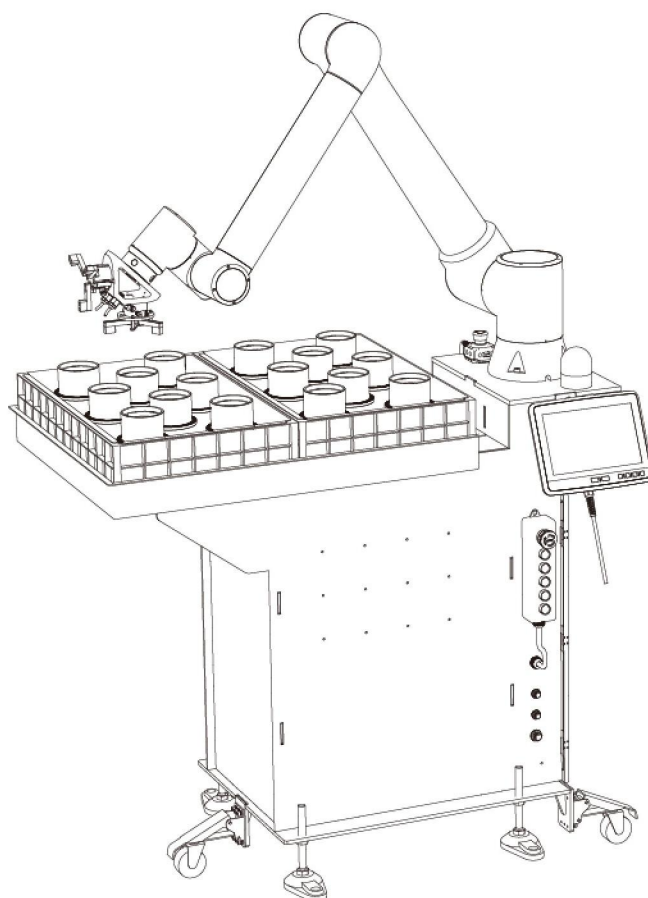


FAIRINO

CNC Loading and Unloading Cap Software User Manual



Fairino (Suzhou) Robot Technology Co.,Ltd

Data Encoding 20200310



Contents

1 Foreword.....	1
2 Disclaimer.....	1
2.1 Disclaimer regarding external equipment (CNC equipment).....	1
2.1.1 Risk of device incompatibility.....	1
2.1.2 Communication anomalies and data loss.....	2
2.1.3 CNC machine operation permissions.....	2
2.2 Disclaimer regarding operational specifications.....	2
2.2.1 Operator qualification requirements.....	2
2.2.2 Risk of improper operations.....	2
2.2.3 Teaching and Calibration Precision.....	3
2.2.4 Workflow Development and Validation.....	3
2.3 Disclaimer Regarding System Operation Status.....	3
2.3.1 Abnormal status handling.....	3
2.3.2 Emergency handling.....	3
2.4 Disclaimer Regarding Software Updates and Data Backup.....	4
2.5 Final right of interpretation.....	4
3 Start Quickly.....	4
3.1 Access to the CNC loading/unloading Cap software.....	4
3.2 CNC Loading and Unloading Parameter Configuration.....	6
3.2.1 Security Point Configuration.....	6
3.2.2 Tool Configuration.....	7
3.2.3 Machine Configuration.....	9
3.2.4 Function Signal Configuration.....	11
3.2.5 Loading and Unloading Method Configuration.....	12
3.2.6 Material loading and unloading formula configuration.....	15

3.3 Robot Manual Teaching	18
3.3.1 Manual demonstration and recording of points	18
3.3.2 View points	19
3.4 Quick addition of CNC loading and unloading workflows	20
3.4.1 Introduction to Simple Workflow Instructions	20
3.4.2 Perform operations on workflows	25
3.4.3 Run a Workflow	26
4 Role Entry and Permissions	26
5 System initial interface	28
5.1 Current Workflow Name	29
5.2 Robot operating status	29
5.3 Current tool coordinate system	29
5.4 Current workpiece coordinate system	30
5.5 Robot Control Area	30
5.6 System Status	31
5.7 Robot global velocity	32
5.8 Robot Current Mode	32
5.9 Current logged-in role	32
5.10 Menu bar	33
6 Home	33
6.1 Data Monitor	33
7 Config	34
7.1 Safety	34
7.1.1 Safety Point	34
7.1.2 Main Program Configuration	35
7.2 Tools	36
7.2.1 Tool Coordinate System Calibration	39
7.3 Machine	41

7.3.1 Communication method	44
7.4 I/O Signals	55
8 Feeding	58
8.1 Methods	58
8.1.1 Tray	58
8.1.2 Conveyor	65
8.2 Recipe	68
8.3 Point Table	77
8.4 Workflow	79
8.4.1 Toolbar	81
8.4.2 Flow control	82
8.4.3 Movement Instruction	88
8.4.4 Advanced Instructions	93
8.4.5 Auxiliary Axis	95
8.4.6 Signal	98
8.4.7 Safety	99
8.4.8 Machine	100
8.4.9 Modbus	103
8.4.10 Data	105
8.4.11 Comment	106
9 System	107
9.1 Data backup	107
9.2 System log	108
9.3 System Settings	108
9.4 Advanced Parameters	109
9.4.1 Configurable signal	109
9.4.2 Extended shaft	118

9.4.3 Modbus TCP	130
------------------------	-----

1 Foreword

First of all, thank you very much for choosing our company's FR series collaborative robot products and the CNC loading/unloading Cap software. Our products have undergone multiple rounds of careful design and testing to ensure that they can meet your various needs.

Please carefully read [the Collaborative Robot User Manual](#) and the CNC Loading/Unloading Cap Software Manual to ensure that you can use our products correctly and achieve the best possible user experience. If you encounter any issues during use, please refer to [the Troubleshooting section in the Collaborative Robot User Manual](#), or contact our after-sales support staff. We truly appreciate your support and trust, and look forward to providing you with even better service and products in the future.

2 Disclaimer

This statement applies to FAIRINO Robot's (hereinafter referred to as "we" or "us") CNC Loading and Unloading Cap Software (hereinafter referred to as "the Software") and the equipment systems used in conjunction with it (hereinafter referred to as "the System"). Before using the System, please be sure to read this statement carefully. Use of the System constitutes your full understanding and acceptance of all the contents of this statement.

2.1 Disclaimer regarding external equipment (CNC equipment)

2.1.1 Risk of device incompatibility

This system is capable of communicating and cooperating with third-party CNC devices (including, without limitation, FANUC, SYNTEC, and similar systems). Our warranty is limited solely to the proper operation of data interaction between this software and CNC devices under standard communication protocols. We disclaim all liability for the performance, reliability, security, transmission delays, data correctness, or any damages or losses resulting from defects, faults, or failures of the third-party CNC equipment.

2. 1. 2 Communication anomalies and data loss

We disclaim all liability for any communication abnormalities, loss of control, data loss, or processing errors resulting from, or caused by, discrepancies in CNC equipment protocol versions, hardware failures, network outages, electromagnetic interference, or any other such factors beyond our reasonable control.

2. 1. 3 CNC machine operation permissions

Operation, maintenance, parameter adjustments, and safety device settings for CNC equipment must be carried out by duly qualified and authorized personnel. We assume no responsibility or liability for any incidents or accidents resulting from operator error on the CNC equipment side, inappropriate parameter settings, or malfunctioning safety devices.

2. 2 Disclaimer regarding operational specifications

2. 2. 1 Operator qualification requirements

Operators of this system are required to have adequate knowledge of industrial robot and CNC equipment operations and to have completed the official training program offered by our company or its authorized representatives. The user assumes full liability for any and all consequences resulting from operation of this system by individuals who have not received such training or who lack the requisite qualifications.

2. 2. 2 Risk of improper operations

It is strictly prohibited to run any workflow without first confirming the safety of the working area, setting a safety point (pHome), correctly configuring the tool/workpiece coordinate system, checking the gripper signals, or verifying the machine status. We shall not be held liable for any accident resulting from failure to comply with this software user manual or on-site safety operating procedures, including but not limited to overspeed operation, forcible intervention in a running program, bypassing safety checks, and similar violations.

2. 2. 3 Teaching and Calibration Precision

The following operations — robot point teaching, tool coordinate system calibration, and workpiece coordinate system calibration—all depend on the operator's proficiency and judgment. Any production anomalies, collisions, or equipment damage resulting from inaccurate teaching, calibration errors, or failure to follow standard procedures shall be the responsibility of the user.

2. 2. 4 Workflow Development and Validation

The workflows (including logic such as loops, conditional branches, signal controls, wait conditions, etc.) are written and configured by operators or technicians. We shall not be held liable for any equipment malfunction, damage, or personal injury caused by programming logic errors, improper condition settings, or incorrect variable assignments. It is strongly recommended that, after the initial deployment or any program modification, each step be verified in manual mode before switching to automatic operation.

2. 3 Disclaimer Regarding System Operation Status

2. 3. 1 Abnormal status handling

This software provides system status monitoring (Normal / Warning / Error) and logging functionalities, but does not cover all potential faults. Users should remain vigilant to system alerts at all times and must immediately cease operation upon receiving a warning or error notification to investigate the root cause. We assume no liability for any consequences arising from failure to heed system alarms or delays in addressing anomalies.

2. 3. 2 Emergency handling

In the event of an emergency (such as collision, personnel entering a hazardous area, abnormal sounds, etc.), the operator shall immediately press the emergency stop button, rather than relying on software stop functions. Any losses exacerbated by failure to take prompt physical emergency stopping measures shall be borne solely by the user.

2.4 Disclaimer Regarding Software Updates and Data Backup

We recommend that users regularly back up system data (including recipes, points, workflows, coordinate systems, etc.). We shall not be held liable for any data loss or configuration failure resulting from failure to back up data or from improper operations during software updates or system restoration.

Software version updates may result in changes to functionality or the user interface. Users are advised to thoroughly review the most current user manual. We assume no responsibility for any operational errors arising from failure to familiarize oneself with the updated version.

2.5 Final right of interpretation

We reserve the sole and final right to interpret this disclaimer and reserve the right to amend it without prior notification. The most current version shall be as set forth in the software release or as posted on our official website.

3 Start Quickly

3.1 Access to the CNC loading/unloading Cap software

- 1) After successful WebApp login, navigate to the "System Settings – Plugin Configuration" interface and select the plugin file named "TendingCap-v1.0.0plugin" for import. Once imported, click the "Enable" button in the "Operations" section of the list below.

Important: The WebApp software version is V3.9.8 or above.

- 2) In the WebApp, click the "Auxiliary Applications" navigation bar to expand the submenu; locate the "TendingCap" submenu and click it to access the login interface of the CNC loading/unloading Cap software, or open the Google Chrome browser and visit the target URL: <http://192.168.58.2:3000/frcap/e36e0670-6ee9-11f1-8003-efb9f455c642/#/>.
- 3) Users select an operation role based on their position. Roles currently include Operator and Technician. Operators can log in directly without a password, while Technicians must enter a password for authentication; the default

password is 1234.

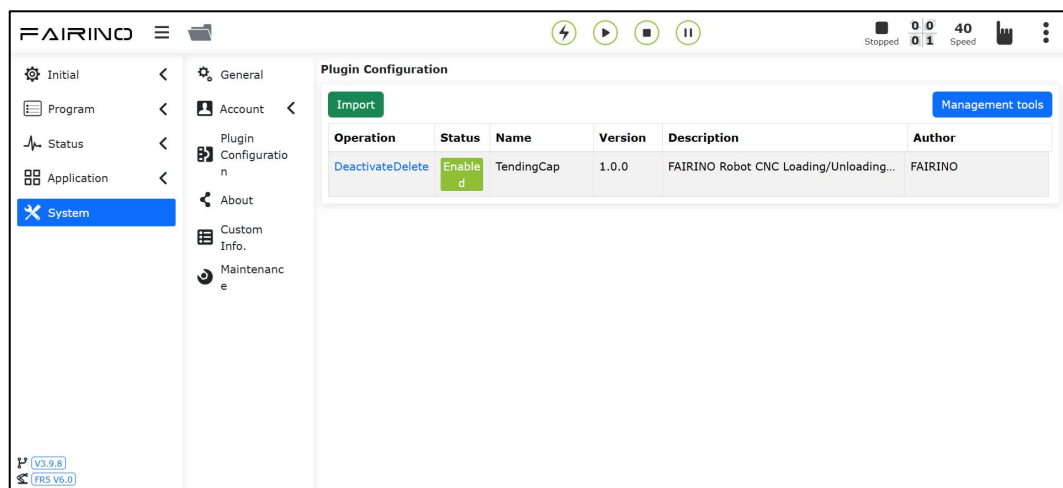
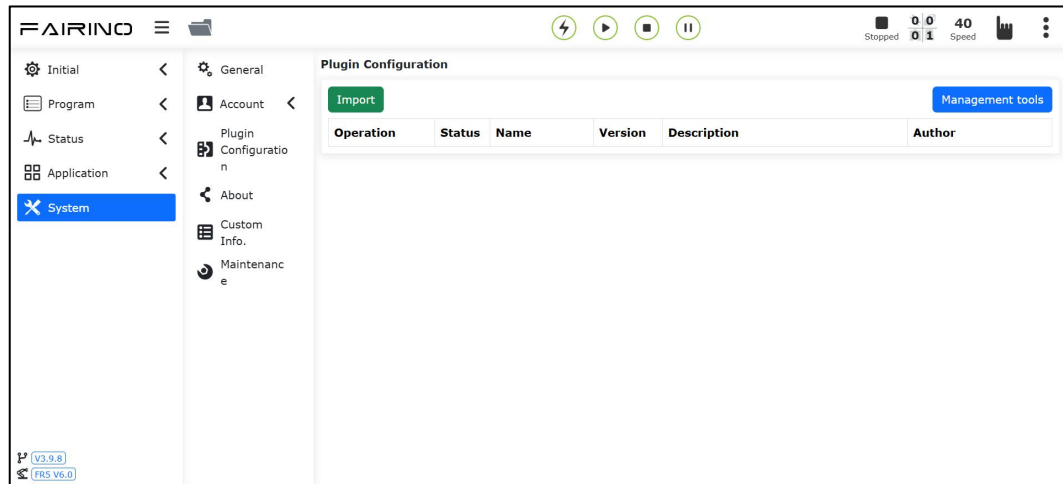


Figure 3-1 WebApp Import of the CNC Material Handling Cap Plugin Package

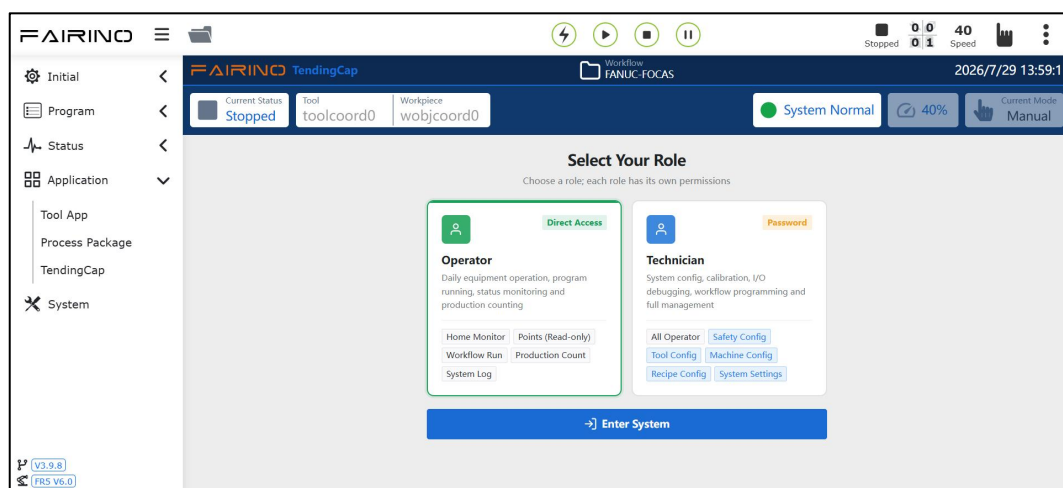


Figure 3-2 In the WebApp, navigate to the login page for the CNC material loading/unloading Cap software.

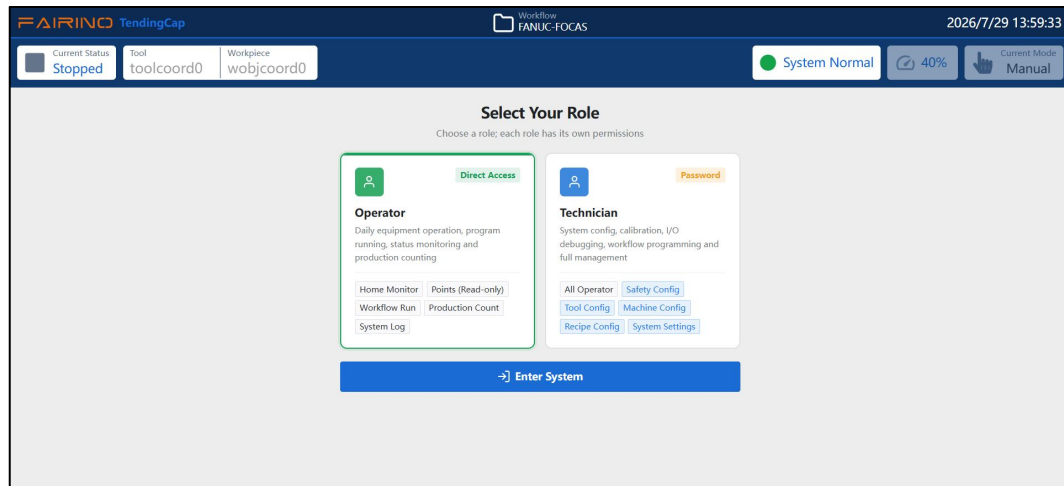


Figure 3-3 Enter the direct access address in your browser to access the CNC material handling Cap software.

3.2 CNC Loading and Unloading Parameter Configuration

Before creating a CNC loading/unloading workflow, configure safety points, tools, machine tools, and function signal parameters. Once these basic settings are configured, select the tool in the loading/unloading method and choose the machine tool and loading/unloading method in the recipe to complete recipe creation. This allows you to directly generate workflow content based on the recipe in subsequent workflows.

3.2.1 Security Point Configuration

Perform the teaching of safety points in "Configuration - Safety"; the name for these points is fixed as "pHome". Click the "Configuration" button to send the current robot pose as safety point data.

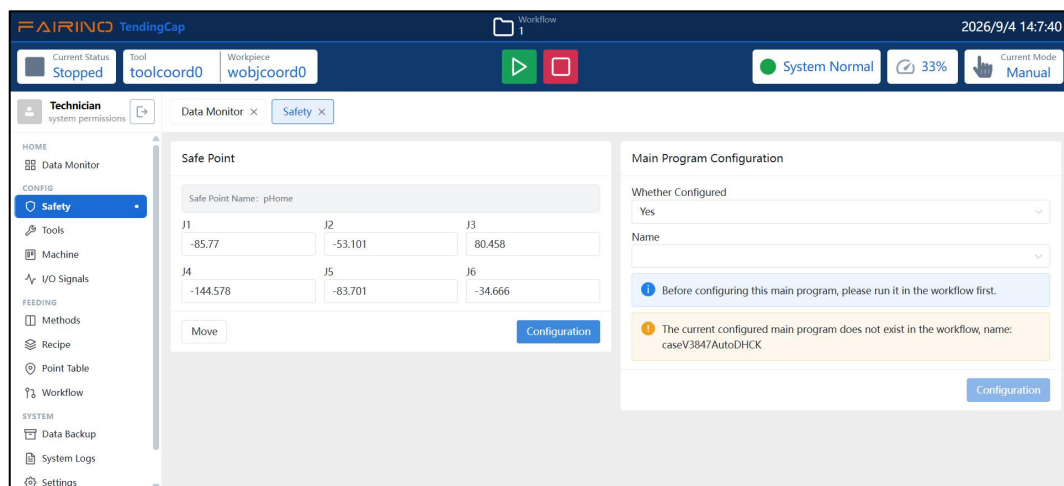


Figure 3.2.1-1 Security Point Configuration

3.2.2 Tool Configuration

To configure the end tool in "Configuration - Tools", follow these configuration steps:

- 1) Click the "Add" card to access the "End Effector" section; enter the tool name (e.g., "End Tool") based on your functional requirements, add grippers (e.g., add a loading gripper and a unloading gripper), select the gripper type (single or double cylinder), and configure the signals.
- 2) Click the "Next" button to enter the "End Load and Center of Gravity" section; use the tool to select the load and set its weight and center of gravity.
- 3) Click the "Next" button to access the "Tool Coordinate System" section, where you can configure the tool coordinate systems for each gripper; if you wish to select a non-default tool coordinate system, click the "Calibrate" button to perform a four-point method tool coordinate system calibration.
- 4) Click the Complete button to add a tool, or click the Edit button in the upper-right corner of the content card to modify it.

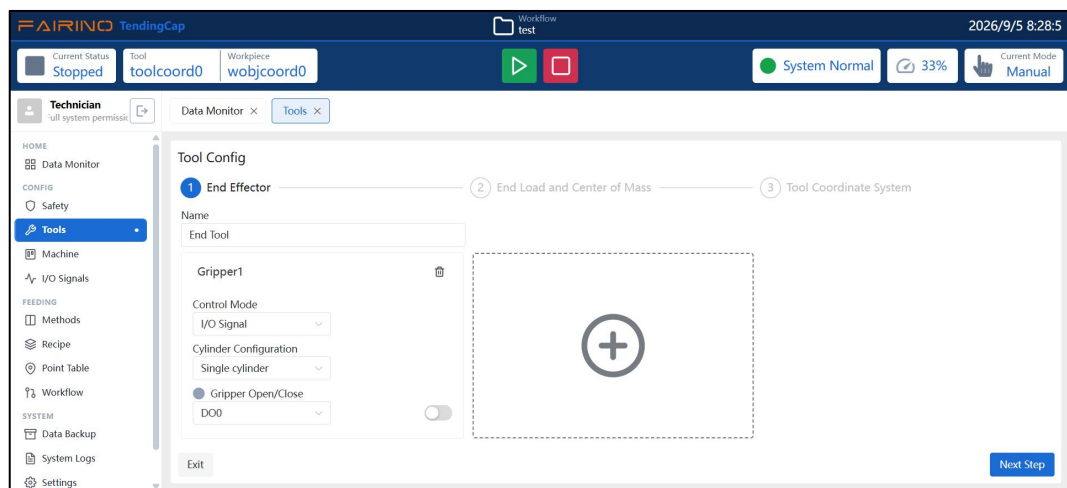


Figure 3.2.2-1 Tool Configuration - End Effector

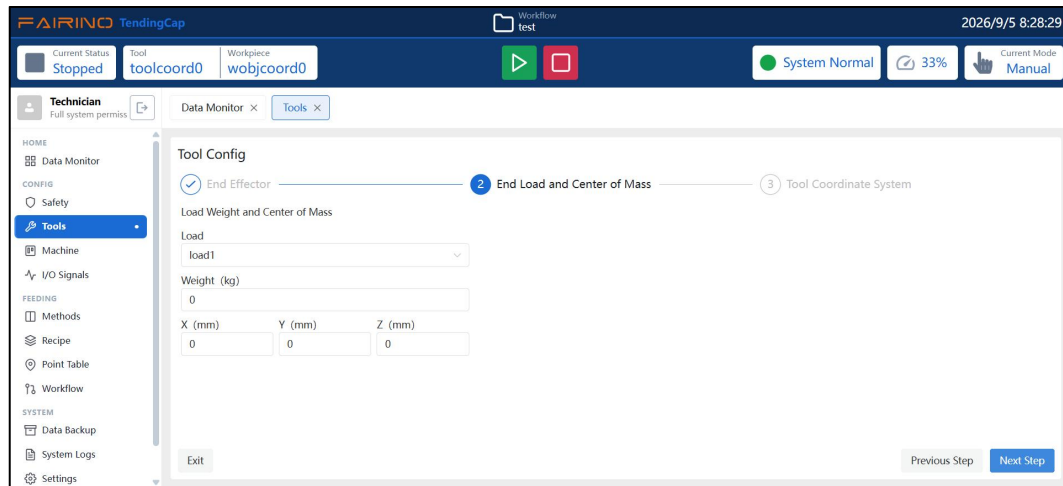


Figure 3.2.2-2 Tool Configuration – End Effector Load and Center of Gravity

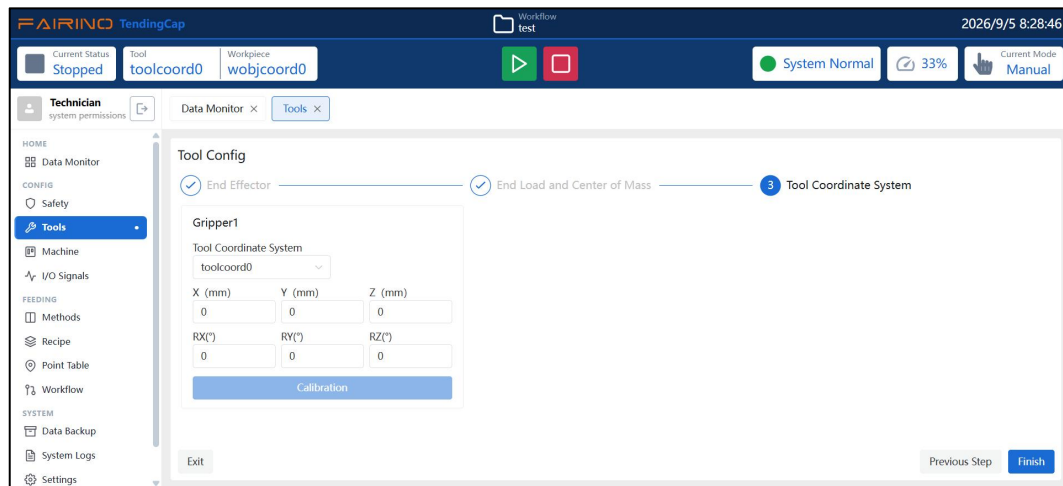


Figure 3.2.2-3 Tool Configuration — Tool Coordinate System

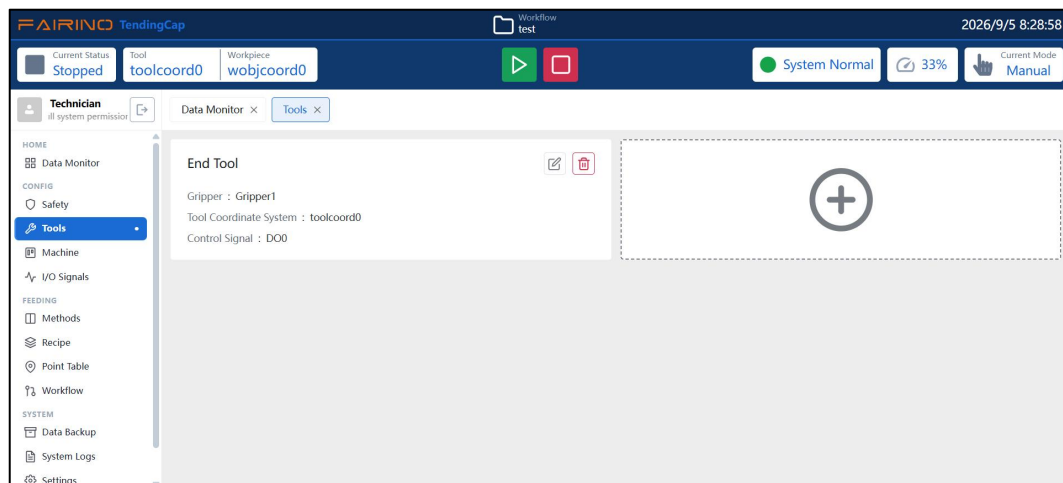


Figure 3.2.2-4 Tool configuration completed

3.2.3 Machine Configuration

To configure the end tool in "Configuration – Machine Tool", follow these steps:

- 1) Click the "Add" card to access "Machine Tool Parameters"; enter the machine tool's name (e.g., "FANUC") based on your functional requirements, and select the machine tool system (available options: FANUC, Xinda, and Other).
- 2) Click the "Next" button to enter Communication Configuration, where you select the communication method based on your machine tool system and configure relevant signals. Common signal functions include "Machine Door Opening/Closing", "Chassis Release/Fixing", and "Process Start". If you choose FOCAS or the New Generation protocol, you must upload a pre-written protocol file in Lua format with a name starting with "CtrlDev_CNC". Taking the "FOCAS" communication protocol as an example, in the configuration interface, click the "Edit" button to access the protocol editing page and upload the protocol. After successful upload, select the protocol and click "Configure"; under "Device Connection," you can view the configured protocol files. Click the "Load" icon to initiate communication. Once the connection is established, perform debugging preparations in the "Device Debugging" section on the right side.
- 3) Click the "Complete" button to add a machine tool; click the "Edit" button in the upper right corner of the content card to make modifications.

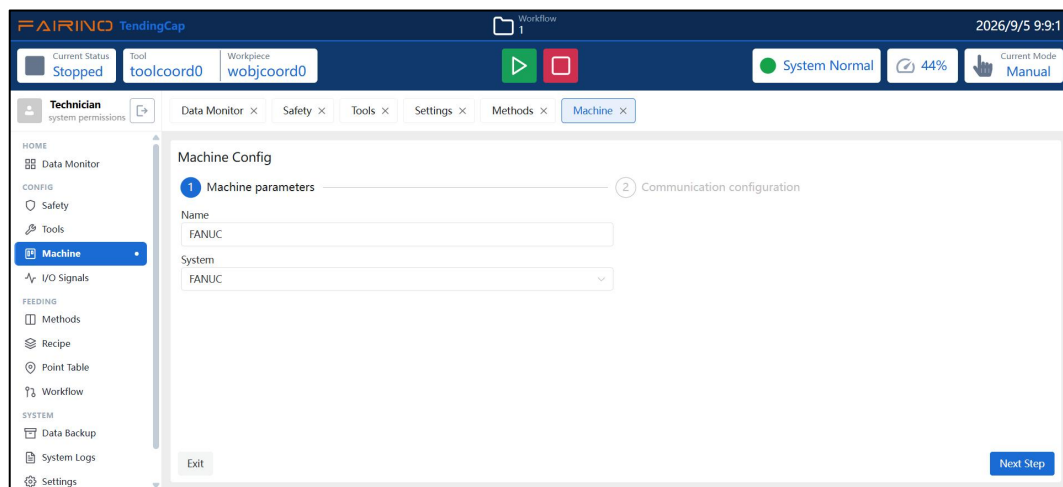


Figure 3.2.3-1 Machine Parameters

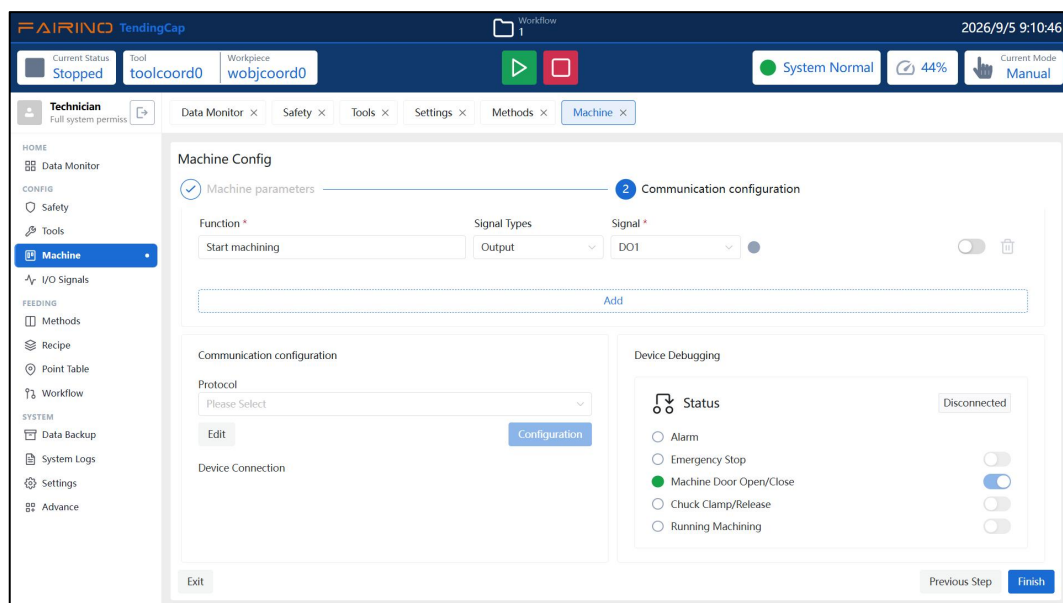
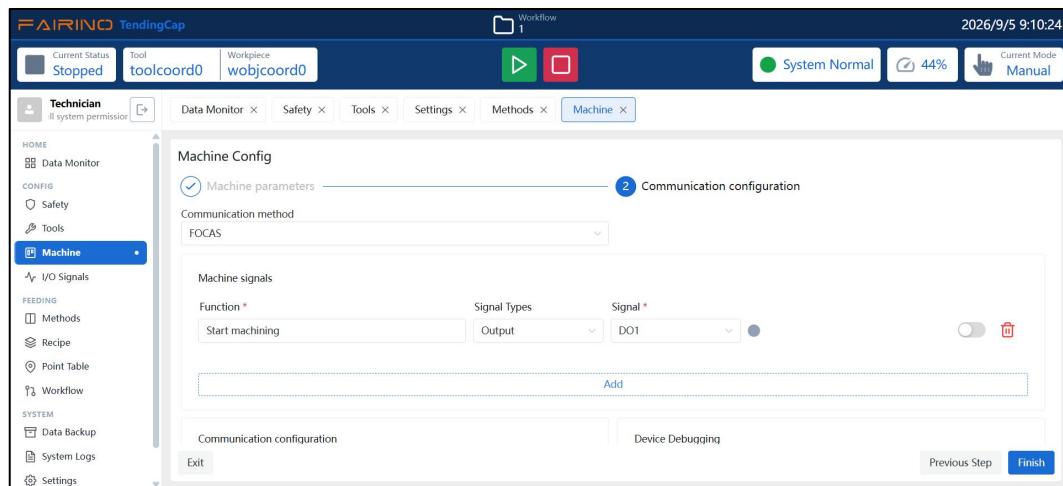


Figure 3.2.3-2 Communication Configuration

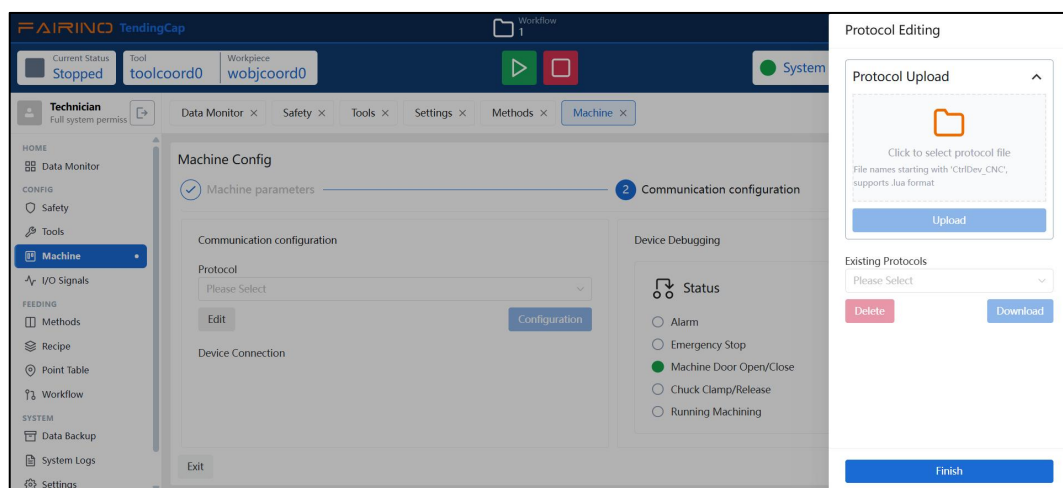


Figure 3.2.3-3 Agreement Editing

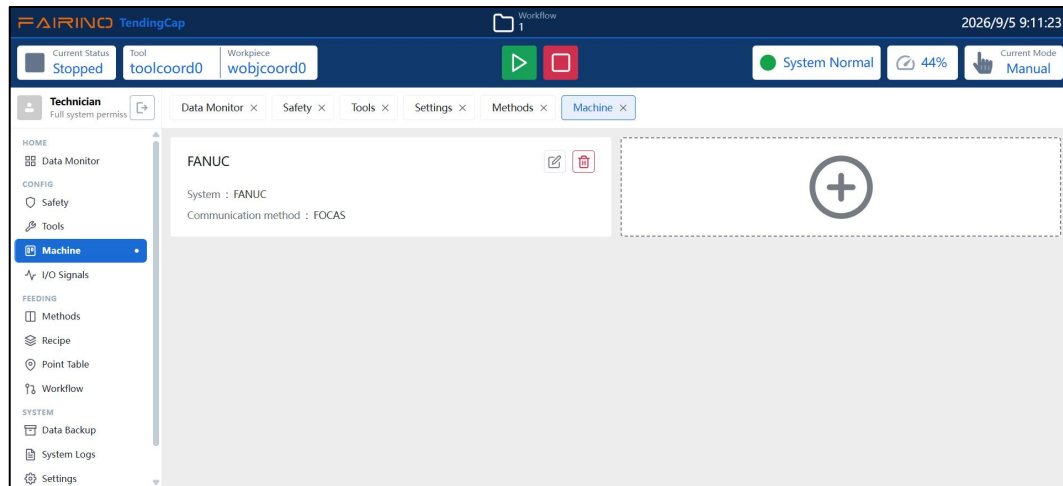


Figure 3.2.3-4 Machine Configuration

3.2.4 Function Signal Configuration

To configure custom functional signals, navigate to "Configuration - Functional Signals," click the "Add Custom Function" button at the bottom, which will open the "Add Custom Function" drawer; enter a name based on your operational scenario, and select the signal type and signal. For example, during the loading and unloading process, an output signal can be used to display red, yellow, and green indicator lights to inform the operator of the machining status, while an input signal can be used to indicate whether the workpiece is in position during the machining process.

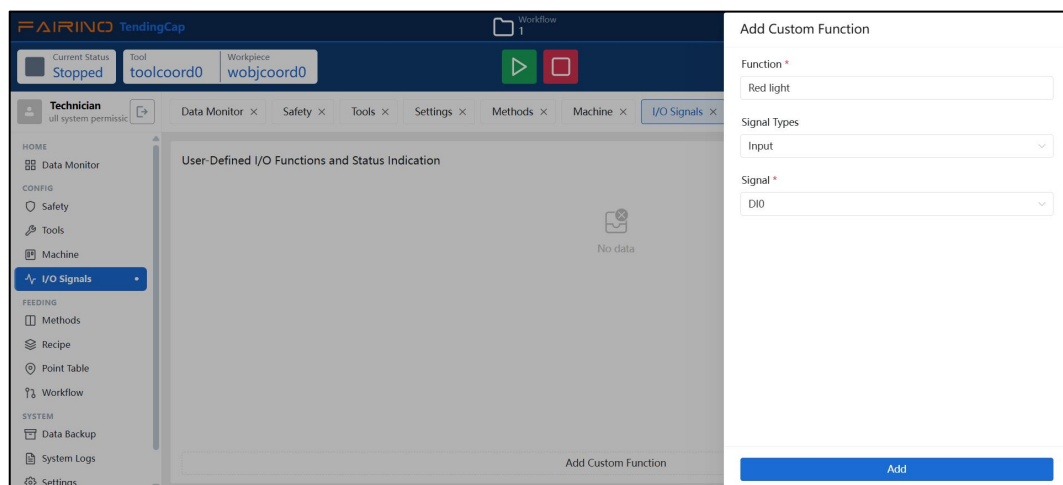


Figure 3.2.4-1 Functional Signal Configuration - Adding Custom Functions

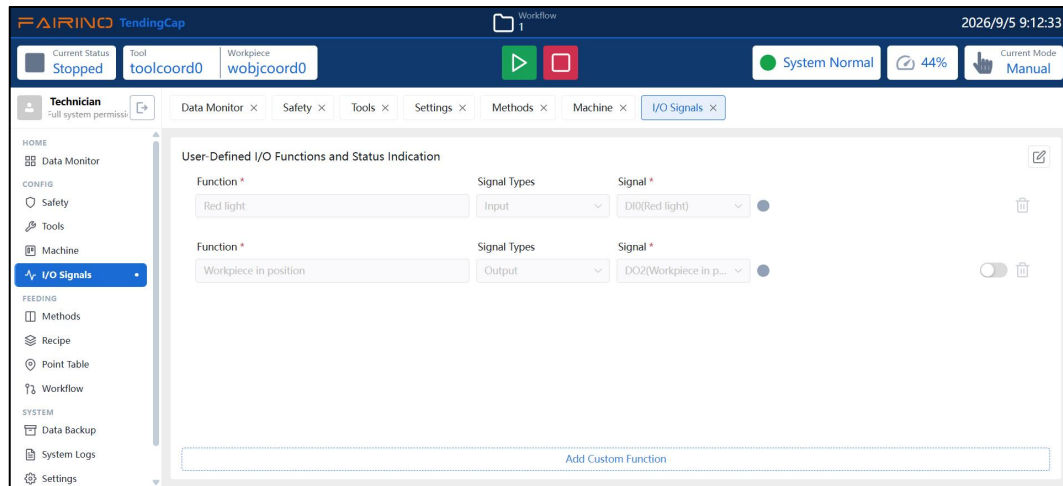


Figure 3.2.4-2 Function Signal Configuration

3.2.5 Loading and Unloading Method Configuration

The "Loading/Unloading – Method" section allows for configuring loading and unloading processes using two methods: material trays and conveyor belts; these methods can be combined together.

The tray-based method employs a three-point-defined matrix as its primary working configuration, with the steps as follows:

- 1) Click the Add tab to enter Plate Parameters, enter a name based on functional requirements, select a tool, and choose whether to use an extended axis.
- 2) Click the Next button to enter the Material Disk Coordinate System section. Select the workpiece coordinate system for the material disk. For non-default systems, click Calibrate to set the workpiece coordinates using either the Origin – X-axis – Z-axis or Origin – X-axis – XY-plane methods.
- 3) Click the Next button to enter the tray array and configure the matrix tray's number of rows, columns, layers, layer height (unit: mm), transition point offsets (X, Y, Z, unit: mm), motion path, and configuration method—point teaching using the three-point method.
- 4) Click the Complete button to add a tray. Click the Edit button in the upper-right corner of the content card to modify it.

