running value;

◆ Delete: To delete checked variable. You can assign the value to the variable and update
the variable value through the setValue command at runtime of program.

◆ **Variables have the following types:**

(1) int variable: Range is −32768-32767.

(2) double floating point value: float-point number (decimal).

(3) string: character string.

(4) pos_position: position variable, the default value is [0,0,0,0,0].

(5) pos_joint: point-position variable, the default value is [0,0,0,0,0].

◆ **Whether to Save Last Running Value:** In the pop-up box of editing a single variable, if "Save" option is checked, the variable value will be recorded and saved in a configuration file in real time and the saved value rather than the initial value will be used when running the script again.

4.8.2 Current Variable

The current variable list includes the variables used by the user when editing the program; when the user debugs the script, the current value of the variable will be updated to the assigned value in real time according to the execution of the script. The user can verify whether the assigned value of the variable is correct while debugging the script.

4.9 Debug and Run

After programming, check the program debugging and confirm whether the program runs successfully; in case of running failure, there will be corresponding prompts.
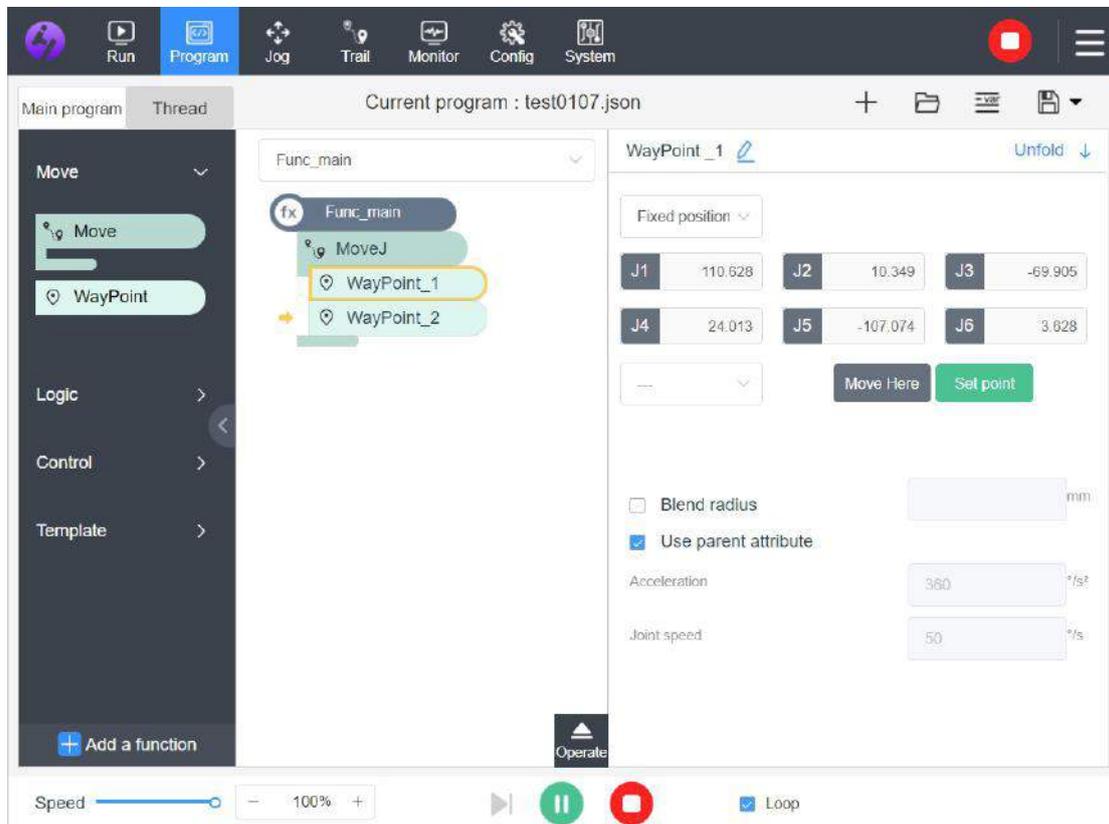
4.9.1 Run

After complete the program editing, select the function to be debugged in the function drop-down box to run it.(The debugging will run from Func_main)

◆ If the current position of the robot **is the starting position of the program**, the system will start to run the program directly after "Run ▶ " button is clicked;

◆ If the function of **testing "program starting point" is enabled** in the configuration, then, if the current position of the robot **is not the starting position of the program**, a window will pop up after you click "Run ▶ " button. The program won't run until the robot moves to the starting position. **Skip the process if this feature is not enabled.** Here are the steps of the

process:
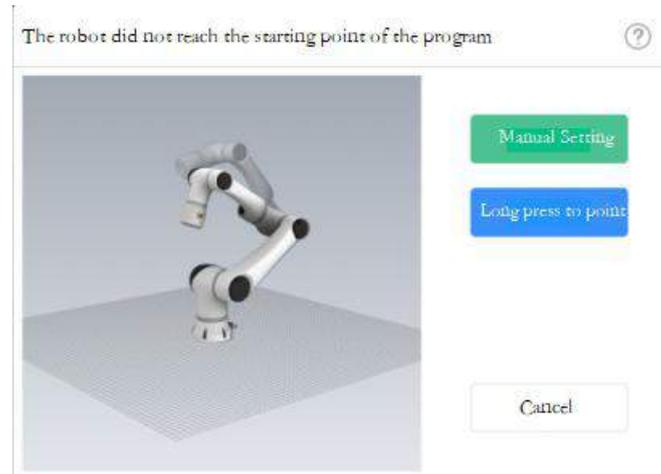
A. **3D model in pop-up window:** The animation from the current position to the starting position is displayed in the model to facilitate the user to understand the moving path of the robot from the current position to the starting position. The user can judge whether this path has a potential safety hazard accordingly so as to choose whether to press "Manual Setting" button to make the robot reach the safe position or press "Press and Hold to Point" button so that the robot automatically reaches the target position, which can save time.

B. **Manual Setting:** Jump to the inching interface to get the robot to move to the starting point; this function is usually used in the scene where there are many obstacles around in order to ensure safety; there is no guarantee that the robot can reach the starting point after the star zero-force teaching due to the deviation, so it is necessary to let the robot automatically back to the starting point by pressing "Press and Hold to Point" button after the problem of security is solved;

C. **Press and Hold to Point:** When you click this button, the robot will automatically arrive at the starting point of the program from the current position. When it arrives at the position, a 3S countdown will appear in the window, and the program will automatically start running after 3S and the pop-up window will disappear;

D. **Cancel:** Cancel running the program. This operation can also be performed during the countdown.

Note 1: If there is an instruction in the program with no parameter set, it cannot be executed successfully;

When the user clicks [Run] button, a pop-up window prompts that there are instructions not set in the program and the program cannot be successfully executed. At the same time, instructions in question will be highlighted in the program node to be easy for the user to find and modify them.

Note 2: If the program has a parameter error, the program may be not executed.

4.9.2 In Execution

After successful execution of the program, we can view the change of the variable value in the window at the lower right corner; when the current instruction is executed, it is followed by a yellow arrow.
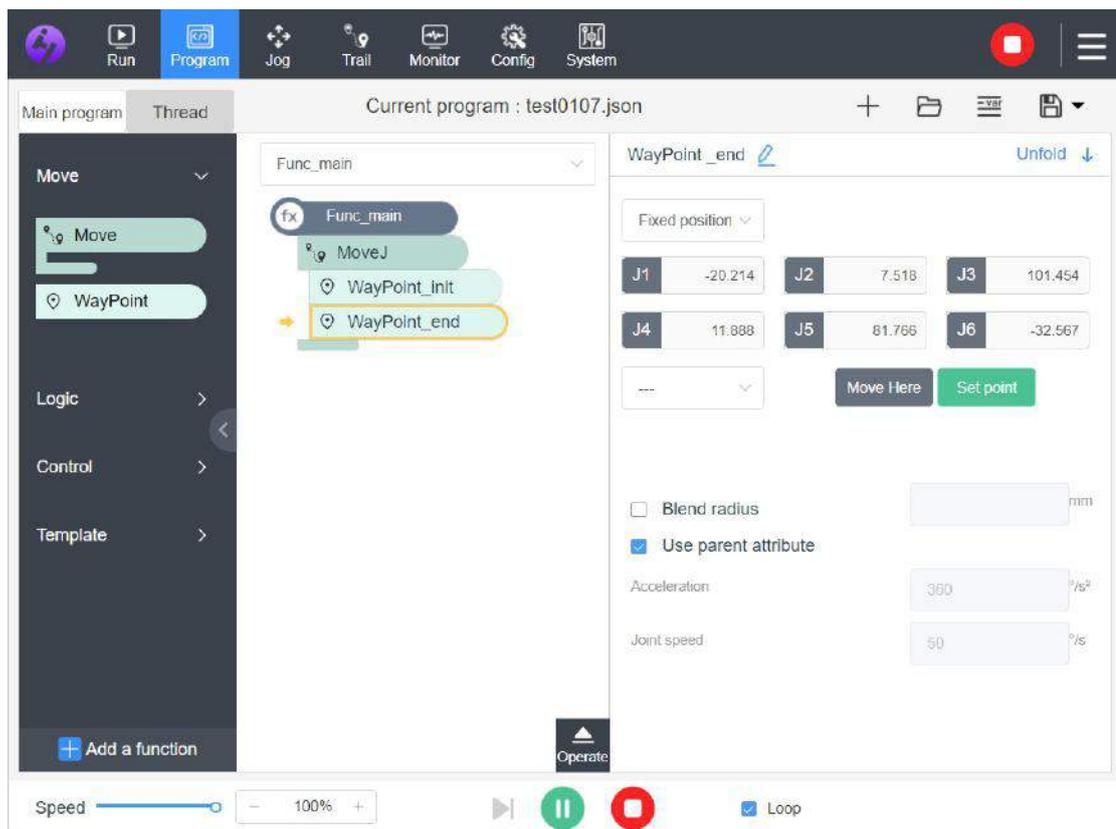
◆ **Pause:** After the program runs, you can click [Pause ⏸] button to pause the program. At this point, the button will change back to the "Run" button and if you click it again to

continue to execute the program.

◆ **Stop:** After the program runs, you can click [Stop ⬛] button to stop the execution of the program.

4.10 Program Example

In robot work, the most basic and commonly used command is the movement command. Here is a simple pick-and-place program. (The current end tool is clamping jaw.)