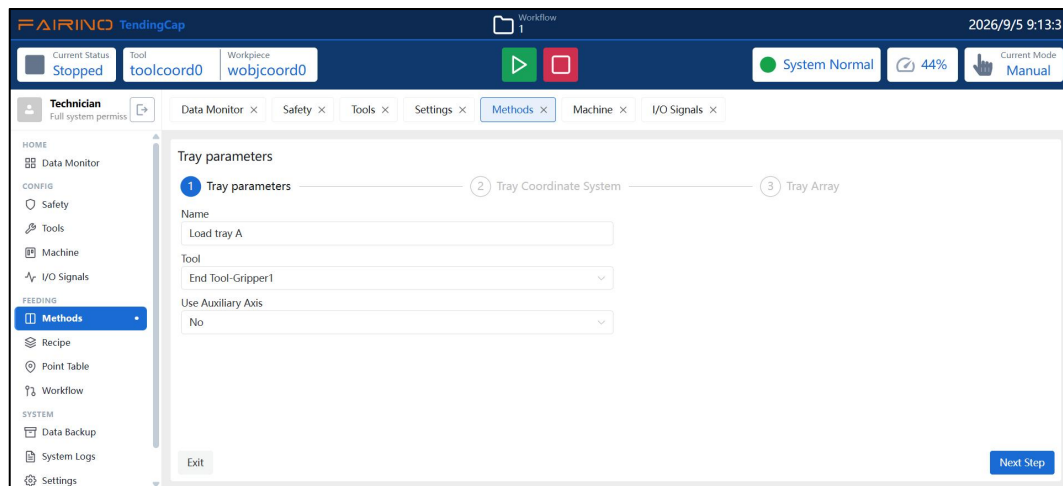


Figure 3.2.5-1 Material Disk Configuration – Material Disk Parameters

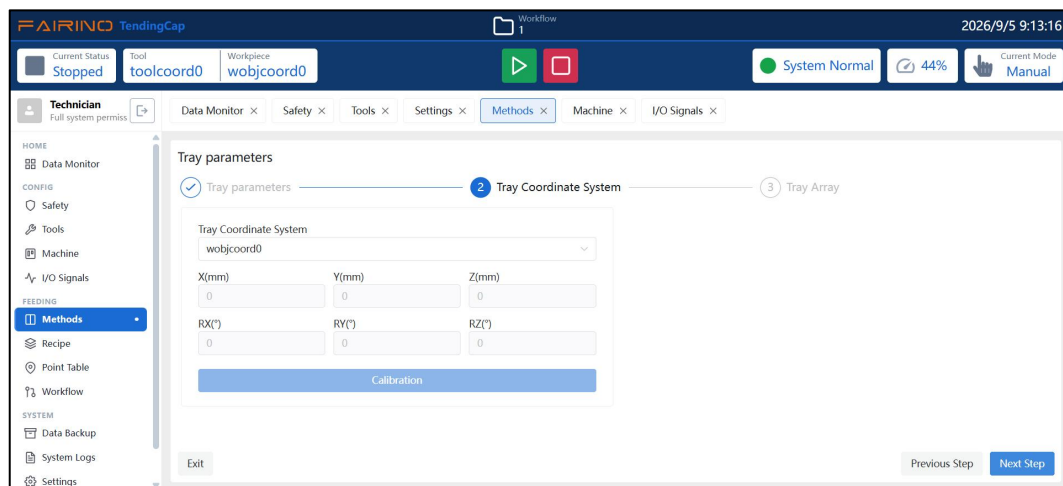


Figure 3.2.5-2 Material palleting method – Material pallet coordinate system

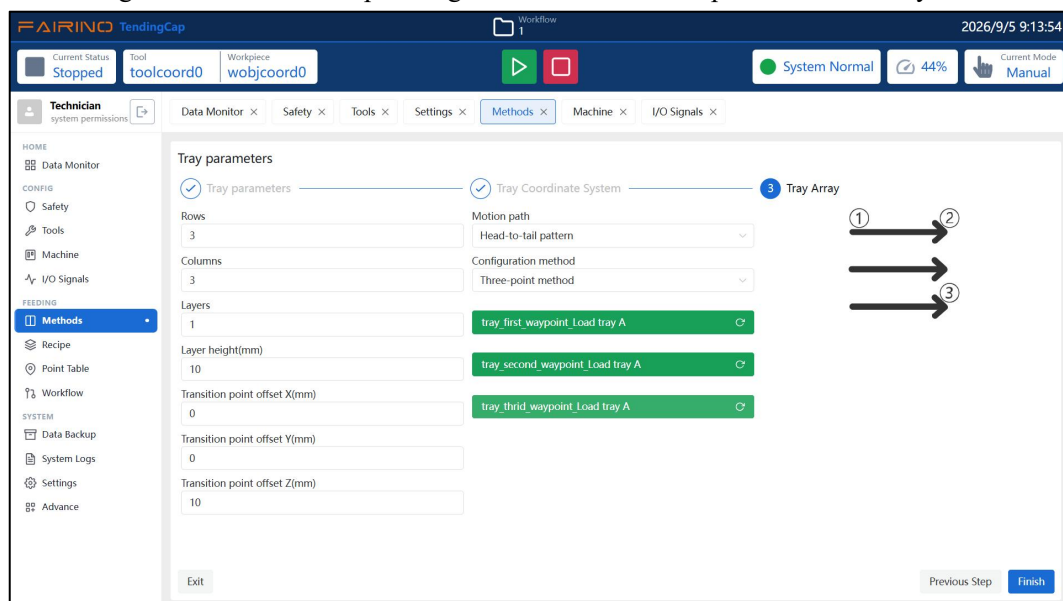


Figure 3.2.5-3 Material Disk Method – Material Disk Array

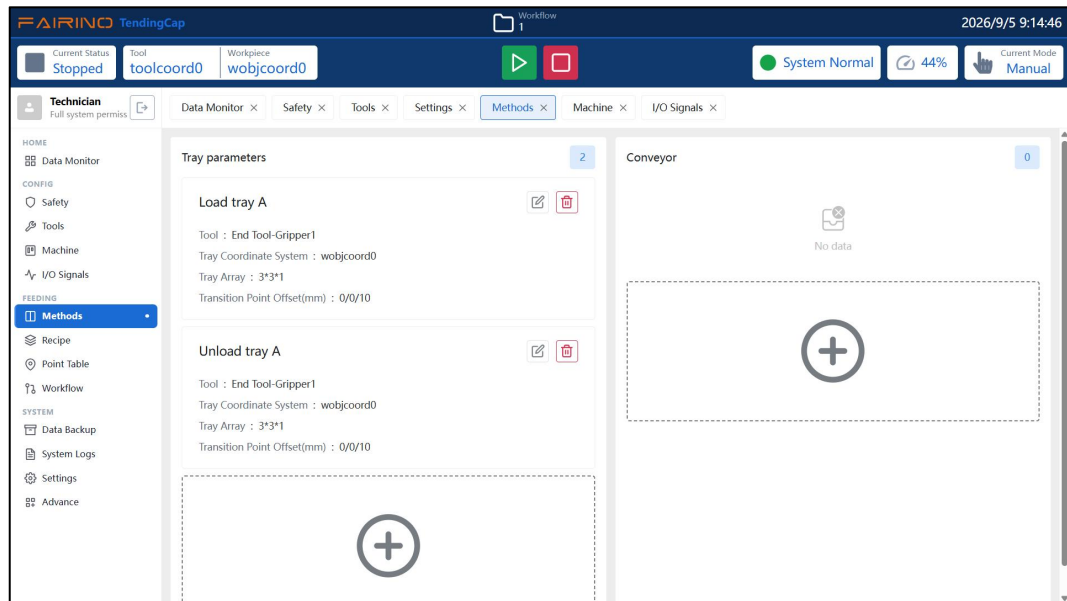


Figure 3.2.5-4 Material tray method

The steps for configuring the conveyor belt setup are as follows:

- 1) Click the Add tab to enter Belt Parameters, enter a name based on functional requirements, and select a tool.
- 2) Click the "Next" button to enter "Position and Signal". Set the positioning points for workpiece loading/unloading on the conveyor belt, enter the transition point offsets (X, Y, Z in mm), and add the input signal for the conveyor belt—for example, the signal triggered when workpieces reach the loading/unloading positions.
- 3) Click the Complete button to add the conveyor belt.

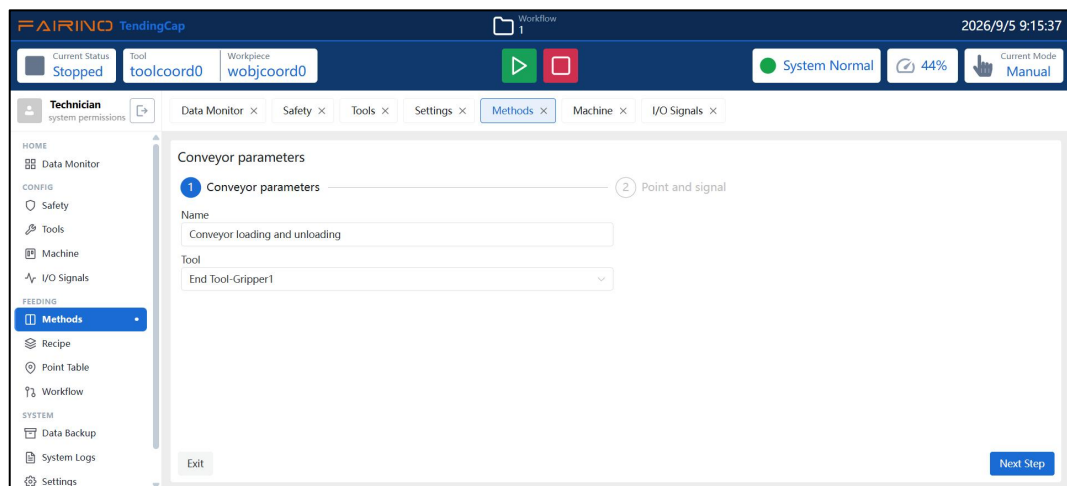


Figure 3.2.5-5 Conveyor Belt Method – Parameters for the Conveyor Belt

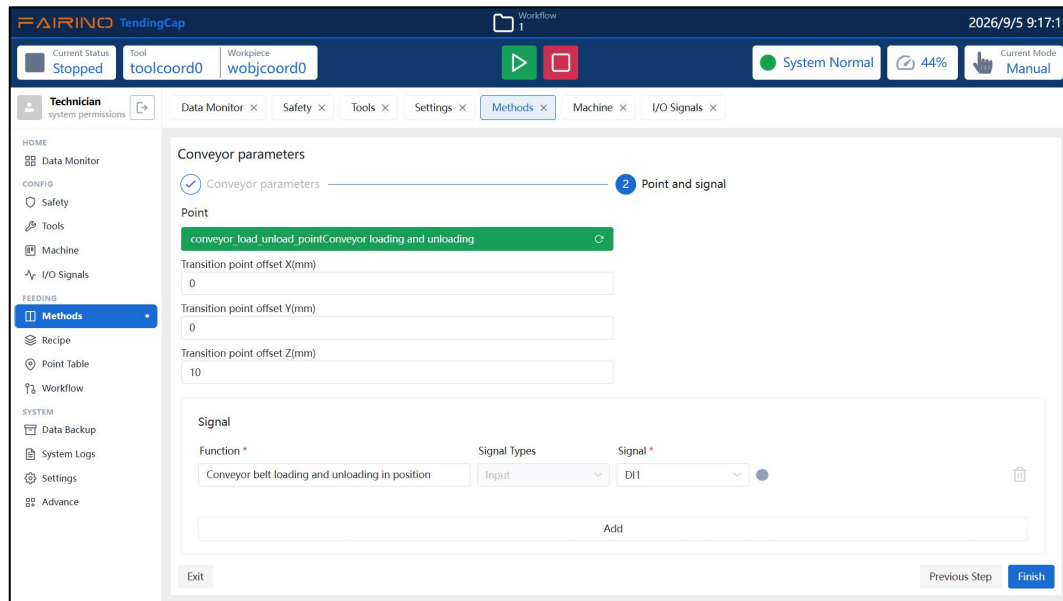


Figure 3.2.5-6 Conveyor belt method – Positioning and signaling

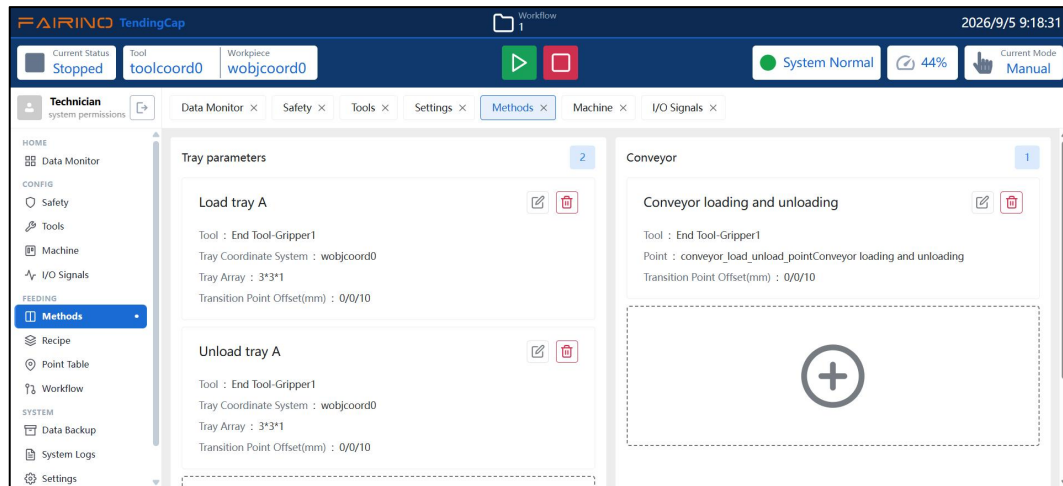


Figure 3.2.5-7 Conveyor Belt Method

3.2.6 Material loading and unloading formula configuration

Configure the recipe in "Material Loading – Recipe" with the following steps:

- 1) Click the "Add" card to enter the "Recipe Parameters" section; enter a name based on your functional requirements, select the operating mode—material loading/unloading, only loading, or only unloading, and specify whether to use an extended shaft.
- 2) Click the "Next" button to enter the "Workstation Layout" page, where you can select the machine tool and loading/unloading method (material tray/conveyor belt).

- 3) Click Next to enter Material Loading/Unloading Configuration. Click the Add Material Loading/Unloading button at the bottom; data for the selected method will appear on the right side. Check the option and click Add at the bottom to confirm.
- 4) Click the "Next" button to enter the "Process Flow" section, where you can set the core process steps as: "Material pickup from tray/ conveyor belt - Material feeding to machine - Material ejection from machine - Material release from tray/ conveyor belt". Based on the configuration of the first three steps, clicking "Add Step" will display different content, with the data within this content being automatically populated according to the configuration. By clicking "Insert Step Here", you can add additional process steps such as "Secondary Positioning" and "Custom Process" based on your specific scenario. Within each step, clicking "Add Sub-step" will take you to the sub-step addition interface, where you can select step types including "Point-to-Point", "Linear Motion", "Control Signal", "Wait Signal", and "Matrix Movement".
- 5) Click the Complete button to add a recipe, or click the Edit button in the upper-right corner of the content card to modify it.

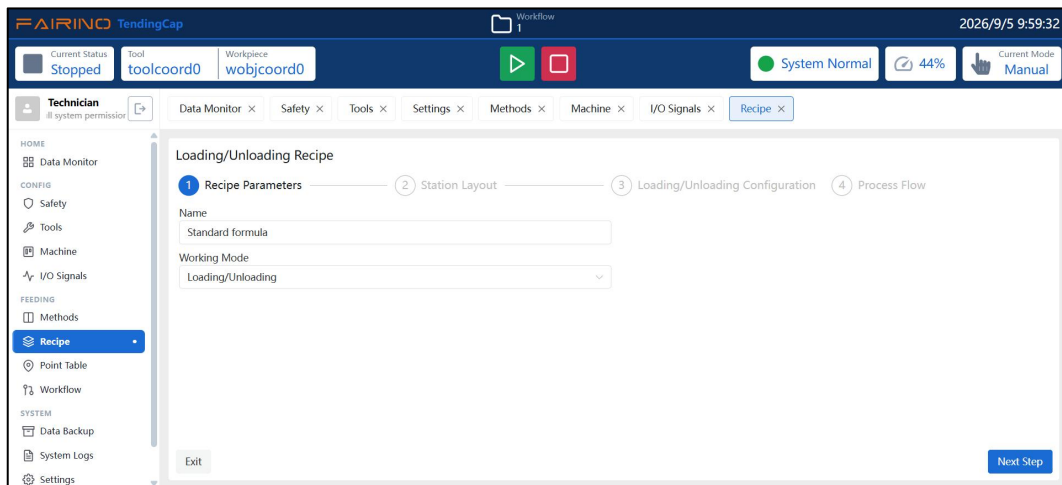


Figure 3.2.6-1 Recipe Parameter

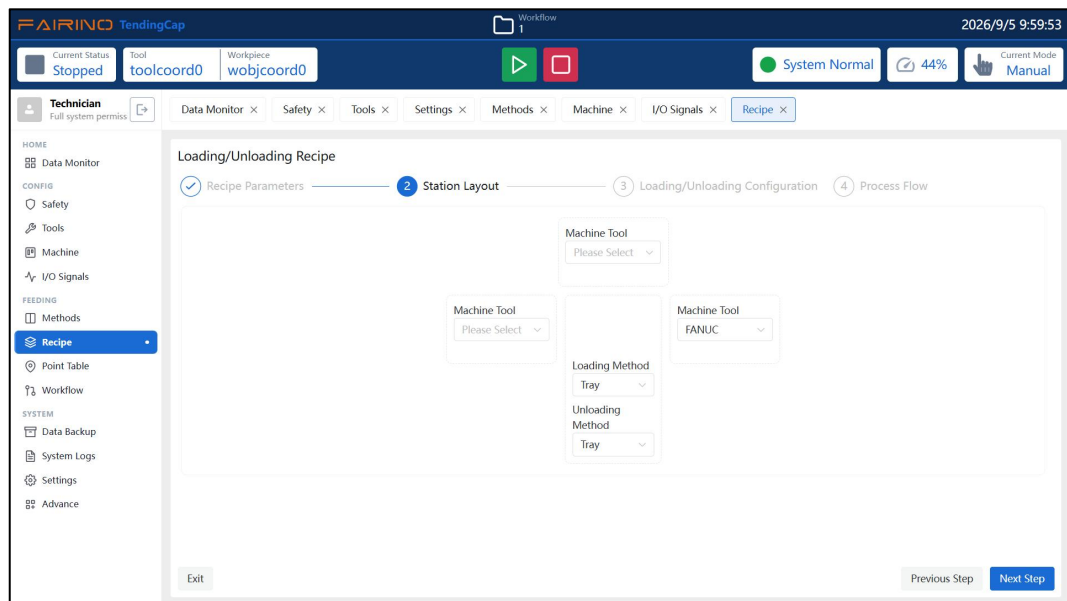


Figure 3.2.6-2 Process Layout for Recipe

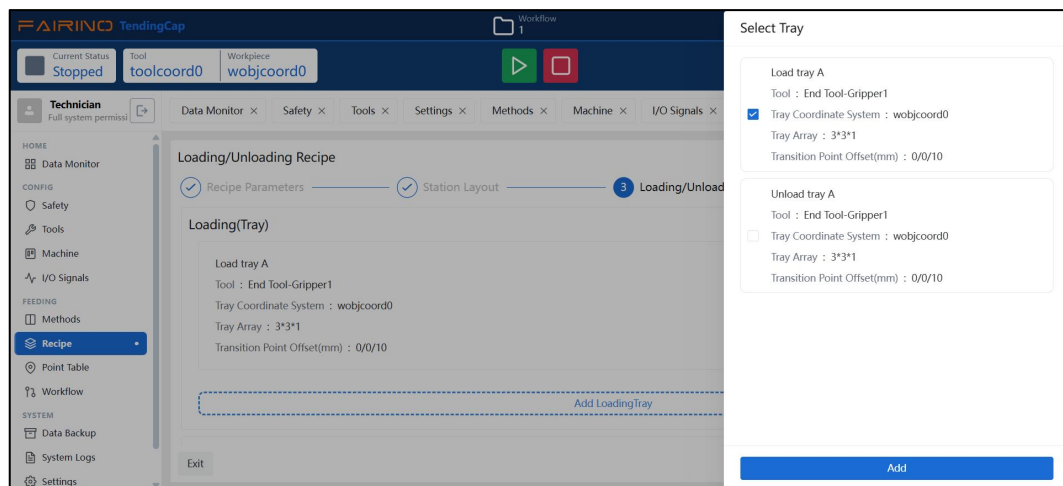


Figure 3.2.6-3 Formula - Material Loading Configuration

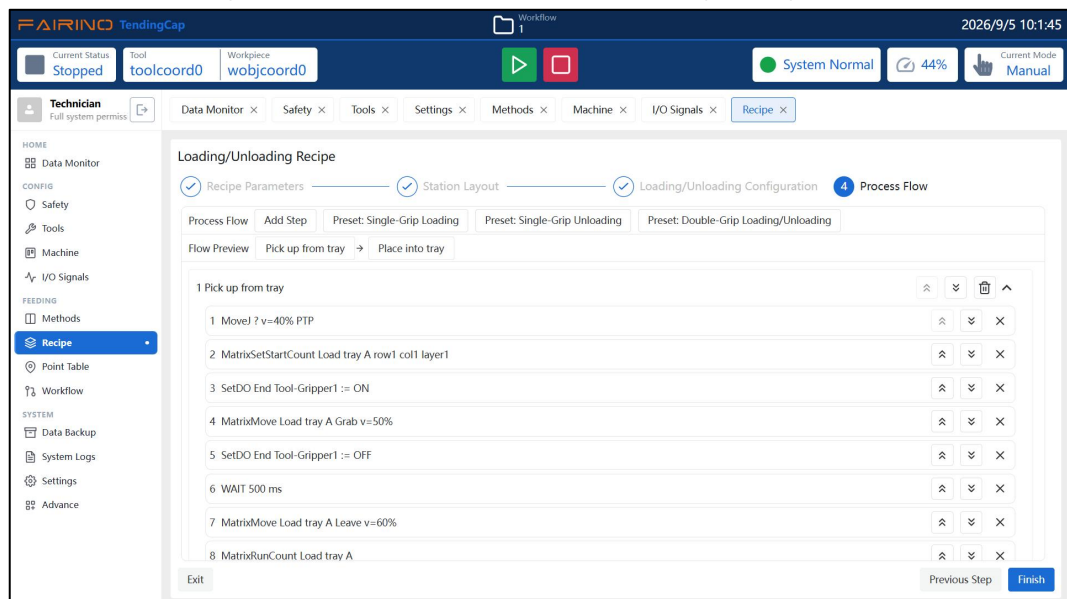


Figure 3.2.6-4 Recipe Process Flow

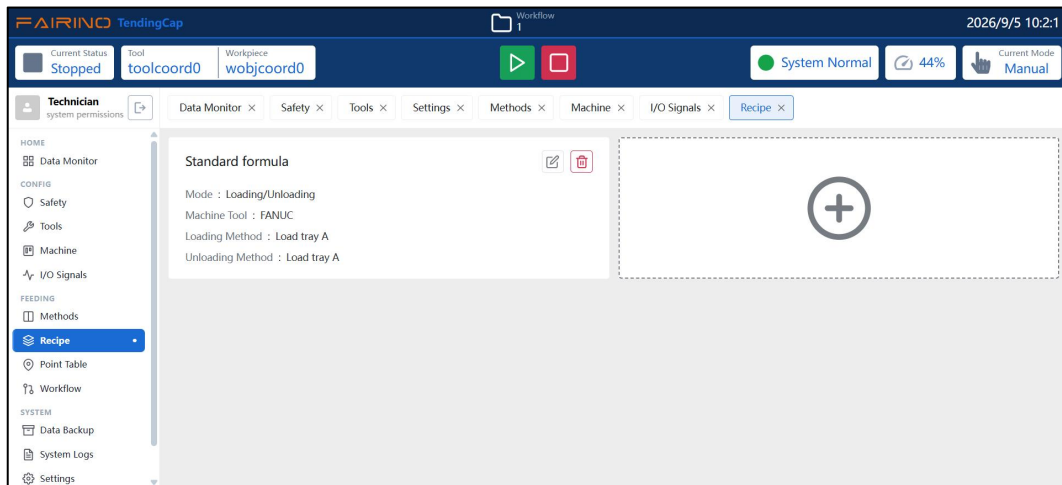


Figure 3.2.6-5 Recipe

3.3 Robot Manual Teaching

3.3.1 Manual demonstration and recording of points

Manual demonstration includes two methods:

- 1) Press and hold the end drag button or tap the top button to switch to drag mode for teaching.
- 2) In the "Loading/Unloading - Position Table" interface, under the "Move Card" section on the right side, switch current mode to manual mode, you can perform movements based on either Cartesian space or joint space.

After the teaching path reaches the target position, you can click the "Record Point" button in the "Loading/Unloading - Point Table" to save the point. When saving the point, the coordinate system for this teaching point will be the coordinate system used by the current robot application.

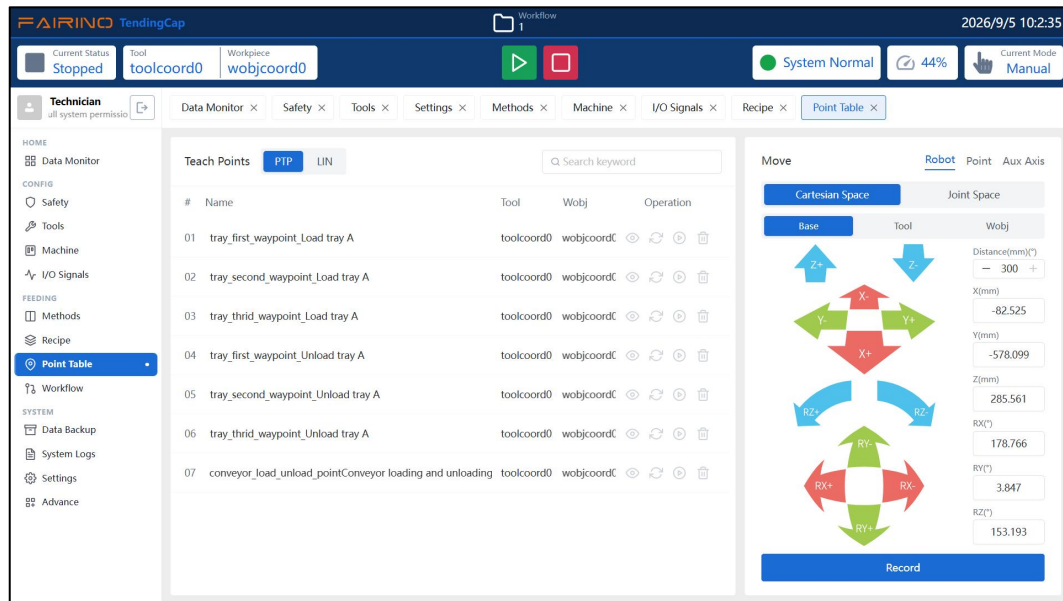


Figure 3.3.1-1 Robot Movement in Cartesian Space

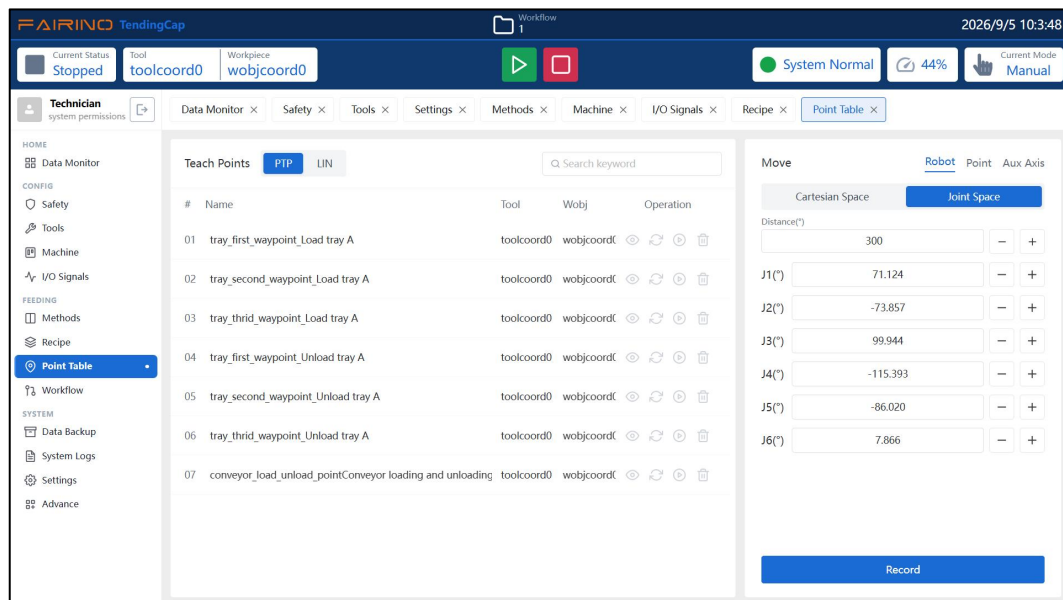


Figure 3.3.1-2 Robot joint-space motion

3.3.2 View points

The "Material Loading/Unloading - Point Table" displays all recorded point data. After entering each data entry, the operation buttons become highlighted in sequence: View, Update, Run Stepwise, and Delete. Clicking a point name opens an input field where you can enter a new name to rename it. In the upper-left corner of the table, select the Run Stepwise mode (PTP - Point-to-Point or LIN - Linear), and enter filtering criteria in the upper-right corner to filter and view points.

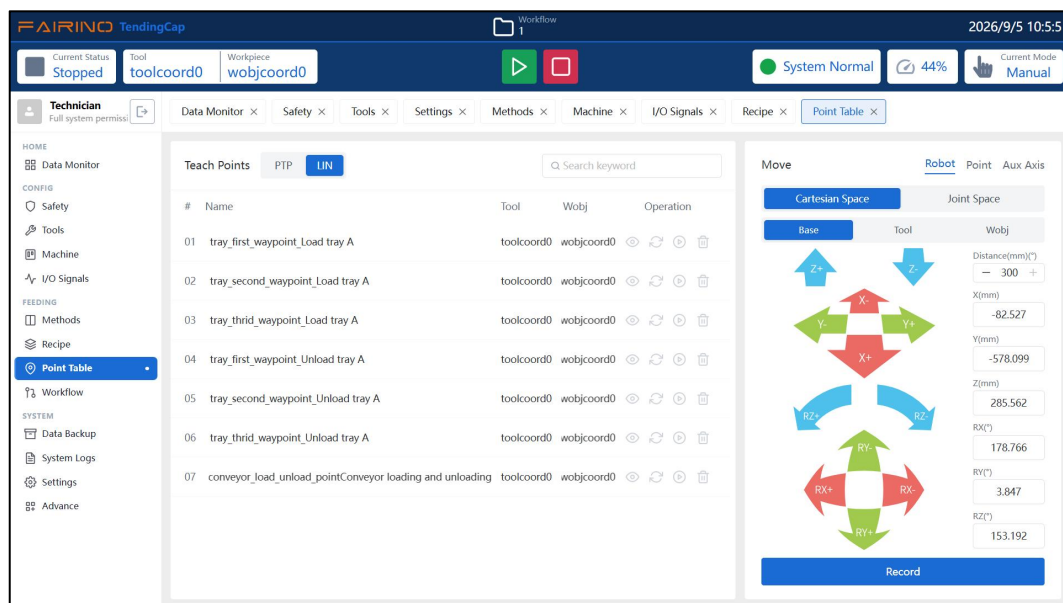


Figure 3.3.2-1 Point Table

3.4 Quick addition of CNC loading and unloading workflows

3.4.1 Introduction to Simple Workflow Instructions

3.4.1.1 MoveJ

Click "MoveJ" to open the "Edit MoveJ" interface on the right. Select the destination point, enable smooth transition, and enter the smooth transition time to ensure a seamless movement from one point to the next. The specific path is the optimal path automatically planned by the motion controller.

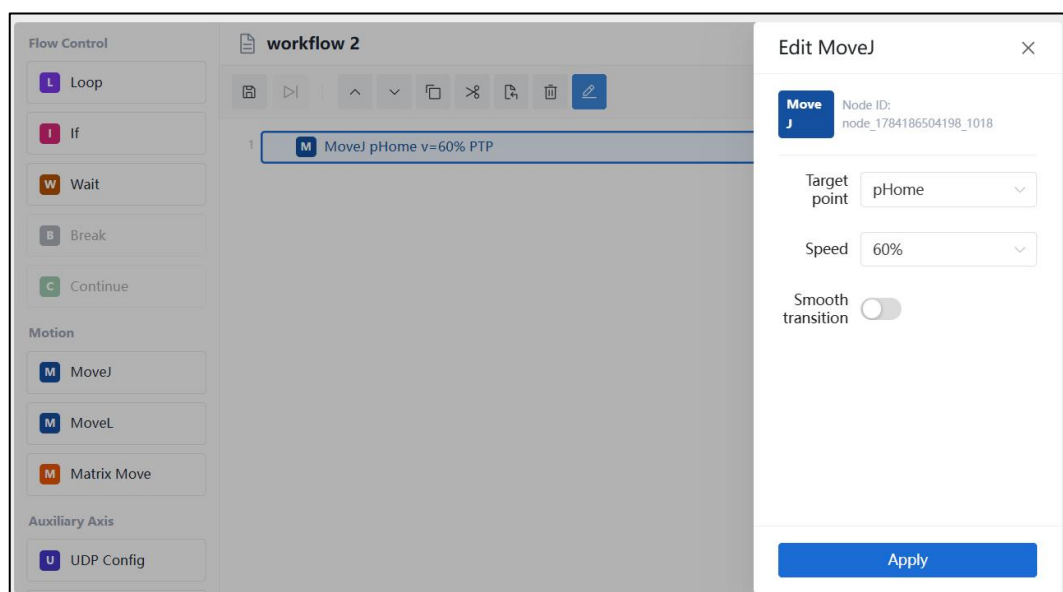


Figure 3.4.1.1-1 MoveJ instruction

3.4.1.2 MoveL

Click "MoveL" to open the "Edit MoveL" interface on the right, select the destination point, enable smooth transition, enter the smooth transition radius, and choose the transition method. The path to the destination is a straight line.

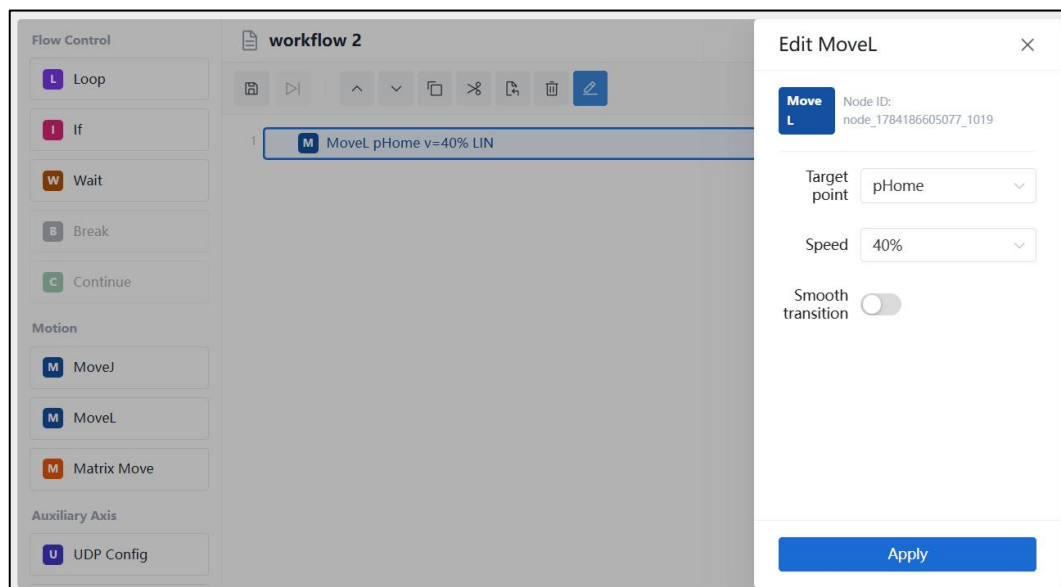


Figure 3.4.1.2-1 MoveL instruction

3.4.1.3 Matrix Move

Click "Matrix Move" to open the "Edit Matrix Move" interface on the right, select the tray and execution action, and the execution action scenario is as follows.

- **Matrix Movement:** When the robot moves to load/unload or grasp/release a selected pallet using an extension shaft, perform synchronous motion planning for the conveyor belt's extension shaft.
- **Matrix operation count:** Counting the row, column, and layer of the robot's loading/unloading operation.
- **Set starting count:** Define where the robot begins loading/unloading operations (row, column, and layer).
- **Get matrix count:** Determine the row, column, and layer of the robot's loading/unloading operation.
- **Total processing count +1:** Increase the total number of robot loading/unloading operations by 1.
- **Total cleared processing operations:** Total robot loading/unloading operations

plus.

- Get total processing count: Get total robot loading/unloading operations.

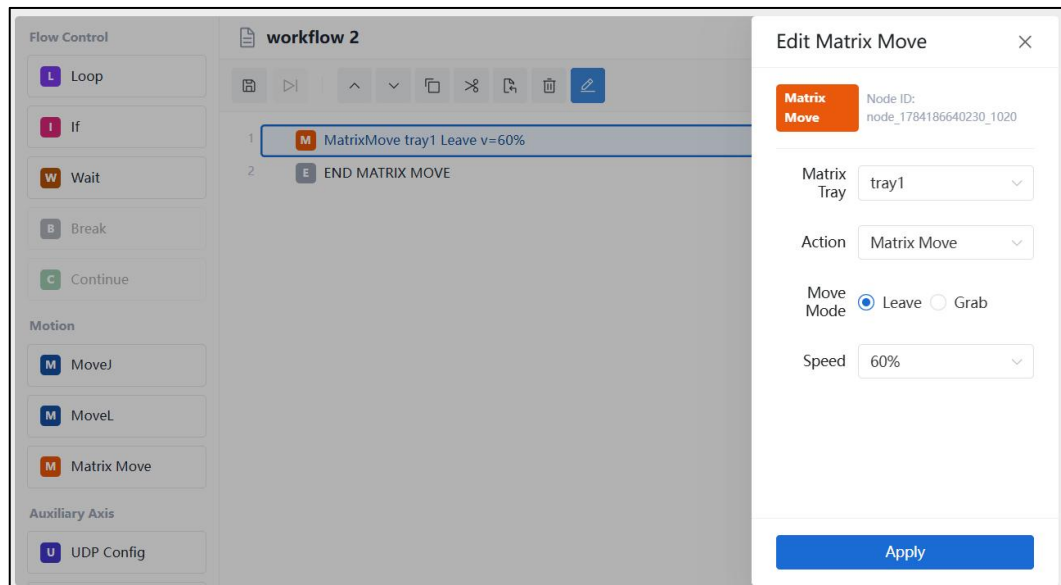


Figure 3.4.1.3-1 Matrix Move

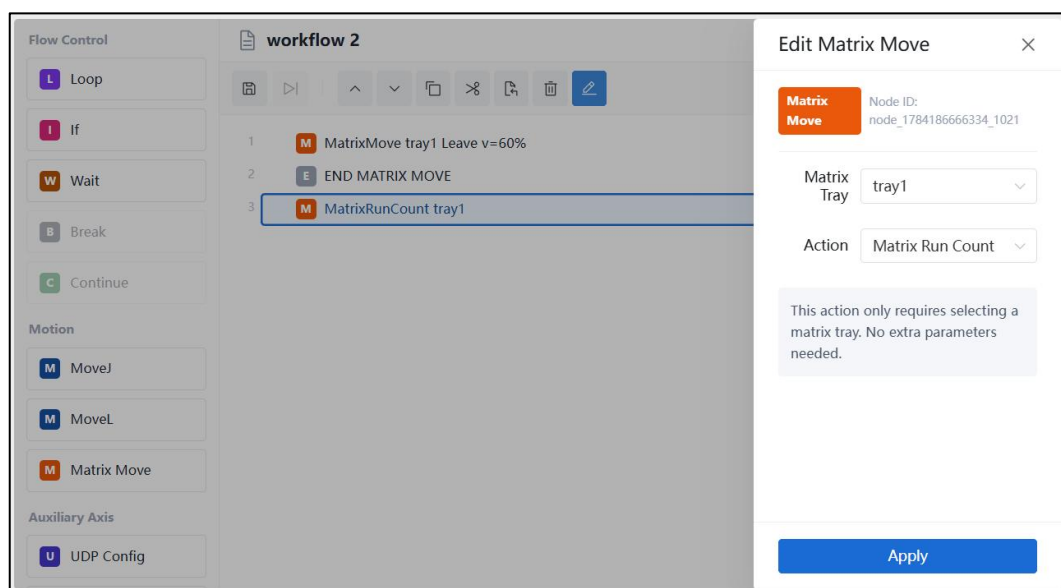


Figure 3.4.1.3-2 Matrix Execution Count

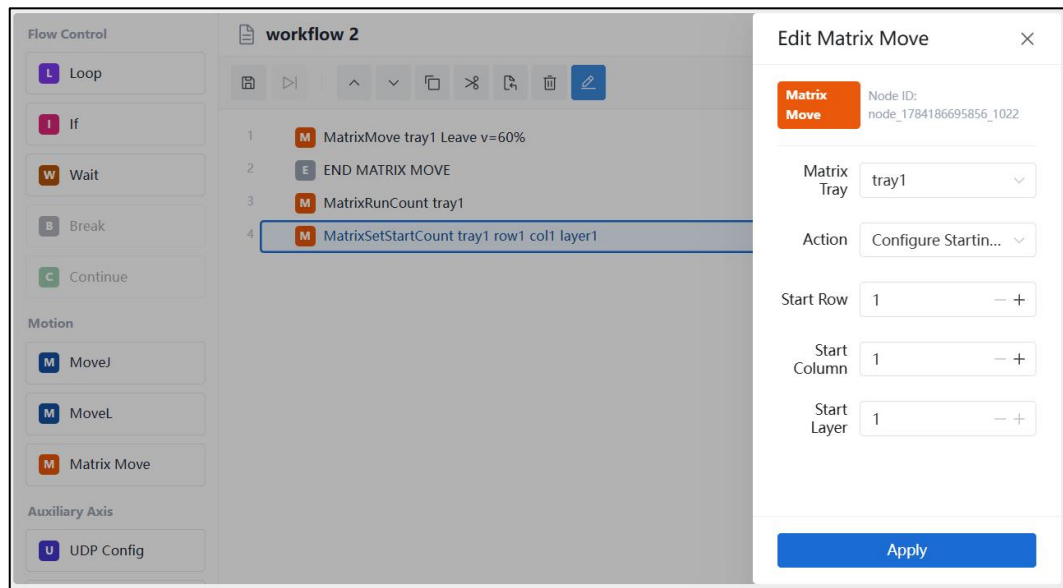


Figure 3.4.1.3-3 Configure starting count

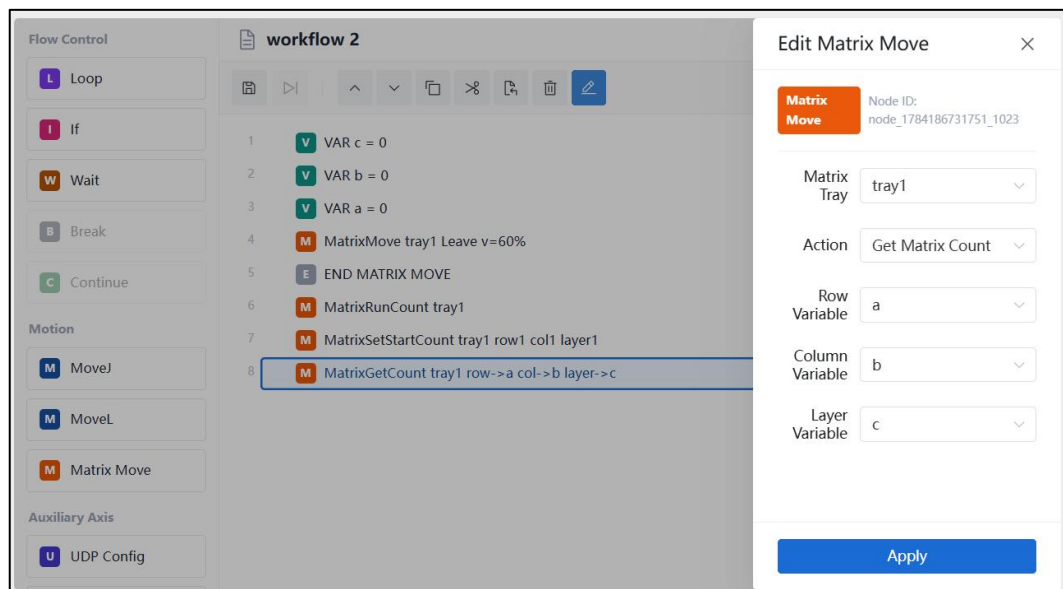


Figure 3.4.1.3-4 Get Matrix Count

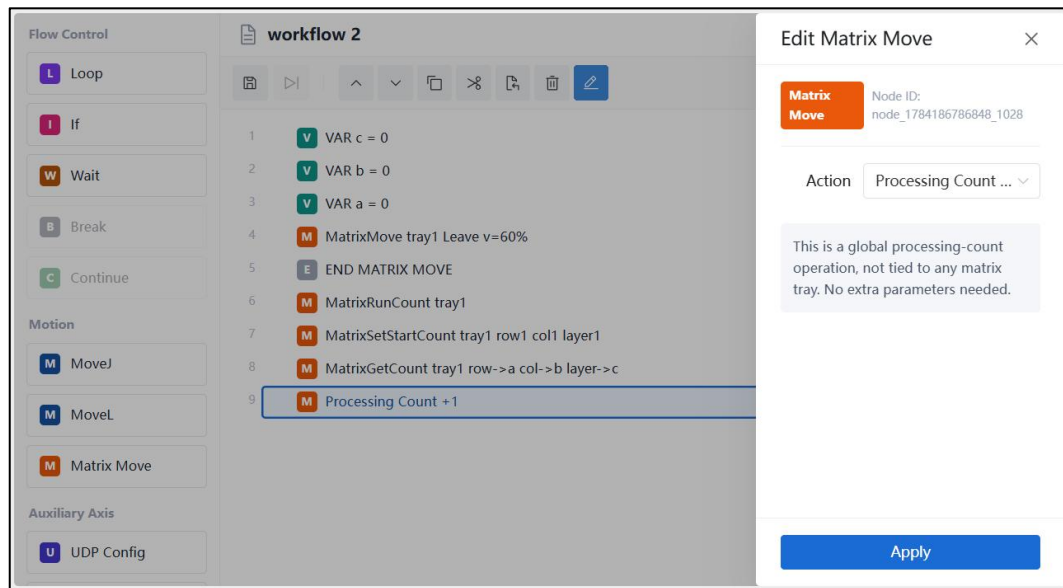


Figure 3.4.1.3-5 Total Processing Count +1

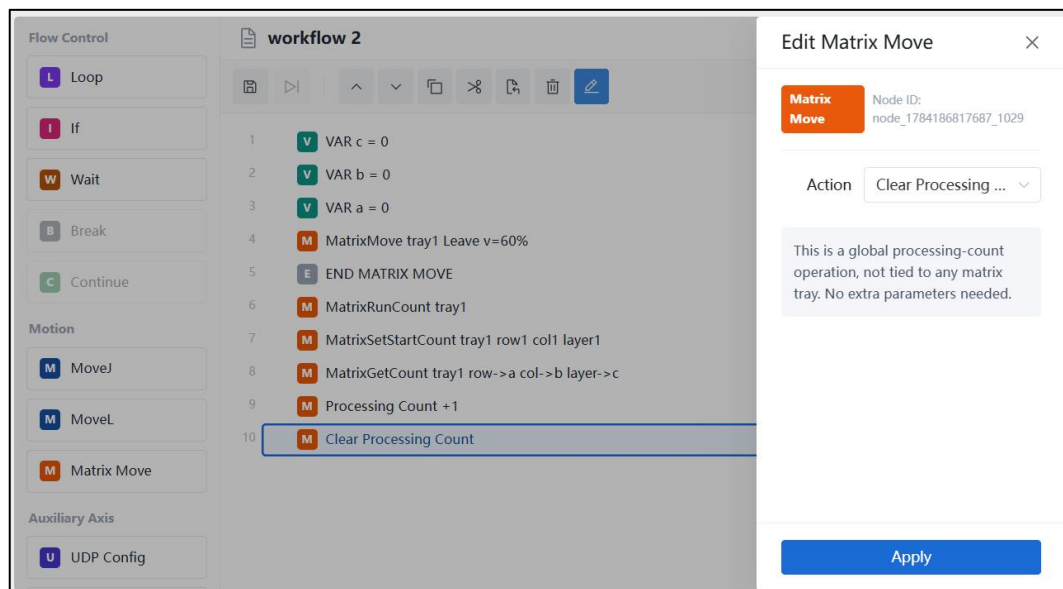


Figure 3.4.1.3-6 Total cleaned processing count

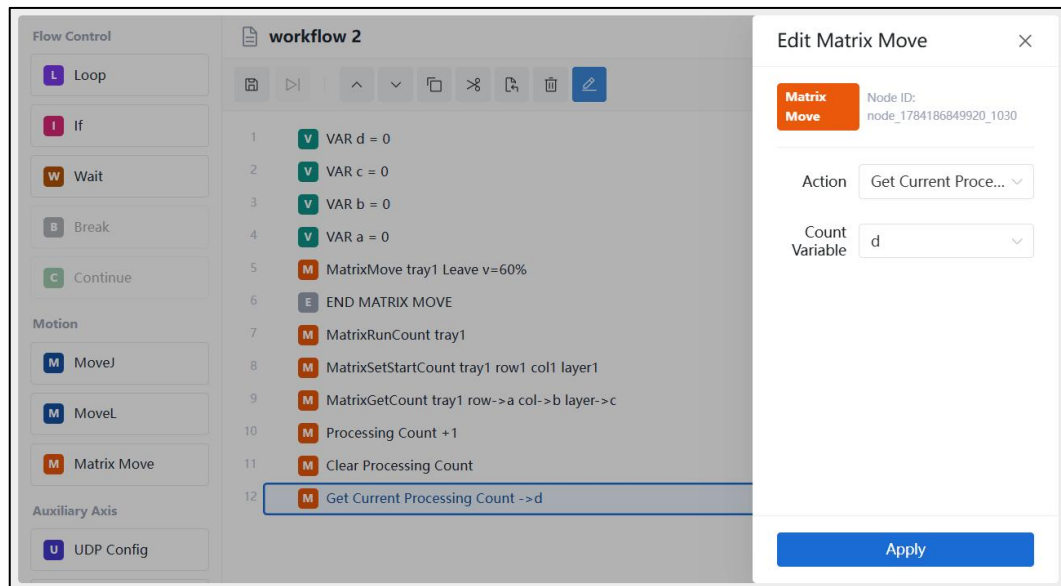


Figure 3.4.1.3-7 Get the total number of current processes

3.4.2 Perform operations on workflows



Name: Save. Function: Save workflow content.