1、 Add a program to the programming interface under enabled condition. [File→ New → Naming Program → Confirm New]

2、 Under the Func_main function, drag a Move command from the command menu to the function node. At this time, WayPoint_1 will be automatically added under the Move node; the motion mode is modified to MoveL. Select an appropriate TCP for the current end clamping jaw (TCP needs to be set in the configuration in advance); the coordinate system feature is Base by default, and set the appropriate motion speed and acceleration for the waypoints under the node.
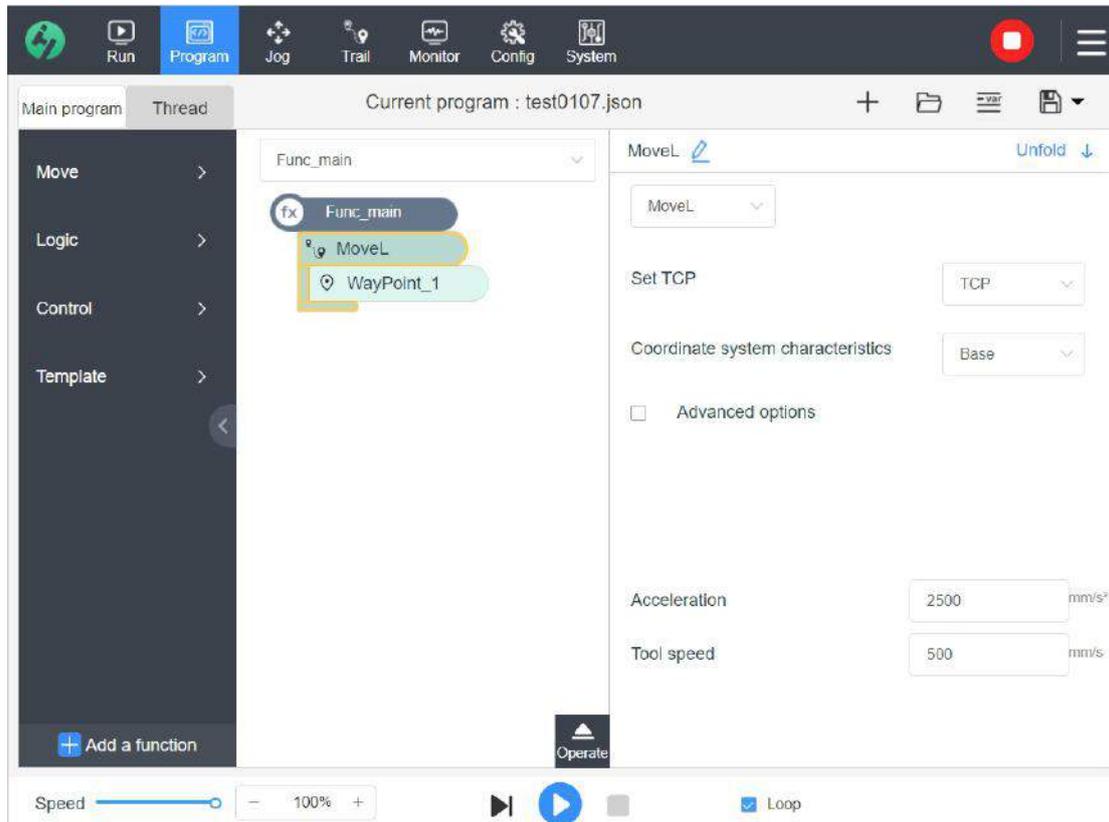
3、Teach WayPoint_1. If the desired point points have been saved in the point list, it can be directly selected for use from the drop-down box of the point list on the current interface. WayPoint 1 is used as the starting point of the robot.