Name: Run in One Step. Function: executes the selected workflow instruction in a single step.



Name: Move Up. Function: Move the selected workflow instruction up;



Name: Move Down. Function: Move the selected workflow instruction down;



Name: Copy. Function: Copy selected workflow instructions.



Name: Cut. Function: Cut the selected workflow instruction.



Name: Paste. Function: Paste a copied or cut-and-paste workflow instruction.



Name: Delete. Function: Remove the selected workflow instruction.



Name: Edit. Function: Edit selected workflow instructions.

3.4.3 Run a Workflow

The left side primarily displays workflow list data; to create a new workflow, click the "Add" icon in the upper-left corner of the list; select the desired items from the list; then click the "Delete" button at the bottom to remove them.

After selecting any workflow, the workflow instructions and operation bar will be displayed. Click on any keyword on the left side to open the detailed interface for adding a workflow instruction; after filling in the detailed information for each instruction, click the "Apply" button at the bottom to save the added workflow instruction to the selected workflow file on the right side. For instructions that do not require parameters, simply click them to add them to the workflow content area.

Click the Start button to run the workflow; click the Stop button to stop the workflow; click the Pause/Restore button to pause or resume the workflow; during program execution, a running icon will appear immediately before the currently executing workflow node.

In manual mode, select any node in the workflow; clicking the "Run Step-by-Step" button allows the robot to execute this instruction individually, while clicking the "Edit" button enables you to modify the parameters of the workflow instructions.

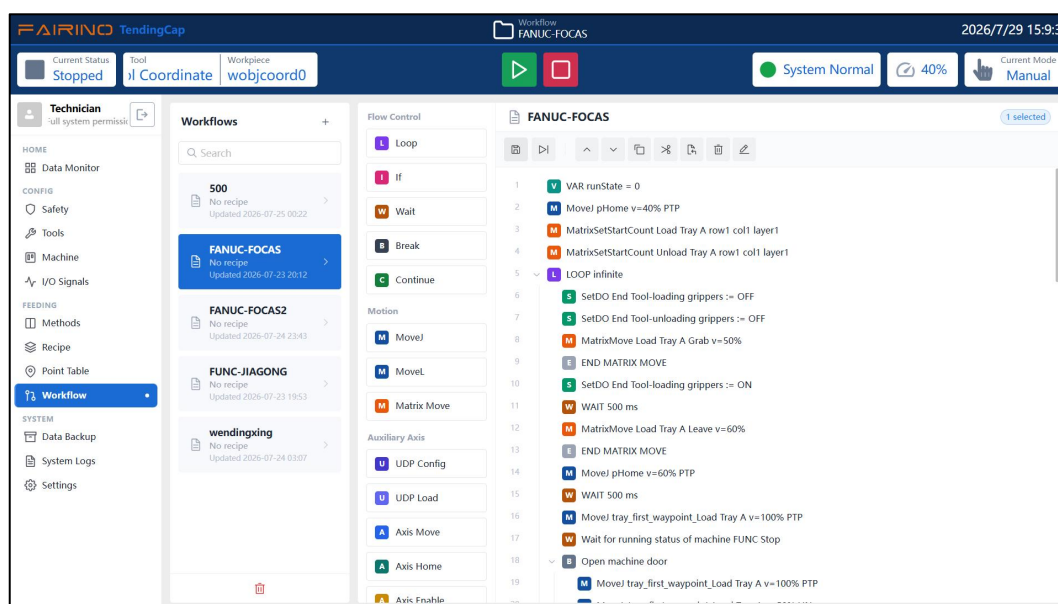


Figure 3.4.3-1 Workflow

4 Role Entry and Permissions

Upon initial launch of the CNC loading/unloading Cap software, users choose

their operating role according to their job function. Each role is granted specific permissions, and there is no need for account registration or login. The current roles are Operator and Technician, with the following permissions:

- **Operator:** Responsible for daily equipment operation, program execution, status monitoring, and production count management. No password verification is required; direct system access is granted.

- **Technician:** All operator permissions, plus system configuration, coordinate calibration, I/O debugging, workflow programming, and all management functions. Password verification is required for access; the initial password is 1234.

Role entry and permissions are as shown in the figure below:

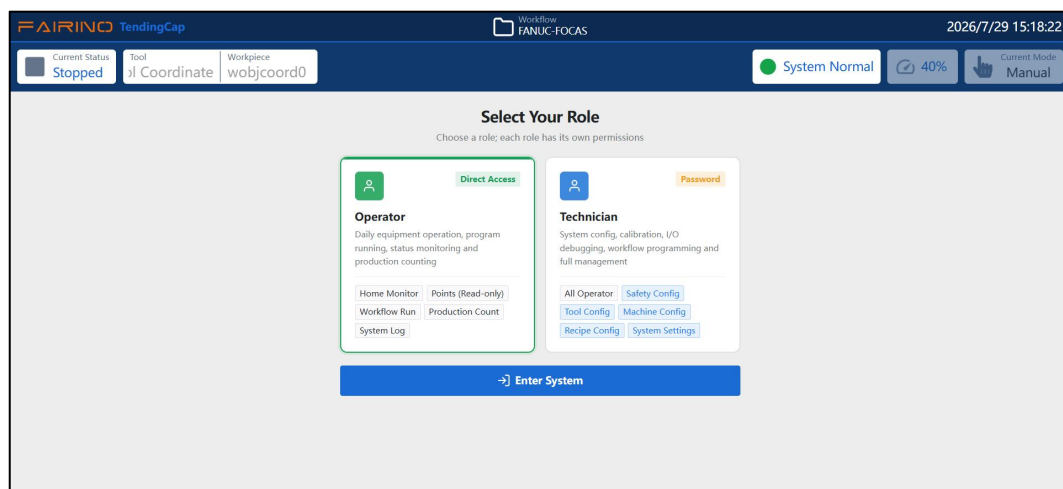
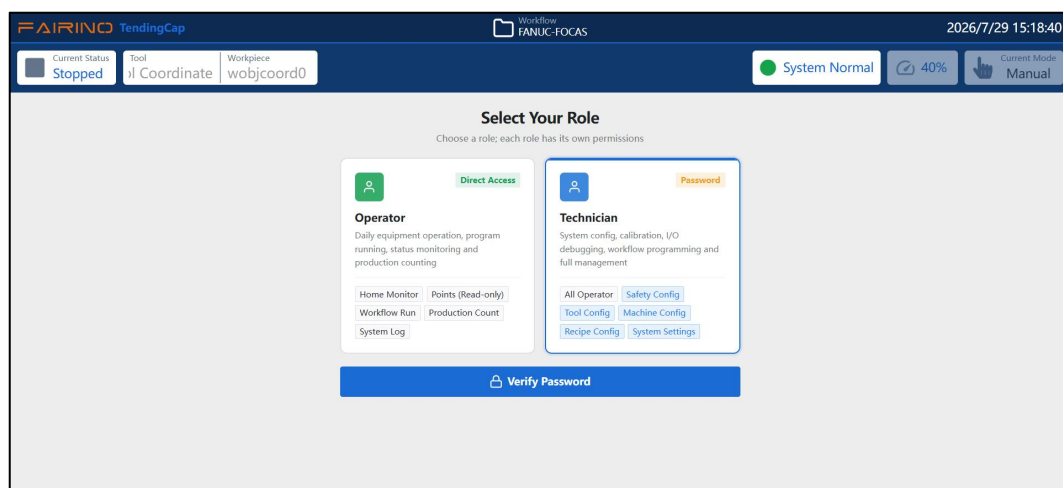


Figure 4-1 Operator role permissions



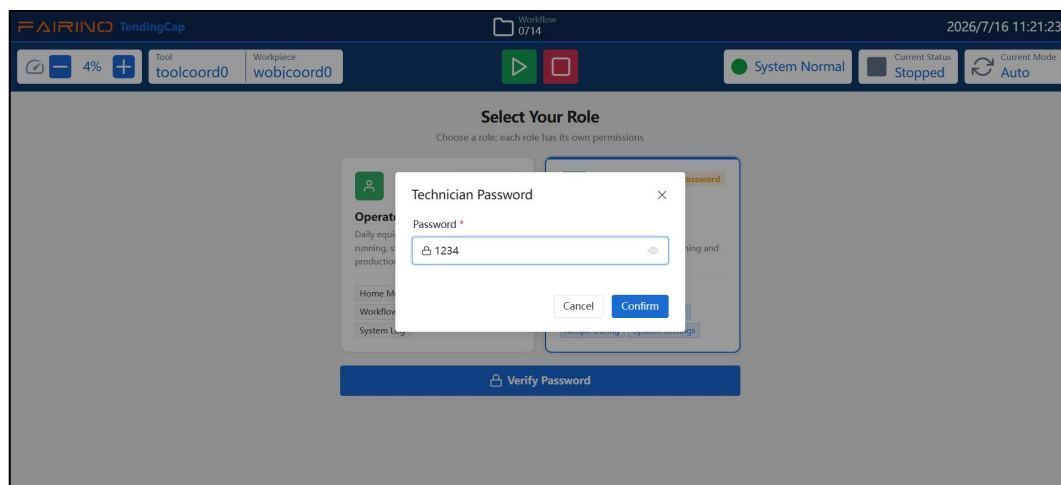


Figure 4-2 Technician role login password verification and permissions

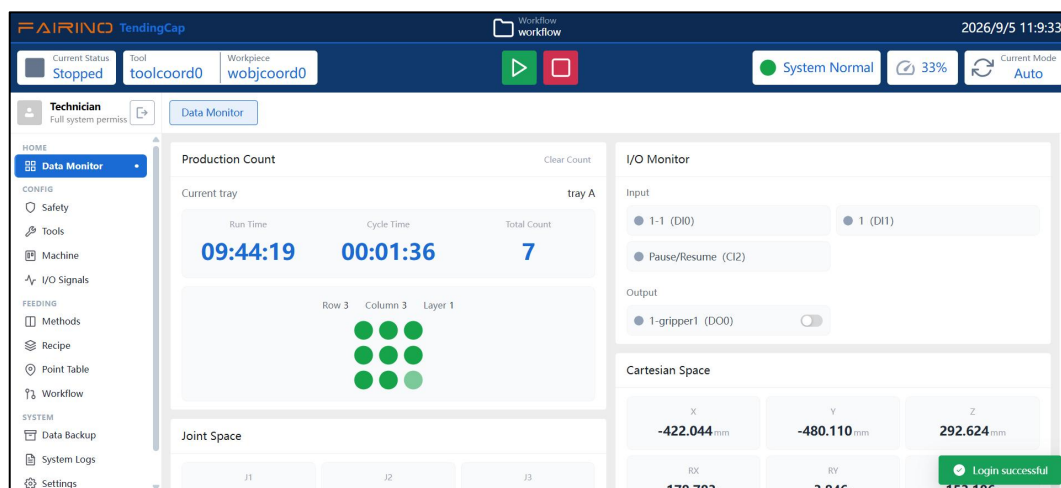


Figure 4-3 Technician role login successful

5 System initial interface

After successful login, enter the "Initial Interface" of the CNC loading/unloading Cap software, which mainly includes:

- Top first layer (from left to right): FAO logo, current running workflow name, current system time.
- Top second layer: robot running status, current tool/workpiece coordinate system, robot control area (start, stop, pause, resume), system status (normal, error, and warning), robot global speed, robot manual/automatic mode.
- Above the menu bar is the currently logged-in role. Click the exit icon on the far right to log out.

- Menu bar and corresponding menu interfaces

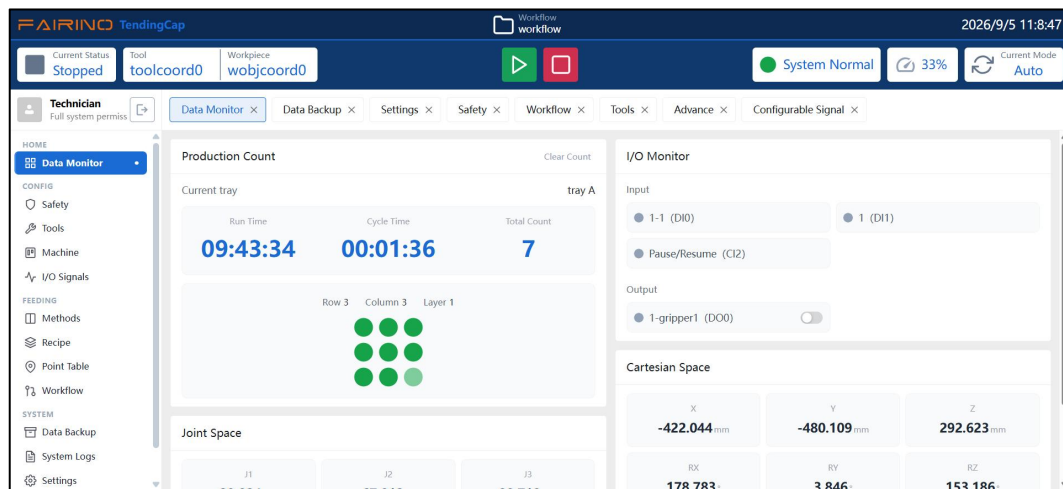


Figure 5-1 The initial interface of the CNC loading/unloading Cap software

5.1 Current Workflow Name

When the user opens a workflow, record the workflow name. When the user closes the interface and re-enters, display the current workflow name.



Name: Current Workflow Name. Function: Click to navigate to the Workflow interface in Loading/Unloading and open the content corresponding to the current workflow name.

5.2 Robot operating status



Name: Running. Function: Displays the operational status of the robot.



Name: Pause. Function: Displays the paused status of the robot.



Name: Stop. Function: Displays the stopped state of the robot.

5.3 Current tool coordinate system

Display the current application's tool coordinate system. Click the dropdown to show all tool coordinate systems, and click on any of them to apply.

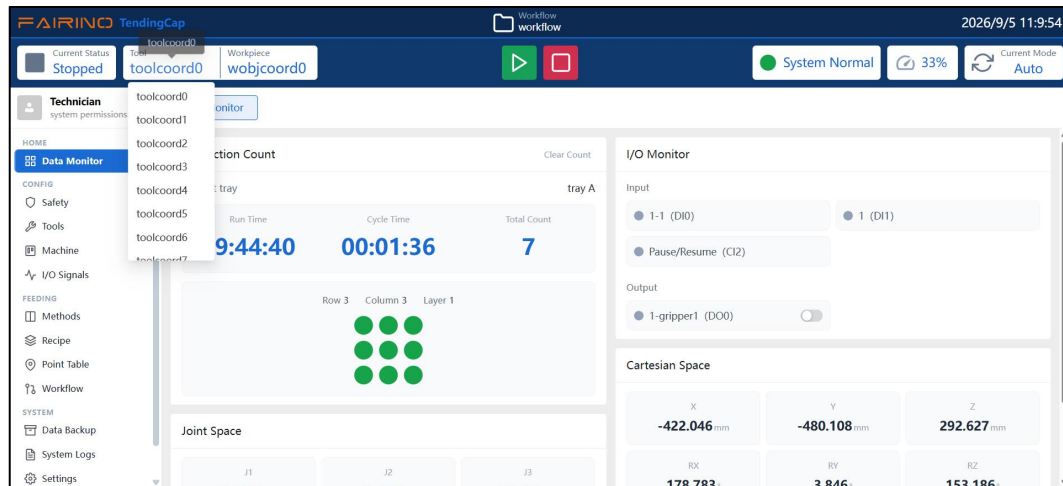


Figure 5.3-1 Current tool coordinate system

5.4 Current workpiece coordinate system

Display the current application's workpiece coordinate system. Click the dropdown to show all workpiece coordinate systems, and click any of them to apply it.

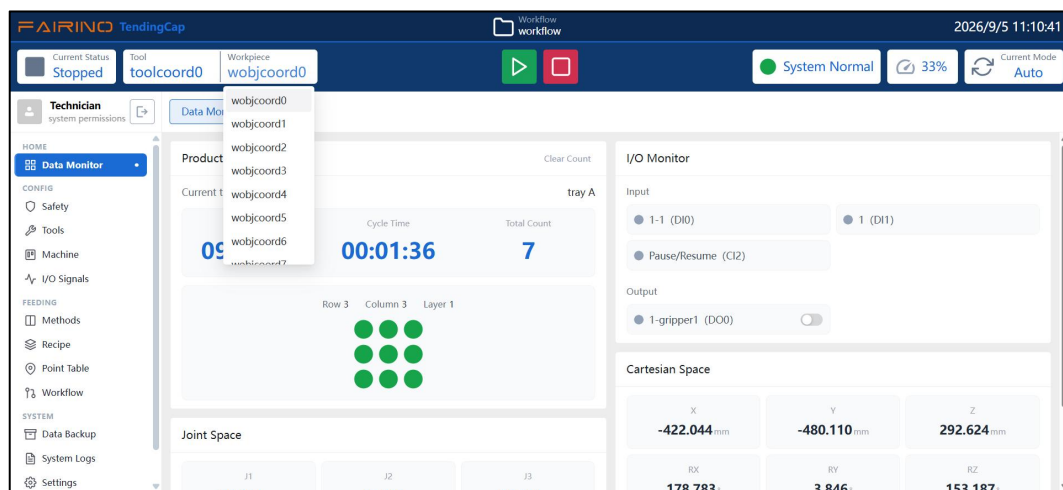


Figure 5.4-1 Current workpiece coordinate system

5.5 Robot Control Area



Name: Start. Function: Upload and start running the workflow.



Name: Stop. Function: Stops the current workflow run.



Name: Pause. Description: Pause the current workflow execution.



Name: Resume. Function: Resume the current workflow execution.

5.6 System Status

Display the current system status: Running Normally, Has Errors, or Has Warnings. When errors or warnings are present, hovering over the status area will display the detailed error and warning messages. Click the Clear button at the bottom to clear some of the errors.



Name: Normal system state. Function: The system is currently normal.



Name: Error/Warning Status. Function: Displays when warnings or errors are present; hides when there are no errors or warnings.

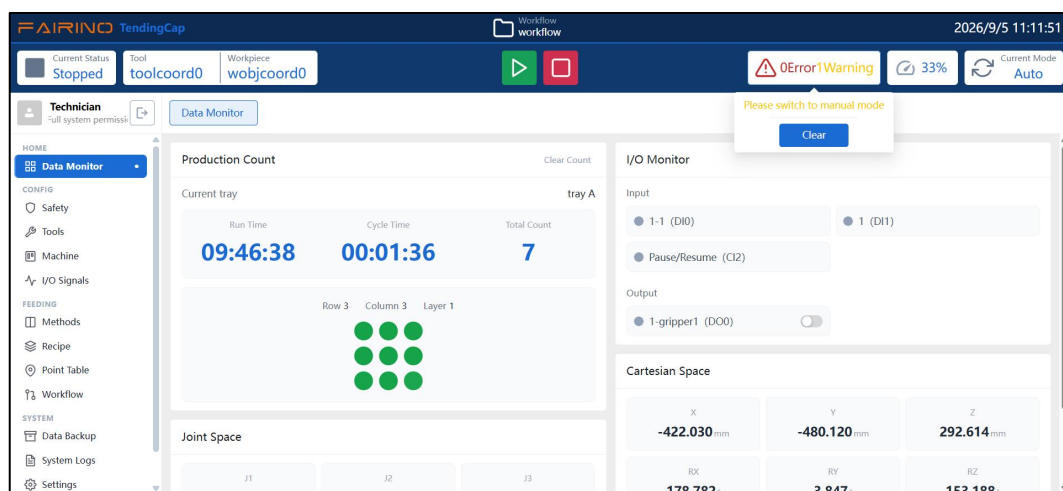
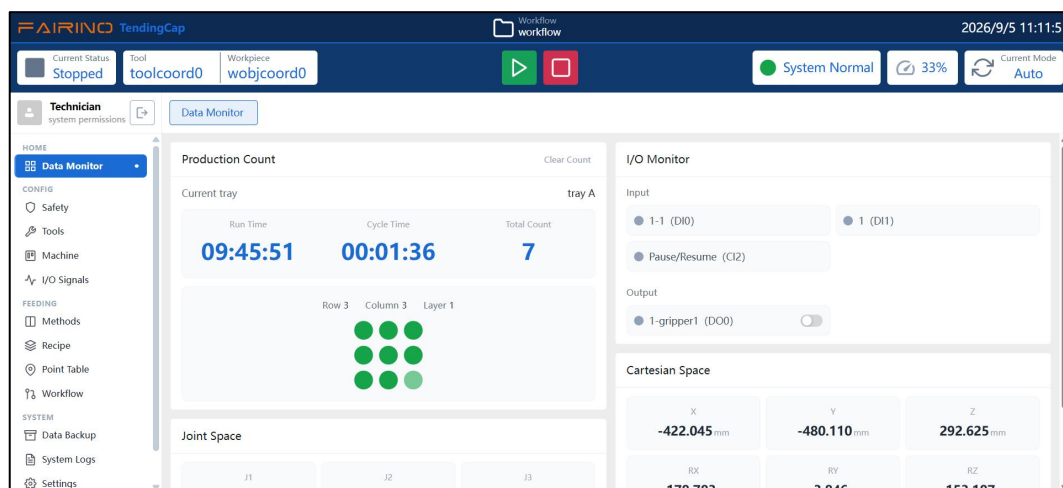


Figure 5.6-1 System Status

5.7 Robot global velocity

After moving to the robot's global speed, display the slider for the robot's global speed, which can be adjusted by sliding, and can also be modified by entering values in the input box.

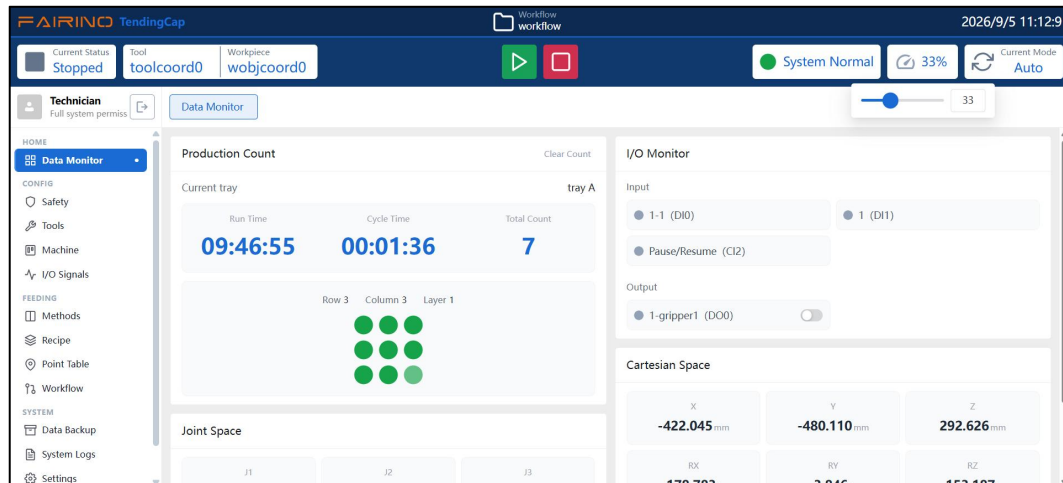


Figure 5.7-1 Robot global velocity

5.8 Robot Current Mode



Name: Automatic Mode. Function: Robot automatic operation mode.



Name: Manual Mode. Function: Robot manual mode for teaching/teach pendant operations.



Name: Drag Mode. Function: The robot's drag mode for performing dragging operations. It cannot be switched to Drag Mode in Automatic Mode; only switchable from Manual Mode. Once the robot enters Drag Mode, it cannot revert to Automatic Mode—switching to Manual Mode will exit Drag Mode.

5.9 Current logged-in role

After the role successfully logs in, it is displayed on the menu bar. Click the rightmost icon to pop up a 'Confirm Logout' prompt.



Name: Logout. Function: Display a logout prompt for secondary confirmation by the user.

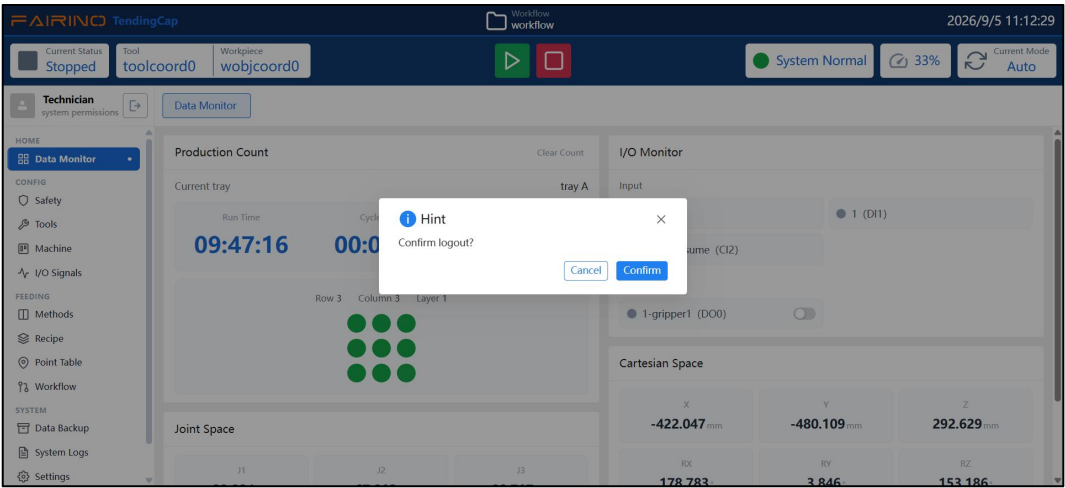


Figure 5.9-1 Confirm exit prompt

5.10 Menu bar

Table 5.10-1 Menu bar table

Menu Type	Menu Bar Name
Home	Data Monitor
	Safety
Config	Tools
	Machine
	I/O Signals
	Methods
Feeding	Recipe
	Point Table
	Workflow
	Data Backup
System	System Logs
	Settings
	Advance

6 Home

6.1 Data Monitor

View real-time status data for loading/unloading production counts, signal monitoring, and joint space and Cartesian space states in Home > Data Monitor. The modules are described as follows:

- Production count: Current loading/unloading formula, running time and cycle duration of the current formula, current row, column, and layer number, indicated by a flashing green light.
- Signal Monitoring: Real-Time Status Display of I/O Corresponding to Each Function During Loading and Unloading Processes.
- Joint space: current data of robot joints J1 to J6.
- Cartesian space: current robot Cartesian pose data X, Y, Z, RX, RY, and RZ.

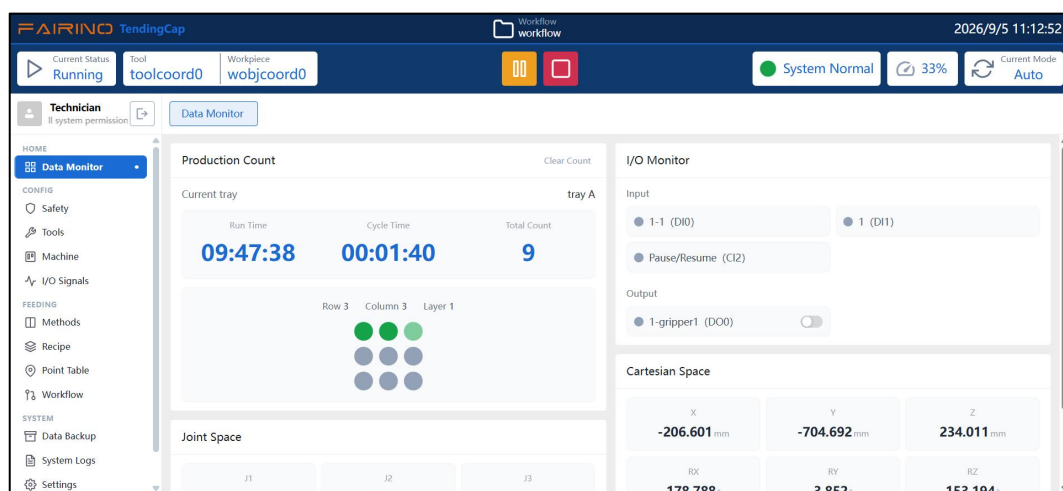


Figure 6.1-1 Data Monitor

7 Config

7.1 Safety

Perform safety point teaching and main program configuration under "Configuration - Security".

7.1.1 Safety Point

The security point name is fixed as "pHome". If the safety point does not exist, prompt "No pHome point data found". Click the "Configure" button to send the current robot pose as the safety point data. After the safety point is successfully configured, click the "Move" button to move to the safety point position for verification.

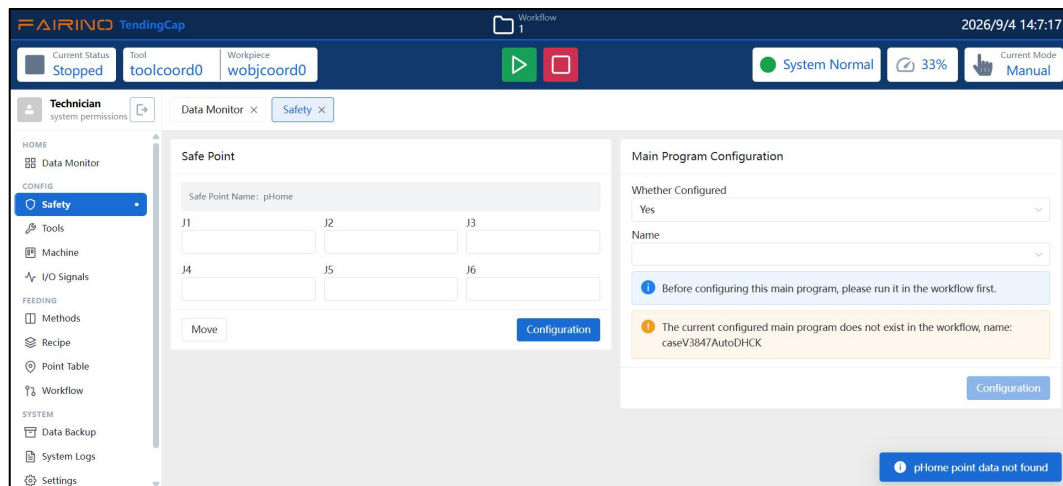


Figure 7.1.1-1 pHome point data not found

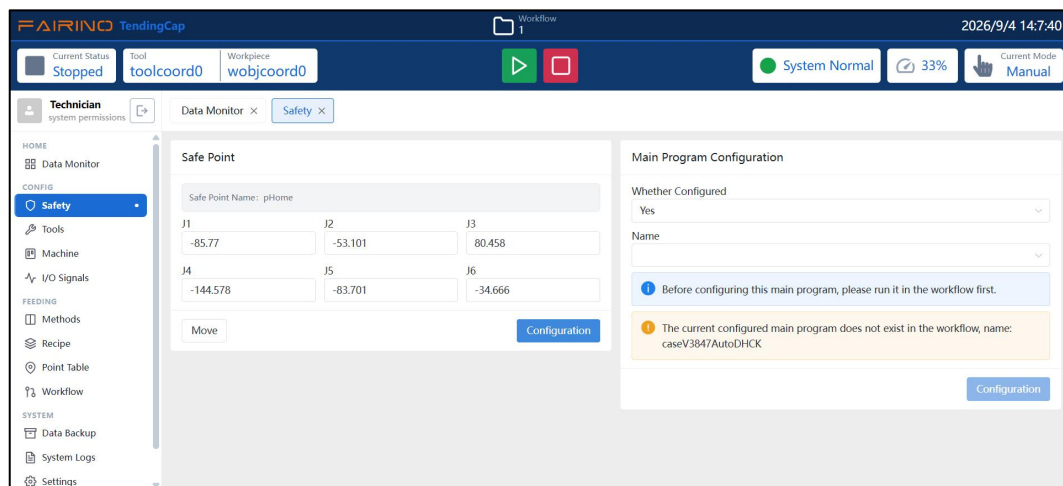


Figure 7.1.1-2 Safety point configured successfully.

7.1.2 Main Program Configuration

The Configuration Main Program can be used in conjunction with the DI Input Signal Configuration Main Program (available under Advanced Parameters). The configured main program should first be tested for safety purposes; after configuring the corresponding DI signal as the "Start Main Program" signal function in the Robot Settings, the user can use this DI signal to initiate the main program.

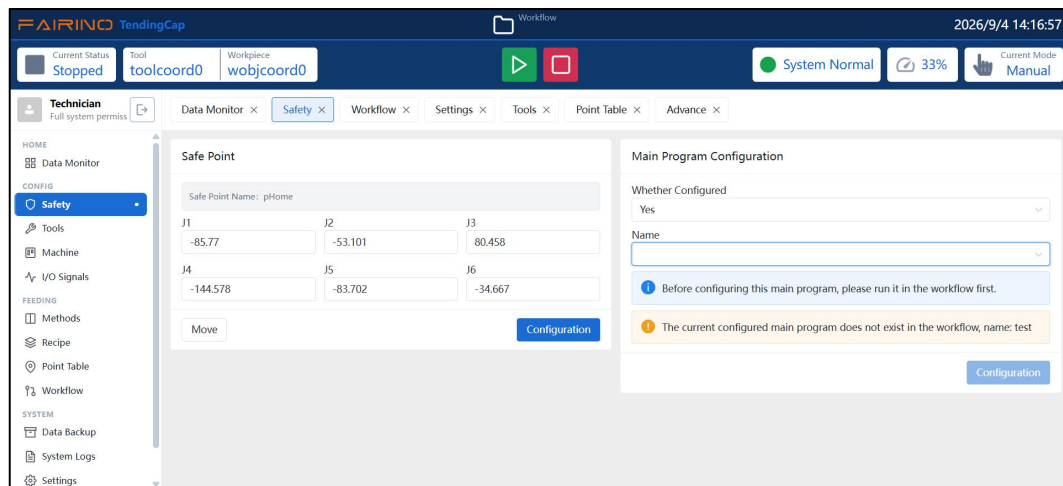


Figure 7.1.2-1 The configured main program is not part of the workflow.

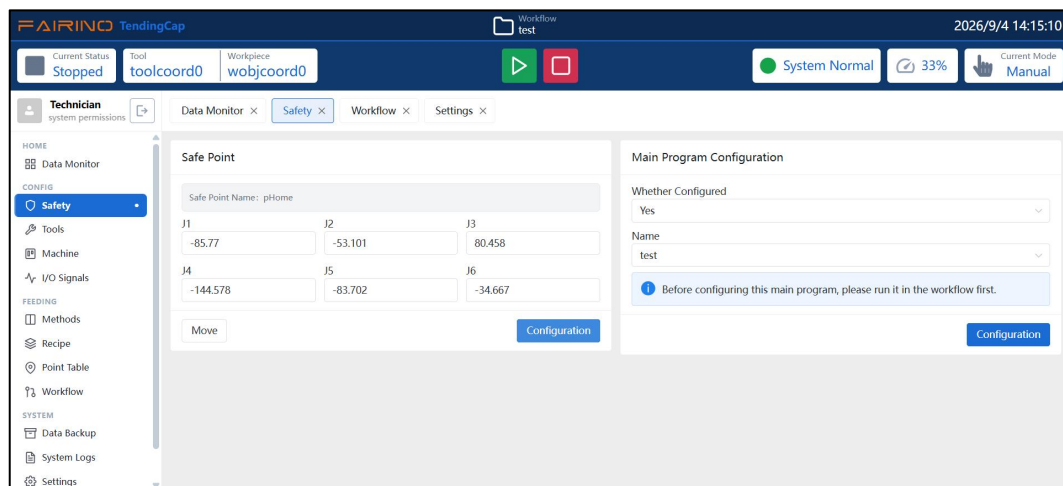


Figure 7.1.2-2 The configured main program is part of the workflow.

7.2 Tools

Configure the end effector in "Config—Tools". Upon entering the page, retrieve the tool list data. When no data exists, display 'No data available'.

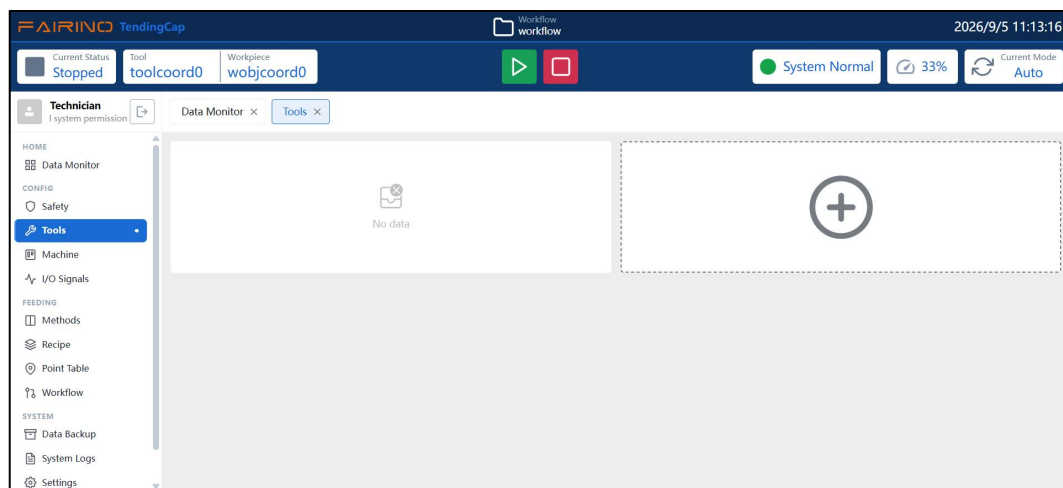


Figure 7.2-1 The tool currently has no data.

Click the 'Add' card to add a new tool in three steps, as described below.

- **End Effector:** Enter the name of the end effector according to the functional requirements; if the name is empty, clicking the "Next" button will display the message "Name cannot be empty." Specify the number of grippers (deletable), configure the cylinder type for each gripper, and, after selecting the single or double cylinder type for the grippers, configure the I/O signals and perform testing.

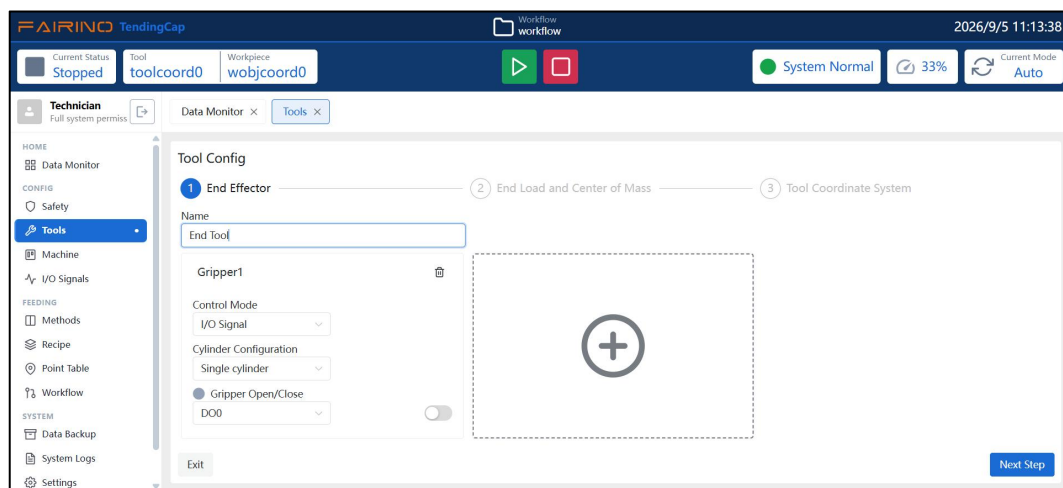


Figure 7.2-2 End effector

- **End load and center of mass:** Select the load number, enter the weight (unit: kg) and the center of mass coordinates X, Y, Z (unit: mm). The load data will be applied after switching the load number.