4、If you do not want the robot to stop in way from WayPoint 1 to WayPoint 2, you can check the blending radius and set the radius parameter. The radius cannot exceed the distance between two points.
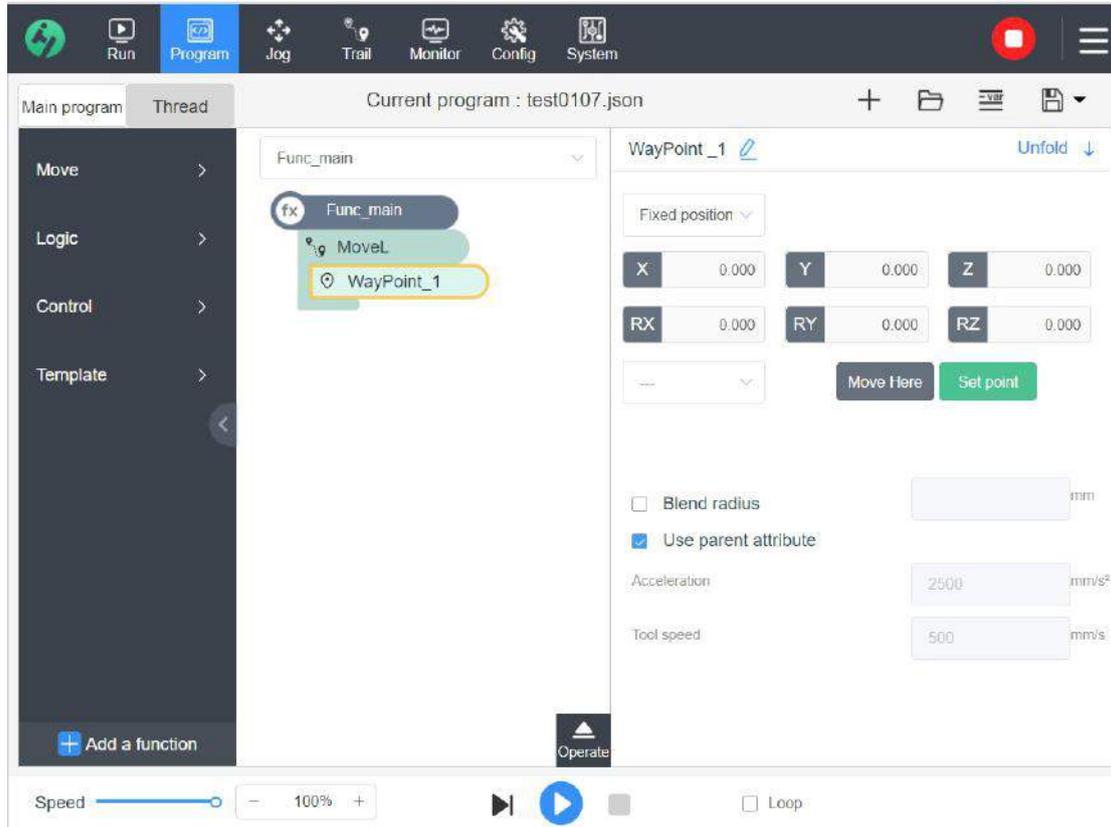
5、Drag a waypoint from the command menu to WayPoint_1 to automatically name it WayPoint_2 and demonstrate it. WayPoint_2 is above the location of the material. In the MoveL motion mode, the robot needs to move from WayPoint_1 to WayPoint_2 to prepare for grabbing the material.
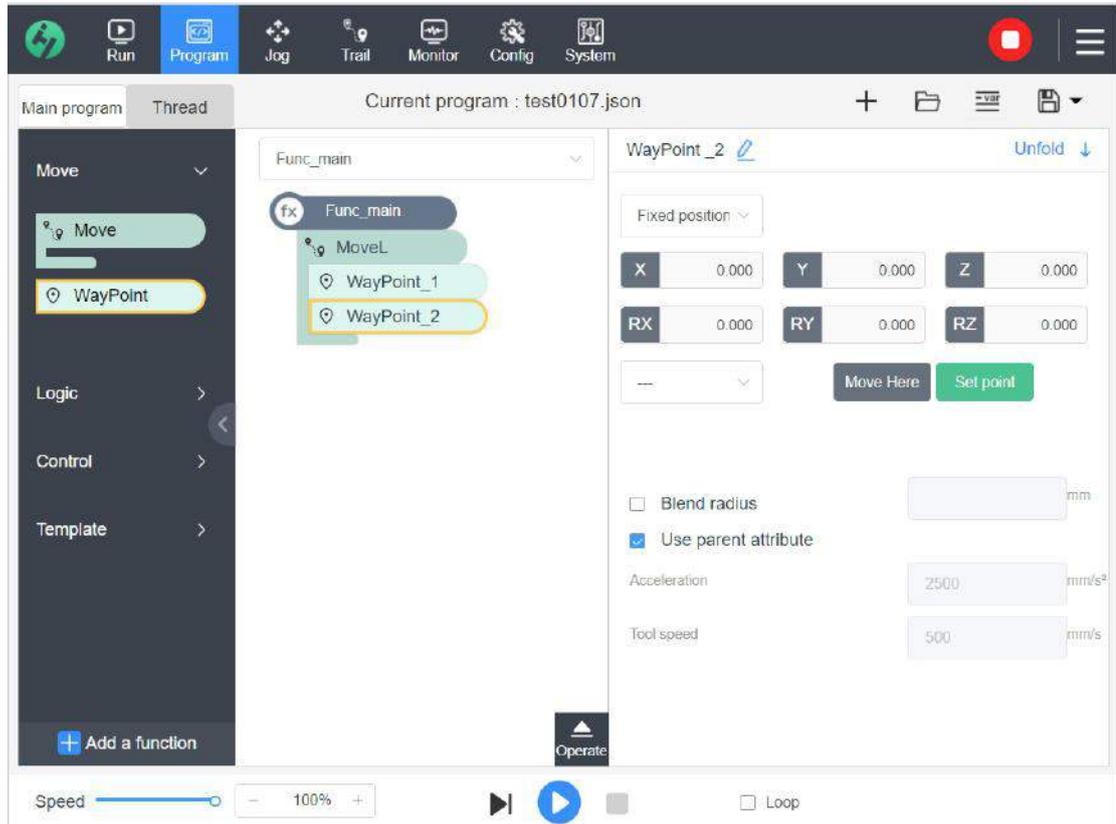
6、Drag a waypoint from the command menu to WayPoint_2 to automatically name it WayPoint_3 and demonstrate it. WayPoint_3 is the target point where the material is located. When the clamping jaw reaches the material location, the jaw is ready to be closed to grab the material.
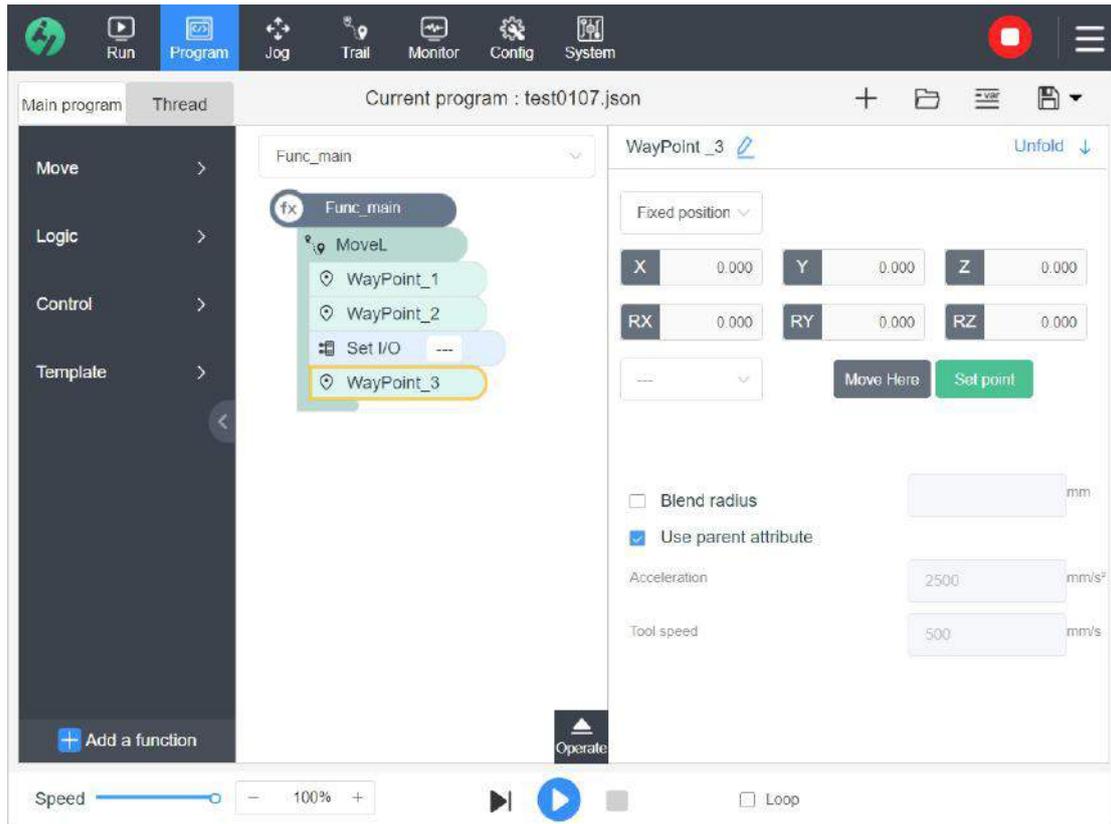
7、Drag Set IO from the command menu to WayPoint_3 and set the electrical cabinet output DO0= High. Close the clamping jaw to pick up the material.
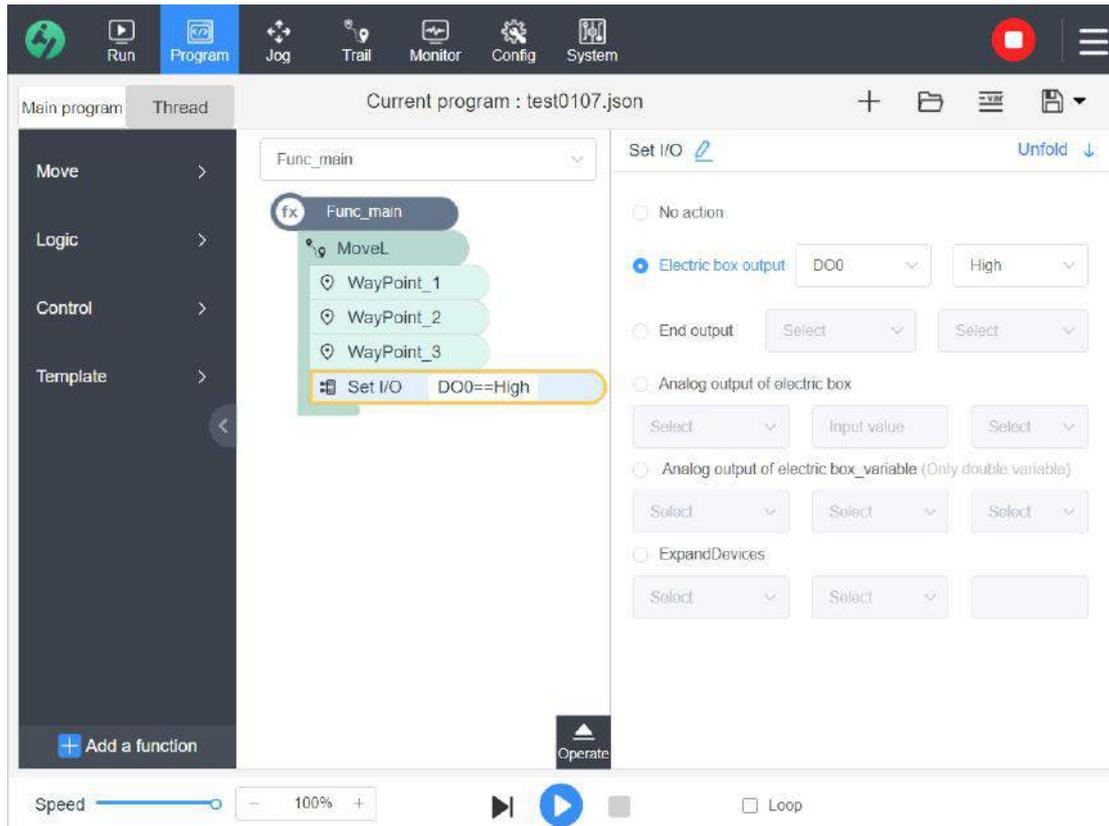
8、Drag a waypoint from the command menu to SetIO to automatically name it WayPoint_4 and demonstrate it. WayPoint_4 is the safe position of transition and the robot will pick up the material and lift it to the safe position.
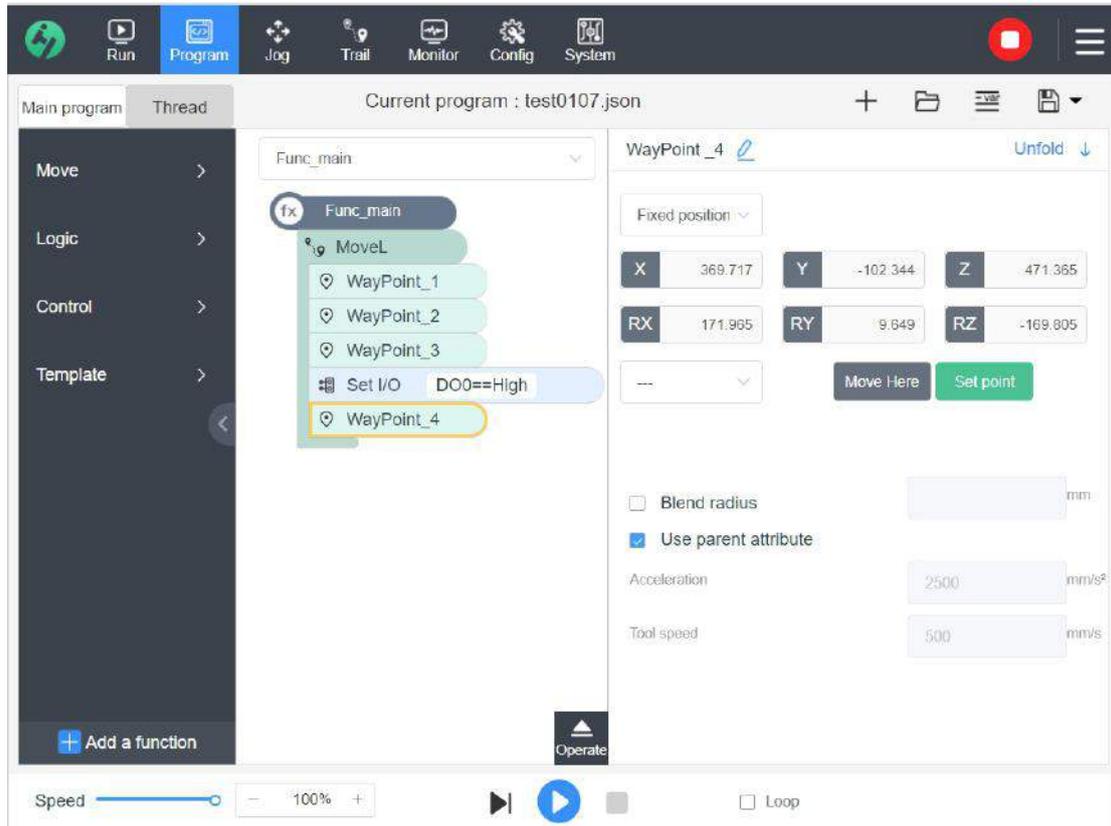
9、Drag a waypoint from the command menu to WayPoint_4 to automatically name it WayPoint_5 and demonstrate it. WayPoint_5 is the final position where the material is placed. The robot needs to move from WayPoint_4 to WayPoint_5 and arrive at the final position where the material is placed.
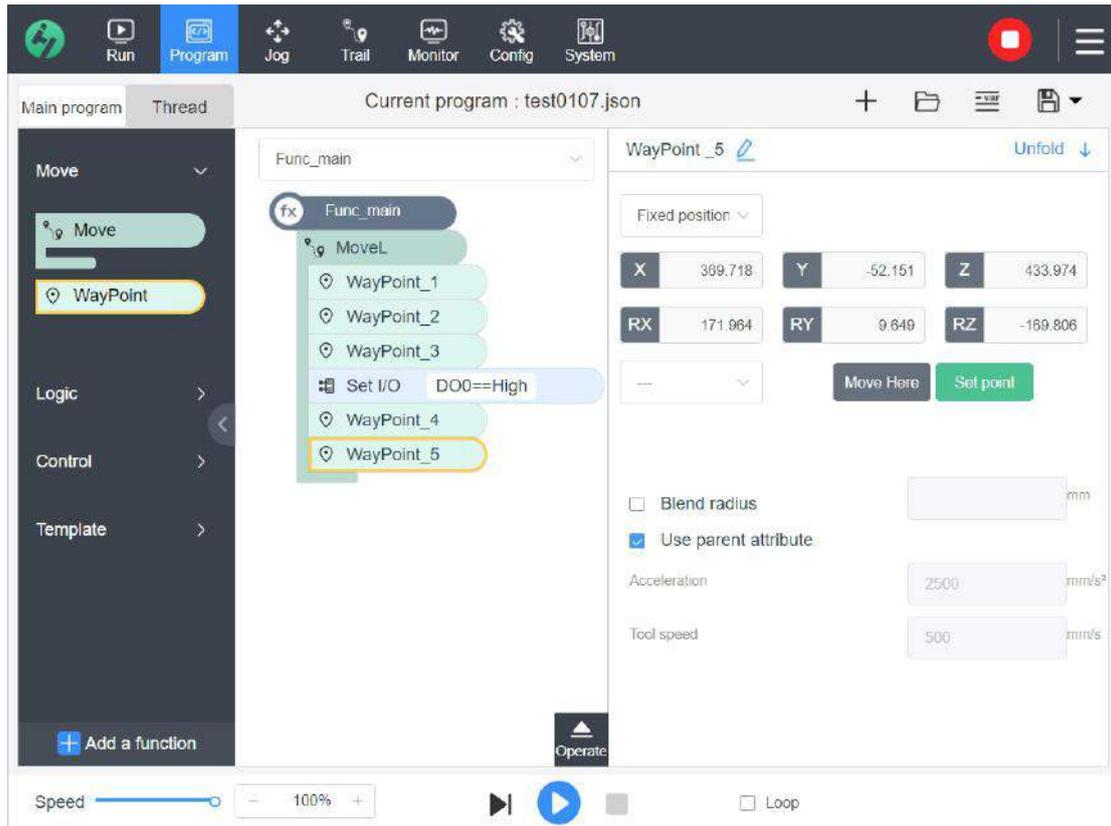
10、 Drag SetIO from the control command of the command menu to WayPoint_5. Set the electrical cabinet output DO0= Low (IO connected with the current clamping jaw is DO0). Open the clamping jaw to place the material.
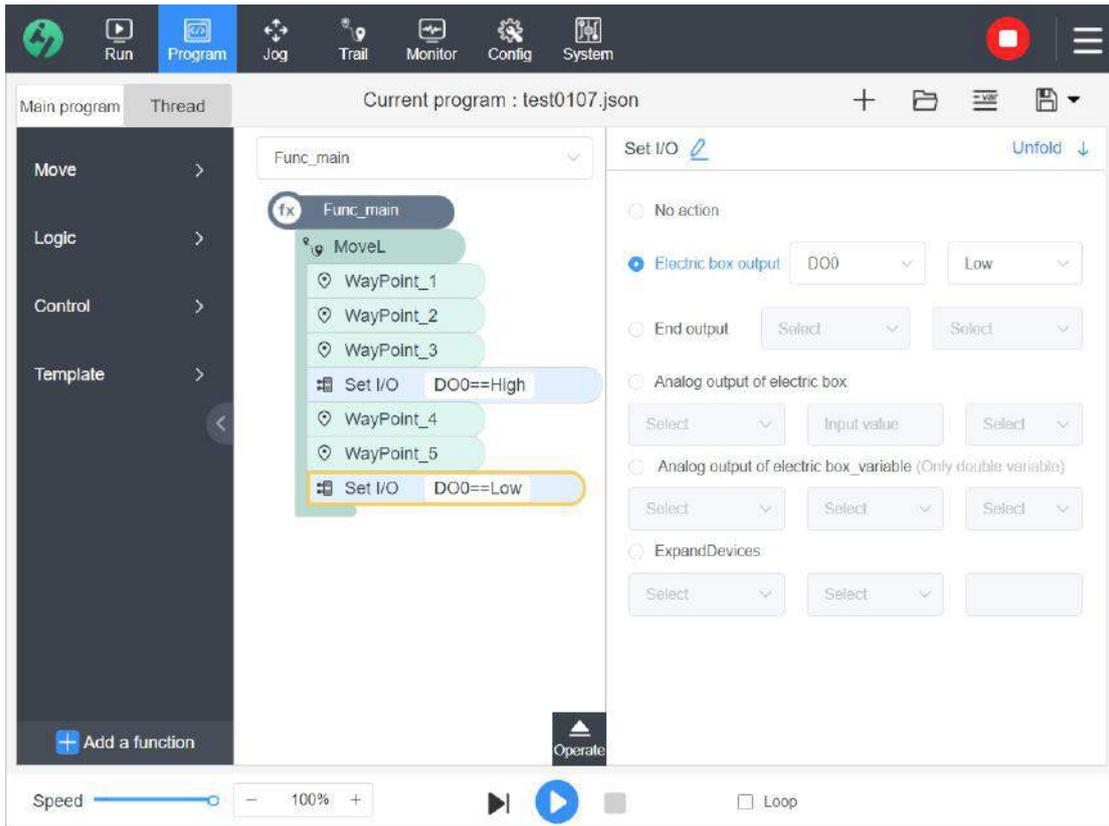
11、The robot returns to the safe position.
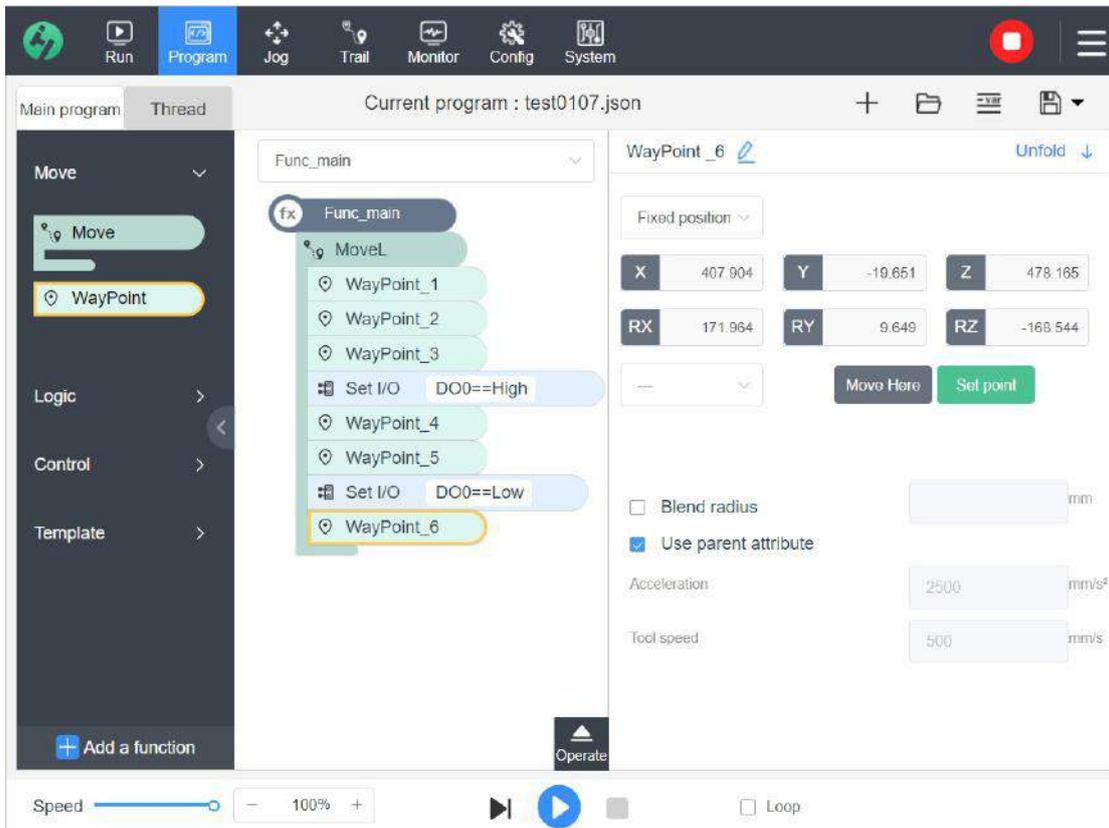
The above is a program for picking up and placing materials and some transitional safety