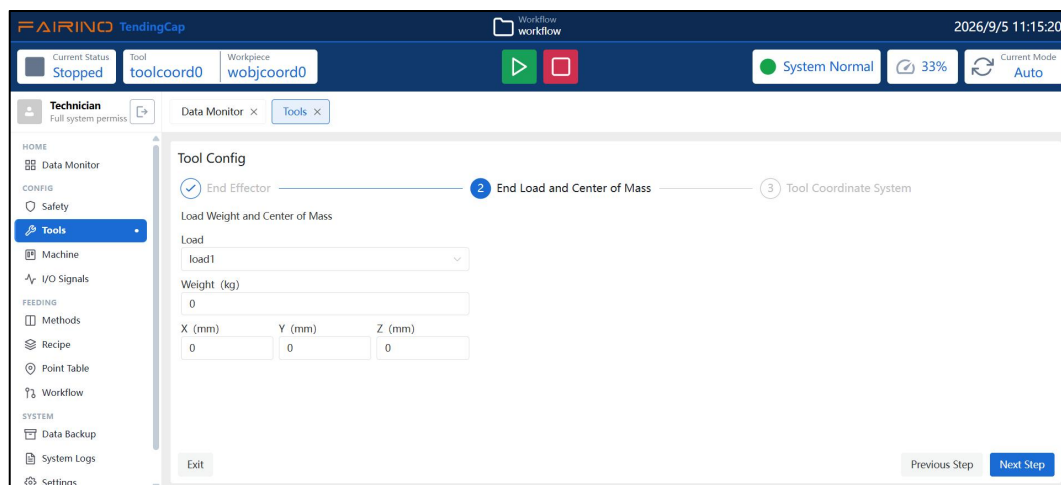


Figure 7.2-3 End load and centroid

● Tool coordinate system: Configure the tool coordinate system corresponding to the gripper, then click the Done button to complete adding the new tool. To switch to a non- default tool coordinate system, tap the “Calibrate” button at the bottom to bring up the calibration interface for the corresponding gripper tool coordinate system. The default calibration method is the “4- point method” ; on this interface, you can also rename the tool coordinate system.

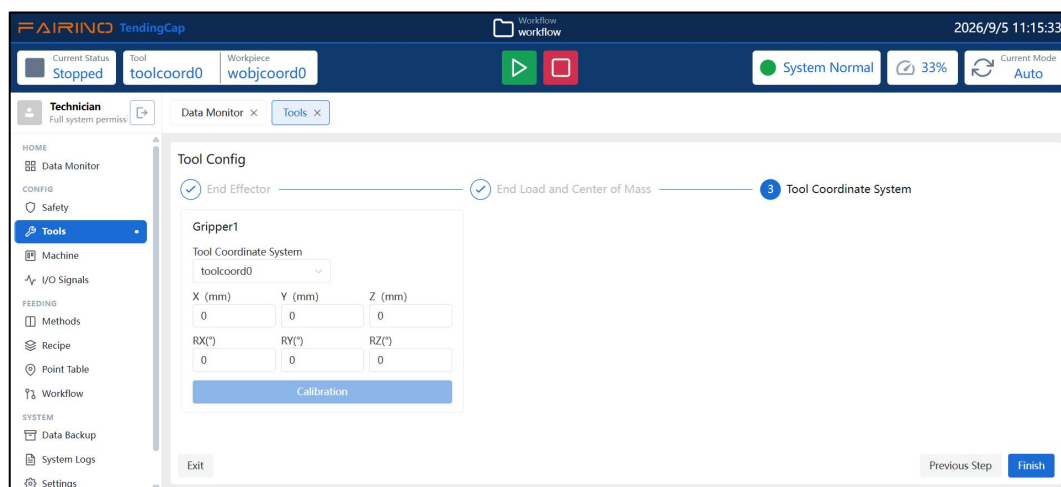


Figure 7.2-4 Tool coordinate system

After clicking the Complete button and successfully adding new data, you will return to the tool homepage, where the data list will be reloaded. Click the Edit button to view or modify the tool's data. Click the Delete button, and a secondary confirmation dialog with the title "Delete End Effector" will appear; click the Confirm button to complete the deletion.

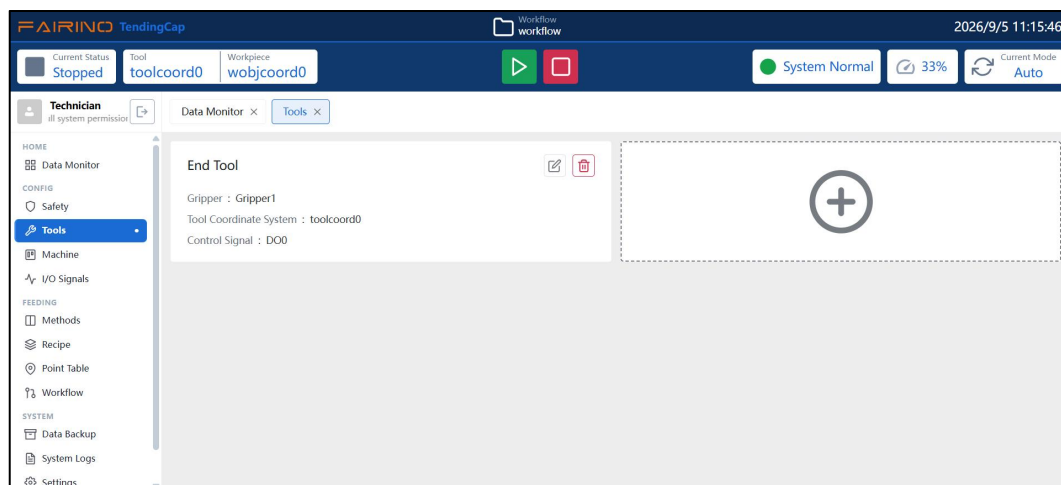


Figure 7.2-5 Tool List

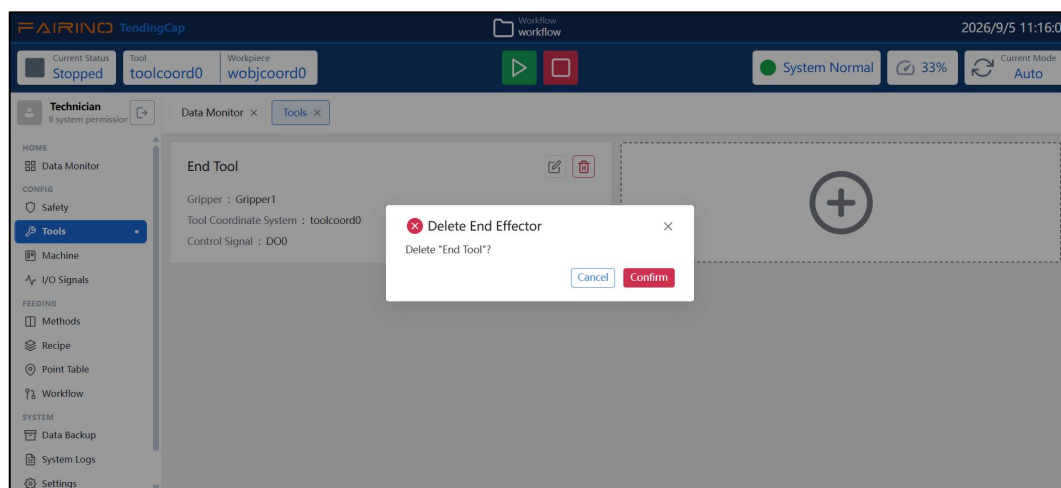


Figure 7.2-6 Confirm before deleting tool

7.2.1 Tool Coordinate System Calibration

The four-point method only calibrates the tool TCP, i.e., the position of the tool center point, and its orientation is by default consistent with that of the end effector. Select a fixed point in the robot workspace, move the tool to the fixed point in three different orientations, and teach Points 1, 2, and 3 in sequence. Then move the tool vertically to the fixed point and teach Point 4. After teaching is complete, click the Calculate button to view the calibration results. Click the Apply button to apply the calculated results to the tool coordinate system.

Important:

- The default tool coordinate system cannot be calibrated. When a non-default tool coordinate system is selected, clicking the calibration button will automatically

apply the default tool coordinate system.

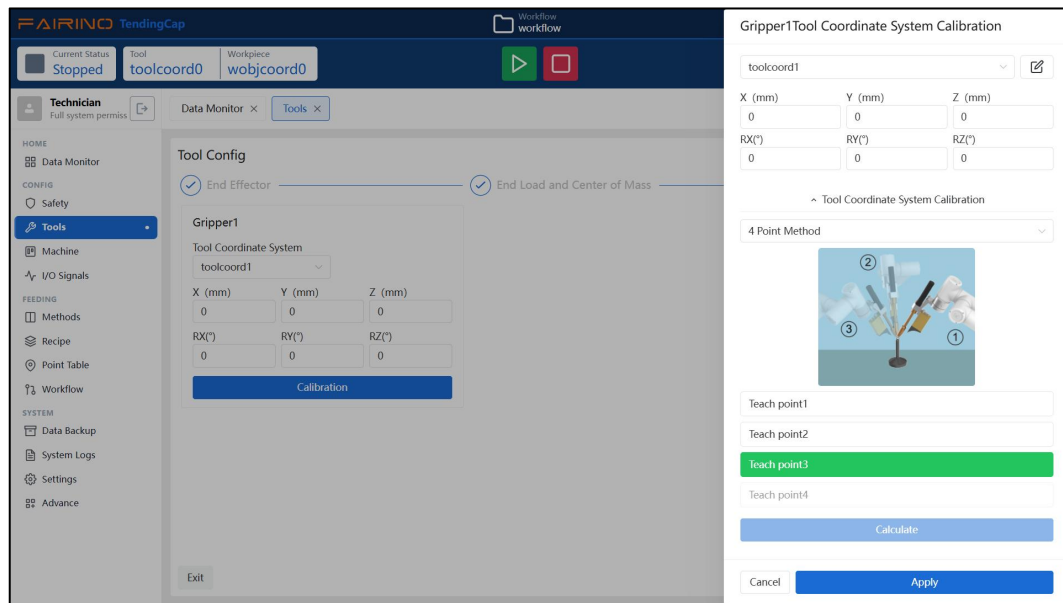


Figure 7.2-7 Tool Coordinate System Four-Point Calibration Points 1, 2, and 3

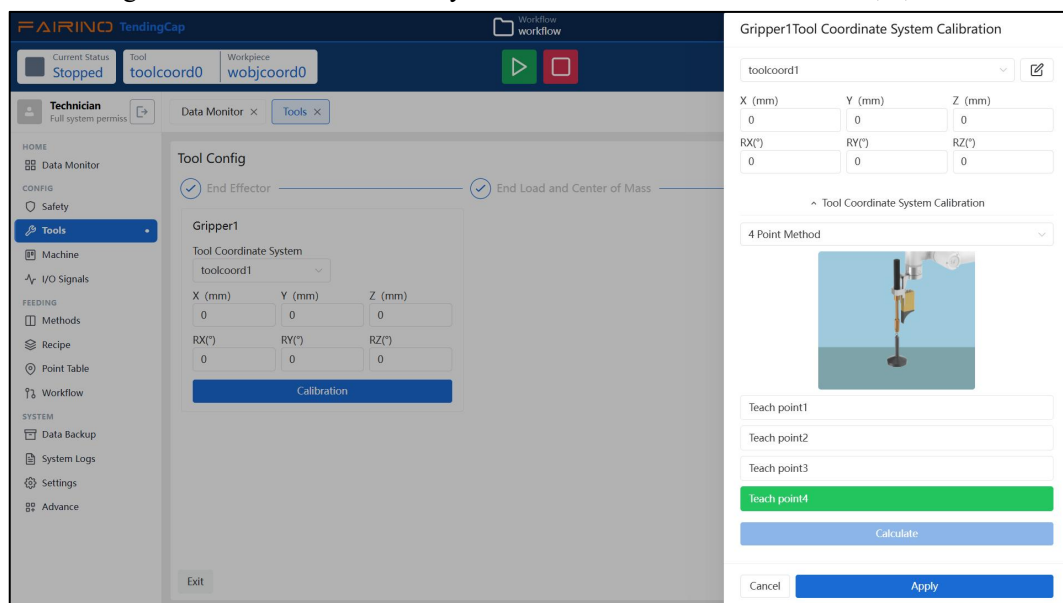


Figure 7.2-8 Tool coordinate system calibration using the 4-point method

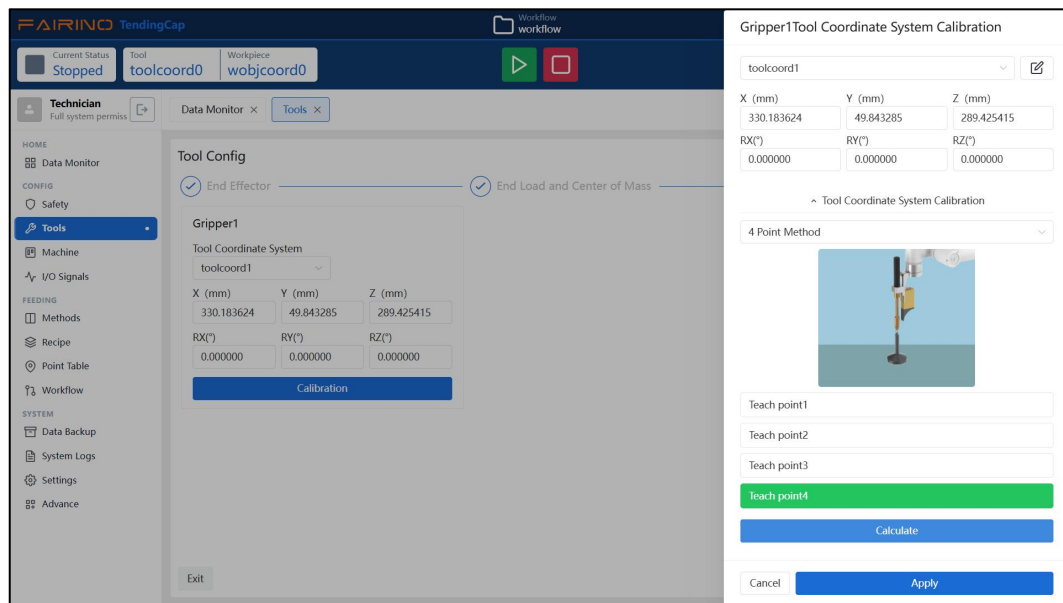


Figure 7.2-9 Tool coordinate system four-point method calibration calculation results

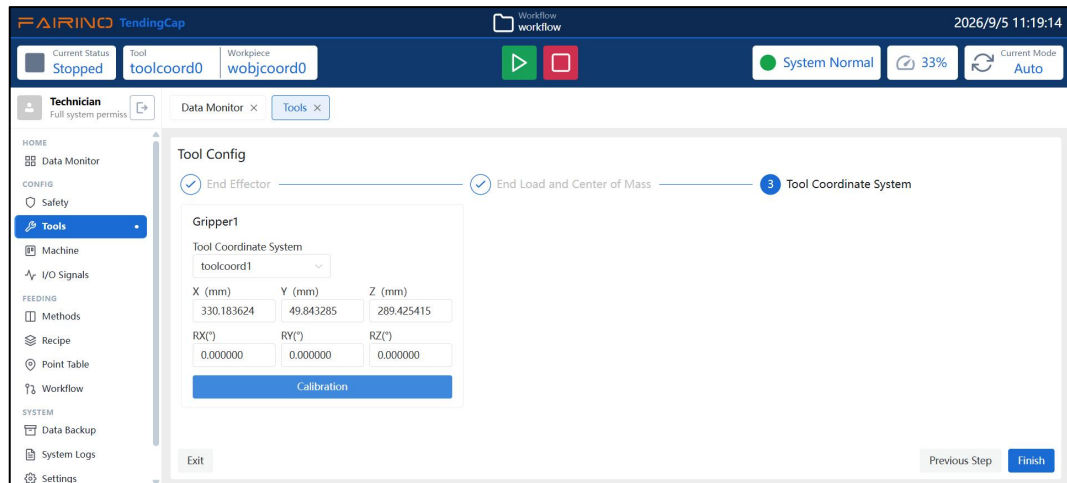


Figure 7.2-10 Application of the Four-Point Method Calibration Results for the Tool Coordinate System

7.3 Machine

In "Configuration - Machine Tool", configure the machine tool. After entering the page, retrieve the machine tool list data. When no data exists, display "No data available".

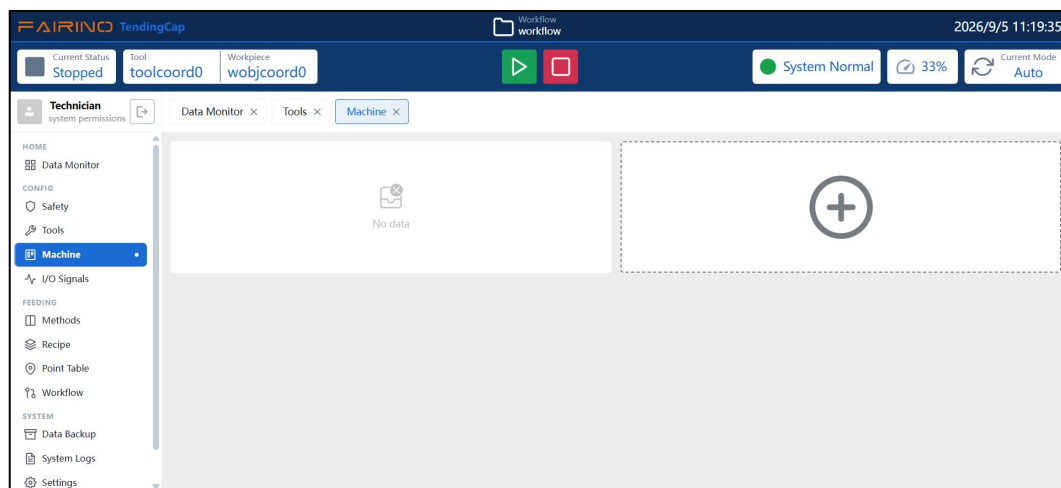


Figure 7.3-1 No machine tool data found.

Click the 'Add' card to create a new machine tool entry, in two steps, as described below:

- **Machine tool parameters:** The user inputs a name based on functional requirements. If the name is left blank and the "Next" button is clicked, a prompt "Name cannot be empty" will appear. Currently, the machine tool supports the following control systems: "FANUC", "SYNTEC", and "Others".

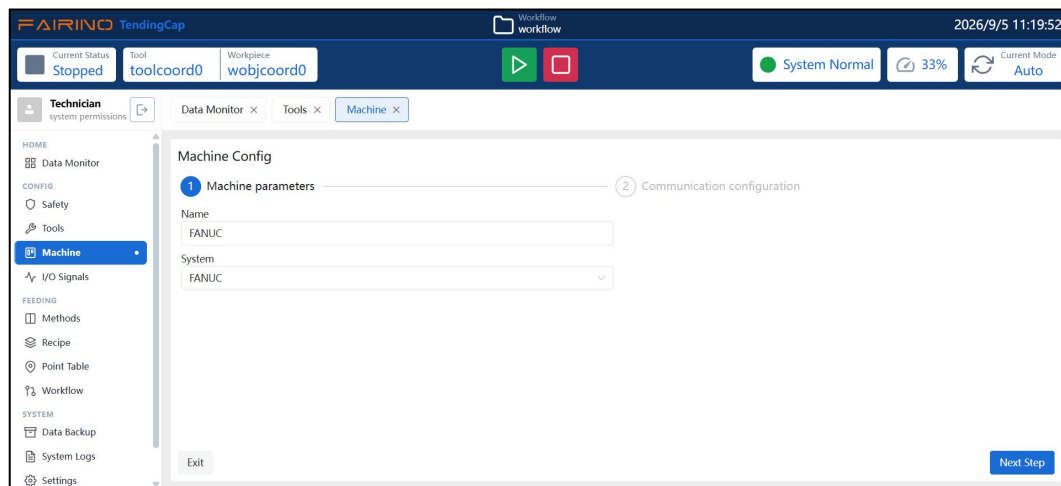


Figure 7.3-2 Machine parameters

- **Communication Configuration:** Depending on the selected system, the communication modes supported by the FANUC system are "I/O" and "FOCAS" (available only on Linux systems); the SYNTEC systems support "I/O" and the "SYNTEC Protocol"; other systems only support "I/O".

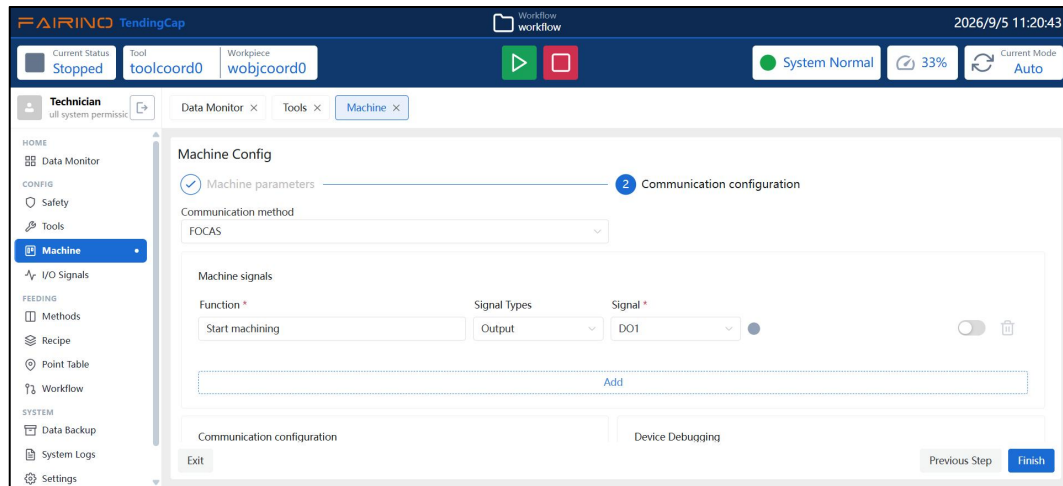


Figure 7.3-3 Communication Configuration – Machine Tool Signals

After successful addition, the homepage displays configured machine tools. Click the "Edit" button in the upper-right corner of each tool configuration card to view or modify details. Click the "Delete" button in the upper-right corner to confirm deletion; click OK to remove the tool.

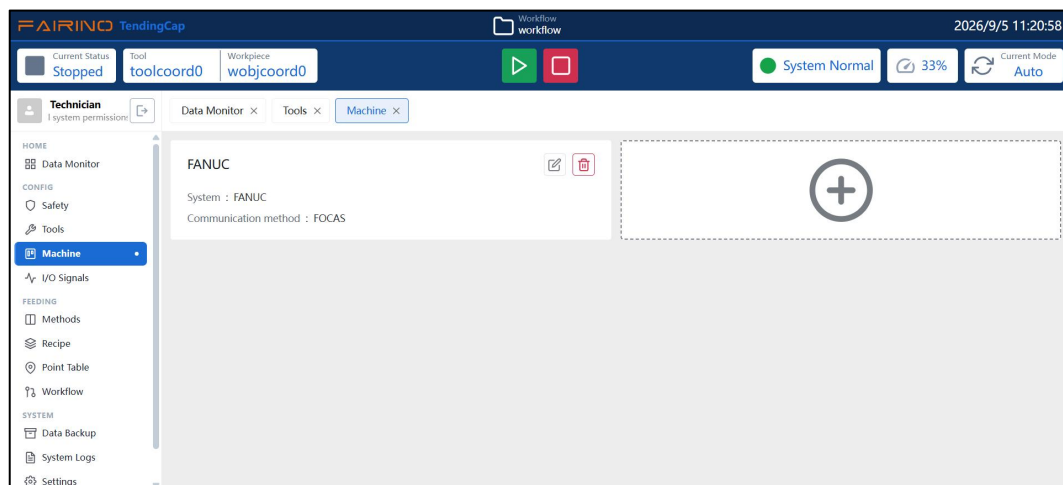


Figure 7.3-4 Machine Tool List

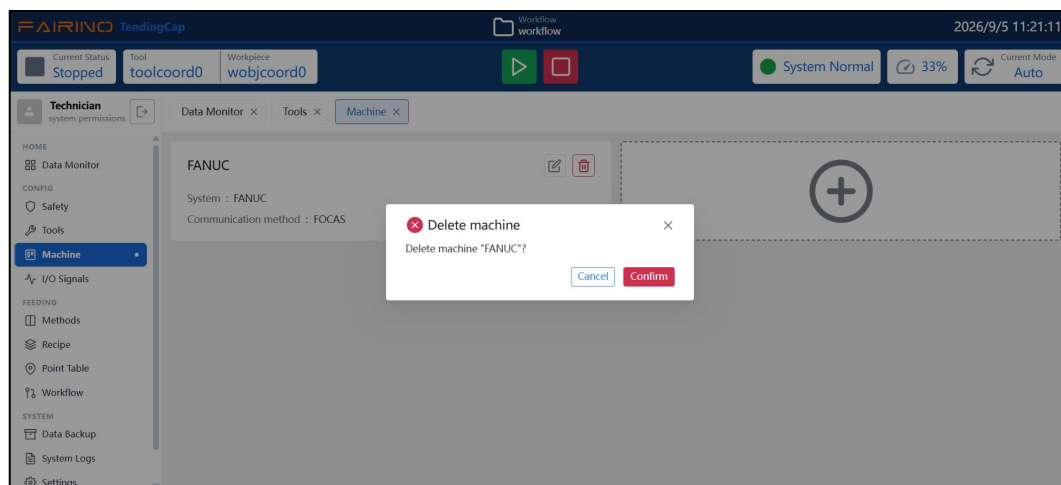


Figure 7.3-5 Machine Tool Deletion Second Confirmation

7.3.1 Communication method

7.3.1.1 I/O

Display only the machine tool signal details. Click the "Add" button at the bottom of each signal to add it, enter the function name, and select the signal type and identifier. For DO signals, use the slider for manual testing and status display; click the "Delete" icon on the right side to remove them. Machine tool signals must not be empty. After successful communication configuration, click the "Complete" button to finalize addition.

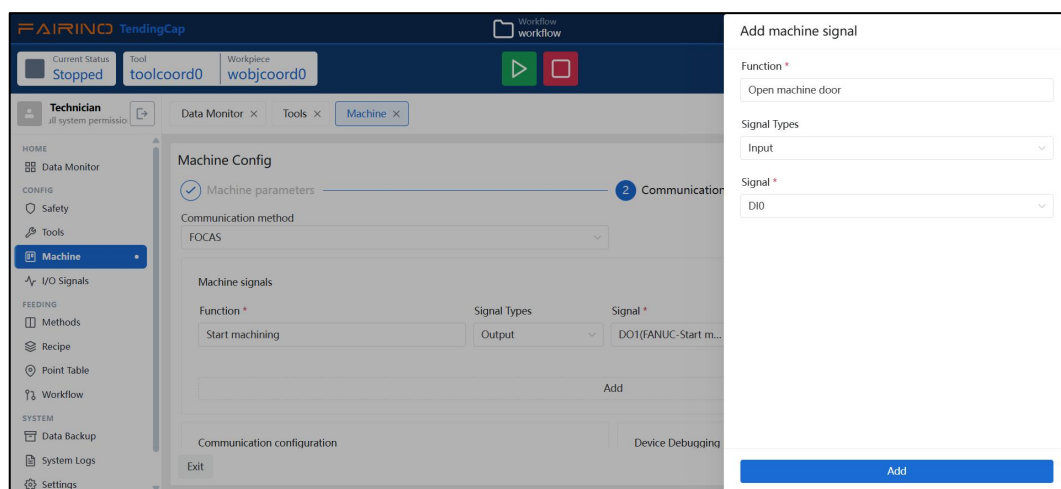


Figure 7.3-6 Communication method – Add Machine Tool Signals

7.3.1.2 FOCAS

Show the machine tool signals, communication configuration for the FOCAS

protocol, and procedures for machine tool debugging.

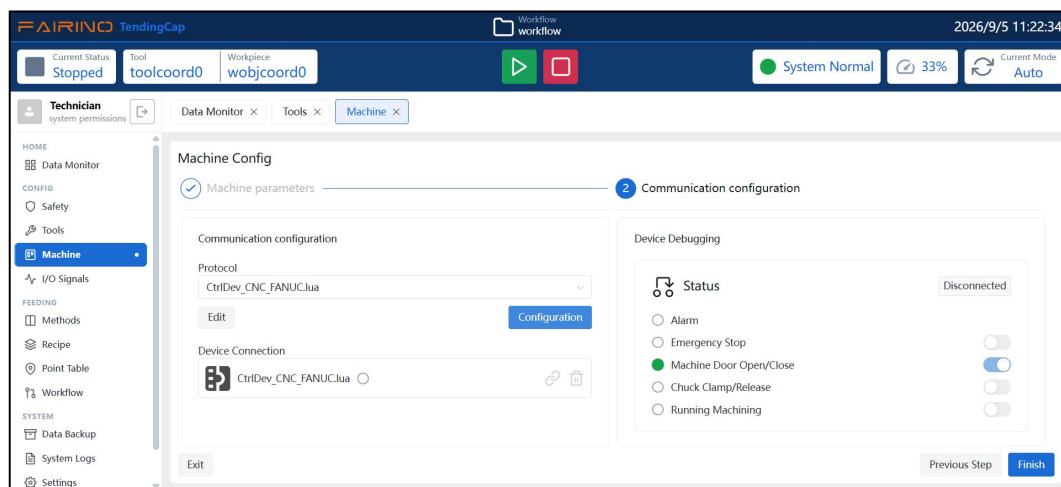


Figure 7.3-7 Communication method – FOCAS

In Communication Configuration, configure protocols, connect devices, and make edits. Details are as follows:

- **Protocol Configuration:** Click the "Configure" button to use the selected protocol file as the open protocol for establishing FOCAS communication with the machine tool.

- **Device Connection:** Displays configured protocol files. The green light indicates the name of the protocol currently connected. Click the "Load/Unload" icon on the far right to connect or disconnect a protocol. Click the "Delete" icon to remove the protocol configuration from display.

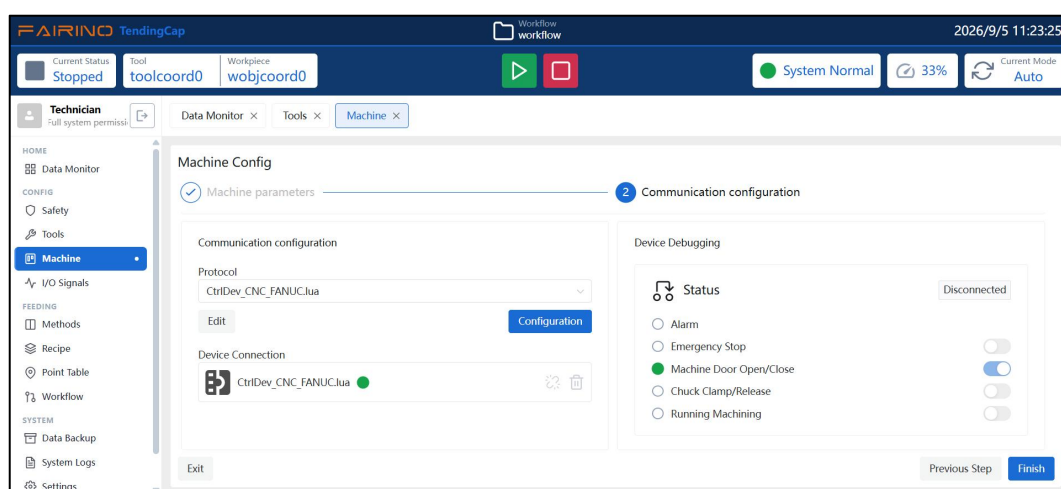


Figure 7.3-8 Communication method – FOCAS Protocol and Device Connection

- **Protocol Editing:** Click the "Edit" button to enter the Protocol Editing interface for uploading, deleting, or downloading protocols. The file name for uploading a

protocol must start with "CtrlDev_CNC" and have the ".lua" extension. Upon successful upload, the existing protocol list will update; select the desired protocol from the list to delete or download it.

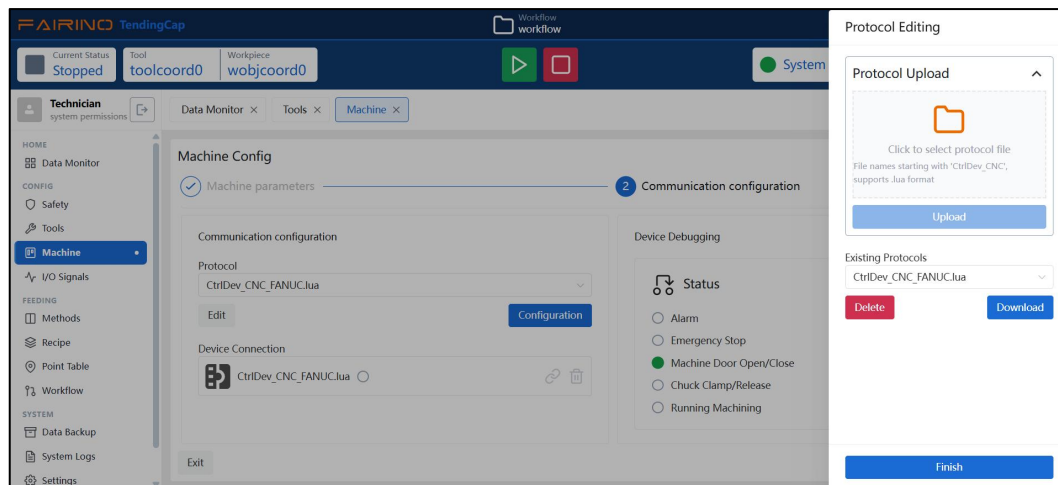


Figure 7.3-9 Communication method – FOCAS Protocol Editing

During equipment debugging, perform machine tool calibration operations and display real-time status feedback.

- Alarm: Gray circles indicate no alarm; green lights indicate an alarm.
- Emergency Stop: The gray circle indicates exiting emergency stop; the green light indicates entering emergency stop. Click the slider to adjust emergency stop settings.
- Machine tool opening/closing: A gray circle indicates the machine is closed, while a green light indicates it is open. Click the slider to adjust the machine's opening/closing settings.
- Chuck engagement/unengagement: The gray circle indicates the chuck is released, while the green light indicates it is engaged. Click the slider to adjust chuck engagement/unengagement.
- Processing Status: Gray circles indicate the machine tool has stopped processing; green lights indicate it is in operation. Click the slider to adjust the machine tool's processing settings.

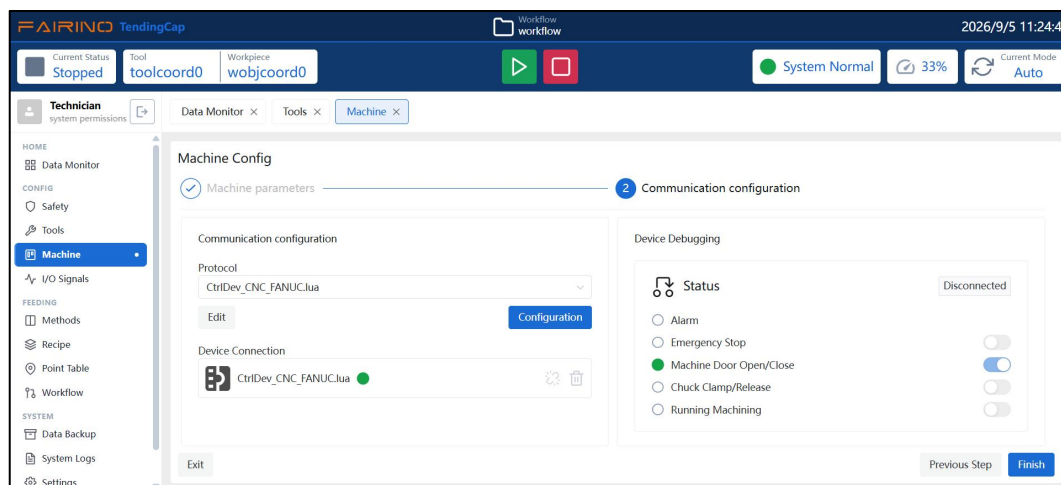


Figure 7.3-10 Communication method – FOCAS Protocol Device Debugging

7.3.1.3 SYNTEC Protocol

Displays the machine tool signals, communication configuration details for the new-generation protocol, and equipment debugging procedures. The communication configuration and debugging content are identical to those of FOCAS.

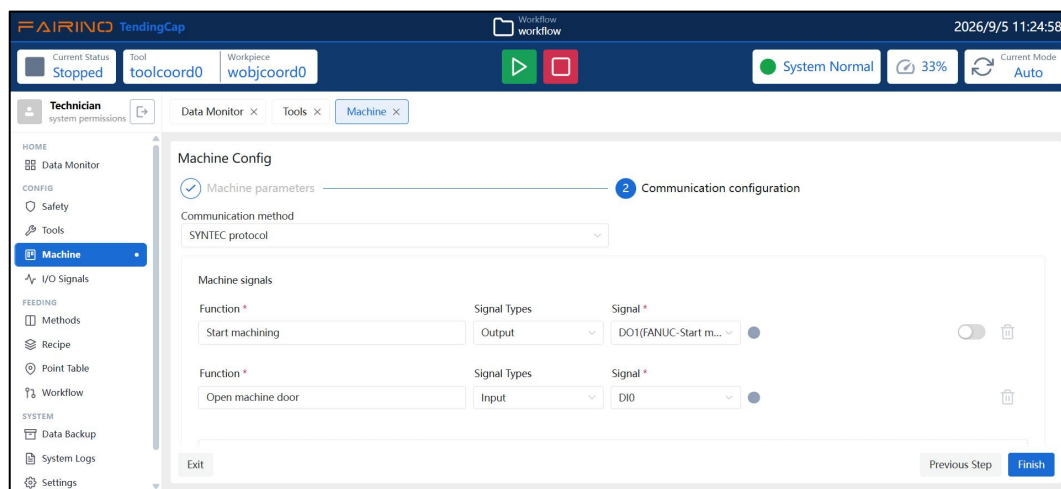


Figure 7.3-11 Communication method – SYNTEC Protocol

● Example Protocol:

- 1 -- CtrlDev_test_syntecCNC_0708.lua
- 2 -- SYNTEC system Modbus TCP communication control (based on the fairino Interaction Description.xls file, sequential addresses)
- 3 -- IP: 192.168.58.100 Port: 502 From Station Number: 100
- 4 --
- 5 -- Address Conversion: R Register → Modbus Holding Register
- 6 -- The 16-bit high value of $RN = N*2$; the 16-bit low value of $RN = N*2+1$
- 7 -- The signal value is written with the lower 16 bits and the upper 16 bits set to 0.


```

8
9  local id = 1
10 ModbusTCPMasterClose(id)
11 rtn = ModbusTCPMasterCreate('192.168.58.100', 502, 100, id)
12 CNCComSet(2, rtn, '192.168.58.100', 502) -- Parameter 1: Machine Tool Manufacturer -
      1. FANUC, 2. SYNTEC Parameter 2: Communication Status
13 SetCNCCConfig(0,1,0,0) -- Open Protocol Options Have Taken Effect
14
15 -- Output array: 11 R values from R1100 to R1110 = 22 U16 values
16 local ctrlValues = {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0}
17
18 -- Pulse Timer:    1. Clamp    2. Release    3. Position 1    4. Position 2    5. Feed
                    completion    6. Reset    7. Stop    8. Start
19 local pulse_timers = {0, 0, 0, 0, 0, 0, 0, 0}
20 local PULSE_TICKS = 10 -- Number of pulse holding periods (10 × 100 ms = 1 second)
21
22 -- The last original signal value used for rising edge detection
23 local prev_run, prev_emerg, prev_chuck, prev_reset, prev_feed_pos, prev_feed_done = 0, 0,
    0, 0, 0, 0
24
25 while(1) do
26
27     -----
28     -- 1. Obtain the robot control signals (9 in total)    --    GetCNCCtrlState Return:
    Run, Emergency Stop, Door, Chopper, Reset, Feed Positioning
29     --                                The spindle moves to enter the machine tool; feeding
    is complete.
30     -----
31     local run, emerg, door, chuck, reset, feed_pos, spindle_jog, robot_enter, feed_done =
    GetCNCCtrlState(2)
32
33     -----
34     -- 2. Pulse signal: Triggered by rising edge → Set to 1 and held → Automatically
    returns to 0
35     --    Level Signal: Direct Transmission
36     -----
37
38     -- R1100 Chuck engagement (Pulse: chuck 0→1 triggered)
39     if chuck == 1 and prev_chuck == 0 then
40         pulse_timers[1] = PULSE_TICKS
41     end
42     local clamp_close = (pulse_timers[1] > 0) and 1 or 0

```

```
43
44 -- R1101 Chuck released (Pulse: Chuck 1 → 0 triggered)
45 if chuck == 0 and prev_chuck == 1 then
46     pulse_timers[2] = PULSE_TICKS
47 end
48 local clamp_open = (pulse_timers[2] > 0) and 1 or 0
49
50 -- R1102 Feed Position 1 (Pulse: feed_pos → 1 triggered)
51 if feed_pos == 1 and prev_feed_pos ~= 1 then
52     pulse_timers[3] = PULSE_TICKS
53 end
54 local feed_pos1 = (pulse_timers[3] > 0) and 1 or 0
55
56 -- R1103 Feed Position 2 (Pulse: feed_pos → 2 triggered)
57 if feed_pos == 2 and prev_feed_pos ~= 2 then
58     pulse_timers[4] = PULSE_TICKS
59 end
60 local feed_pos2 = (pulse_timers[4] > 0) and 1 or 0
61
62 -- R1104 Feed completed (Pulse: feed.done 0→1 triggered)
63 if feed_done == 1 and prev_feed_done == 0 then
64     pulse_timers[5] = PULSE_TICKS
65 end
66 local feed_done_set = (pulse_timers[5] > 0) and 1 or 0
67
68 -- R1107 Machine Tool Reset (Pulse: Reset from 0 to 1 triggered)
69 if reset == 1 and prev_reset == 0 then
70     pulse_timers[6] = PULSE_TICKS
71 end
72 local cnc_reset = (pulse_timers[6] > 0) and 1 or 0
73
74 -- R1109 Process Stop (Pulse: triggered by run 1→0)
75 if run == 0 and prev_run == 1 then
76     pulse_timers[7] = PULSE_TICKS
77 end
78 local stop_proc = (pulse_timers[7] > 0) and 1 or 0
79
80 -- R1110 Processing initiation (Pulse: triggered by run 0→1)
81 if run == 1 and prev_run == 0 then
82     pulse_timers[8] = PULSE_TICKS
83 end
84 local start_proc = (pulse_timers[8] > 0) and 1 or 0
```

```

85
86    -- Level signal (direct transmission)
87    -- R1105: Main spindle movement; R1106: Entry into the machine tool; R1108:
    Emergency stop of the machine tool.
88
89    -----
90    -- 3. Write to Modbus TCP (Function Code 0x10, Base Address 2200,22 U16 bits)
91    --    R1100~R1110 = Modbus addresses 2200~2221
92    --    When an exception occurs, Modbus TCP Master Set Hold Registers returns the
    error string.
93    -----
94    ctrlValues[1] = 0; ctrlValues[2] = clamp_close -- R1100
95    ctrlValues[3] = 0; ctrlValues[4] = clamp_open  -- R1101
96    ctrlValues[5] = 0; ctrlValues[6] = feed_pos1   -- R1102
97    ctrlValues[7] = 0; ctrlValues[8] = feed_pos2   -- R1103
98    ctrlValues[9] = 0; ctrlValues[10] = feed_done_set -- R1104
99    ctrlValues[11] = 0; ctrlValues[12] = spindle_jog -- R1105 electrical level
100 ctrlValues[13] = 0; ctrlValues[14] = robot_enter -- R1106 electrical level
101 ctrlValues[15] = 0; ctrlValues[16] = cnc_reset   -- R1107
102 ctrlValues[17] = 0; ctrlValues[18] = (emerg == 1) and 0 or 1 -- R1108 Emergency stop
    (anti-logic: No emergency stop → 0; Emergency stop → 1)
103 ctrlValues[19] = 0; ctrlValues[20] = stop_proc   -- R1109
104 ctrlValues[21] = 0; ctrlValues[22] = start_proc  -- R1110
105
106 local wret = ModbusTCPMasterSetHoldRegs(id, 2200, 22, ctrlValues, "U16")
107 local comm_err = 0
108 if type(wret) == "string" then
109     comm_err = 1
110 end
111
112 -----
113 -- 4. Read the machine tool status (CNC → Robot)
114 --    Base address 6220,18 U16 fields (each R value occupies 2 bytes: high byte + low
    byte)
115 --    The signal value is at the low bit (odd address).
116 --    The first return value in an exception is the error string.
117 --
118 --    Modbus Address with Subscript R Register Signal
119 --    6220 High/6221 Low    v1/v2    R3110    Clamping Feedback
120 --    6222 High/6223 Low    v3/v4    R3111    Clamp release feedback
121 --    6224 High/6225 Low    v5/v6    R3112    Main spindle positioning
    completed

```

```

122  -- 6226 High/6227 Low    v7/v8    R3113    Main spindle positioning
      completed
123  -- 6228 High/6229 Low    v9/v10    R3114    Request Shipping Feedback
      (1 = Allow Shipping)
124  -- 6230 High/6231 Low    v11/v12    R3115    Machine Tool Alarm
      Feedback
125  -- 6232 High/6233 Low    v13/v14    R3116    Safety Screen Feedback
126  -- 6234 High/6235 Low    v15/v16    R3117    Door opening completion
      feedback
127  -- 6236 High/6237 Low    v17/v18    R3118    Operation Status: 1 =
      Running; 2/0 =Stopped; 3 = Emergency Stop
128  -----
129  local v1, v2, v3, v4, v5, v6, v7, v8,
130      v9, v10, v11, v12, v13, v14, v15, v16,
131      v17, v18 = ModbusTCPMasterGetHoldRegs(id, 6220, 18, "U16")
132
133  if type(v1) == "string" then
134      -- Communication error: all statuses set to 0
135      v1, v2, v3, v4, v5, v6, v7, v8,
136      v9, v10, v11, v12, v13, v14, v15, v16,
137      v17, v18 = 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
138      comm_err = 1
139  end
140
141  local clamp_fb_close = v2  -- R3110 Low: Card clamping feedback
142  local clamp_fb_open  = v4  -- R3111 Low: Card release feedback
143  local spindle_pos1   = v6  -- R3112 Low: Spindle positioning 1 completed
144  local spindle_pos2   = v8  -- R3113 Low: Spindle positioning 2 completed
145  local feed_request    = v10 -- R3114 Low: Request shipping feedback (1 = Allow
      shipping)
146  local cnc_alarm      = v12  -- R3115 Low: Machine tool alarm feedback
147  local safety_curtain = v14  -- R3116 Low: Safety Light Screen Feedback
148  local door_open_fb   = v16  -- R3117 Low: Door opening feedback
149  local run_state      = v18  -- R3118 Low: Operation Status    1 = Running    2/0
      =Stopped    3 = Emergency Stop
150
151  -- Spindle Angle Status: Bit 0 = Positioning 1 completed; Bit 1 = Positioning 2
      completed
152  local spindle_angle = spindle_pos1 + spindle_pos2 * 2
153
154  -- Clamping status: 1 = Clamped, 0 = Released (A clamp feedback value of 1 indicates
      the clamp is engaged)

```

```

155     local chuck_state = clamp_fb_close
156
157     -----
158     -- 5. Report to Robot
159     --     SetCNCStatus(Communication status, Operating status, Door status, Chucks
160         status, --Spindle angle status, Machine tool alarm, Safety light screen, --Feed allowed,
161         Reserve 1, Reserve 2, Reserve 3, Reserve 4)
162     -----
163     SetCNCStatus(comm_err,      -- 0 = Normal communication; 1 = Fault
164                   run_state,    -- 1 = Run 2 = Stop 3 = Emergency Stop
165                   door_open_fb, -- The door is open properly.
166                   chuck_state,  -- 1 = Clamped | 0 = Released
167                   spindle_angle, -- bit0 = Position 1    bit1 = Position 2
168                   cnc_alarm,    -- Machine Tool Alarm
169                   safety_curtain, -- Safety Screen
170                   feed_request, -- Allow shipping
171                   0,            -- obligate 1
172                   0,            -- obligate 2
173                   0,            -- obligate 3
174                   0)           -- obligate 4
175
176     -----
177     -- 6. The pulse timer decrements and saves the previous signal value (for rising edge
178         detection in the next cycle).
179     -----
180     for i = 1, 8 do
181         if pulse_timers[i] > 0 then
182             pulse_timers[i] = pulse_timers[i] - 1
183         end
184     end
185
186     prev_run      = run
187     prev_emerg    = emerg
188     prev_chuck    = chuck
189     prev_reset    = reset
190     prev_feed_pos = feed_pos
191     prev_feed_done = feed_done
192
193     -----
194     -- 7. Connect Check and Exit
195     -----
196     CNCCheckDisConnect()

```

```

194     local stopFlag = GetOpenLUAStopFlag(id)
195     if(stopFlag ~= 0) then
196         ModbusTCPMasterClose(id)
197         SetCNCStatus(0,      -- 0 = Normal communication; 1 = Fault
198                     0,      -- 1 = Run 2 = Stop 3 = Emergency Stop
199                     0,      -- The door is open properly.
200                     0,      -- 1 = Clamped | 0 = Released
201                     0,      -- bit0 = Position 1    bit1 = Position 2
202                     0,      -- Machine Tool Alarm
203                     0,      -- Safety Screen
204                     0,      -- Allow shipping
205                     0,      -- obligate 1
206                     0,      -- obligate 2
207                     0,      -- obligate 3
208                     0)      -- obligate 4
209
210     break
211 end
212
213     sleep_ms(100)
214end

```

- Analysis of the new generation of CNC open protocols:

The new-generation CNC Open Protocol primarily consists of three components.

① **Establish a communication connection.** Specify the protocol ID (the number set when loading an open protocol must match that in the protocol file), CNC IP address, port number, and other parameters, then use the "Modbus TCP Master Create()" command to establish a Modbus TCP connection between the robot and the new-generation CNC system.

② **Write control data to the CNC in a cyclic manner.** When executing the CNC Open Protocol, the system first reads the current CNC control data from the robot controller's internal memory, then packages and writes this data to command the CNC-controlled machine tool. The return value of the GetCNCCtrlState() instruction for reading CNC control data is defined in Table 4.3.1-1; control data can be parsed based on CNC register specifications, and subsequently written in batches to registers R1100 – R1110 via Modbus TCP.

Table 7.3.1-1 GetCNCCtrlState () returned value

Order	Type	Name	Description
1	uint16_t	run	Operation control: 0 – Stop processing; 1 – Start processing (pulse signal, maintained for 1 second)
2	uint16_t	emerg	Emergency stop control: 1 – No emergency stop (safe); 0 – Trigger emergency stop
3	uint16_t	door	Door control: 0 – Close door; 1 – Open door
4	uint16_t	chuck	Chuck control: 1 – Clamping (triggers R1100 pulse); 0 – Releasing (triggers R1101 pulse)
5	uint16_t	reset	Machine tool reset: 0 – No reset; 1 – Reset (pulse signal, maintained for 1 second)
6	uint16_t	feed_pos	Transportation location selection: 1 – Location 1 (pulse R1102); 2 – Location 2 (pulse R1103)
7	uint16_t	spindle_jog	Main axis incremental movement: 0 – Stop; 1 – Incremental rotation (level signal)
8	uint16_t	robot_enter	Enter the machine tool: 0 – Not entered; 1 – Robot has entered the machine tool (level signal)
9	uint16_t	feed_done	Delivery completed; 0 – Not completed; 1 – Delivery completed (pulse signal, maintained for 1 second)

③**Read status data from the CNC in a loop.** The CNC Open Protocol first reads real-time status data (R3110 – R3118) from the CNC via Modbus TCP, then writes the relevant data to the robot controller, enabling the robot to monitor the CNC's real-time status. The protocol defines the CNC status interface parameter SetCNCStatus() as

shown in the table below.

Table 7.3.1-2 SetCNCStatus () Detailed Parameters

Order	Type	Name	Description
1	uint16_t	comm_err	Communication status: 0 - Normal communication; 1 - Communication failure
2	uint16_t	run_state	Operation status: 1 - Running; 2/0 - Stopped; 3 - Emergency stop
3	uint16_t	door_open_fb	Door opening status feedback: 1-Door fully open; 0-Door not fully open
4	uint16_t	chuck_state	Chuck status: 1-Clamped; 0-Released
5	uint16_t	spindle_angle	Spindle angle status (reserved): bit0 - Position 1 completed; bit1 - Position 2 completed
6	uint16_t	cnc_alarm	Machine tool alarm; 1 - Machine tool fault; 0 - No fault
7	uint16_t	safety_curtain	Safety light curtain (reserved); 1 - Light curtain activated; 0 - Light curtain normal
8	uint16_t	feed_request	Allow shipping: 1-Allow; 0-Do not allow
9	uint16_t	reserved1	obligate 1
10	uint16_t	reserved2	obligate 2
11	uint16_t	reserved3	obligate 3
12	uint16_t	reserved4	obligate 4

7.4 I/O Signals

Configure user-defined I/O functions and status indicators in Configuration>

Functional Signals. Upon entering the page, retrieve the list of functional signals. If no data is available, display "No Data".

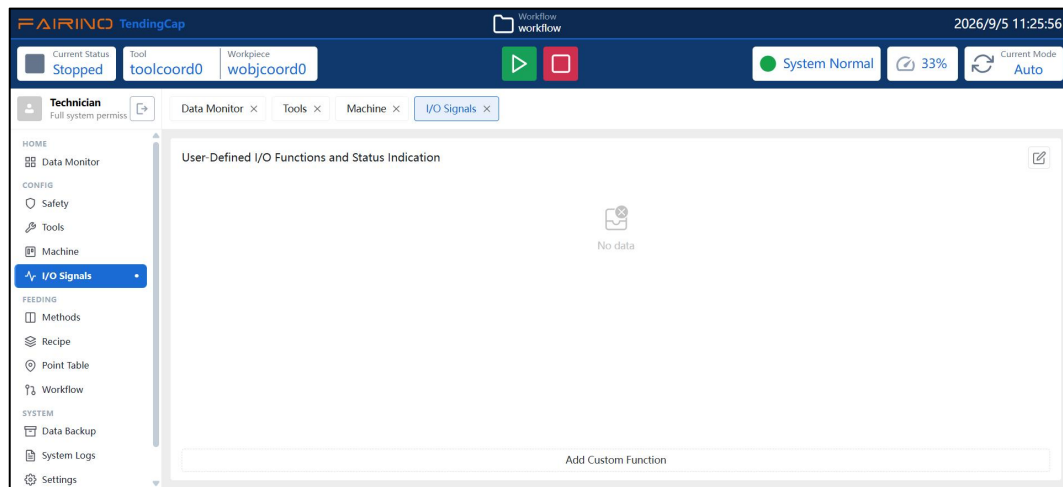


Figure 7.4-1 No data available for function signals

Click the Add Custom Function button at the bottom. A drawer will appear on the right. Enter a name, select the signal type and signal, then click Add at the bottom to complete configuration. Name and signal are required; otherwise, an error message will appear. Once added, the data in the list cannot be modified by default.

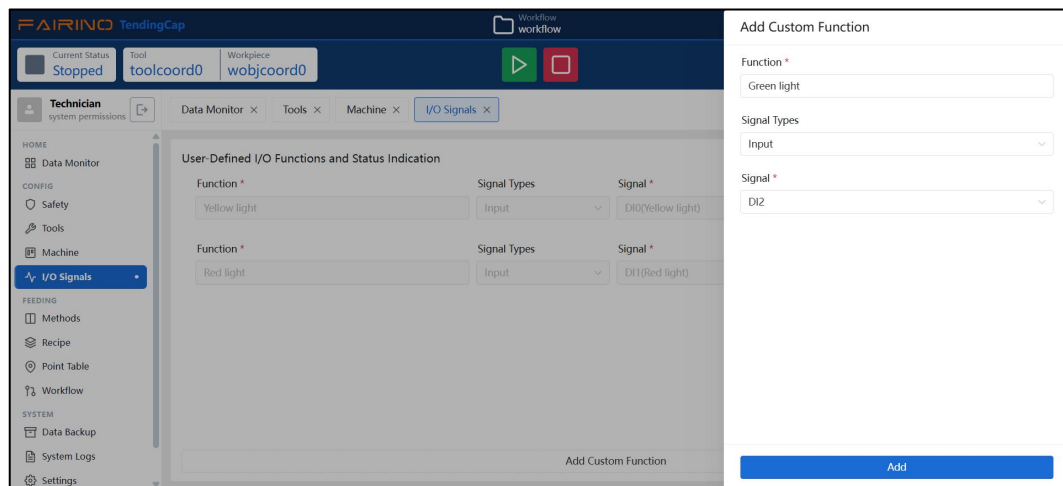


Figure 7.4-2 Add Function Signal

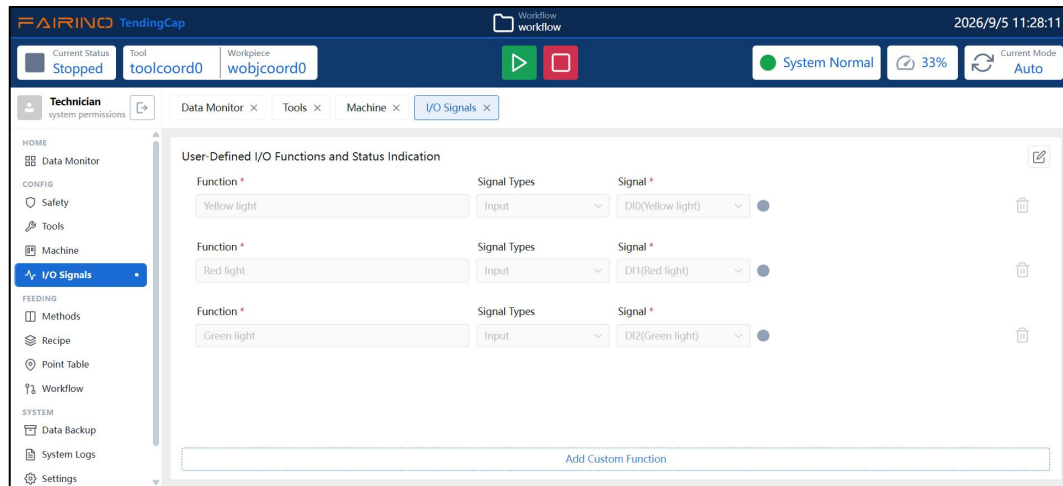


Figure 7.4-3 Function Signal List

Click the "Edit" icon in the upper-right corner to enter editing mode, where you can modify functional signal data. You may also use the slider to test signal reception on the interface; when connected and activated, a green light will illuminate after configuration. At the end of each data entry, click the "Delete" button to display a secondary confirmation prompt for "Delete Functional Signal"; confirm to delete.

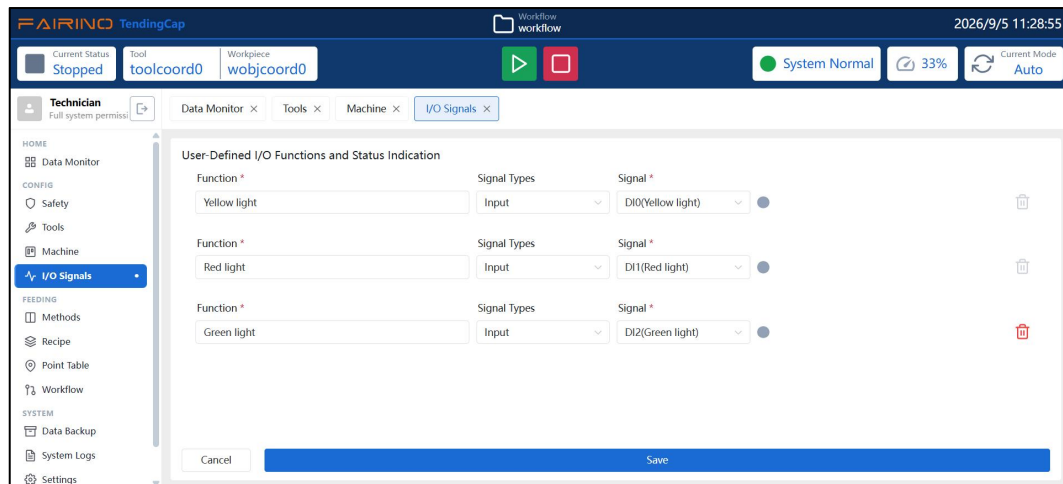


Figure 7.4-4 Function Signature List Editing Status

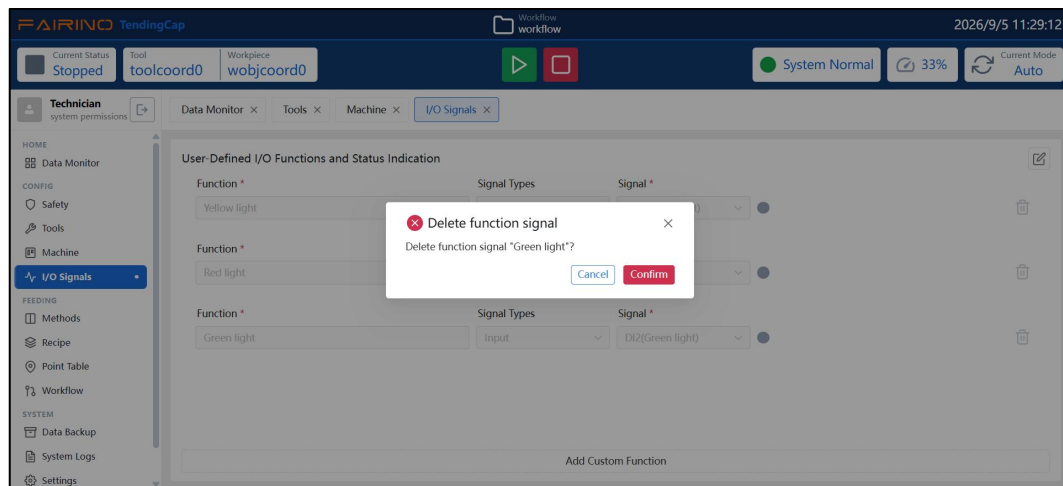


Figure 7.4-5 Delete functional signal data

8 Feeding

8.1 Methods

There are two material loading methods: tray and conveyor belt. After entering the interface, retrieve data for both methods; if no data is available, display "No parameters". Click the add cards at the bottom of each method to enter data.

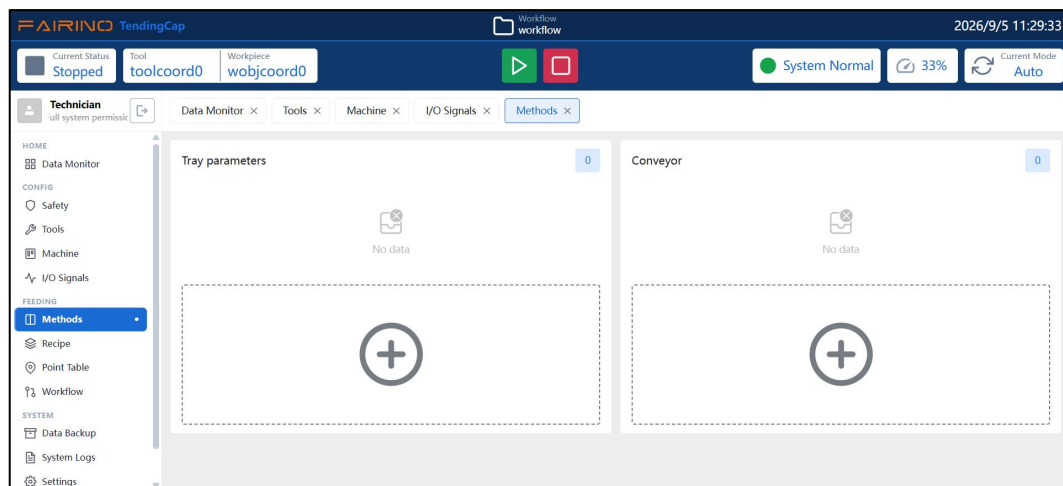


Figure 8.1-1 No data available for this method

8.1.1 Tray

The material tray configuration consists of three steps, as described below.

- Material tray parameters: Enter a name according to the functional

requirements; if the name is empty, clicking the "Next" button will display the message "Name cannot be empty." Select a tool; when you click the "Next" button, the corresponding tool coordinate system will be applied. This ensures that when setting the pallet's positioning points, the teaching points will align with the current tool coordinate system. Choose whether to use an extended axis; if you select "Yes," proceed to the Advanced Parameters section to configure the extended axis coordinate system, configure the extended axis UDP communication, and perform extended axis configuration and testing. When the pallet selected in the workflow uses an extended axis, the matrix movement command will include synchronized motion planning for the extended axis.

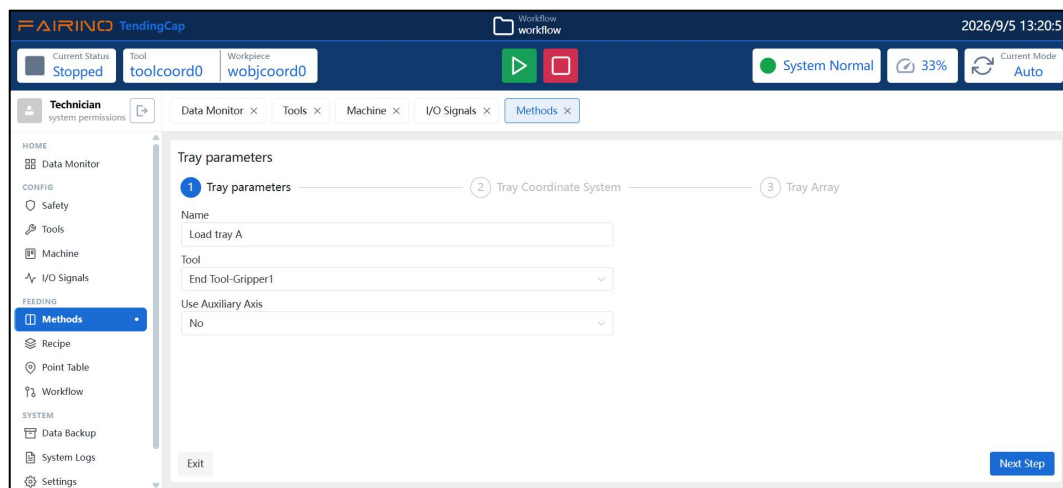


Figure 8.1-2 Material Plate Parameters

- Workpiece coordinate system: Select the workpiece coordinate system and rename or automatically calibrate it.

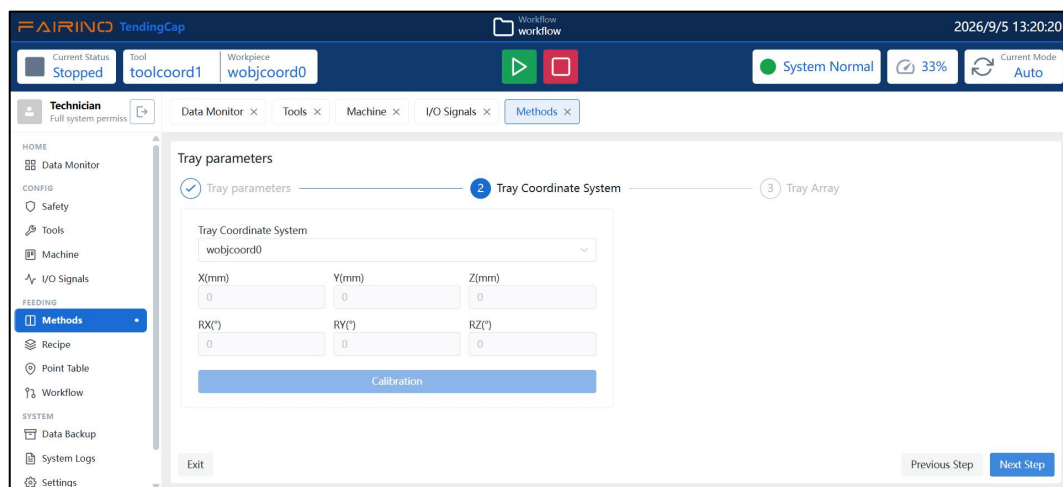


Figure 8.1-3 Material Plate Coordinate System

● **Material Disk Array:** Configure the number of rows, columns, layers, layer height (unit: mm), transition point offsets (X, Y, Z), and motion path settings — including point teaching using the three-point method. Click the "Complete" button to finish adding the array.

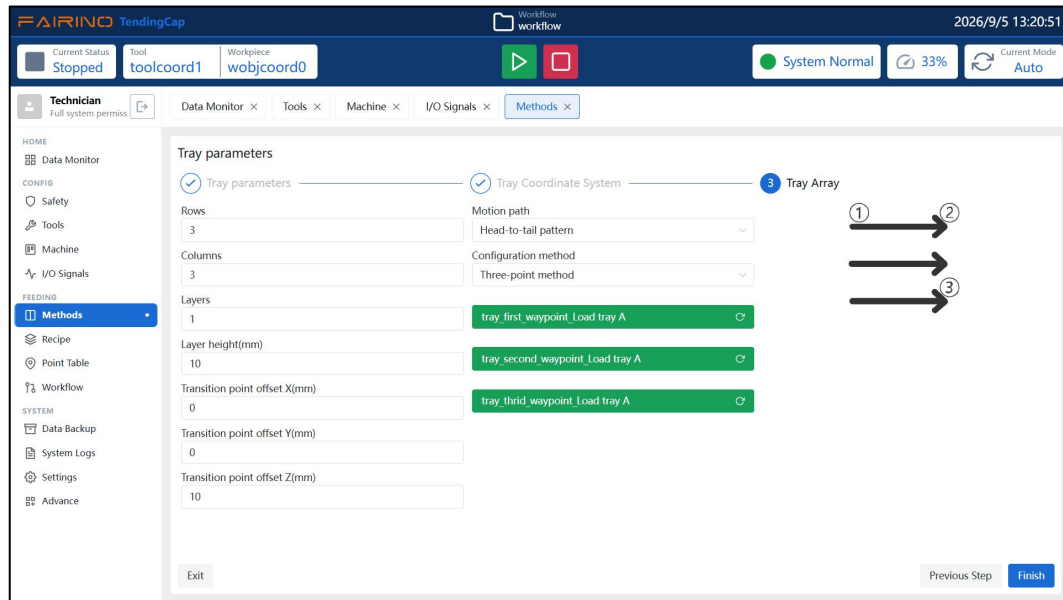


Figure 8.1-4 Material Disk Array

The movement paths are categorized into the head-to-tail walking method and the bow-shaped walking method, as detailed below.

Step-by-step progression: From left to right, complete the first row and return to the left starting point; then repeat for the second row; continue this pattern for the third row, until full coverage is achieved.

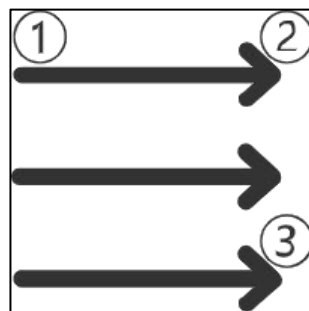


Figure 8.1-5 From start to finish, the walking pattern remains consistent.

Bow-shaped progression: Complete the first row from left to right, then move vertically downward. Next, complete the second row from right to left. Then move vertically downward again, and complete the third row from left to right until the area is fully covered.

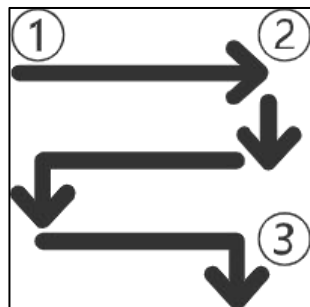


Figure 8.1-6 Arch-shaped movement pattern

After successful addition, the homepage displays configured material trays. Click the "Edit" button in the upper-right corner of each tray configuration card to view or modify details. Click the "Delete" button in the upper-right corner to confirm deletion; click OK to remove the tray.

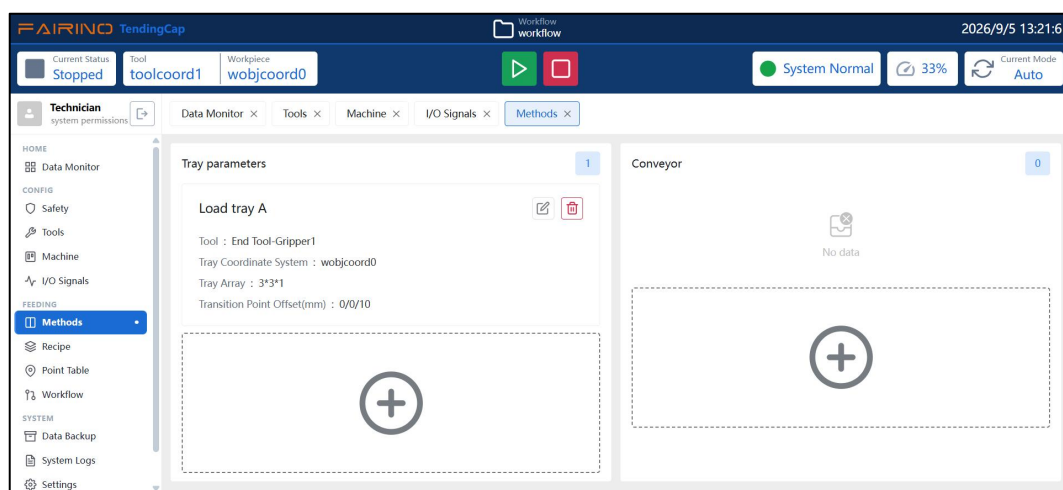


Figure 8.1-7 Tray List

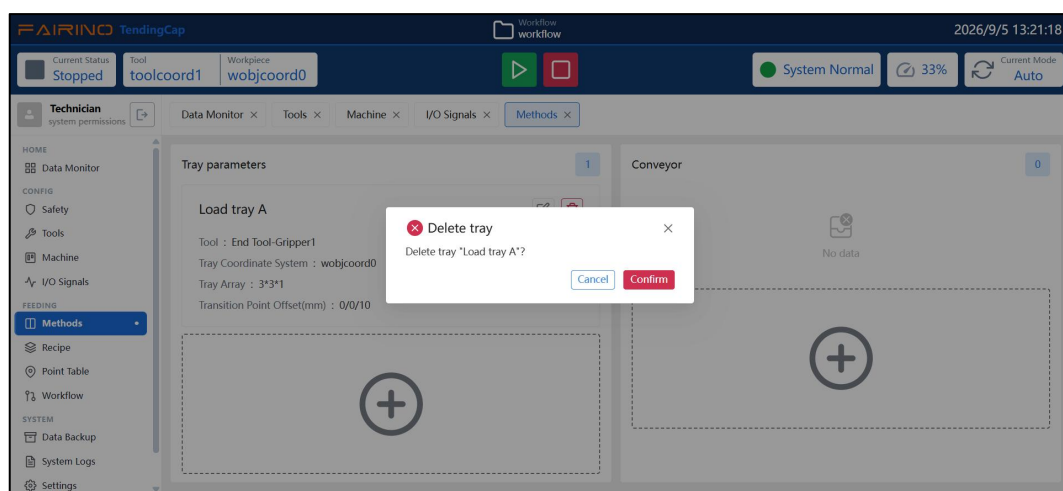


Figure 8.1-8 Secondary confirmation required to delete the material list

After modifying tools and workpieces in the editing interface, you must re-teach

the matrix material inventory positions; otherwise, the changes cannot be completed. When clicking the Complete button, the system will verify whether the tool and workpiece coordinate systems match the matrix points. If they do not match, a prompt will appear: "The parameter point coordinate system has changed. Confirm consistency before completing configuration."

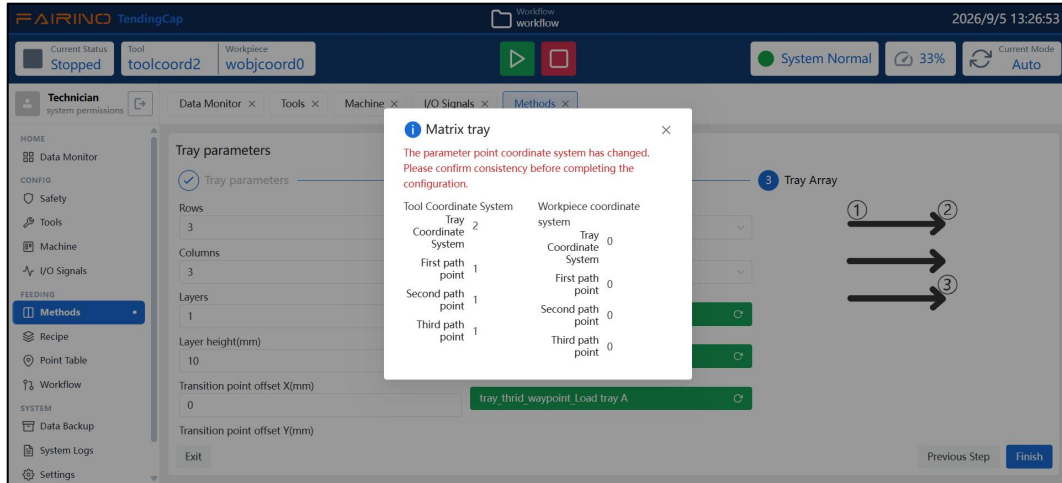


Figure 8.1-9 Warning: The coordinate system for the inventory location has changed

8.1.1.1 Workpiece Coordinate System Calibration

Select a non-default workpiece coordinate system, fix the workpiece position, and choose either the calibration method "Origin-X-Axis-Z-Axis" or "Origin-X-Axis-XY+Plane". The first two points are identical for both methods; only the third point differs: the first method calibrates the Z-axis of the workpiece coordinate system, while the second method calibrates a point on the XY+ plane. Perform calibration as shown in the diagram. After completing the point teaching, click the Calculate button to compute the workpiece pose; upon obtaining the result, click Apply to finalize the calibration.

Important:

- The workpiece coordinate system is calibrated based on the tool coordinate system and must be established upon the existing tool coordinate system.
- The default workpiece coordinate system cannot be calibrated. When selecting a non-default workpiece coordinate system, click the Calibrate button to automatically apply the default system.

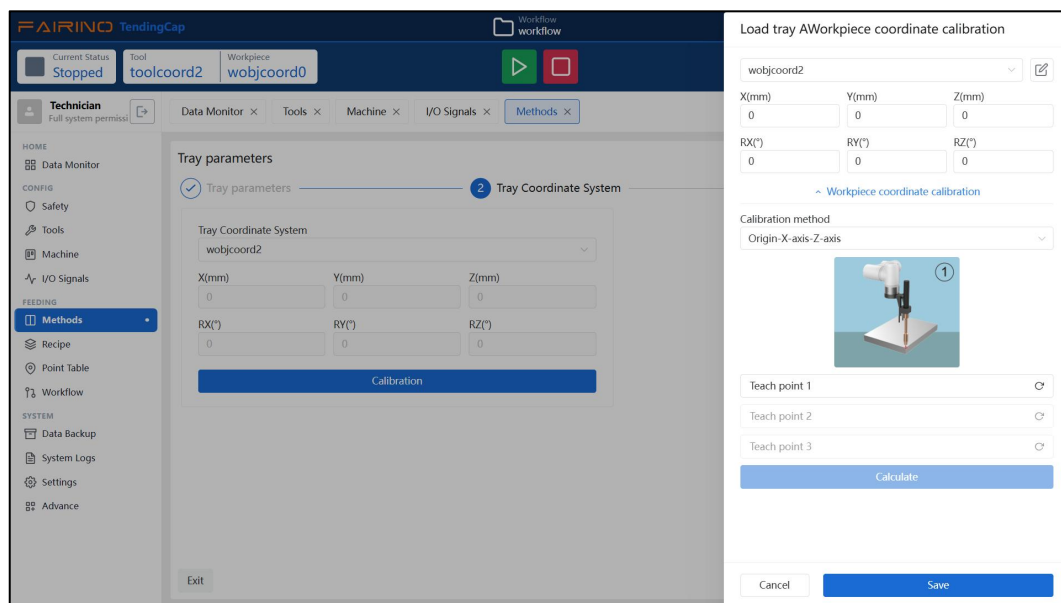


Figure 8.1-10 Material tray coordinate system reference point 1

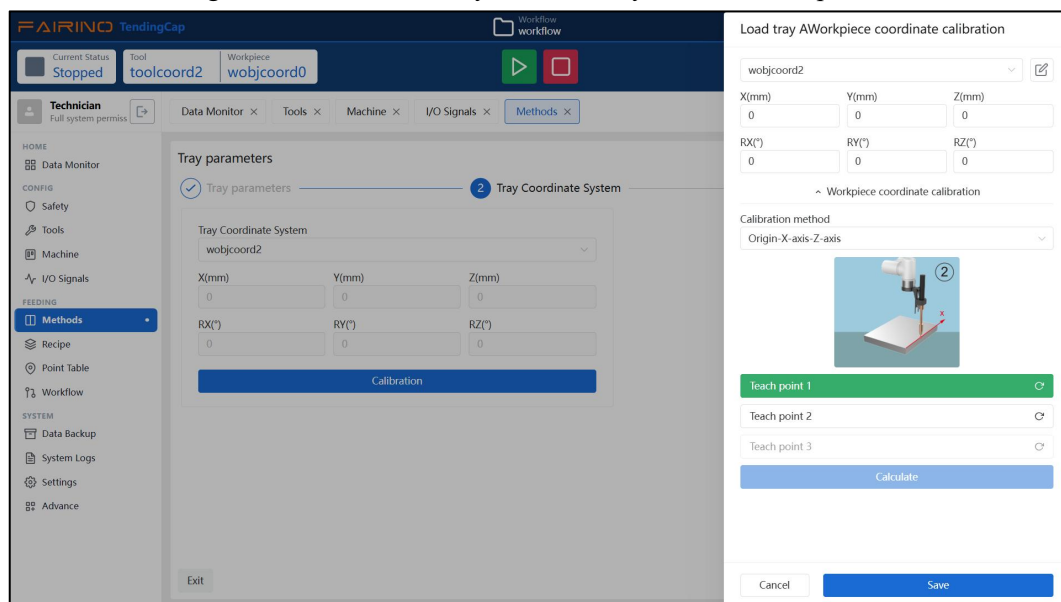


Figure 8.1-11 Material Plate Coordinate System Reference Point 2

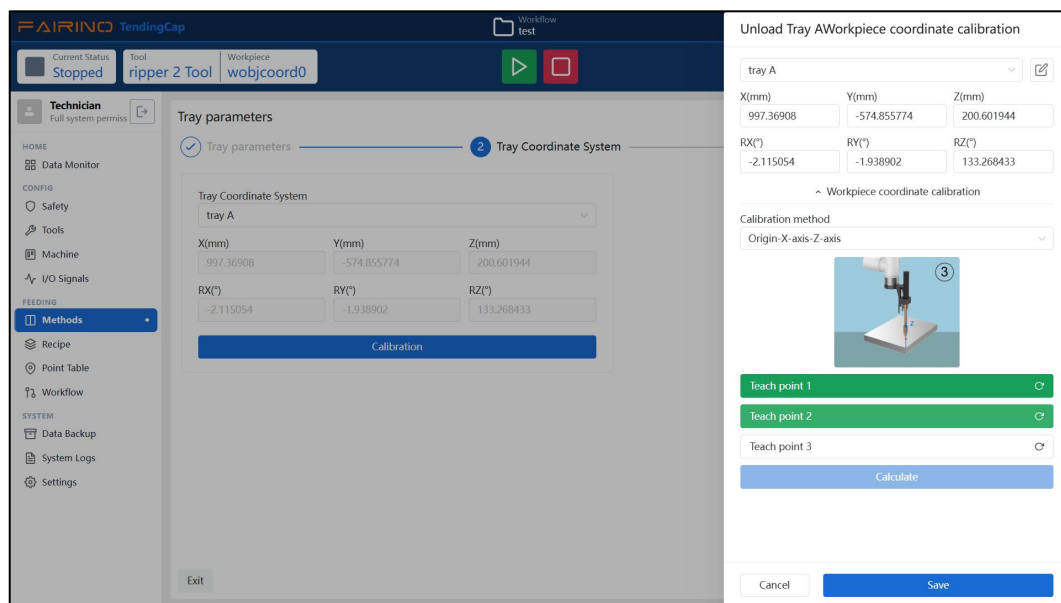


Figure 8.1-12 Material tray coordinate system calibration: Origin - X-axis - Z-axis point 3

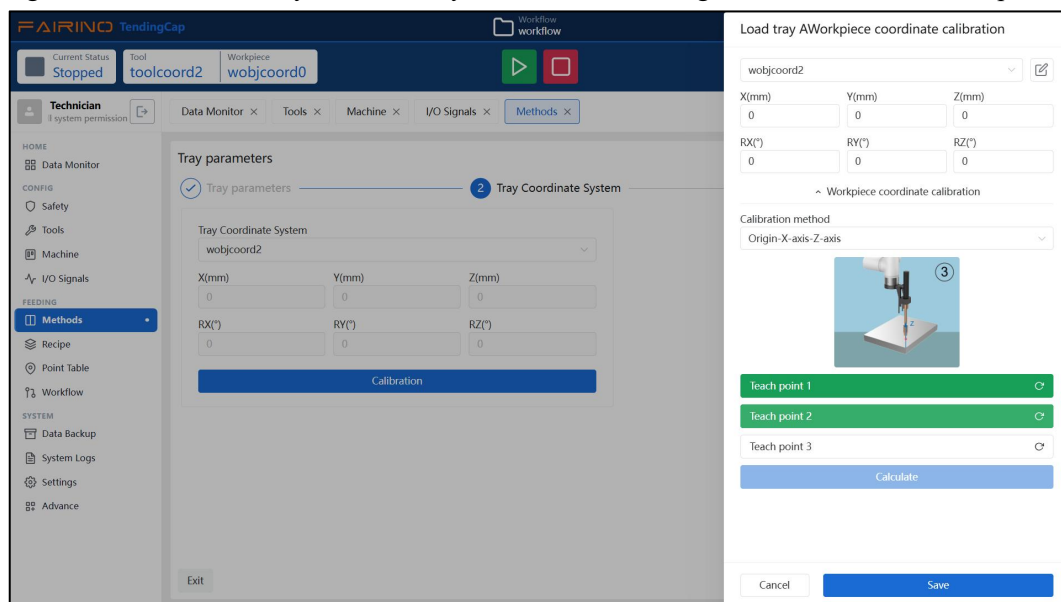


Figure 8.1-13 Material tray coordinate system calibration: Origin - X-axis - XY+ plane point

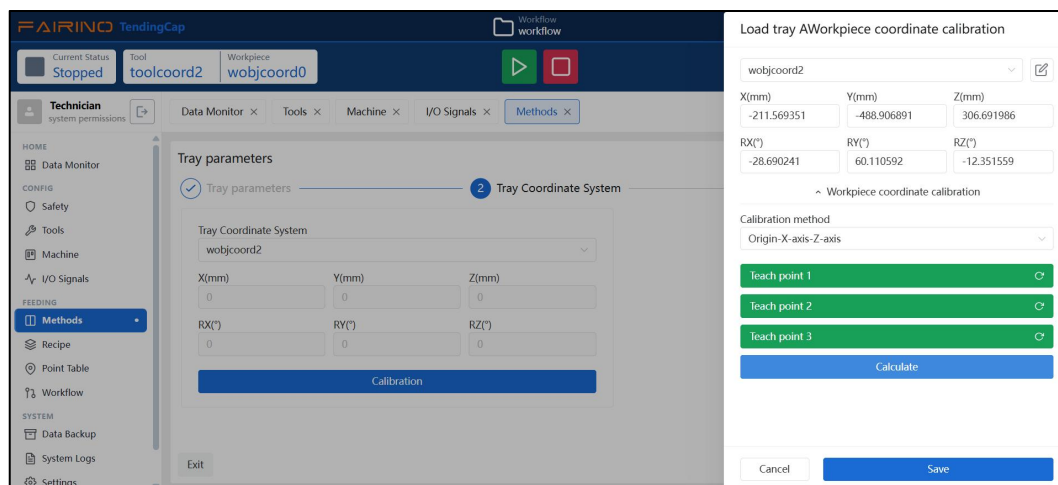


Figure 8.1-14 Results of the material tray coordinate system calibration

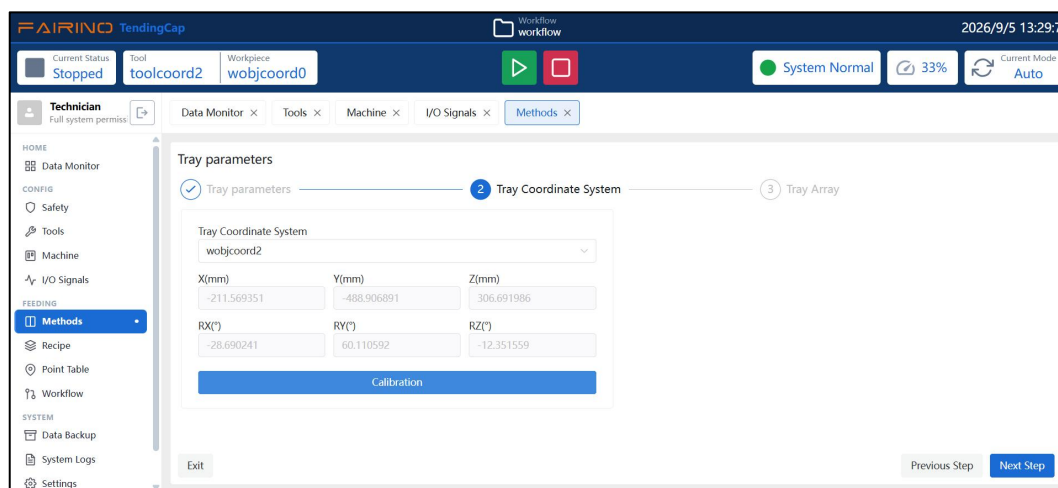


Figure 8.1-15 Material tray coordinate system calibration results saved

8.1.2 Conveyor

The conveyor belt configuration consists of two steps, as described below.