points are added in it to prevent collision with the surrounding obstacles during the MoveL linear motion. After editing the program, click "Run" button to execute the program. After checking the loop option, the program will be executed in way of loop.

Chapter 5 Run

In the running interface, the user can run the saved files directly and this is suitable for ordinary operators to make the robot work quickly;
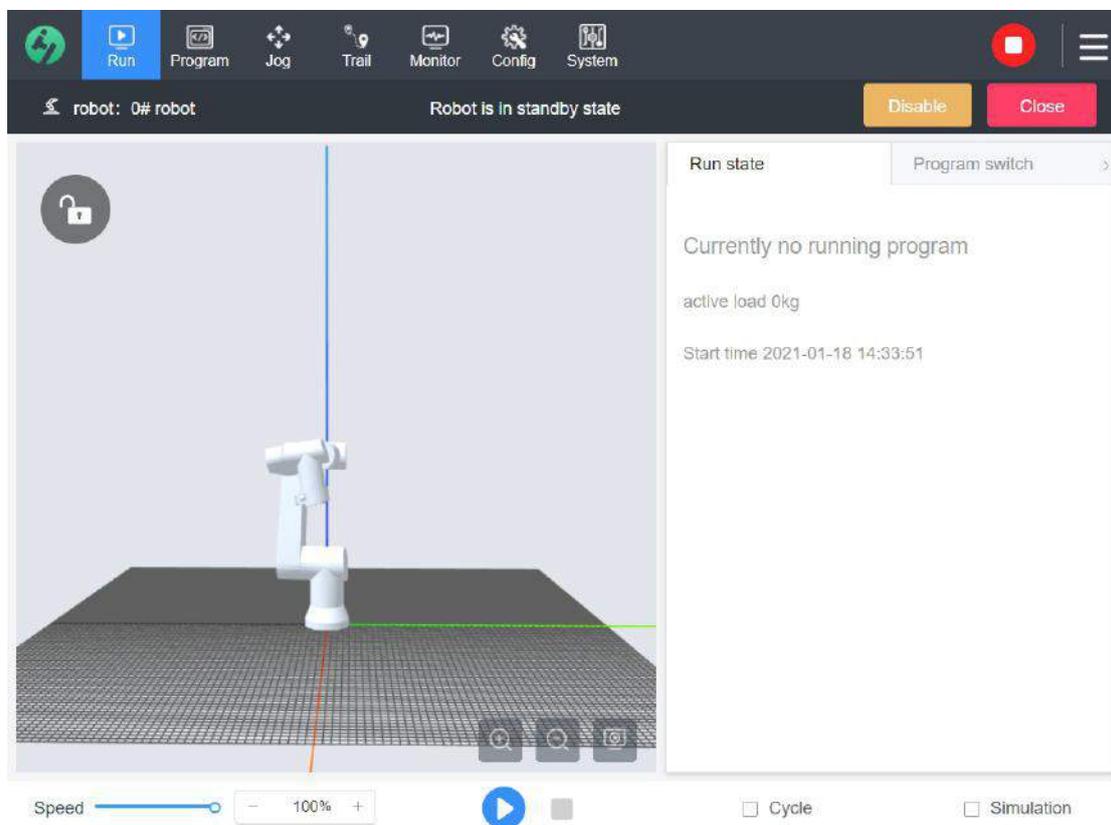
5.1 Initialization of Robot

◆ Name of current robot: The user can modify the robot name in the robot attributes of the "Management" module, convenient for purpose division of multiple robots in the same

production environment.