- **Conveyor belt parameters:** Enter a name based on functional requirements. If the name field is empty, click Next to see the prompt "Name cannot be empty." Select a tool; clicking Next will apply the corresponding tool coordinate system. This ensures that when setting loading/unloading points for the conveyor belt, the teaching points align with the current tool coordinate system.

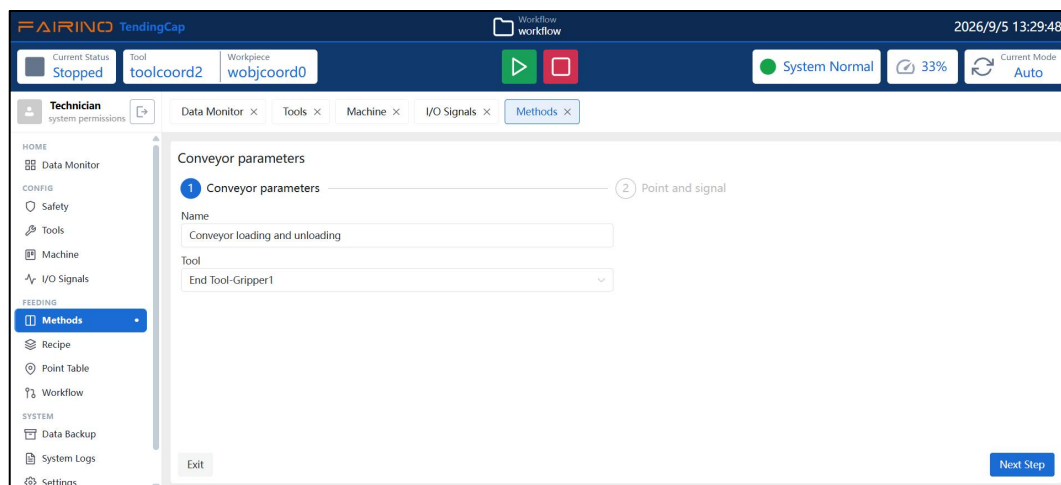


Figure 8.1-16 Conveyor Belt Parameters

● **Position and Signals:** Click the "Set Loading/Unloading Positions" button to teach the current robot pose as the conveyor belt loading/unloading position data, then enter the transition point offsets (X, Y, and Z). Signals may not need to be configured; if you wish to add one, click the "Add Conveyor Belt Signal" drawer on the right. By default, only input signal types are available. After entering the function and signal details, click Add to complete the addition. Finally, click Add to finalize the new conveyor belt configuration.

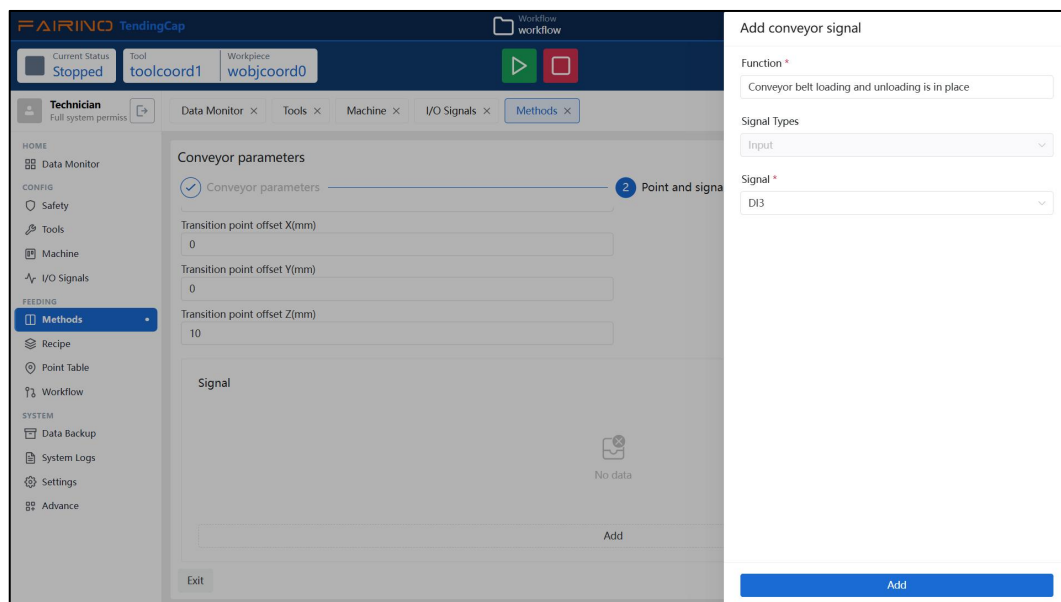


Figure 8.1-17 Signal Addition

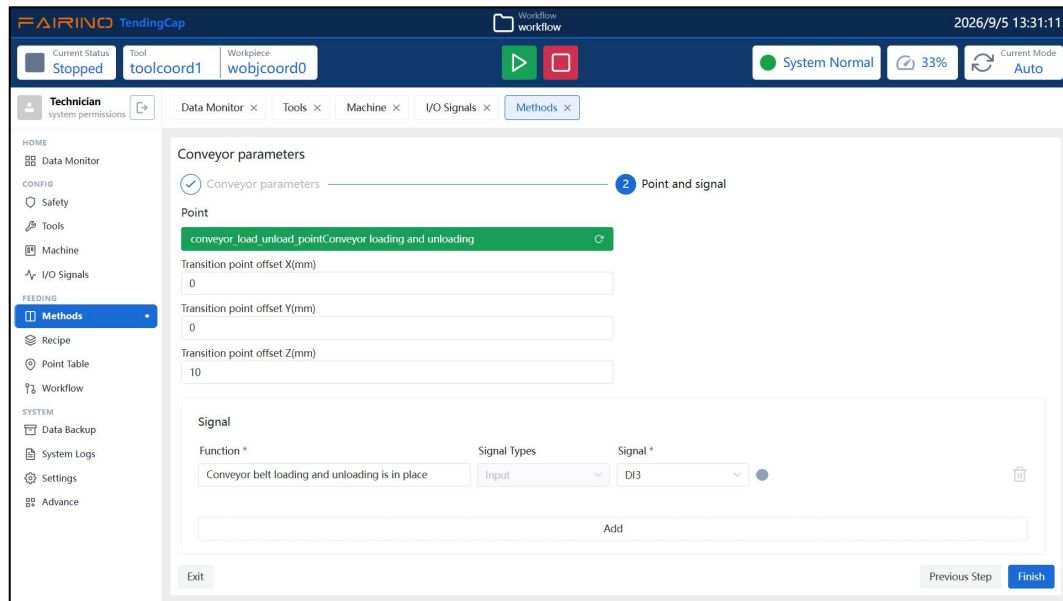


Figure 8.1-18 Point Location and Signal

After successful addition, the homepage displays configured conveyor belts. Click the "Edit" button in the upper-right corner of the conveyor belt configuration card to view or modify details. Click the "Delete" button in the upper-right corner to display a secondary confirmation for "Delete Conveyor Belt"; click confirm to delete.

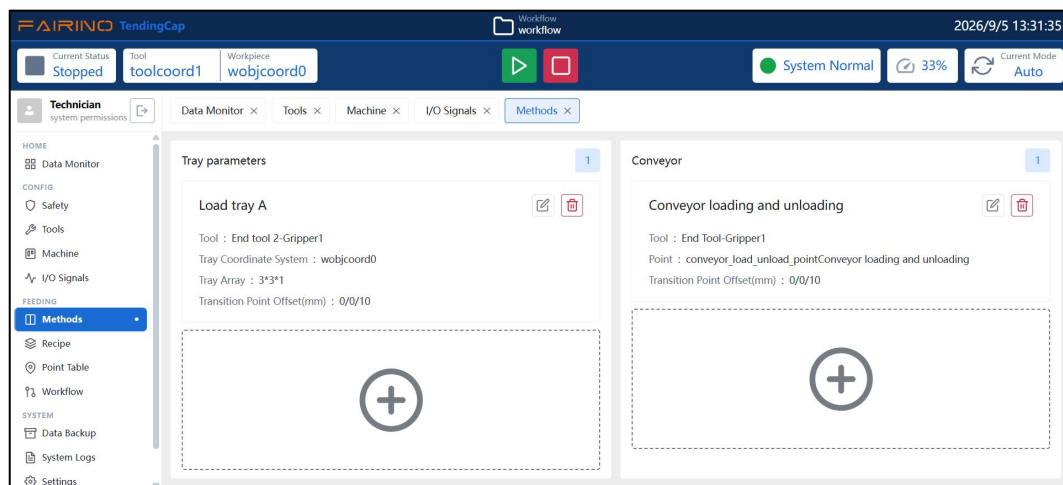


Figure 8.1-19 Conveyor Belt List

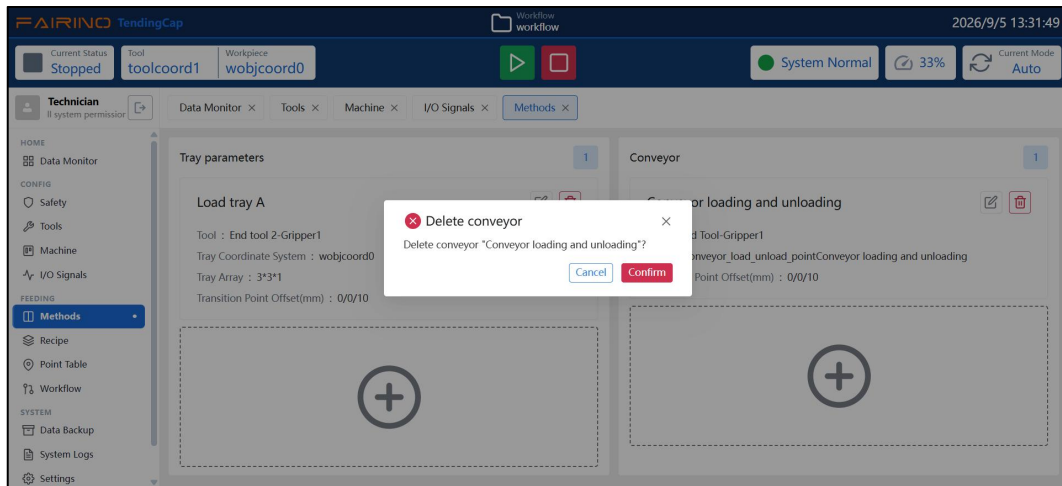


Figure 8.1-20 Delete secondary confirmation for the conveyor belt

After entering the editing interface to modify the tool, you must re-configure the conveyor belt loading/unloading points; otherwise, the modification cannot be completed. When clicking the Complete button, the system will verify whether the tool matches the conveyor belt loading/unloading points. If they do not match, a prompt will appear: "The parameter point coordinate system has changed. Confirm consistency before completing configuration."

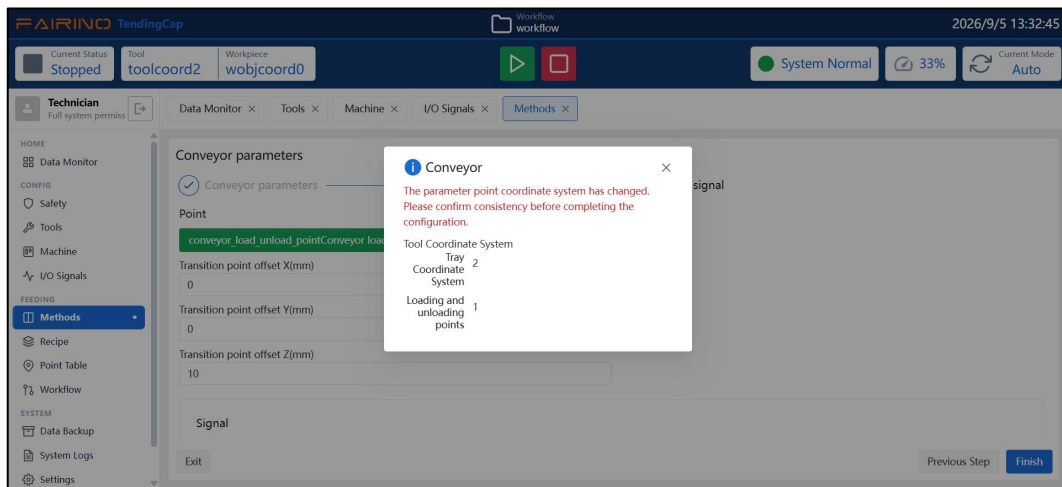


Figure 8.1-21 Warning: The coordinate system for the conveyor belt location has changed

8.2 Recipe

Configure the recipe in the "Material Loading - Recipe" section. Upon entering the page, retrieve recipe list data. If no data is available, display "No Data".

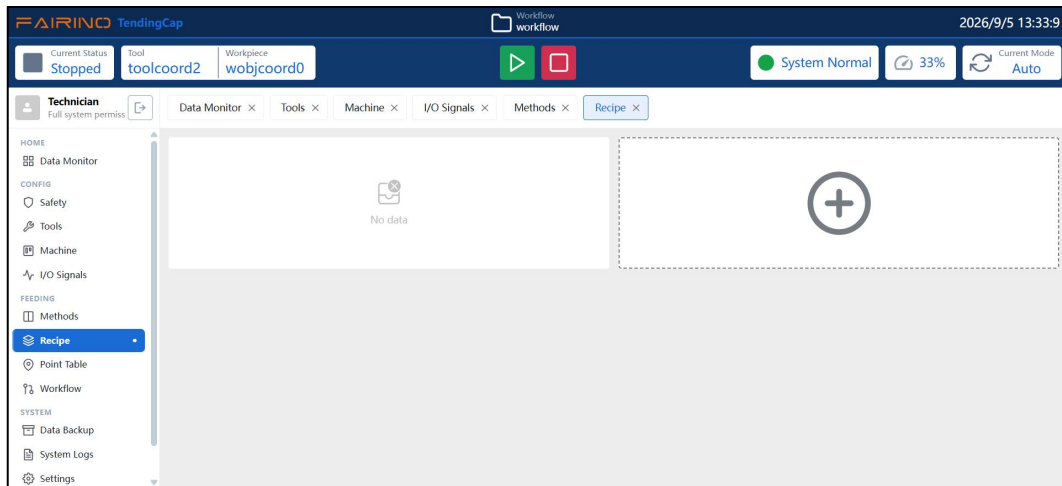


Figure 8.2-1 No formula data available

Click the Add tab to create a recipe in four steps, as follows.

- **Formula Parameters:** Enter the formula name as required. If the name field is empty, click "Next" to see the prompt "Name cannot be empty." Select the operation mode based on your loading/unloading scenario—loading/unloading, loading only, or unloading only. Choose whether to use an extended axis; configuration for extended axis functionality requires setup and connection in the Web App. After confirming usage, the extended axis's connection status, IP address, and port will be displayed.

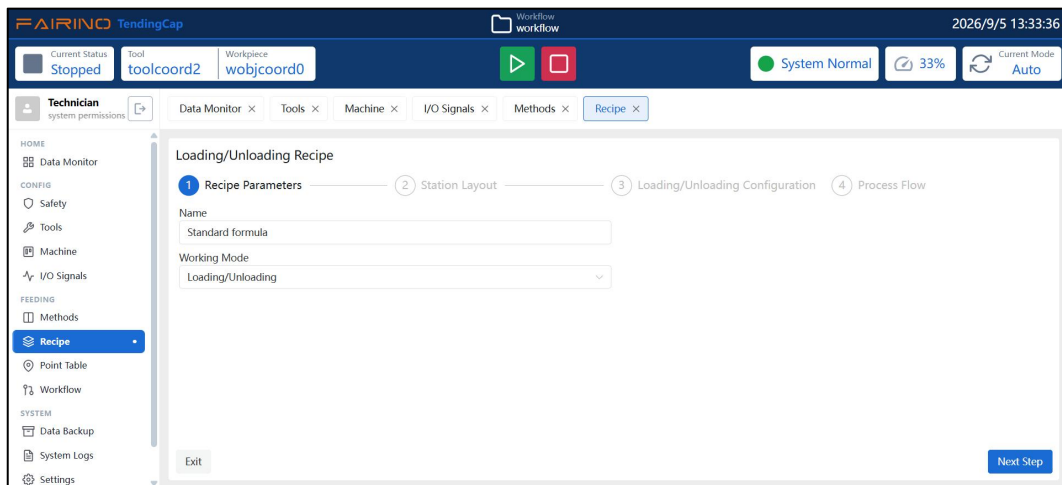


Figure 8.2-2 Formula Parameters

- **Process Layout:** Select machine and loading/unloading methods ("material trays" and "transfer belts").

Important:

- When no machine tool is selected, click the Next button to display the message "Configure at least one machine tool".

➤ When the operation mode in the formula parameters is set to loading/unloading mode, both loading and unloading methods are displayed. If only loading is selected, only the loading method is shown; if only unloading is selected, only the unloading method is displayed.

➤ The selection of loading/unloading method determines the configuration and process steps. Select the appropriate method based on your requirements.

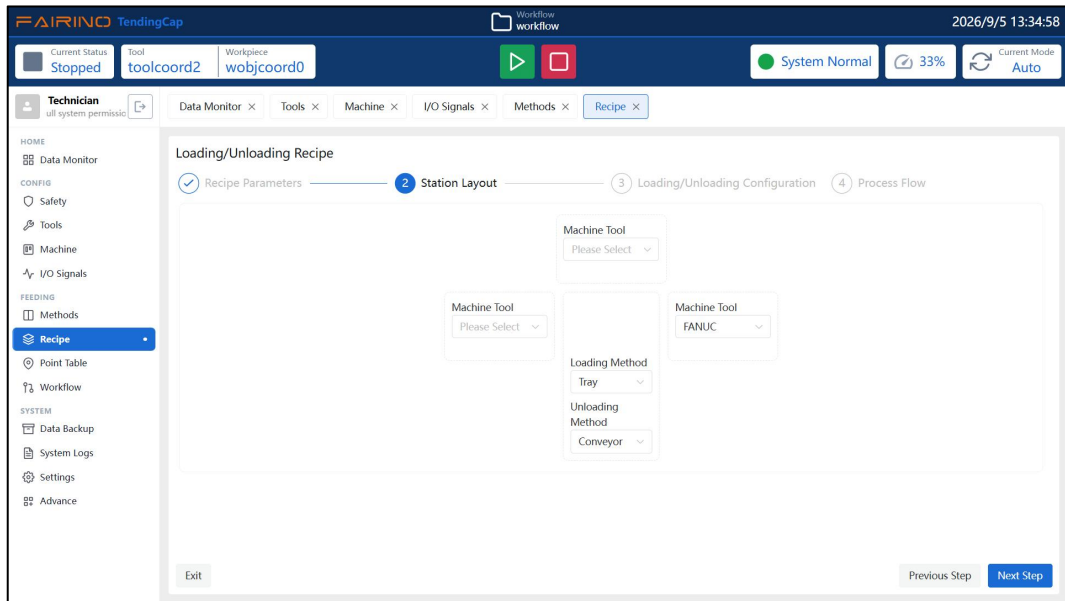


Figure 8.2-3 process layout

● Material loading configuration: Displays the corresponding loading details based on the operational mode of the formula parameters and the material handling method in the process layout. The combination scenarios are as follows:

- Working mode: Load/Unload — Loading from tray, Unloading to tray.
- Working mode: Load/Unload — Loading from tray, Unloading to conveyor.
- Working mode: Load/Unload — Loading from conveyor, Unloading to conveyor.
- Working mode: Load/Unload — Loading from conveyor, Unloading to tray.
- Working mode: Load only— Loading from tray.
- Working mode: Load only— Loading from conveyor.
- Working mode: Unload only— Unloading to tray.
- Working mode: Unload only— Unloading to conveyor.

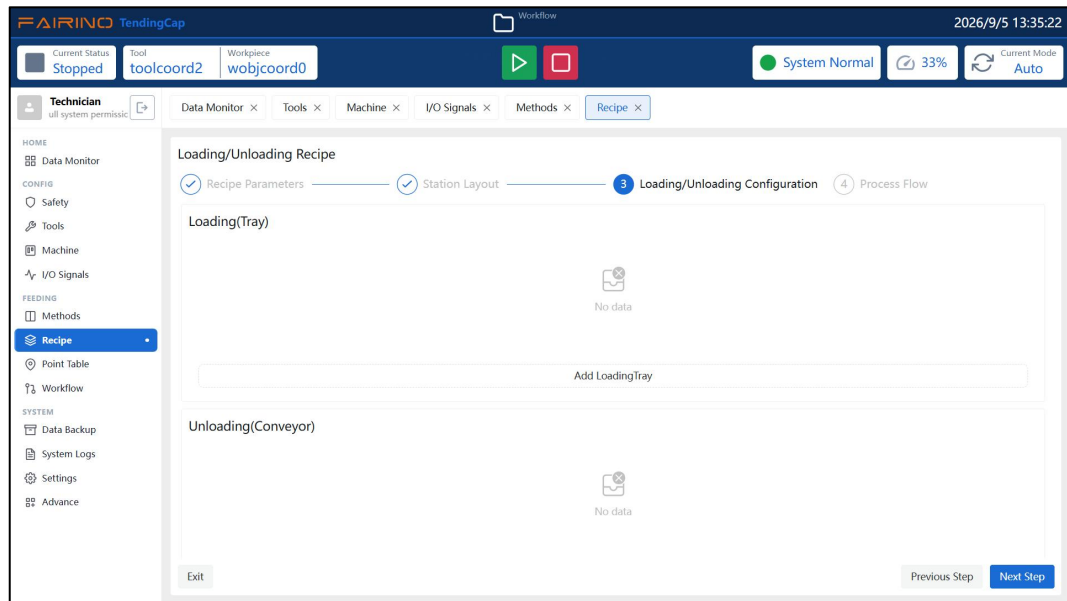


Figure 8.2-4 Work mode — loading and unloading, loading method — tray, unloading method — tray

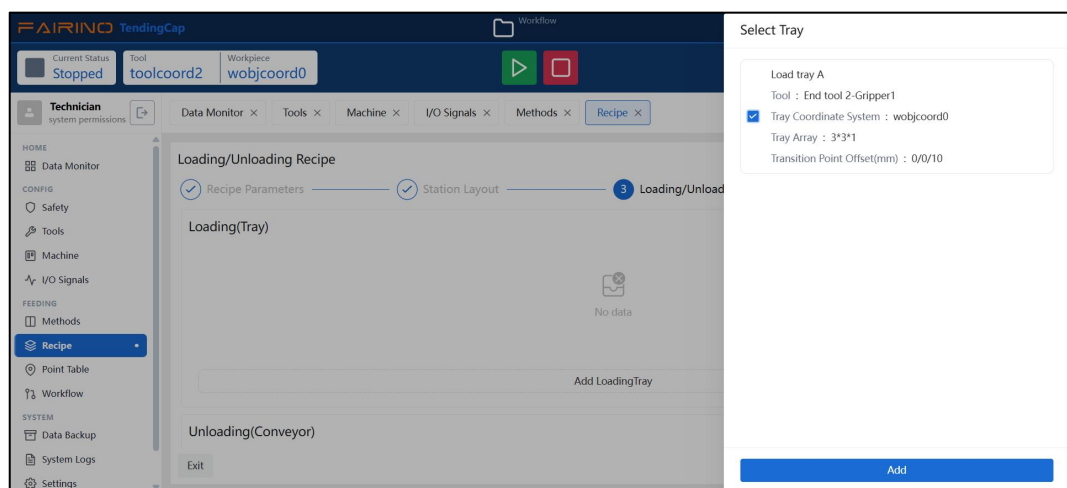


Figure 8.2-5 Add feeding tray

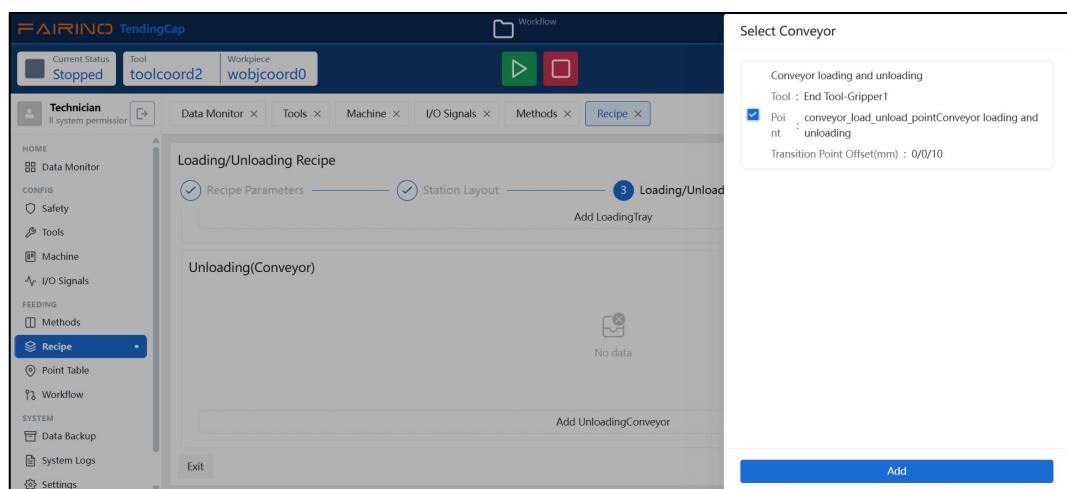


Figure 8.2-6 Add the material feeding conveyor belt

● **Process Flow:** Set the core steps as "Material pickup from tray/belt - Material feeding to machine tool - Material discharge from machine tool - Material release from tray/belt". After configuring the first three steps, clicking "Add Step" will display corresponding content, with data automatically populated based on configuration settings. Clicking "Insert Step Here" allows you to add core steps or integrate additional process steps such as "Secondary Positioning" and "Custom Process" according to specific requirements. Within each step, click "Add Sub-step" to select from available options including "Point-to-Point", "Linear Motion", "Control Signal", "Wait Signal", or "Matrix Movement". Once the process flow is complete, click the "Complete" button to finalize the addition.

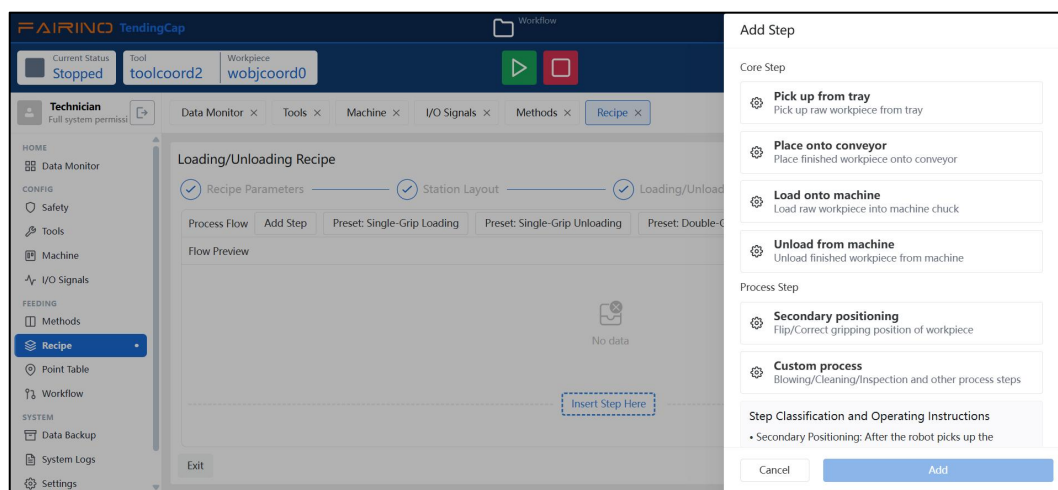


Figure 8.2-7 Add Step / Insert Step Here

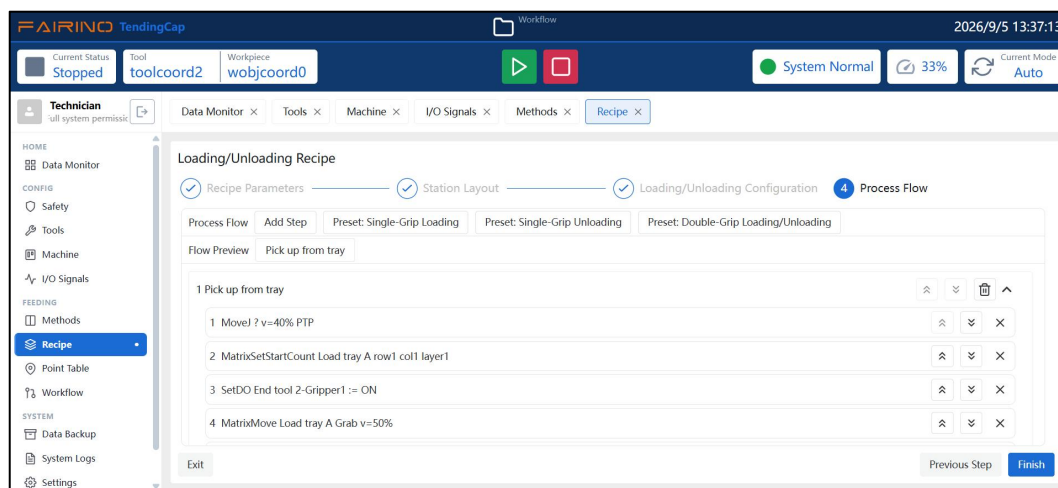


Figure 8.2-8 Add Step / Insert Step Here - Example: Material Disk Extraction

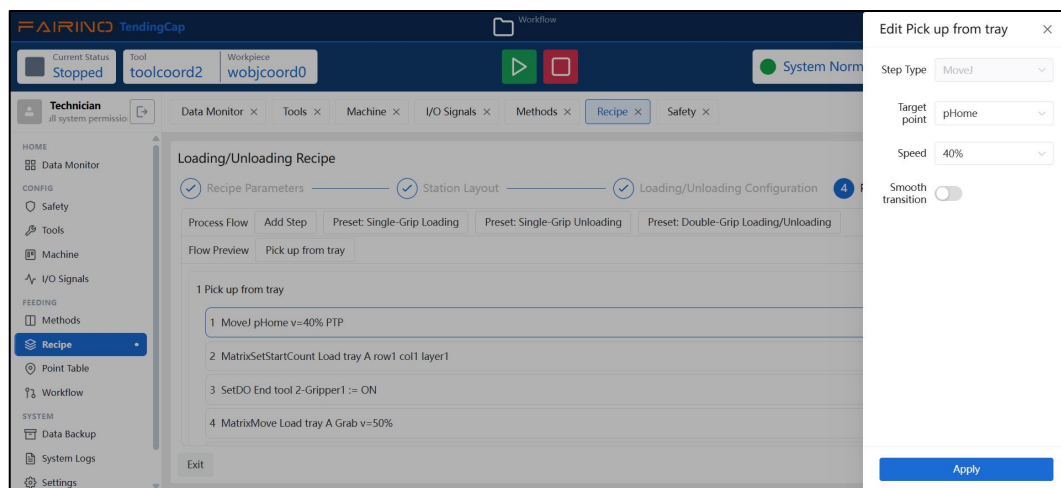


Figure 8.2-9 Add Sub-step

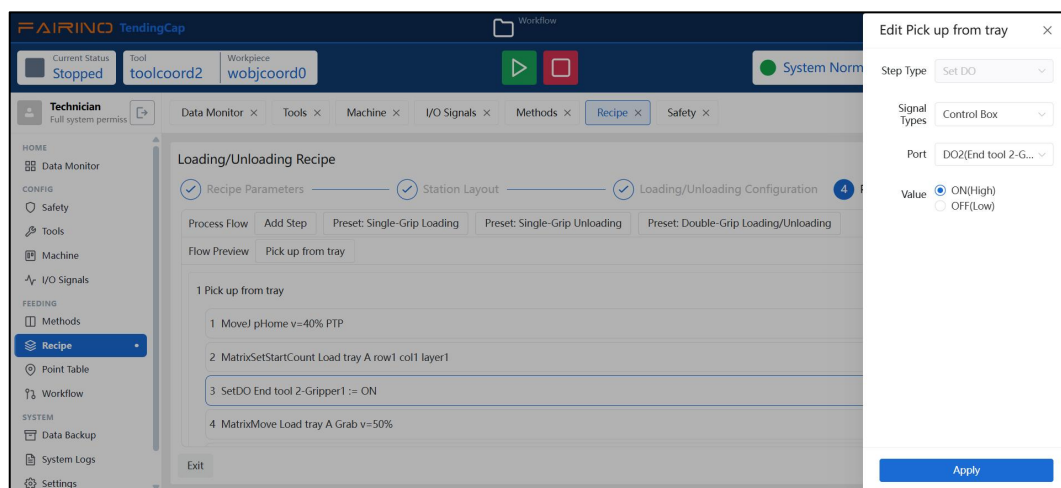


Figure 8.2-10 Select sub-step to edit and modify

After successful addition, the homepage displays configured recipes. Click the Edit button in the upper-right corner of the recipe configuration card to view or modify details. Click the Delete button in the upper-right corner to confirm deletion; click OK to remove the recipe.

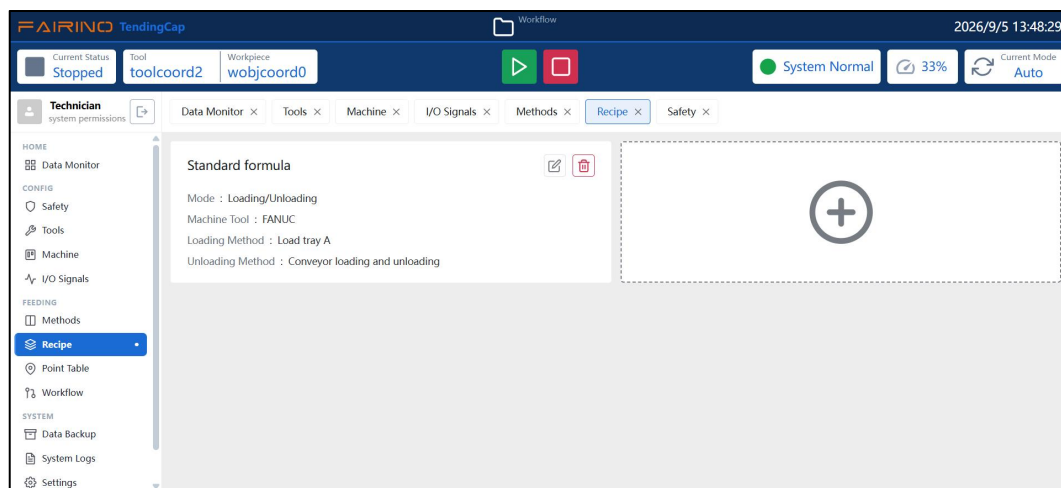


Figure 8.2-11 Formula List

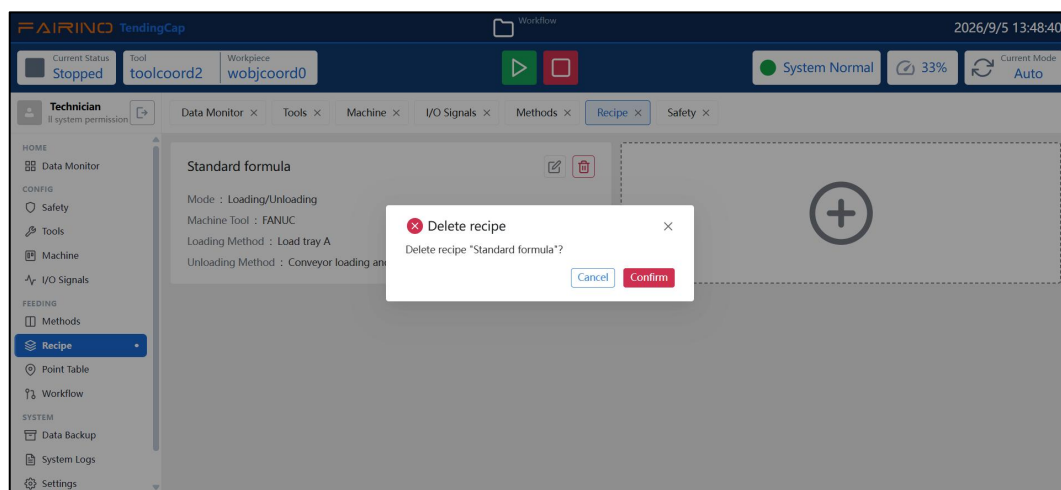


Figure 8.2-12 Formula deletion requires two-factor confirmation

During the workflow process, click the button for each step under "Process Preview" to automatically scroll the corresponding content to the top for easy viewing. Users can move steps up or down, delete them, or fold/unfold them as needed; they can also move sub-steps up or down, delete them, or edit them by selecting the drawer on the right side of the step bar.



Name: Move Up. Function: Move a process or sub-step up.



Name: Move Down. Function: Move a process or sub-step down.



Name: Fold. Function: Fold and hide process content.



Name: Expand. Function: Display process content in detail.



Name: Process deletion. Function: Delete process content.



Name: Sub-step deletion. Function: Delete sub-step content.

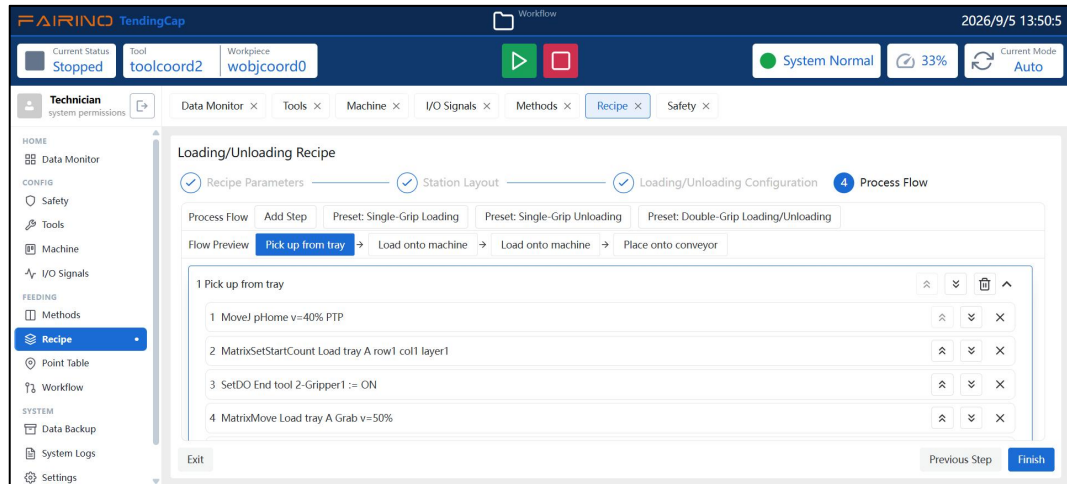


Figure 8.2-13 Material retrieval from the storage tray

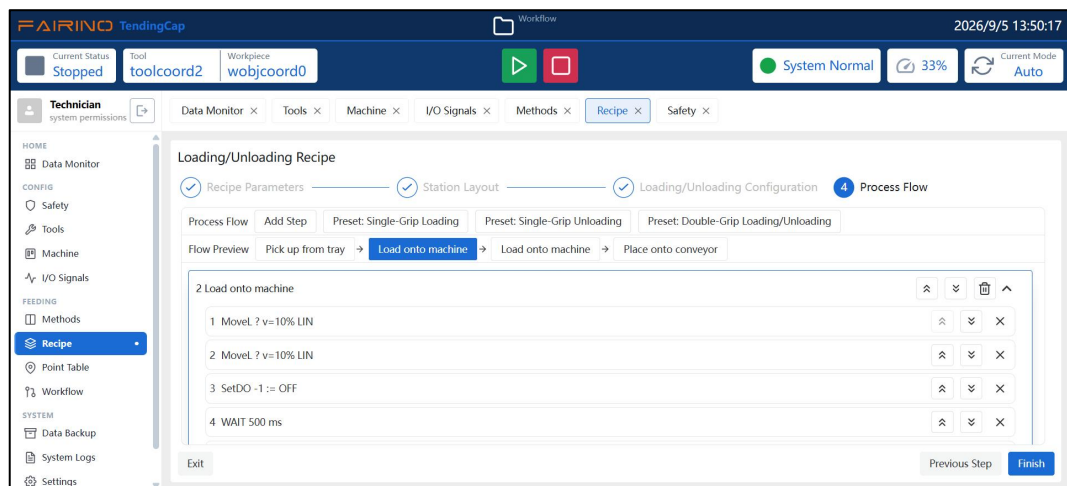


Figure 8.2-14 Material loading on the machine tool

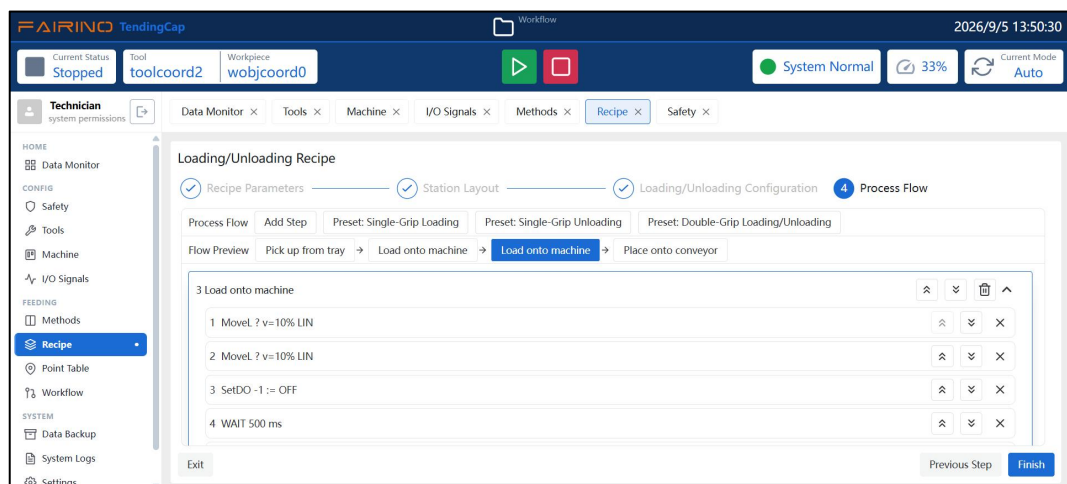


Figure 8.2-15 Material cutting on the machine tool

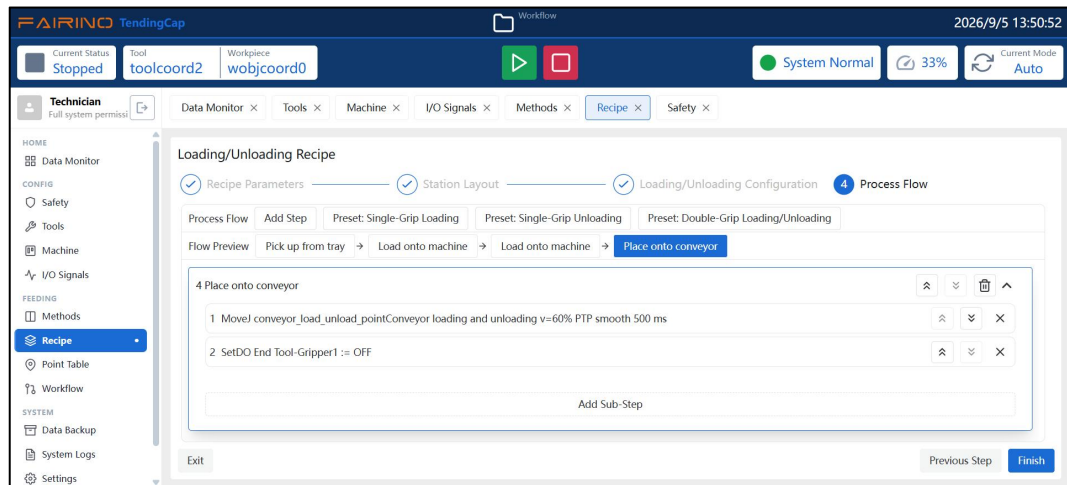


Figure 8.2-16 Conveyor belt discharge

When there is existing content in the workflow, click a preset instruction in "Workflow" and a prompt will appear: "The workflow already has content. Generating again will overwrite existing content. Continue?" Click OK to generate. If no content is selected, the preset workflow content will be displayed directly on the interface.