◆ Boot, power-on and enabling operation of robot: <u>Please see "1.3 Initialization of Robot"</u>
<u>for details</u>

5.2 Select a program for running
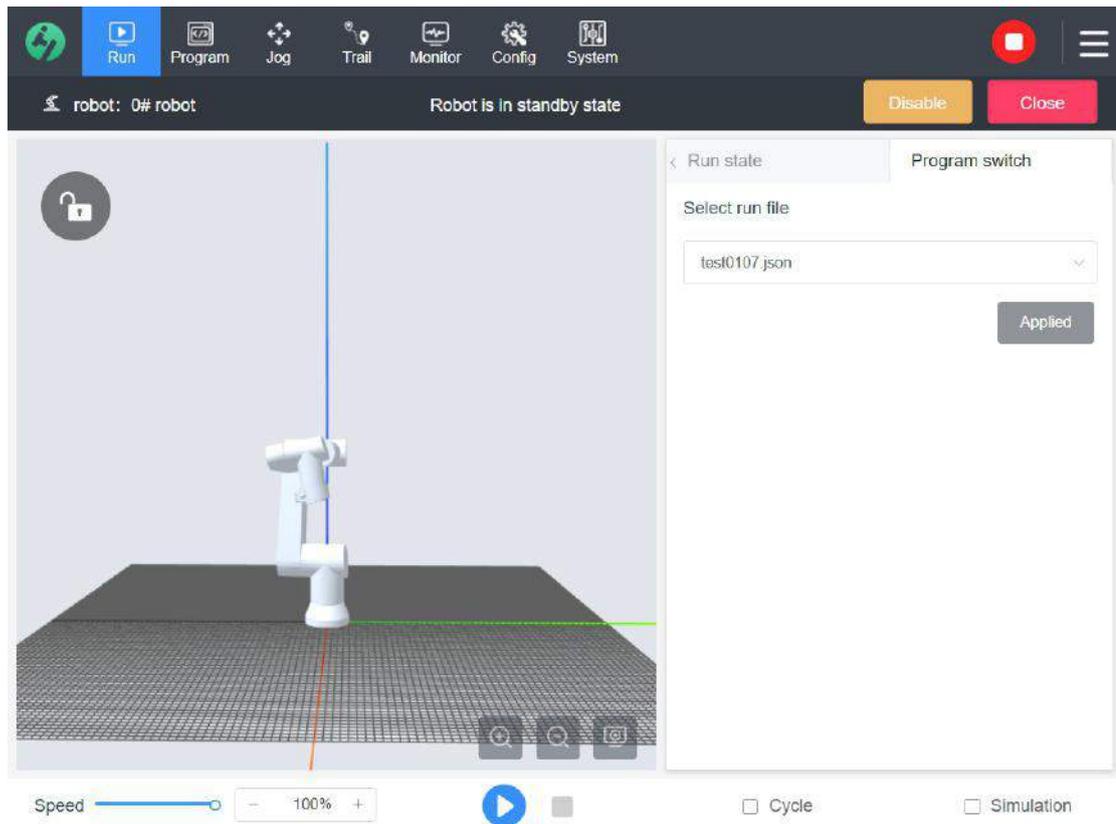
Select the file to be executed from the drop-down box and apply it to the robot. If a program has been set as the "Default Program", the system will automatically load the default program after power on, and the name of the loaded default program will be displayed in the drop-down box;

◆ **Run:** After the program loading is completed, if the current position of the robot is the starting position of the program, the system will start to run the program directly when the user clicks "Run ▶ " button;

◆ If the function of testing "program starting point" is enabled in the configuration, then, if the current position of the robot is not the starting position of the program, a window will pop up after you click "Run ▶ " button. The program won't run until the robot

moves to the starting position. If this function is not enabled, then skip this process. **(See 4.9.1 Operation Description for details)**

◆ **Pause Running:** After the program runs, you can click [Pause] button to pause the program

◆ **Stop Running:** After the program runs, you can click [Stop] button to stop the execution of the program

Note: If the program contains an enabling instruction and the system needs to execute the program when it is not enabled, it is necessary to manually enable the program to return to the starting point to run.

5.3 Running Status

The state display of the robot during operation facilitates the understanding of the working time of the robot, including the name of the script file that the robot is running, the payload, running time, execution time, number of loops, etc.;

5.4 3D Model of Robot

It will be displayed synchronously with the robot; please see "2.10 3D Model" for detailed introduction
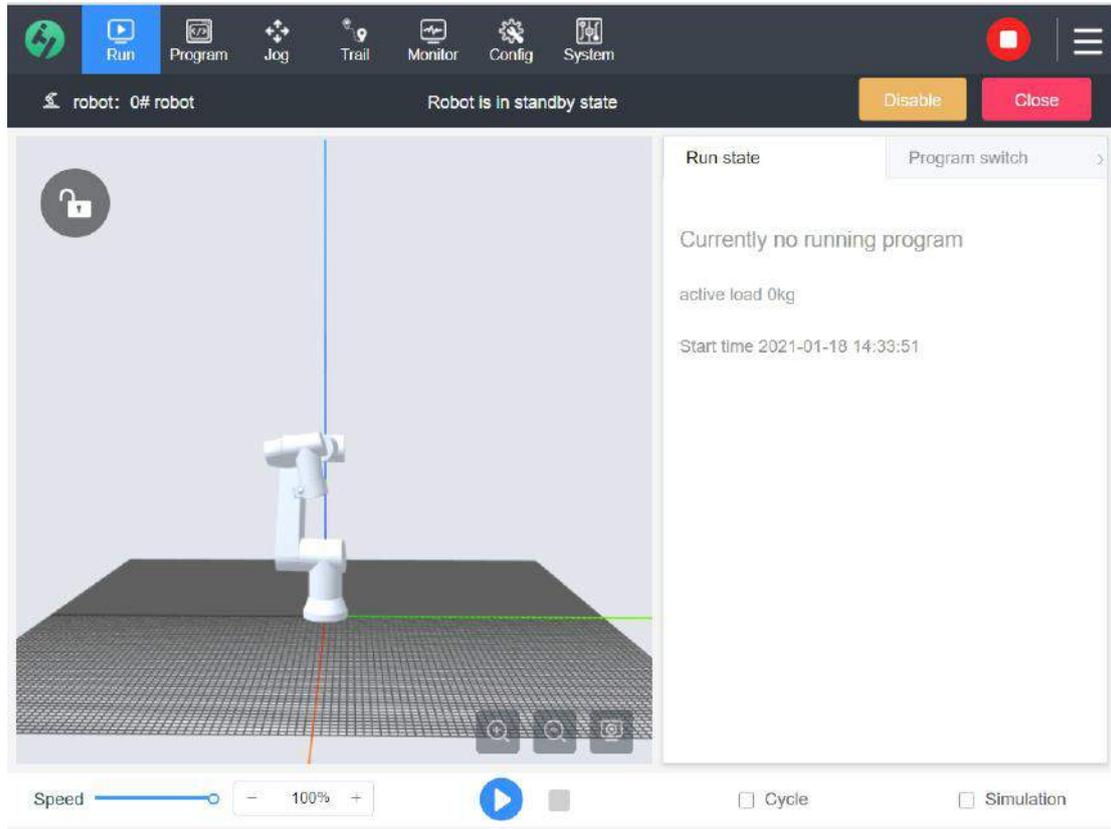
5.5 Lock

After the safety lock, the interface cannot be operated, and will recovery after being unlocked with the password in order to effectively prevent the robot from being misoperated during operation;
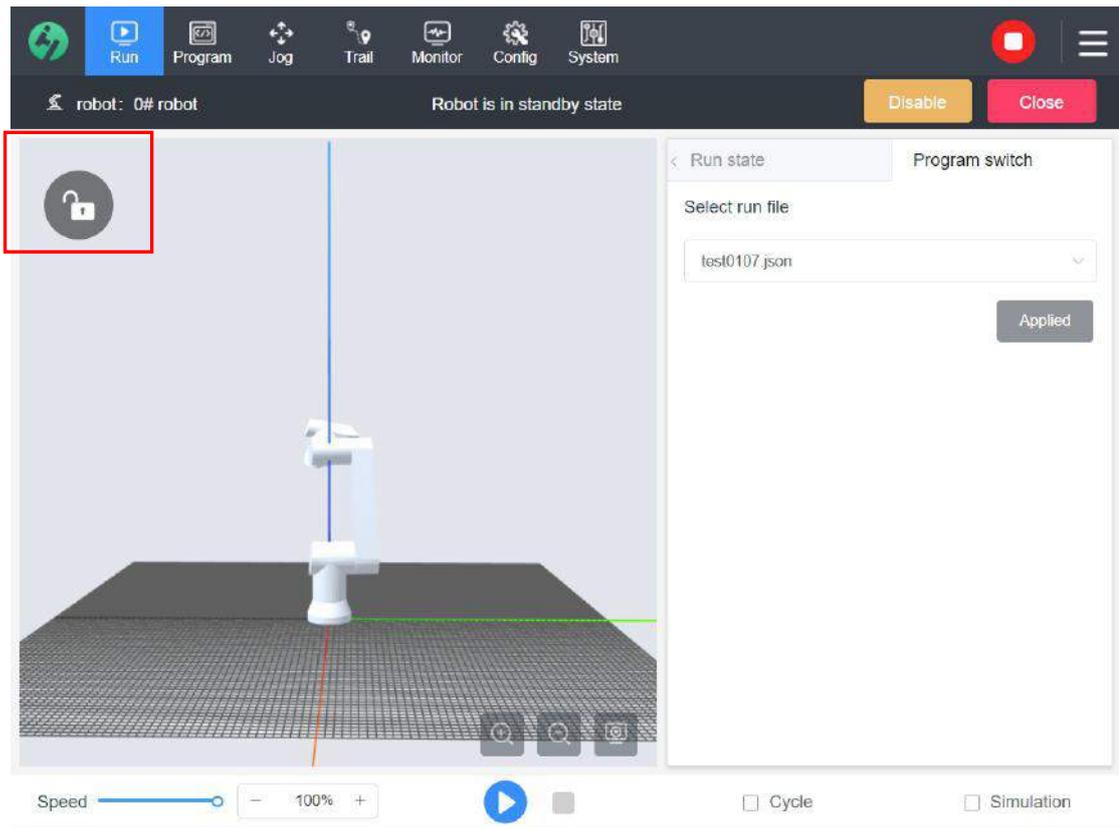
5.6 Bottom Operating Space

In this operating space, the use can adjust robot movement speed, run/pause/stop the script, switch simulated/real robot

Real robot: It indicates that the system is connected with the robot hardware. It is necessary to pay special attention to the position safety of the robot and its surrounding environment when teaching.

Simulated robot: It indicates that the system is connected with the simulated robot, and its function is no different from that of the real robot. We do not need to consider the safety of robot hardware and this is easy for quick demonstration.