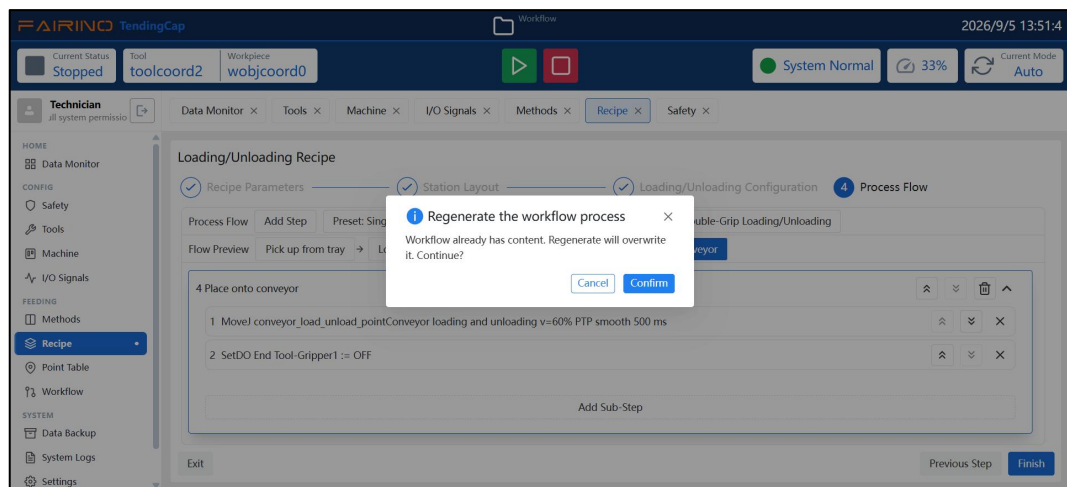


Figure 8.2-17 Regenerate workflow prompt

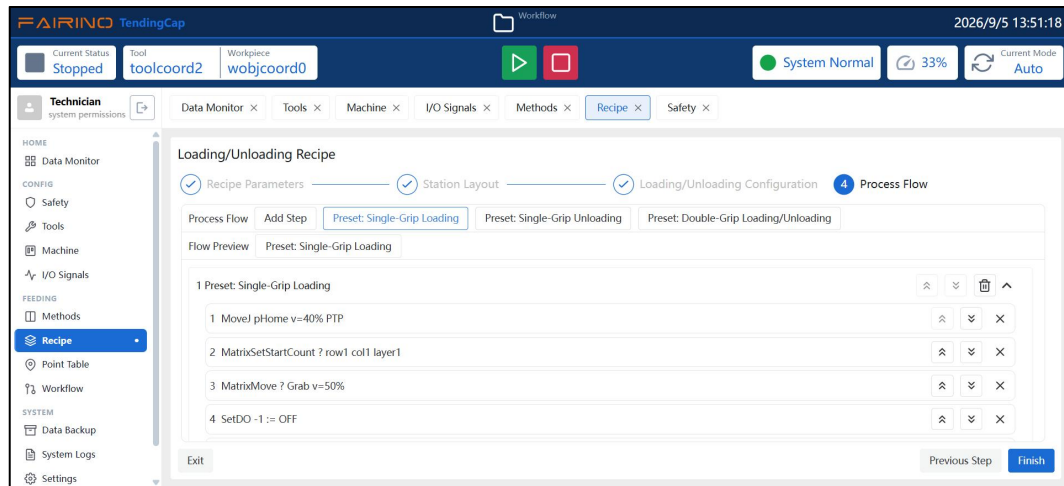


Figure 8.2-18 Confirm and generate the preset workflow

8.3 Point Table

In the "Material Handling – Point Table" section, you can view, update, perform single-point operation tests, delete, and record point data.

- The left "Teaching Point" card displays and views point data. At the top of the card, you can select the "PTP/LIN" mode for single-point operation, filter table data, and perform point recording based on the current robot pose. The point table supports renaming, viewing, updating, testing with single-point operation, and deleting point names.

- On the right, the "Move" tab displays the Cartesian space for robot movement (based on Base/Tool/Wobj). The interface features button groups to switch between corresponding Cartesian pose (X, Y, Z, RX, RY, RZ) and joint (J1 – J6) data. Long-pressing a button controls the robot to move linearly along the X, Y, or Z axes or rotate around RX, RY, or RZ.

- On the right side, the "Move" tab displays the joint space for robot movement (single-axis point motion); operate the left-side increment/decrement buttons to control robot motion. In manual mode and joint coordinate system mode, rotate a specific robot joint. When the robot exceeds its motion range (soft limit) and stops, use single-axis point motion for manual operation to move it out of the restricted position. Single-axis point motion is faster and more convenient than other operation modes for rough positioning or large-scale movements. Set the "Move Distance" parameter (the maximum travel distance when holding the button; input range: 0 – 300). Pressing the

increment/decrement buttons controls robot movement; releasing the button immediately stops motion, while holding it continuously causes the robot to move until reaching the set threshold before stopping.

- Right-side "Move" Card – Position: Enter the tool's coordinate position data to calculate the joint position; click the "Move" button at the joint position to have the robot execute the calculated joint movement.

- Right-side "Move" tab – Expansion Axis: After successfully establishing a UDP connection, perform enable/disable and forward/reverse tests, as well as zero-point calibration.

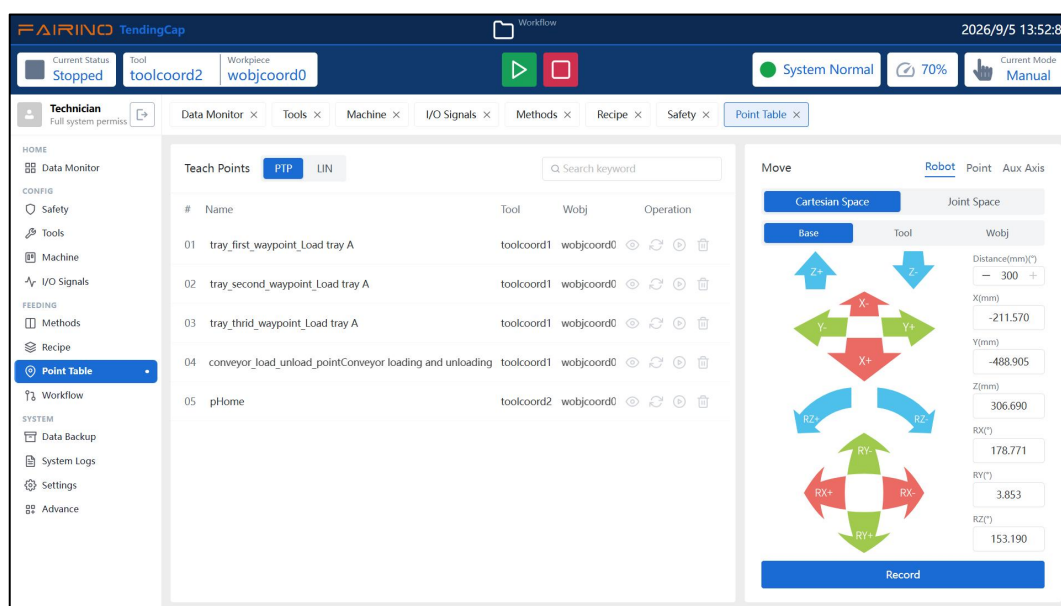


Figure 8.3-1 Point Table

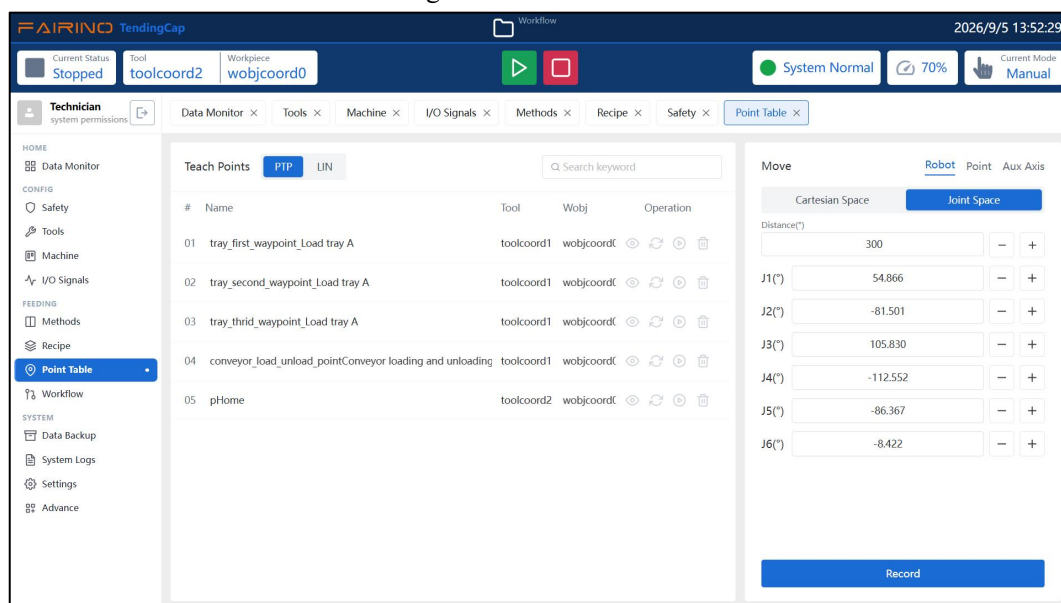


Figure 8.3-2 Joint space

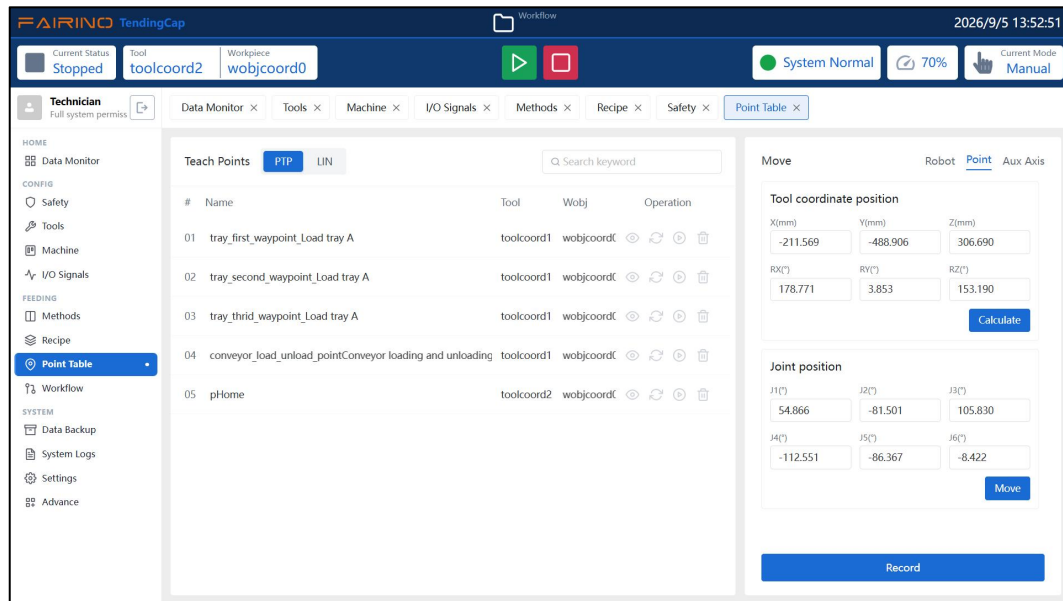


Figure 8.3-3 Tool coordinate position calculation – Joint position movement

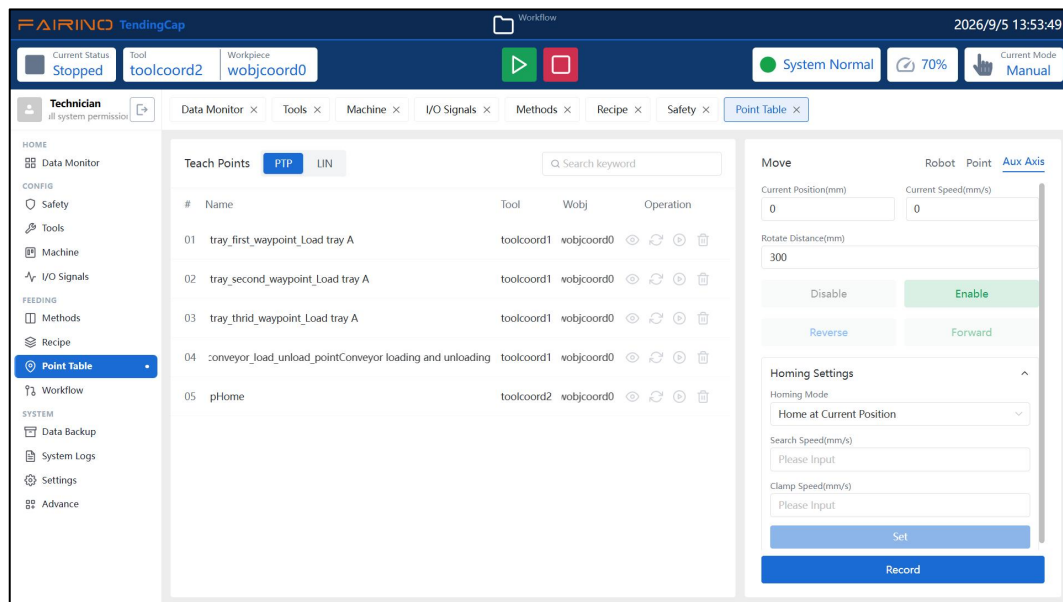


Figure 8.3-4 Extension Axis

8.4 Workflow

Configure the workflow in the "Material Handling – Workflow" section. Upon entering the page, retrieve workflow list data. If no data is available, display "No Workflows".

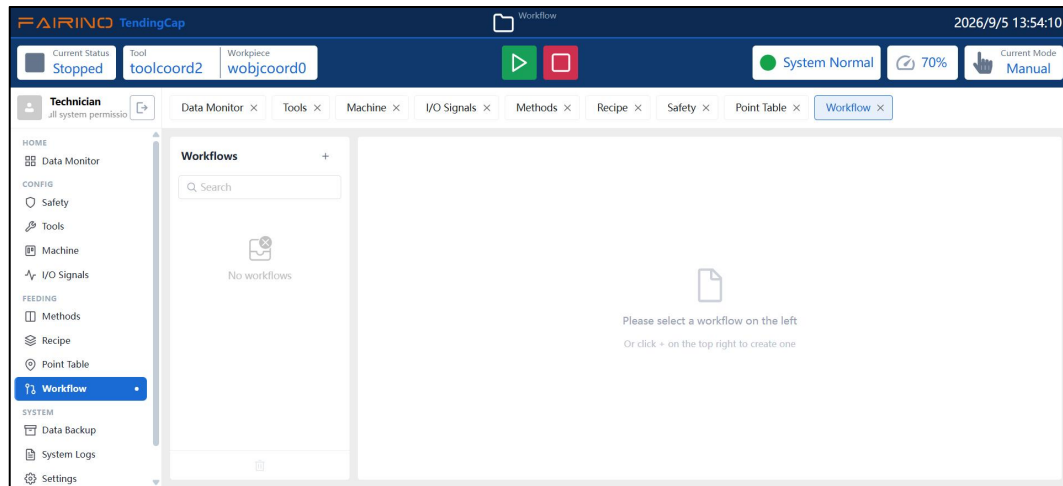


Figure 8.4-1 No workflow data

When adding a workflow, you can create a new one from a blank template or copy an existing one, and link it to an existing recipe. Then click the Create button to complete the workflow design. After successful creation, click the Generate by Recipe button to generate the workflow content based on the recipe's workflow steps. The workflow content is displayed in a tree structure.

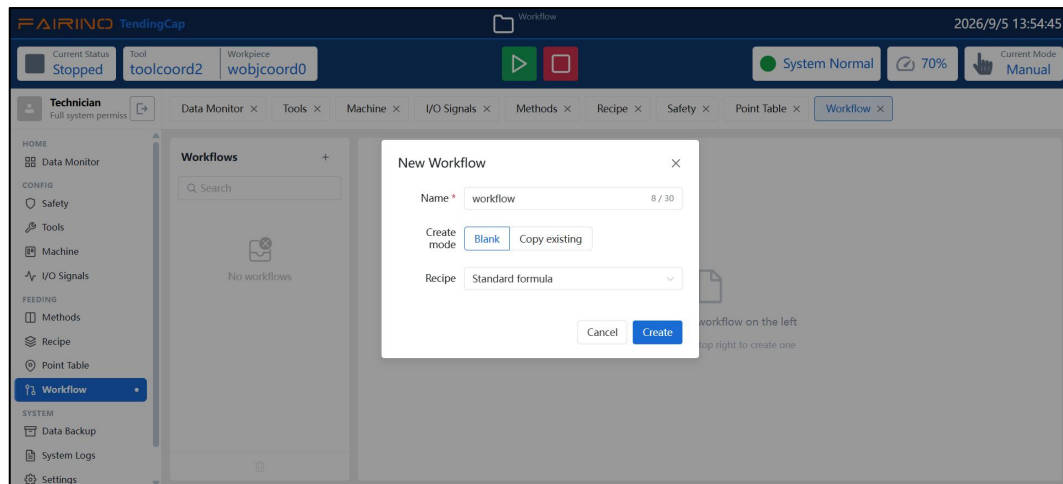


Figure 8.4-2 Create New Workflow Blank

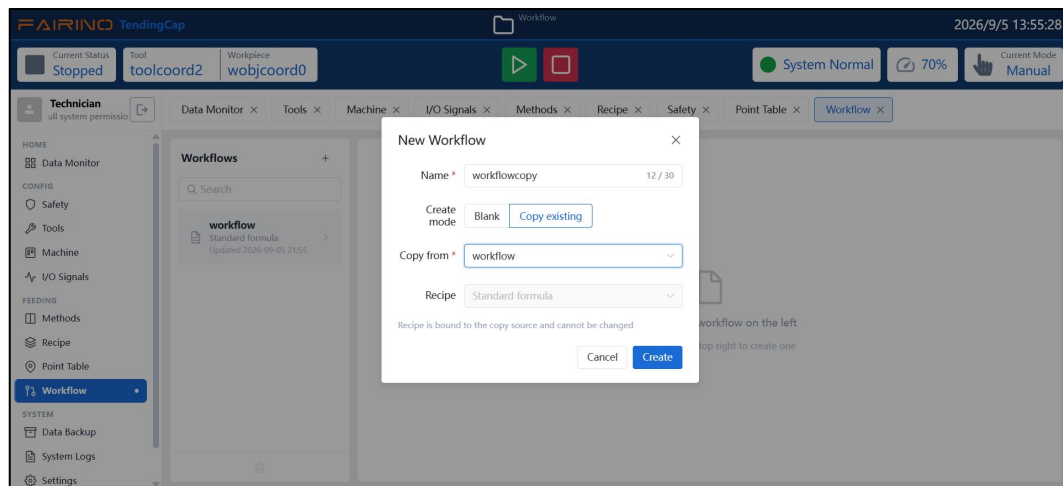


Figure 8.4-3 Workflow copies existing items and creates new ones

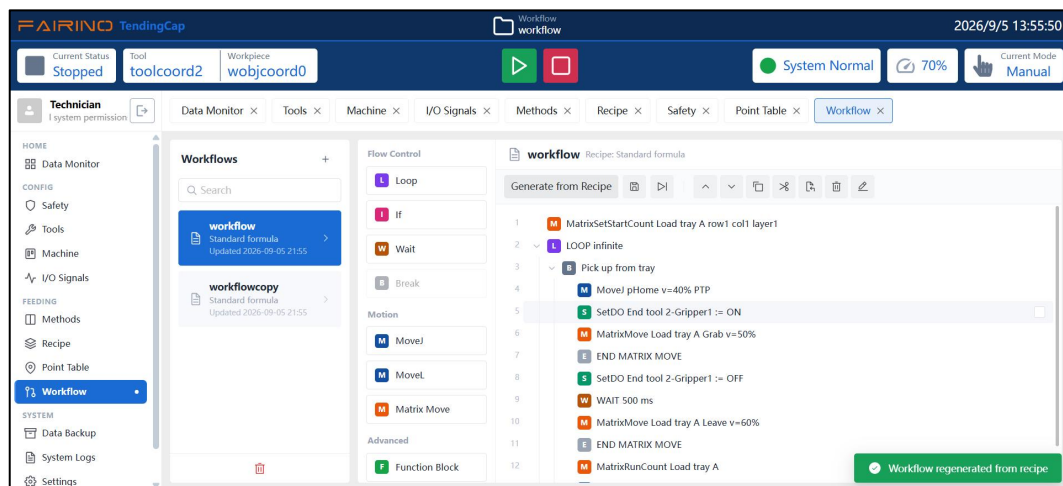


Figure 8.4-4 Workflow generated based on the recipe

8.4.1 Toolbar



Name: Save. Function: Save workflow content.



Name: Run in One Step. Function: executes the selected workflow instruction in a single step.



Name: Move Up. Function: Move the selected workflow instruction up;



Name: Move Down. Function: Move the selected workflow instruction down;



Name: Copy. Function: Copy selected workflow instructions.



Name: Cut. Function: Cut the selected workflow instruction.



Name: Paste. Function: Paste a copied or cut-and-paste workflow instruction.



Name: Delete. Function: Remove the selected workflow instruction.



Name: Edit. Function: Edit selected workflow instructions.

8.4.2 Flow control

8.4.2.1 Loop

Click "Loop" to open the "Edit Loop" interface on the right. The loop scenario is as follows:

- infinite loop
- Limited iterations: Number of iterations to input
- Condition-based loop: Select the signal state or variable comparison type based on requirements, and add conditions

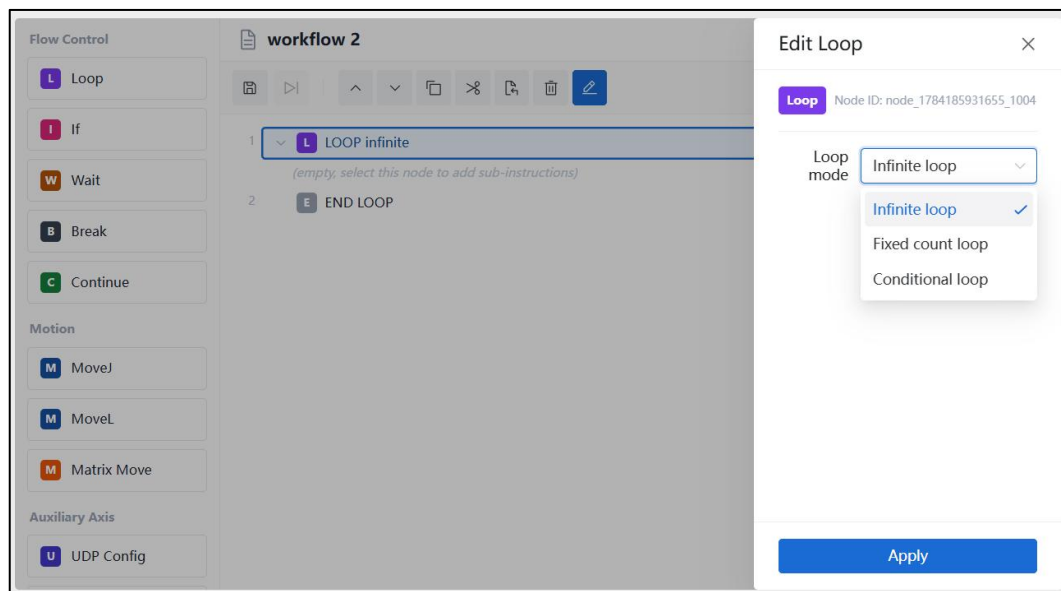


Figure 8.4.2.1-1 Loop instruction

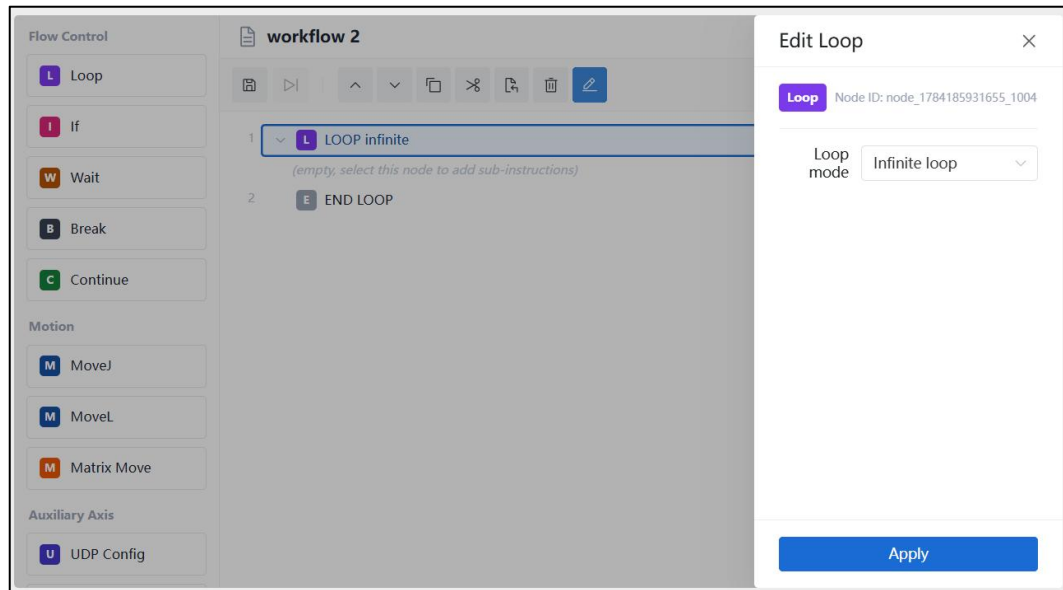


Figure 8.4.2.1-2 Loop Instruction – Infinite Loop

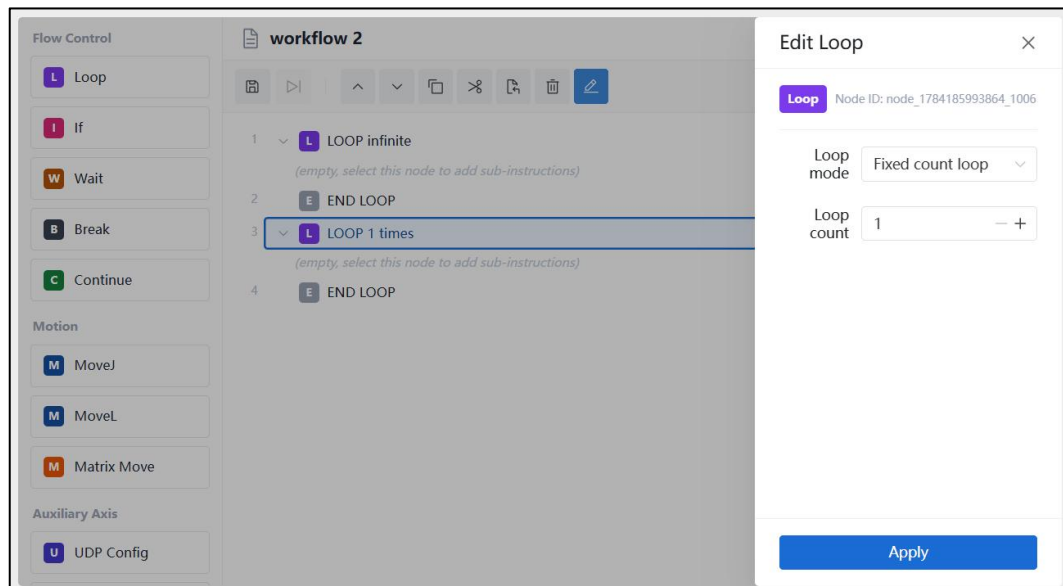


Figure 8.4.2.1-3 Loop Instruction – Limited Number of Loops

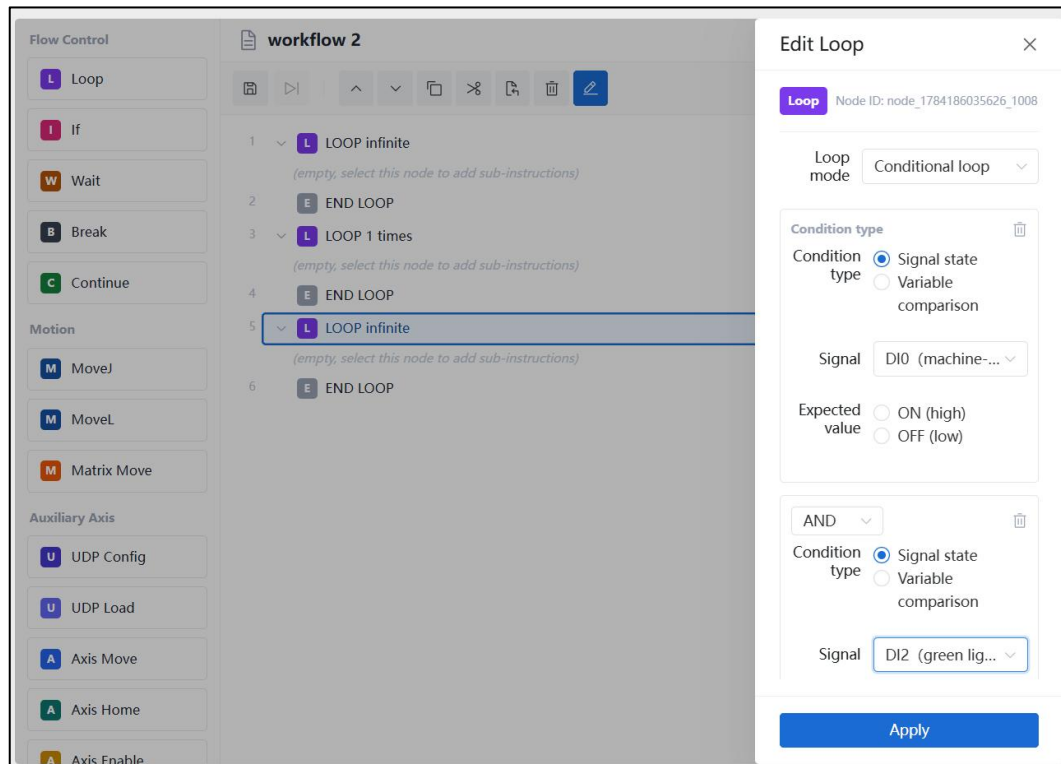


Figure 8.4.2.1-4 Loop Instruction – Loop if the condition is met

8.4.2.2 If

Click "If" to open the "Edit If" interface on the right. The scene is as follows:

- Add Condition: Add a judgment condition
- Add Otherwise Clause: Add additional logic for judgment
- Add Otherwise: No configuration required; executed when none of the other branches meet the conditions.

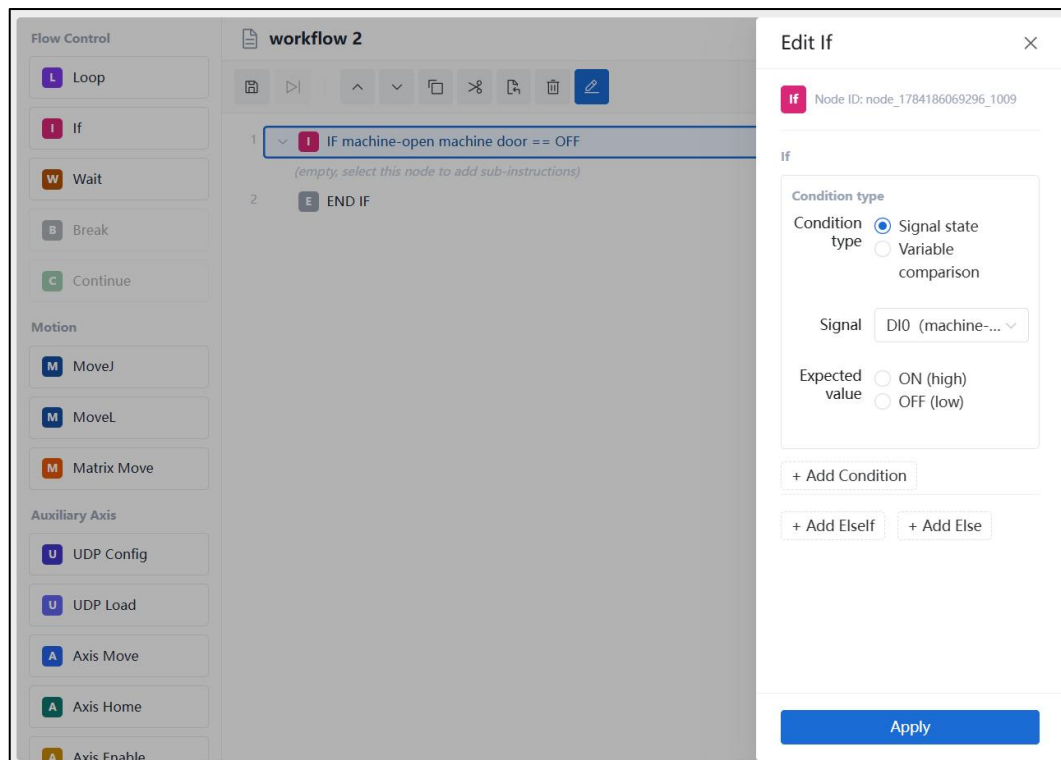


Figure 8.4.2.2-1 if instruction

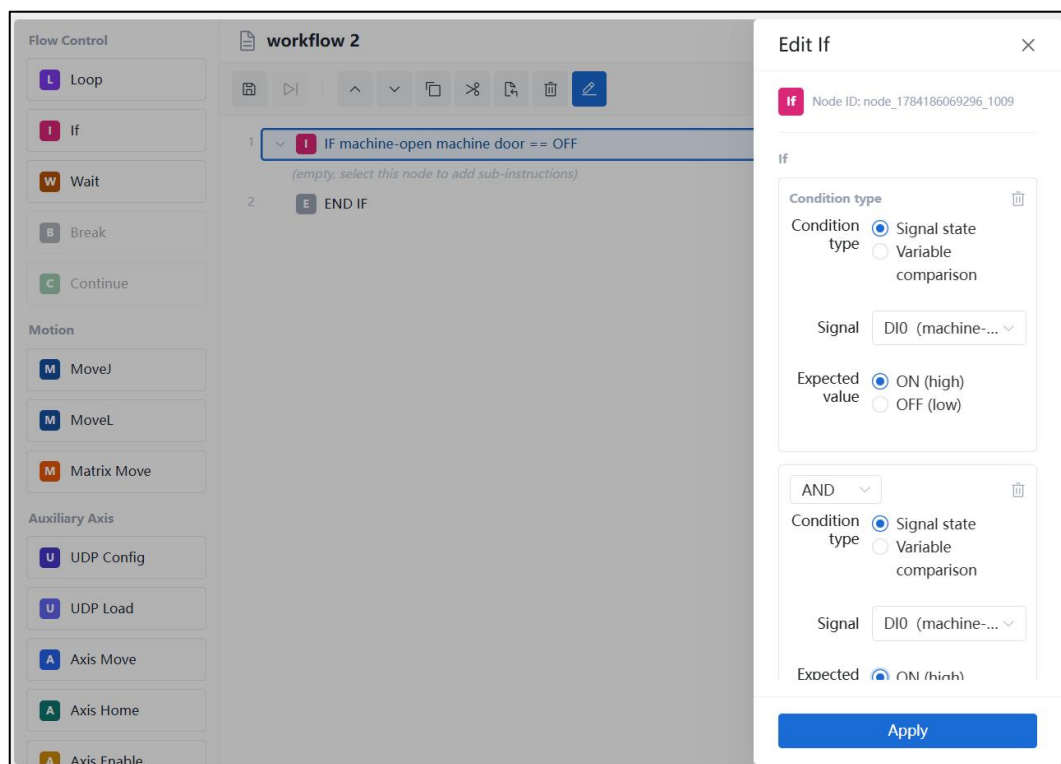


Figure 8.4.2.2-2 If Instruction – Add Condition

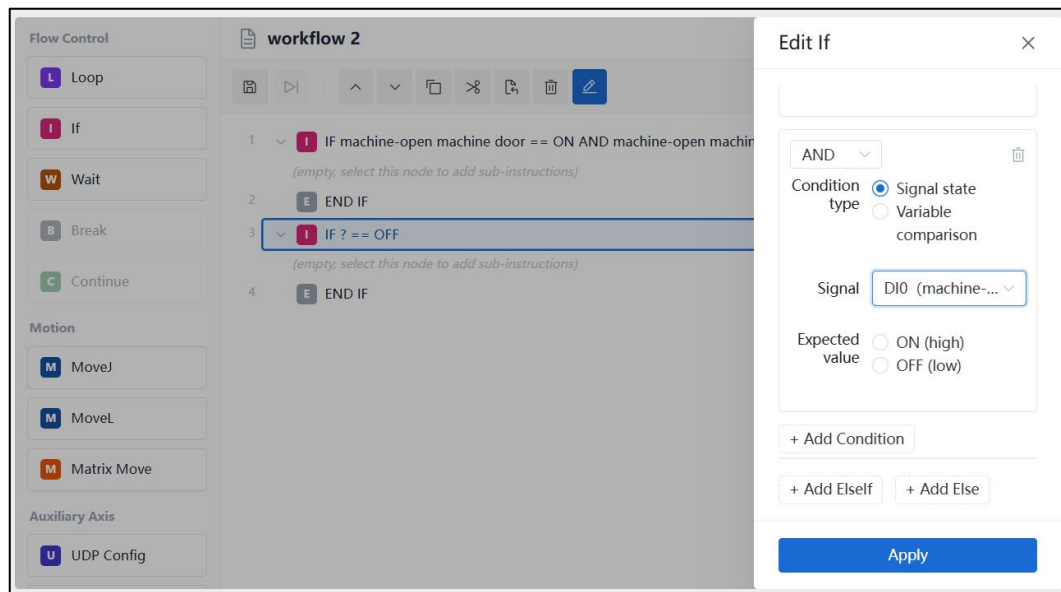


Figure 8.4.2.2-3 If Instruction – Add Otherwise Judgment

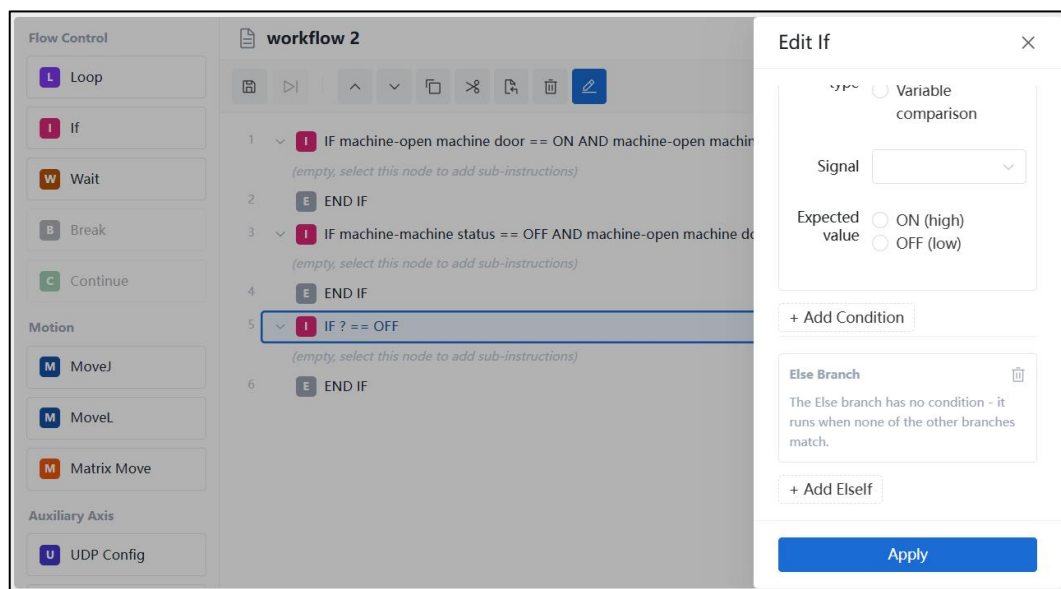


Figure 8.4.2.2-4 If Instruction – Add Otherwise

This instruction requires basic programming knowledge. For assistance, please contact us.

8.4.2.3 Wait

Click "Wait" to open the "Edit Wait" interface on the right, and enter the delay duration in milliseconds

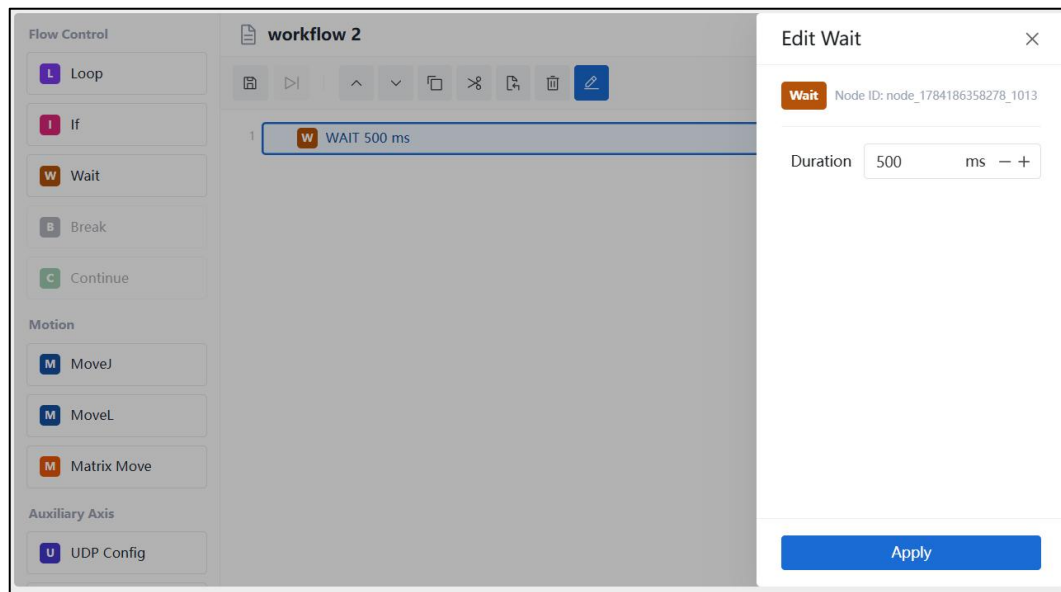


Figure 8.4.2.3-1 Wait Instruction

8.4.2.4 Break

Click "Break" to open the "Edit Break" interface on the right and use it with loop instructions.

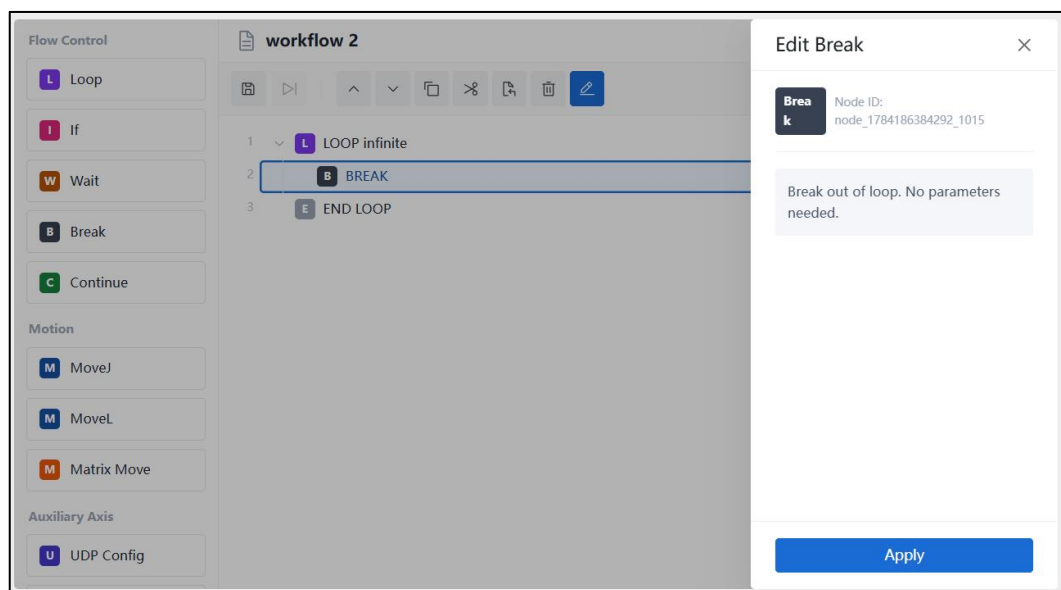


Figure 8.4.2.3-1 Breakinstruction

8.4.2.5 Continue

Click "Continue" to open the "Edit Continue" interface on the right and use it with the loop instruction.