Chapter 6 Monitoring

In this interface, you can monitor and set the real-time I/O signal of the electric cabinet. The interface displays the current state of I/O, including the I/O state while the program is running. When the program stops running, all output signals remain in its states. Due to the limited refresh frequency of the interface, signals that change very quickly may not be displayed correctly.

6.1 Electrical Cabinet IO

6.1.1 Input and Output
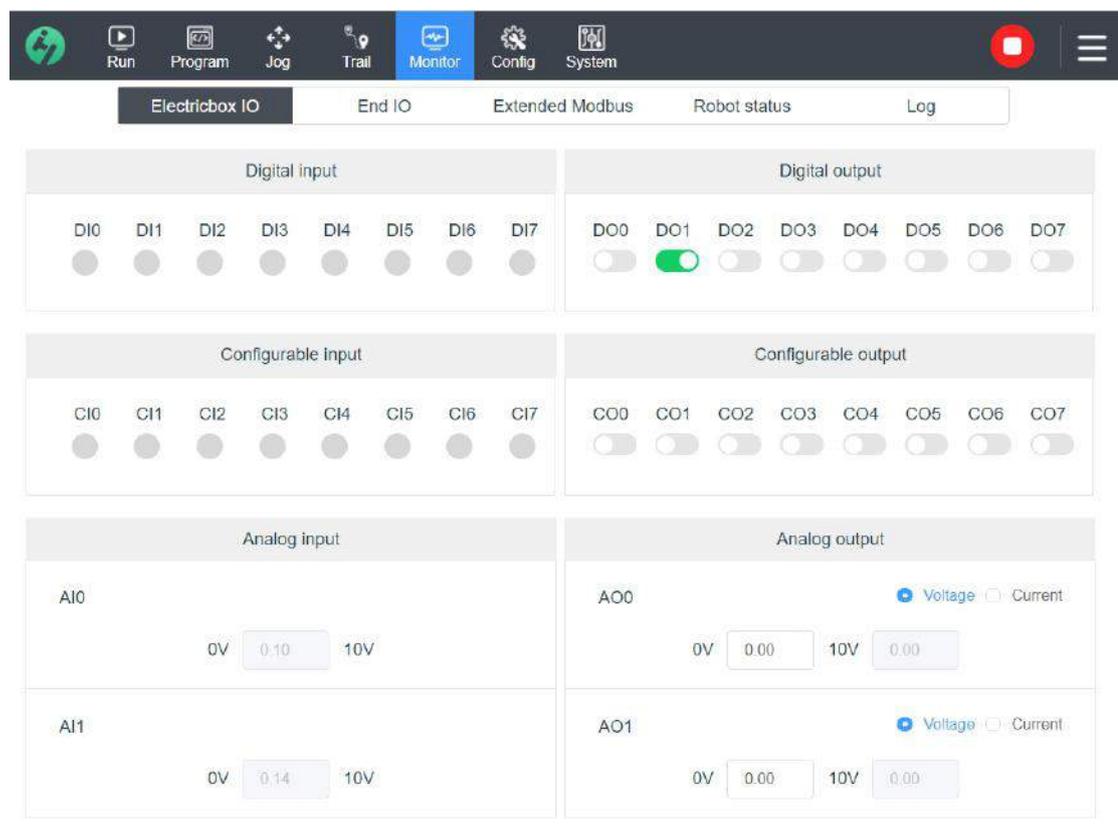
◆ General input: DI0 ~ DI7, 24V power supply. If the input is high level, the corresponding circle will turn blue, while the gray circle indicates low level.

◆ Configurable input: CI0 ~ CI7, 24V power supply. If the input is high level, the corresponding circle will turn blue, while the gray circle indicates low level.

◆ Analog input: 2 channels, analog current range: [4-20 mA], analog voltage range: [0 -

10V]

◆ General output: DO0 ~ DO7, 24V power supply. Slide the corresponding output switch ⬤ to set the configurable output high level; at the moment, the switch turns green ⬤ .

◆ Configurable output: CO0 ~ CO7, 24V power supply. Slide the corresponding output switch ⬤ to set the configurable output high level; at the moment, the switch turns green ⬤ .

◆ Analog output: 2 channels, analog current range: [4-20 mA], analog voltage range: [0 - 10V]

6.2 End IO
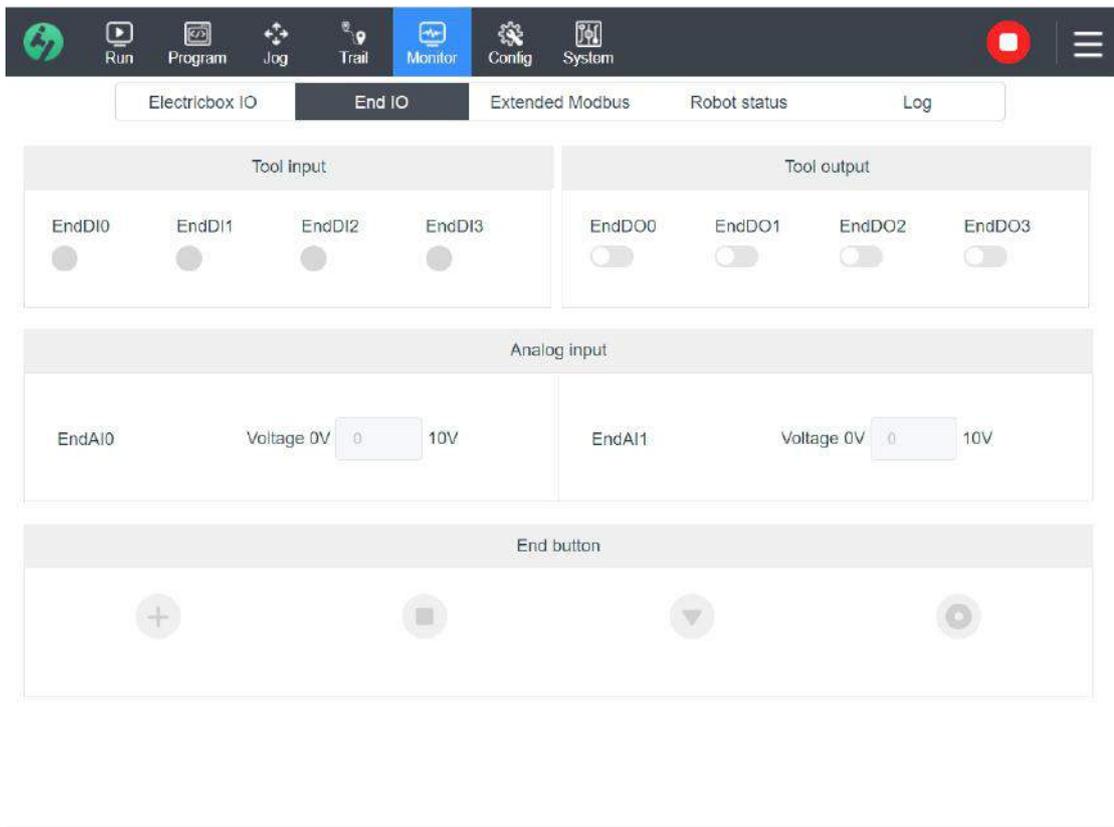
◆ Tool IO: Tool inputs EndDI0~EndDI3, corresponding to IO input signal of the robot end; the input is high level and the corresponding circle turns green;

◆ Tool outputs EndDO0~EndDO3, corresponding to IO output signal of the robot end; slide the corresponding output switch ⬭ to set the configurable output high level; at the moment, the switch turns green 🟢.

◆ Analog input: 2 channels, analog voltage range: [0 - 10V].

◆ Function button status: Identify the corresponding button status of new end IO; for example, if you press "Zero-Force Demonstration" button at the end, the button here changes from gray ⚪ to blue 🔵.

6.3 Expand Modbus
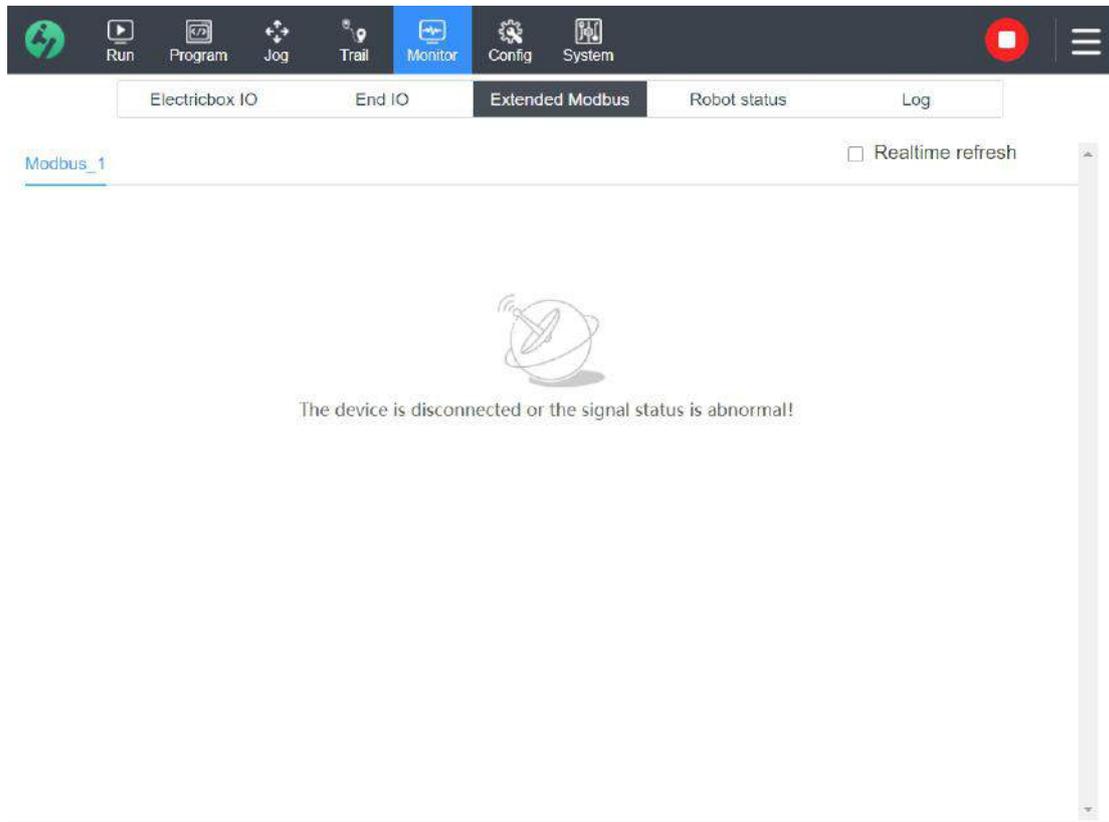
When a Modbus communication device is added to the configuration, the added device signal is displayed here to monitor the status. Only successfully connected Modbus devices are shown here. If the added device is not connected successfully, it will not be shown here; in this case, please check the device parameters in the configuration are correct and reconnect it.

Check "Real-Time Refresh" button to view the latest data; if it jumps to other interface, the refresh box will be unchecked automatically to save system resources, and Modbus data will not be updated;

6.4 Robot Status
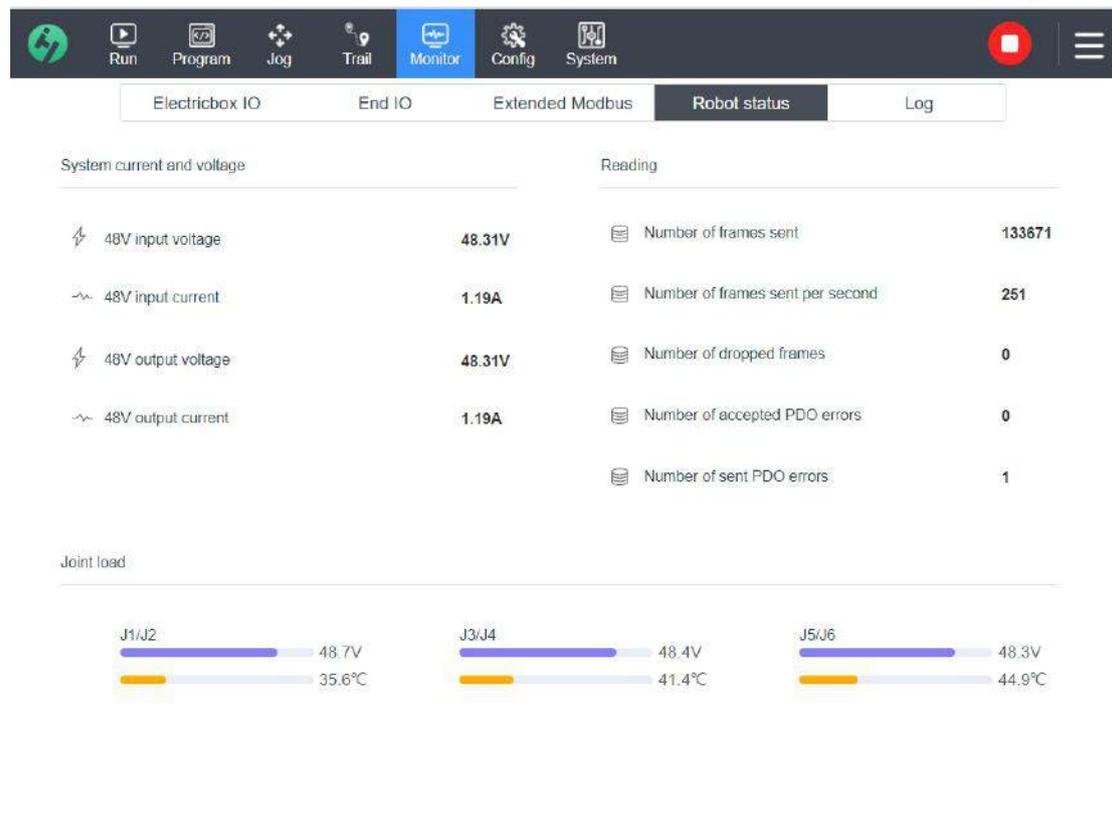
◆ Display the power supply situation of the robot's electric cabinet in real time;

◆ Display the relevant information on each robot joints, including temperatures of motor and electronic equipment as well as currents and voltages at joints.

6.5 Log Record
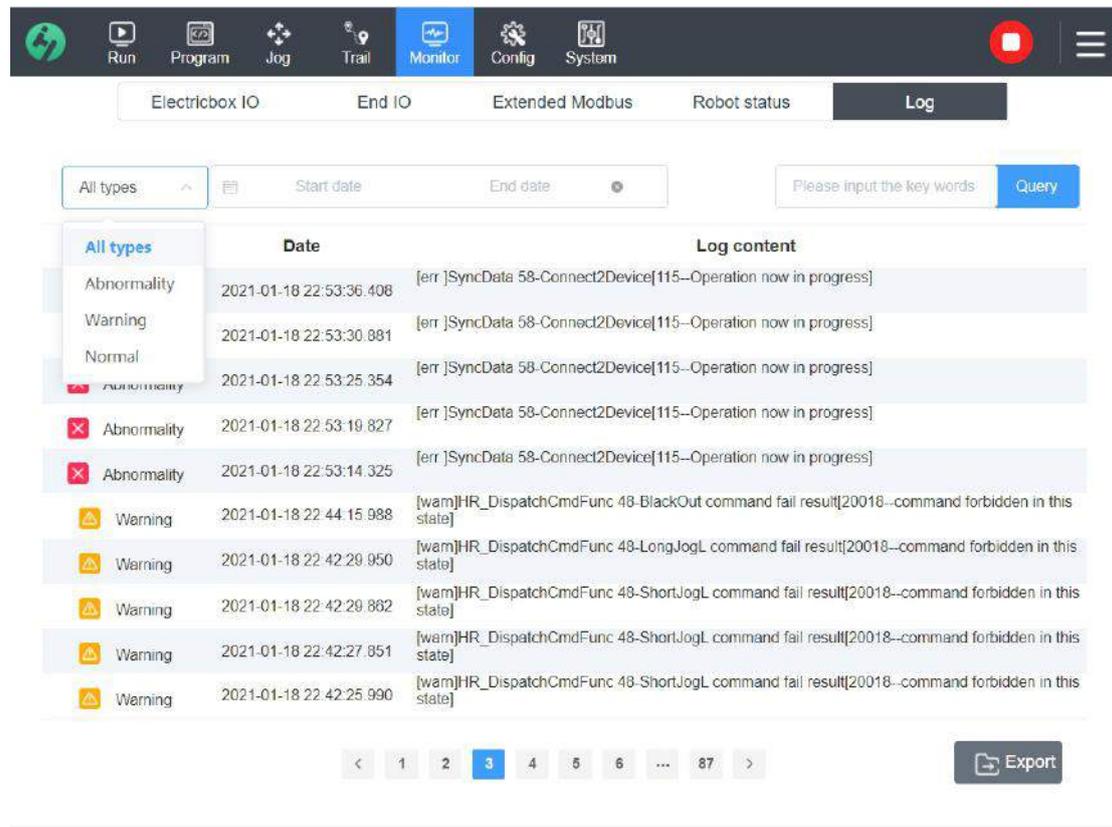
Display the severity classification of the message, the arrival time of the log record message and the message content.

◆ **Message content includes:**

1) Command sender: client;

2) Command sent: cmd;

3) Error Code: For example, Error:1035 or err:1041. See the communication protocol file for the error code information;

4) Error interpretation: For example, RS: DCS Status Error;

◆ **Type:** The type of error includes severe abnormality, common abnormality, warning, normal; the log list can be filtered and inquired.

◆ **Export:** Export all logs to the external device; in general, the logs can be exported and sent

to the technical support to review and resolve the problem when the user has a question.

Chapter 7 Management

7.1 User Management

Users with administrator permission can edit the module. When forgetting the password,the operator needs to ask the administrator to provide or modify the password.

Permissions include: administrator and operator

◆ Add User: Click "Add User" button and enter the name, permission and phone number in the pop-up window, and click "OK" button to confirm it, then the use is added successfully;

◆ Modify Permission: Click   , enter the name, permission and phone number in the pop-up modification window and then click "OK" button to confirm it;

◆ Delete User: Check the block in front of the name and click "Delete" button;

Note: The information in the first line is the super administrator account initially configured in the system which cannot be deleted or modified; other accounts can be deleted and modified.

7.2 System Update
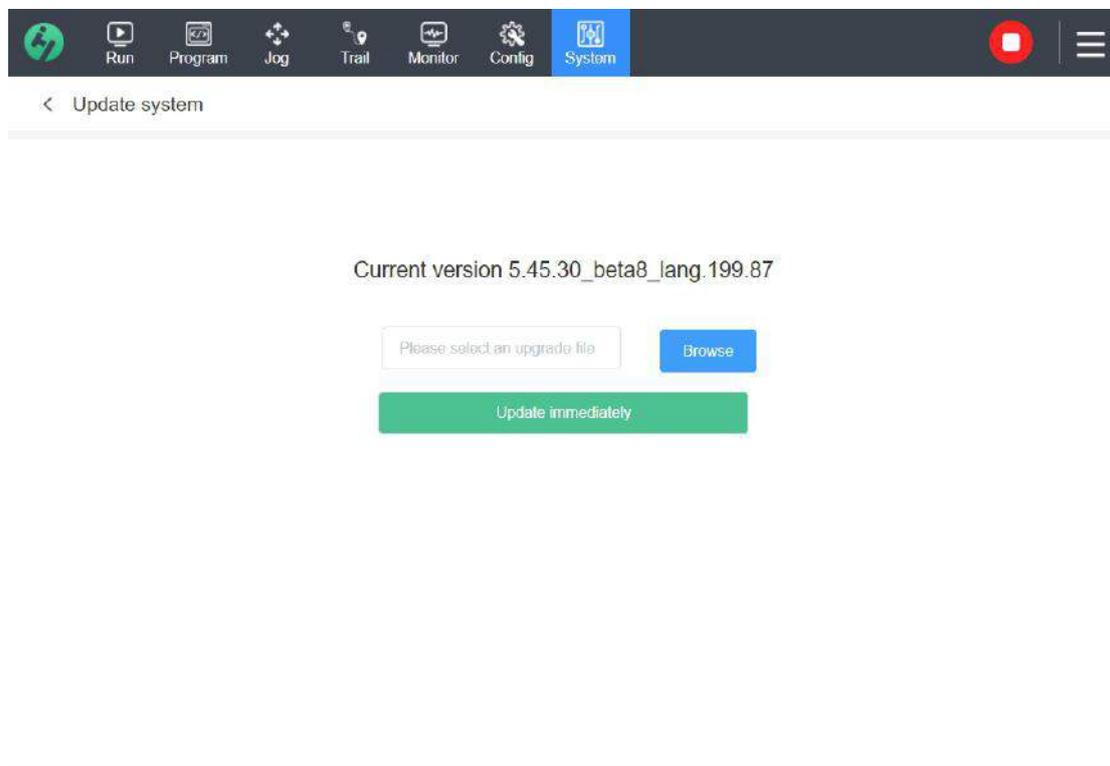
[Browse]: Open the folder and select Update Pack;

[Upgrade Now]: Click [Upgrade Now] button to automatically update the software system; the process will last 2-3 minutes;

Note: Please save relevant data before updating, for the system will be restarted after updating.

7.3 IP Setting

After the robot is normally connected to the Internet, it will automatically obtain the available network port.

Chapter 8 Troubleshooting

8.1 Emergency Stop

If the user predicts that there will be a hidden danger when the robot is moving, please press the Emergency Stop button immediately; the robot stops working after the "Emergency Stop" button is pressed and it can be reset according to the prompt. After reset, the robot is in the **de-energized** state, and the user needs to power on the robot to get it return to the normal working state.
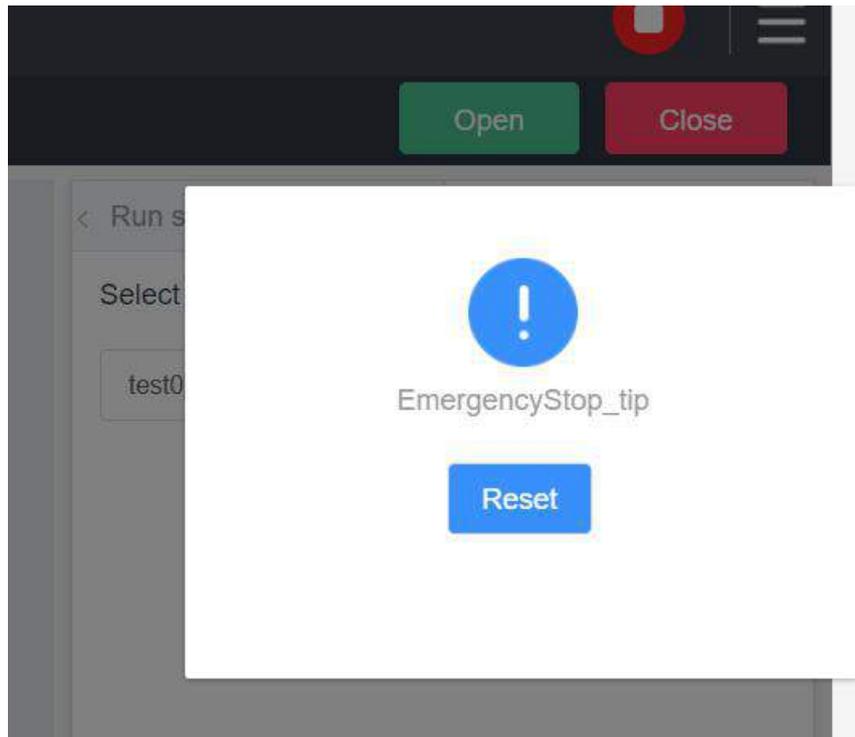
Reset Steps:

1)   Rotate the red Emergency Stop button on the demonstrator clockwise firstly.

2)   Click the [Reset] button in the pop-up window to clear the error.

3)   Go back to the homepage and click the "Power on" button to get the robot to return to normal state.

Note: If you do not rotate the Emergency Stop button and directly click [Reset] button, the system will pop up a prompt of "Reset failed", and the user still needs to follow the reset process to get it to return to normal.

8.2 Safety Collision Stop

When the robot encounters an obstacle, it will judge whether to stop moving according to the feedback force of the obstacle. The magnitude of the force is determined by the configuration level of the safety limit. The lower the safety level is, then the larger feedback force will be required to stop the robot; the higher the safety level is, the smaller the feedback force will be

required to stop the robot.

After stopping, it can be restored to normal by reset. For the sake of safety, please ensure that the robot will not meet an obstacle when reset so as not to trigger the safety collision function and to stop again.
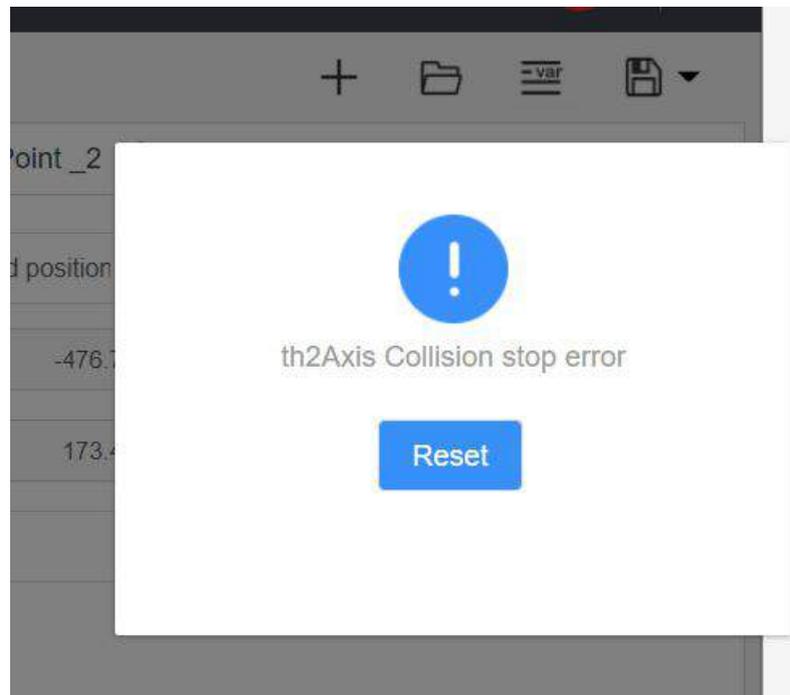
8.3 Trigger Safety Light Curtain

Start the security light curtain monitoring in the safety limit, configure the safety IO signal to stop or pause; when the IO signal is monitored, the system will trigger the protective stop, and pop up a prompt.

◆ [Safety Light Curtain Stop]: When the safety IO signal is detected, the system stops the movement of the robot and displays pop-up window prompt that the robot is in the state of safety protection and it is necessary to reset it and clear the fault before the robot can be operated again. After reset, the power supply of the robot is not cut off.

◆ [Safety Light Curtain Pause]: When the safety IO signal is detected, the script will be suspended and it is necessary to reset it and clear the fault before the current script can continue to run.
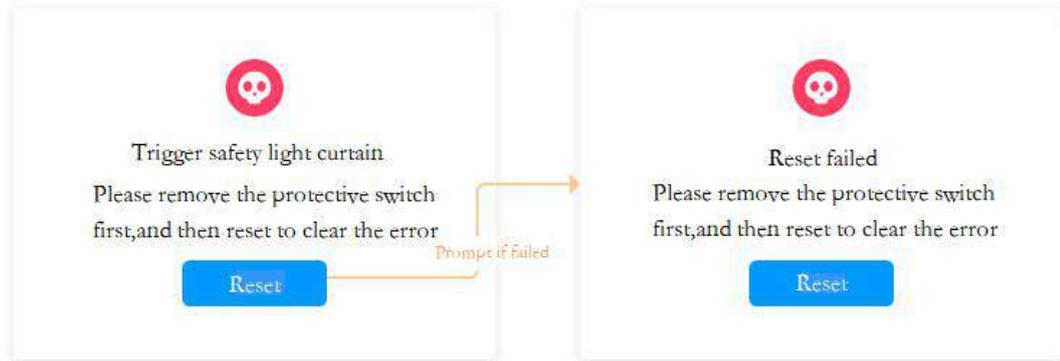
Reset Steps:

1)   Firstly, release the protection switch.

2)   Click the [Reset] button in the pop-up window to clear the error.

Note: If you do not release the switch and directly click [Reset] button, the system will pop up a prompt of "Reset failed", and the user still needs to follow the reset process to get it to return to normal.

8.3 Beyond Safety Space

During the movement of the robot, if the movement exceeds the limited safe range, a prompt window will pop up, and the user needs to press the "Restore" button to get the robot to return to the safe range. After recovery, the pop-up window will disappear and the system will return to normal.
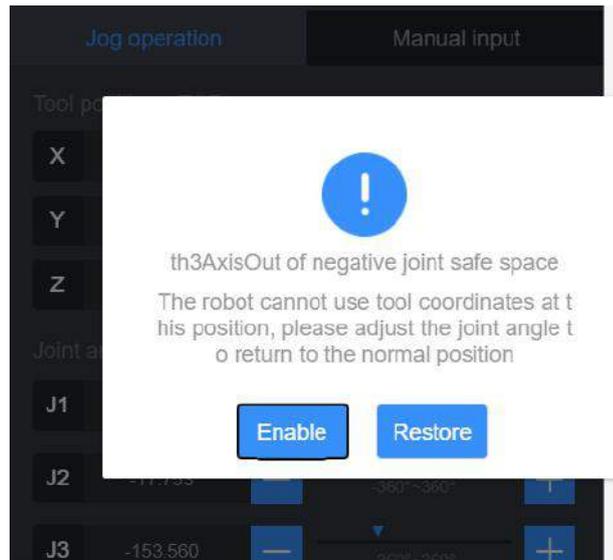
8.4 Singular Point Status

If the robot is in the singular point position in the inching process, the system will pop up a prompt:
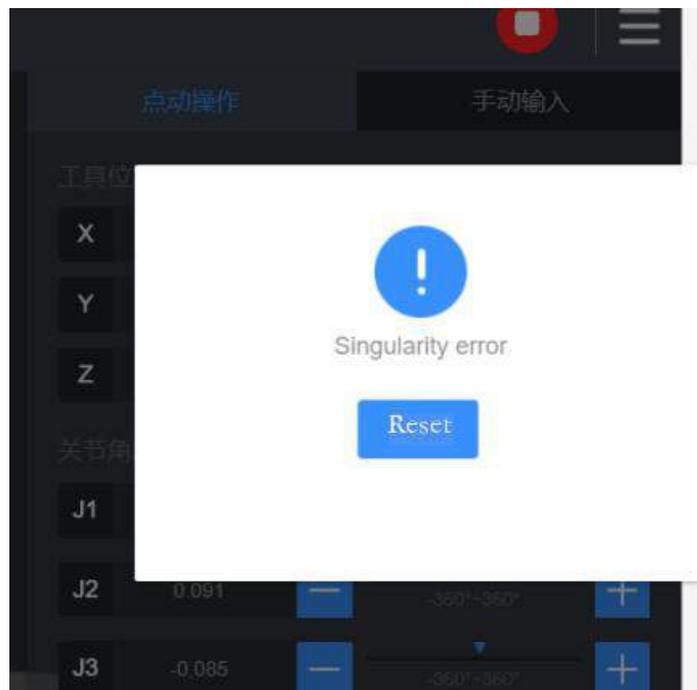
Release Singular Point:

1) Click [Reset] button and clear the error; the pop-up window will disappear;

2) Adjust any axis of J1~J6 so that the robot's attitude

is not at the singular point position.

Note: When singular point is encountered, the user can only adjust the joint angle to make the attitude of the robot not at the position of singular point after reset. The adjustment of the tool position direction (X,Y,Z,RX,RY,RZ) is invalid at the moment.