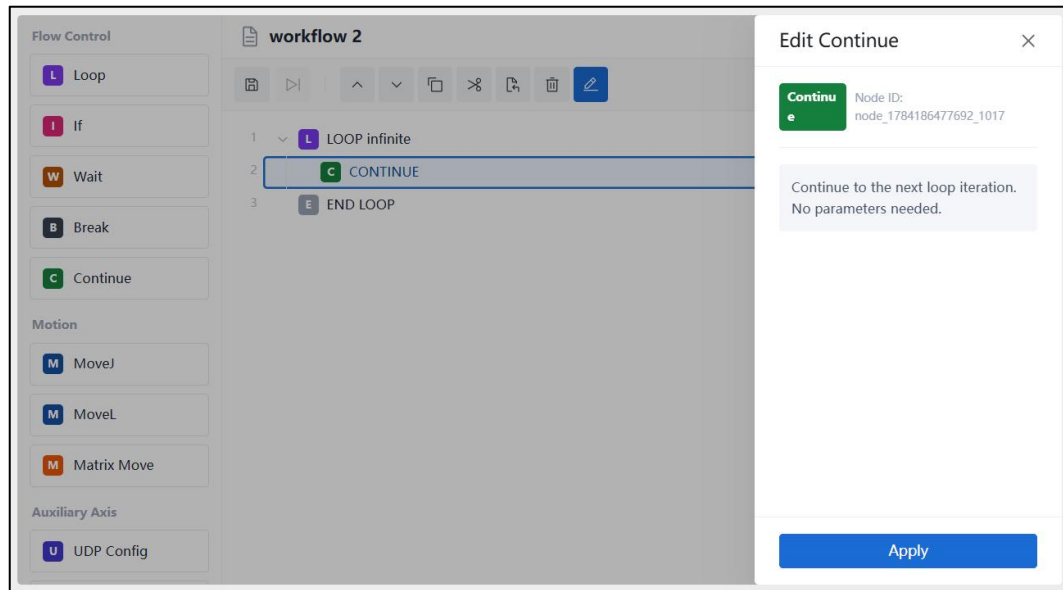


Figure 8.4.2.5-1 Continue Instruction

8.4.3 Movement Instruction

8.4.3.1 MoveJ

Click "MoveJ" to open the "Edit MoveJ" interface on the right. Select the destination point, enable smooth transition, and enter the smooth transition time to ensure a seamless movement from one point to the next. The specific path is the optimal path automatically planned by the motion controller.

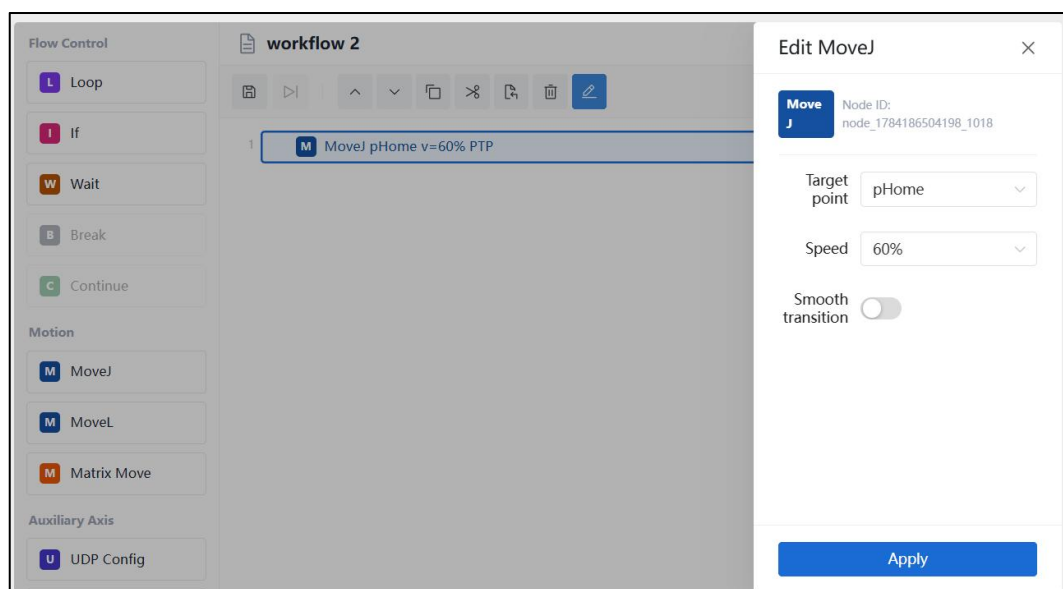


Figure 8.4.3.1-1 MoveJ instruction

8.4.3.2 MoveL

Click "MoveL" to open the "Edit MoveL" interface on the right, select the destination point, enable smooth transition, enter the smooth transition radius, and choose the transition method. The path to the destination is a straight line.

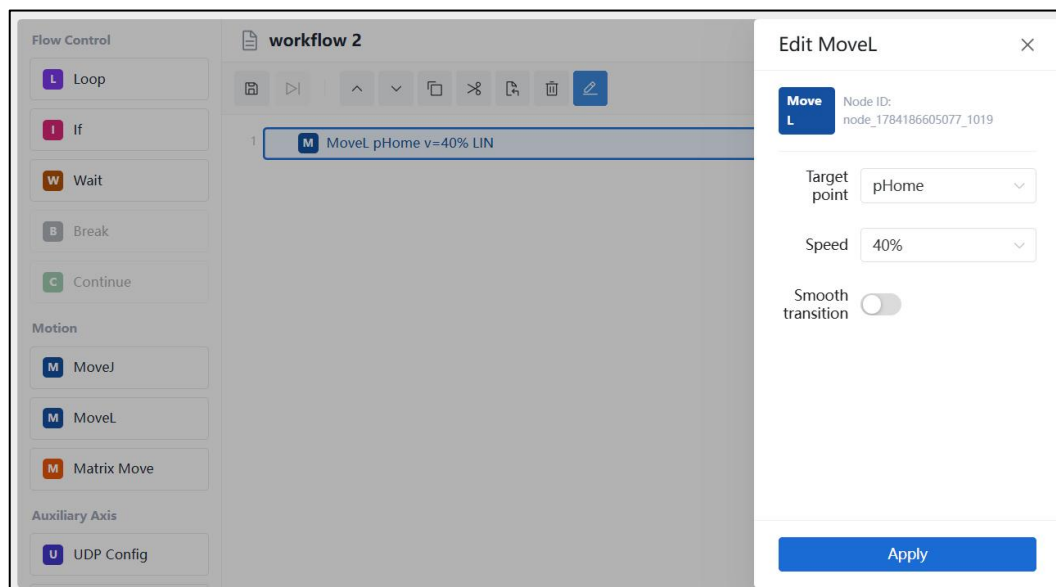


Figure 8.4.3.2-1 MoveL instruction

8.4.3.3 Matrix Move

Click "Matrix Move" to open the "Edit Matrix Move" interface on the right, select the tray and execution action, and the execution action scenario is as follows.

- **Matrix Movement:** When the robot moves to load/unload or grasp/release a selected pallet using an extension shaft, perform synchronous motion planning for the conveyor belt's extension shaft.

- **Matrix operation count:** Counting the row, column, and layer of the robot's loading/unloading operation.

- **Set starting count:** Define where the robot begins loading/unloading operations (row, column, and layer).

- **Get matrix count:** Determine the row, column, and layer of the robot's loading/unloading operation.

- **Total processing count +1:** Increase the total number of robot loading/unloading operations by 1.

- **Total cleared processing operations:** Total robot loading/unloading operations

plus.

- Get total processing count: Get total robot loading/unloading operations.

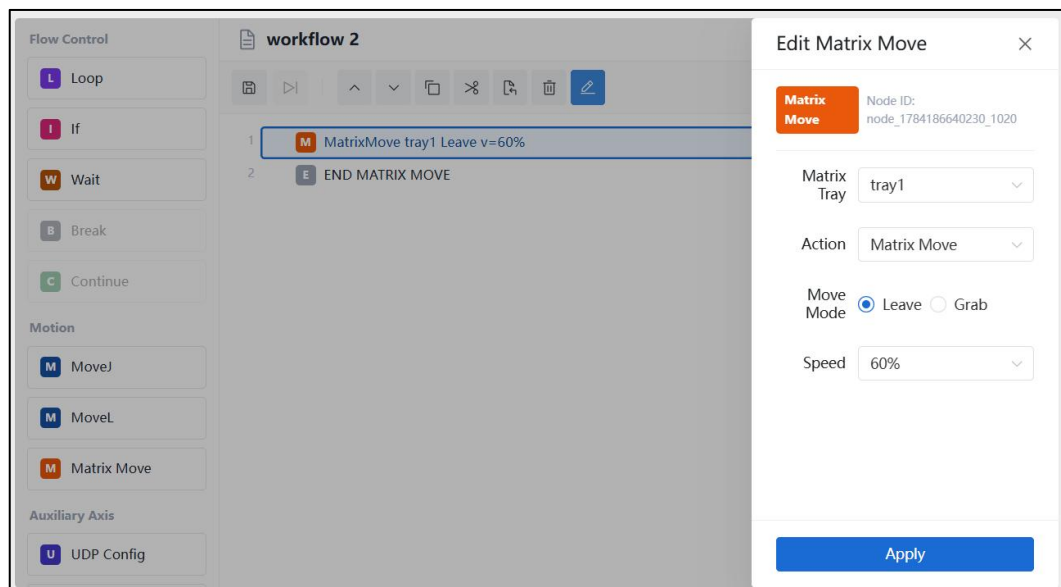


Figure 8.4.3.3-1 Matrix Move

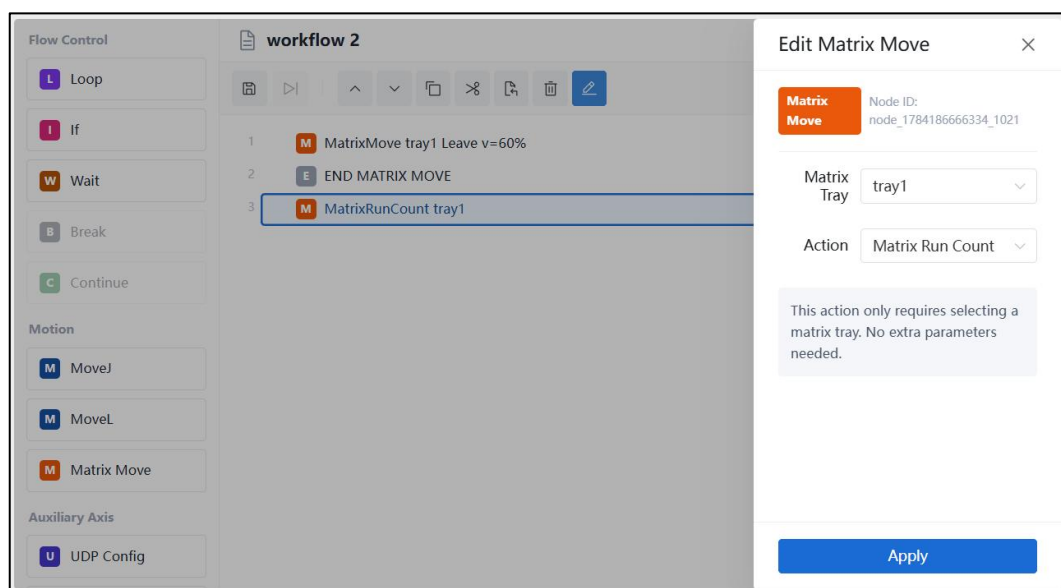


Figure 8.4.3.3-2 Matrix Execution Count

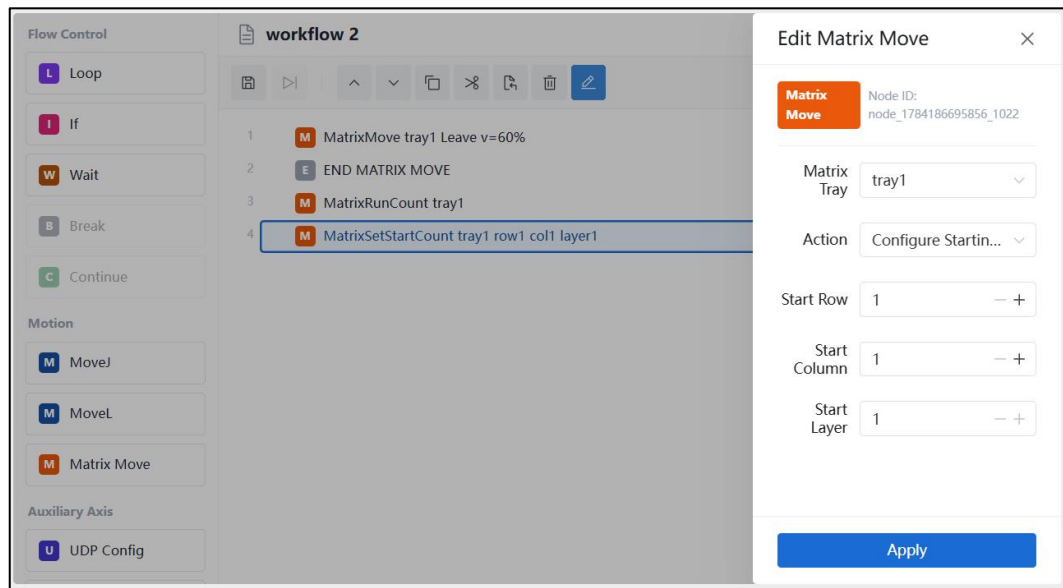


Figure 8.4.3.3-3 Configure starting count

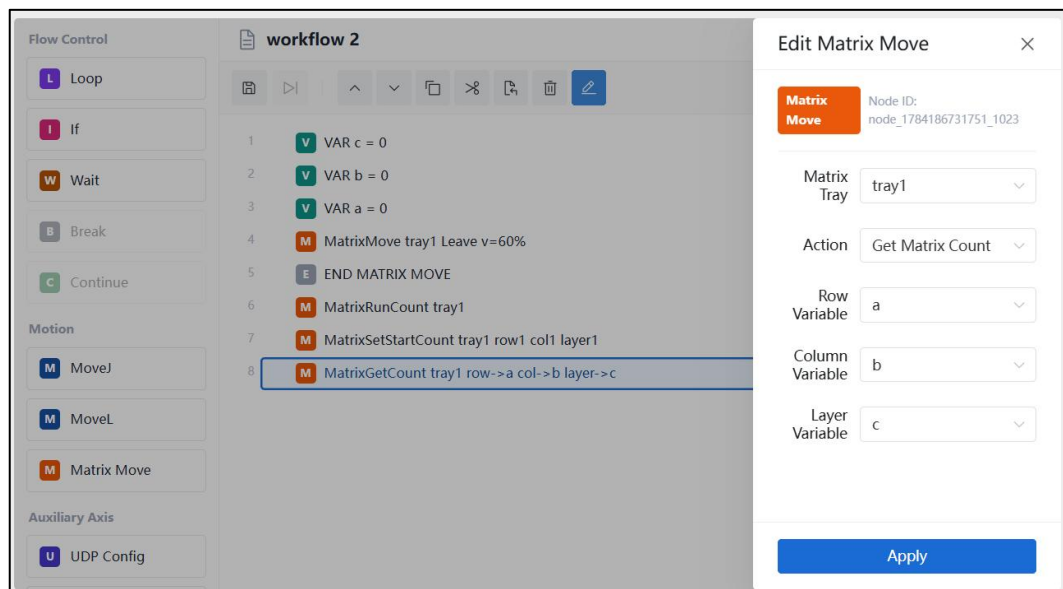


Figure 8.4.3.3-4 Get Matrix Count

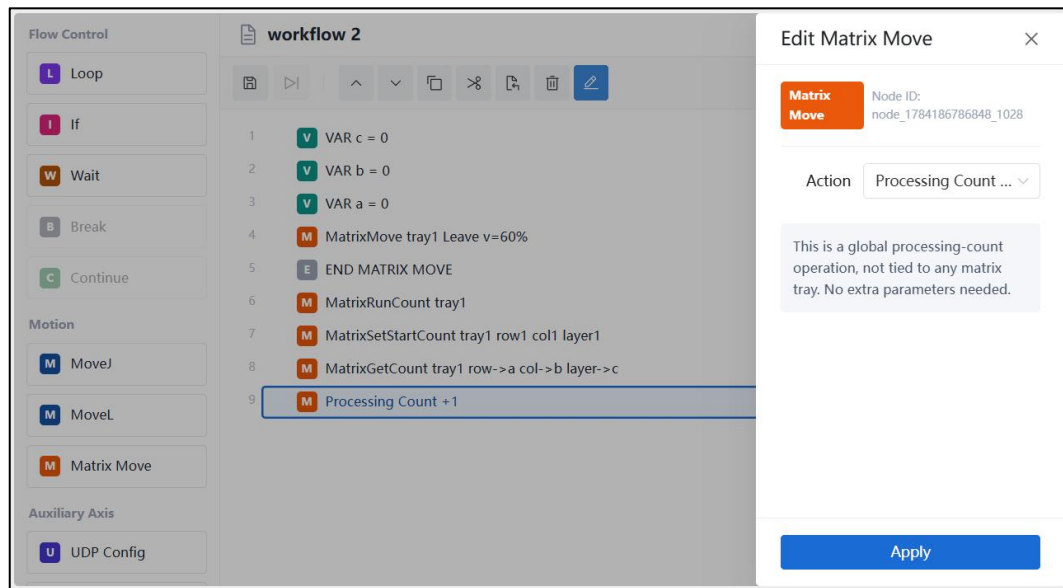


Figure 8.4.3.3-5 Total Processing Count +1

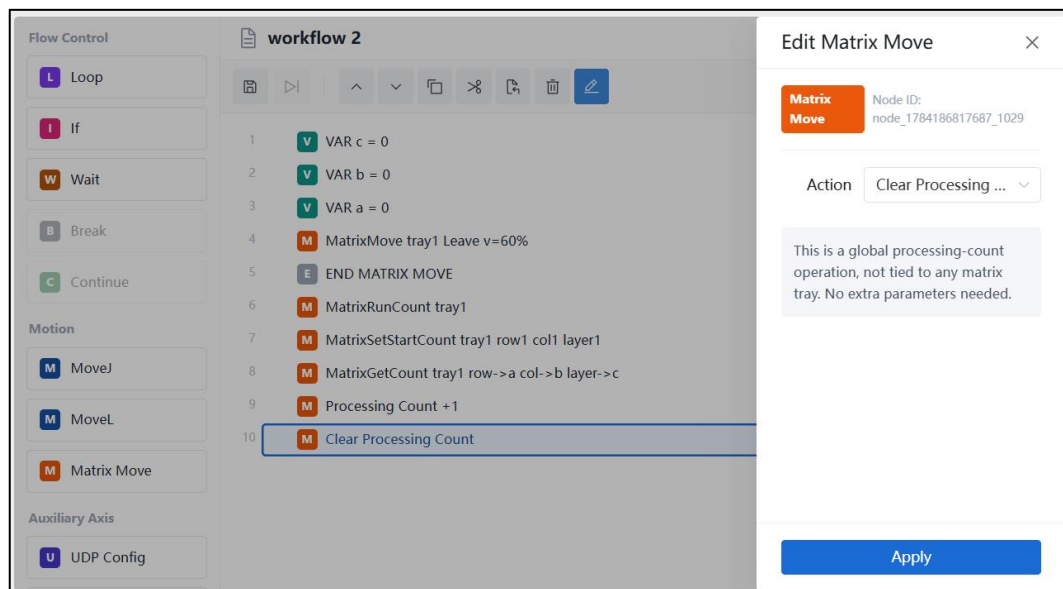


Figure 8.4.3.3-6 Total cleaned processing count

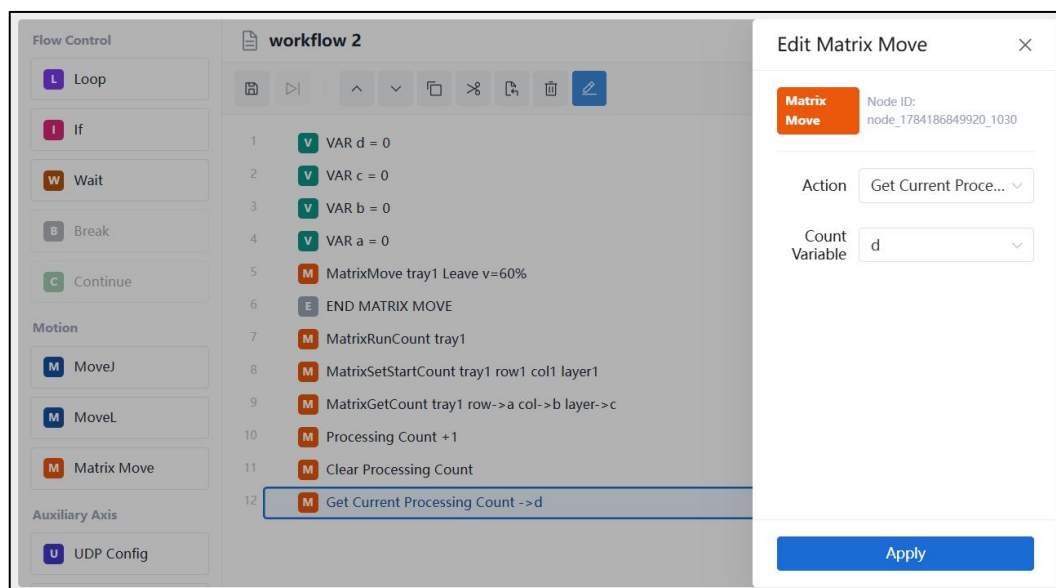


Figure 8.4.3.3-7 Get the total number of current processes

8.4.4 Advanced Instructions

8.4.4.1 Functional block

Click "Function Block" to open the "Edit Function Block" interface on the right side, where you can add, view, copy, or delete function blocks.

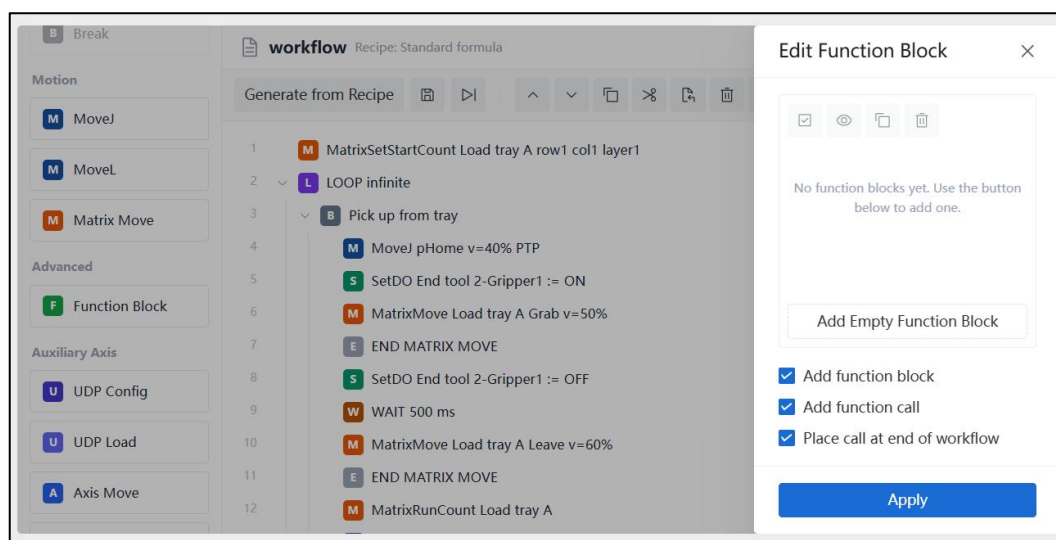


Figure 8.4.4.1-1 Function Block Interface

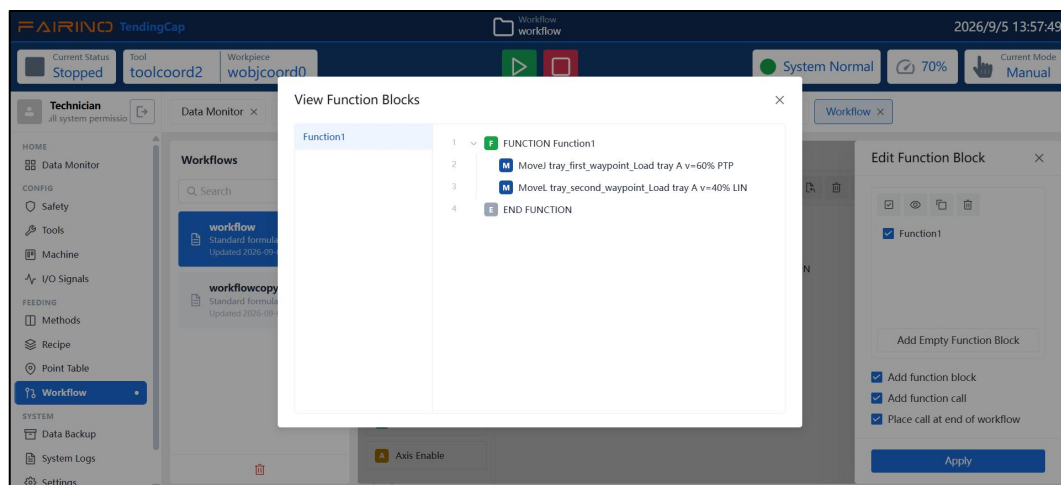


Figure 8.4.4.1-2 Functional Block Content View Interface

After adding function block list data, select the desired function block and apply the following operations: "Add Function Block," "Add Function Execution," or "Place Function Block at the End of the Workflow."

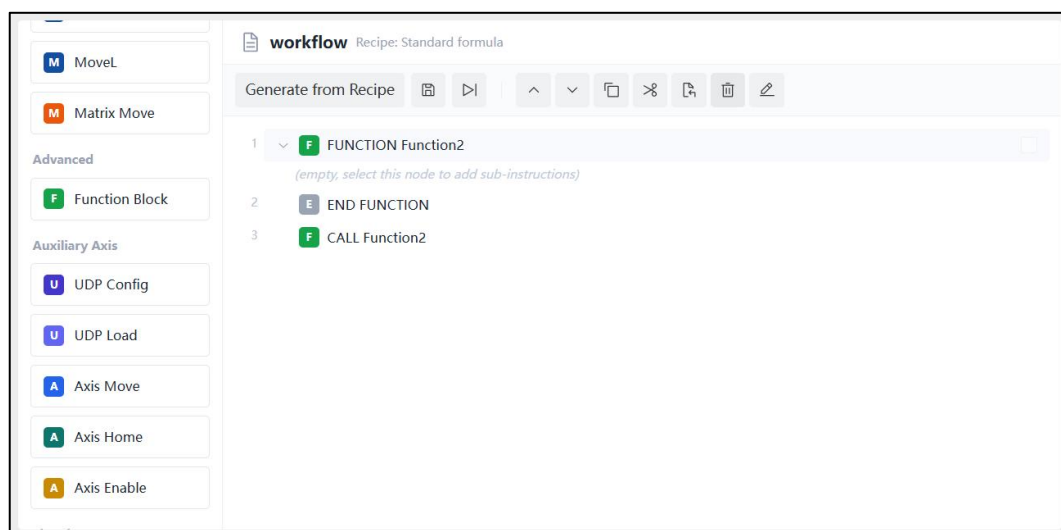


Figure 8.4.4.1-3 Functional Block Application

Add a function block to the workflow content; by checking the function block, you can add any workflow to it for service multiplexing.

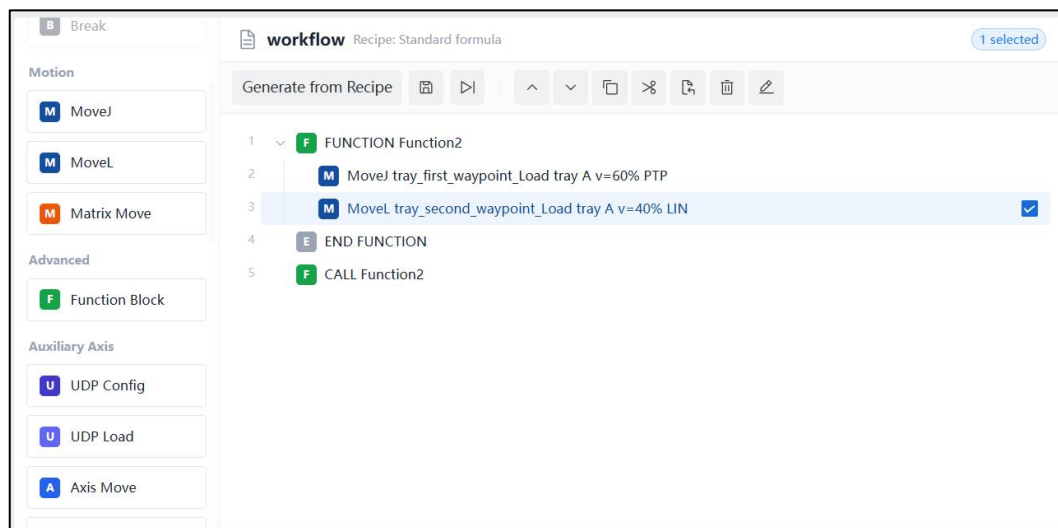


Figure 8.4.4.1-4 Function Block Content Addition

8.4.5 Auxiliary Axis

For scenarios using extended shafts, the combination mode consists of a controller and PLC (UDP).

8.4.5.1 UDP Config

Click "UDP Communication Configuration" to open the "Edit UDP Communication Configuration" interface on the right.

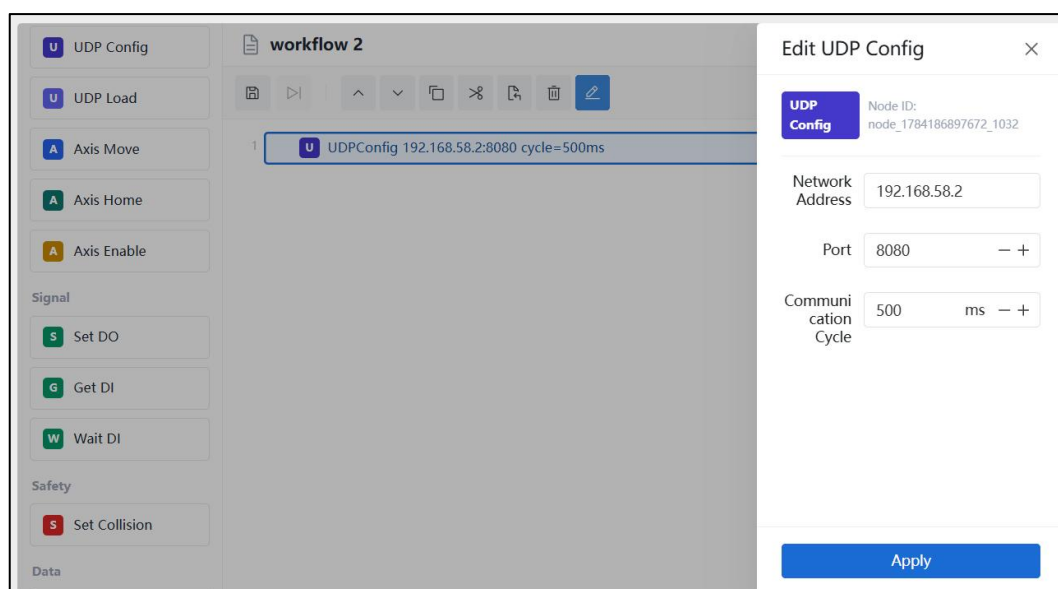


Figure 8.4.5.1-1 UDP Communication Configuration Command

8.4.5.2 UDP Load

Click "UDP Communication Load" to open the "Edit UDP Communication Load" interface on the right.

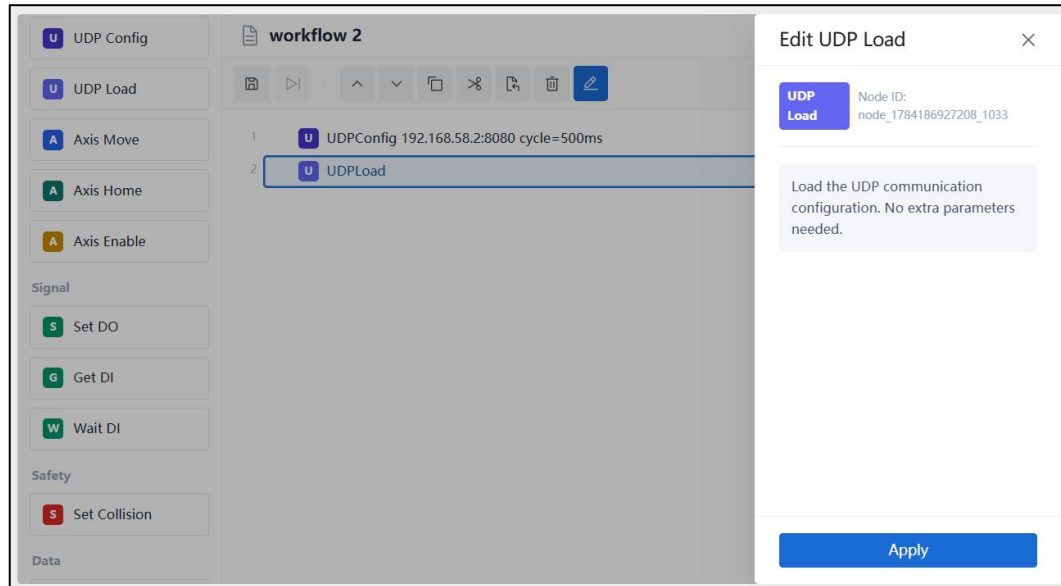


Figure 8.4.5.2-1UDP Communication Loading Instruction

8.4.5.3 Axis Move

Click "Extend Axis Movement" to open the "Edit Extend Axis Movement" interface on the right.

When used with a point-to-point command, it breaks down spatial movement along the X-axis into movements along external axes. Select "Synchronous" as the motion mode and choose the destination point.

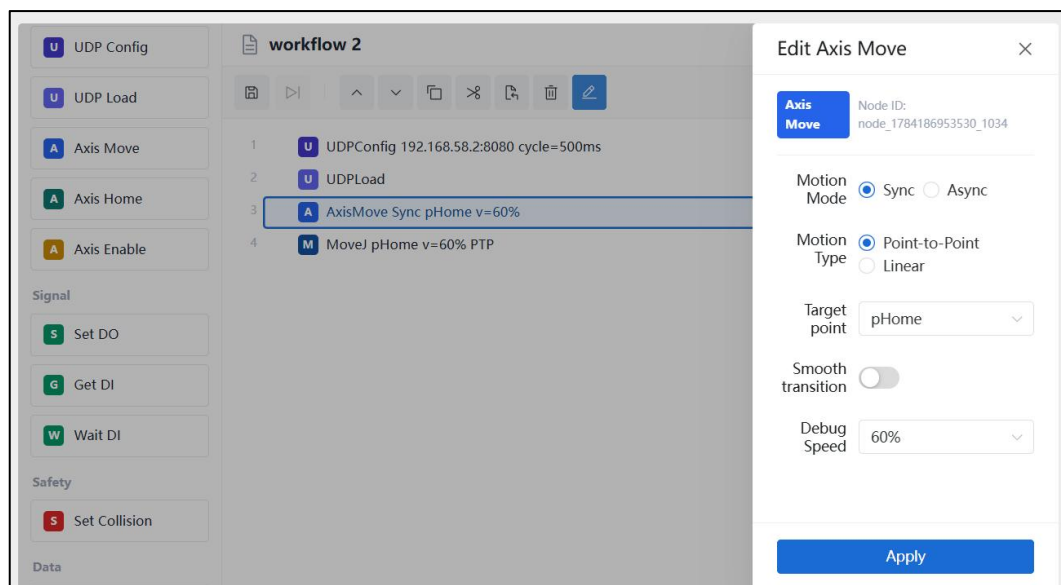


Figure 8.4.5.3-1 Extension Axis Movement Command

8.4.5.4 Axis Home

Click "Set Axis Zero" to open the Edit Axis Zero interface on the right.

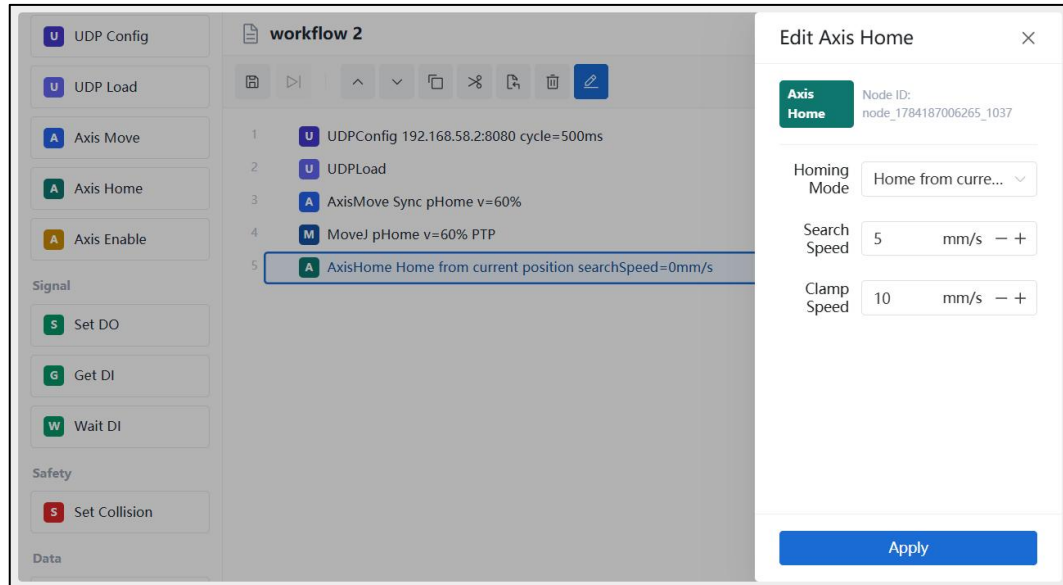


Figure 8.4.5.4-1 Extension Axis Zero Command

8.4.5.5 Axis Enable

Click "Enable Expansion Axis" to open the "Edit Enable Expansion Axis" interface on the right.

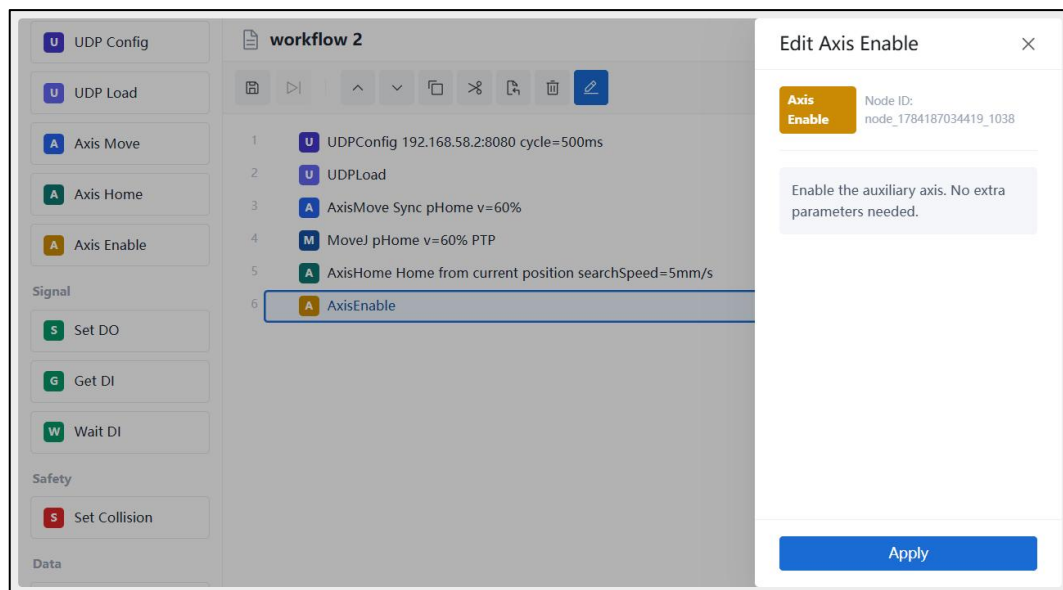


Figure 8.4.5.5-1 Extension Axis Enable Command

8.4.6 Signal

8.4.6.1 Set DO

Click the Control Signal Switch to open the Edit Control Signal Switch interface on the right. Select the signal port and signal value to turn it on/off. Select signal type, signal port, and signal ON/OFF value. Supported signal types: control box, terminal, and extended PLC.

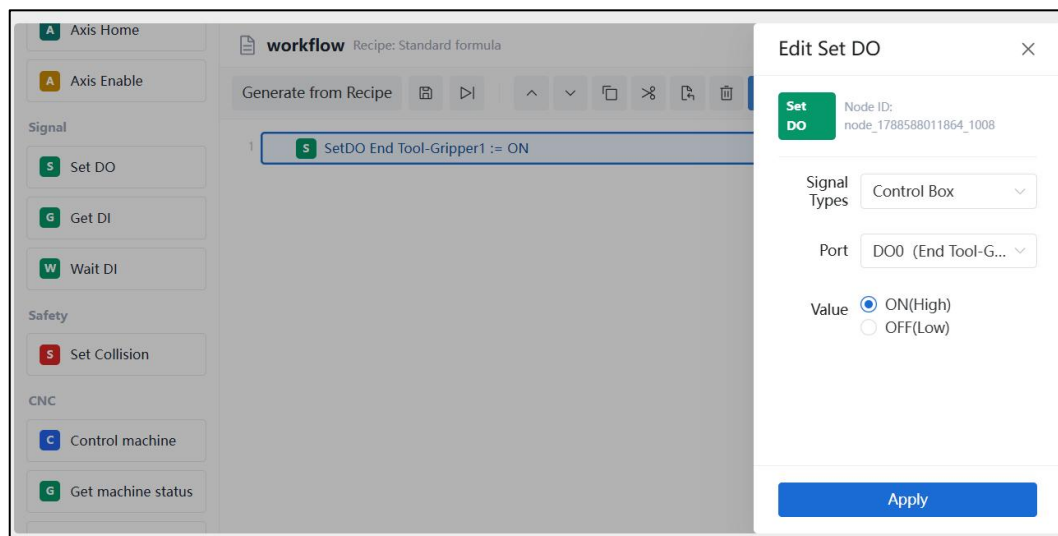


Figure 8.4.6.1-1 Control Signal Switching Instruction

8.4.6.2 Get DI

Click "Get Signal Switch" to open the "Edit Get Signal Switch" interface on the right. Select the signal port and input variable value. Select signal type, signal port, and signal ON/OFF value. Supported signal types: control box, terminal, and extended PLC.

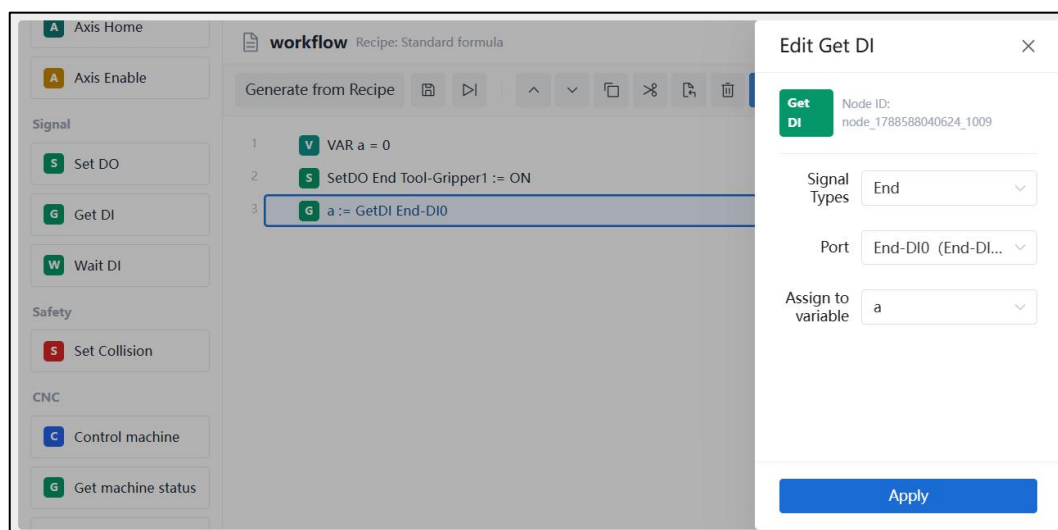


Figure 8.4.6.2-1 Obtain the signal switching instruction

8.4.6.3 Wait DI

Click "Wait Signal" to open the Edit Wait Signal interface on the right. Select a signal port, choose whether to enable or disable the target value, and select the timeout handling option (stop error reporting, wait for execution, or continue waiting). For stop error reporting and wait for execution, enter a timeout duration. Select signal type and signal port; choose the desired ON/OFF state; select the wait timeout handling method (stop with error, wait for execution, or continuously wait); for "Stop with Error" and "Wait for Execution" modes, specify an input timeout duration. Supported signal types: Control Box, End-Point PLC, and Extended PLC.

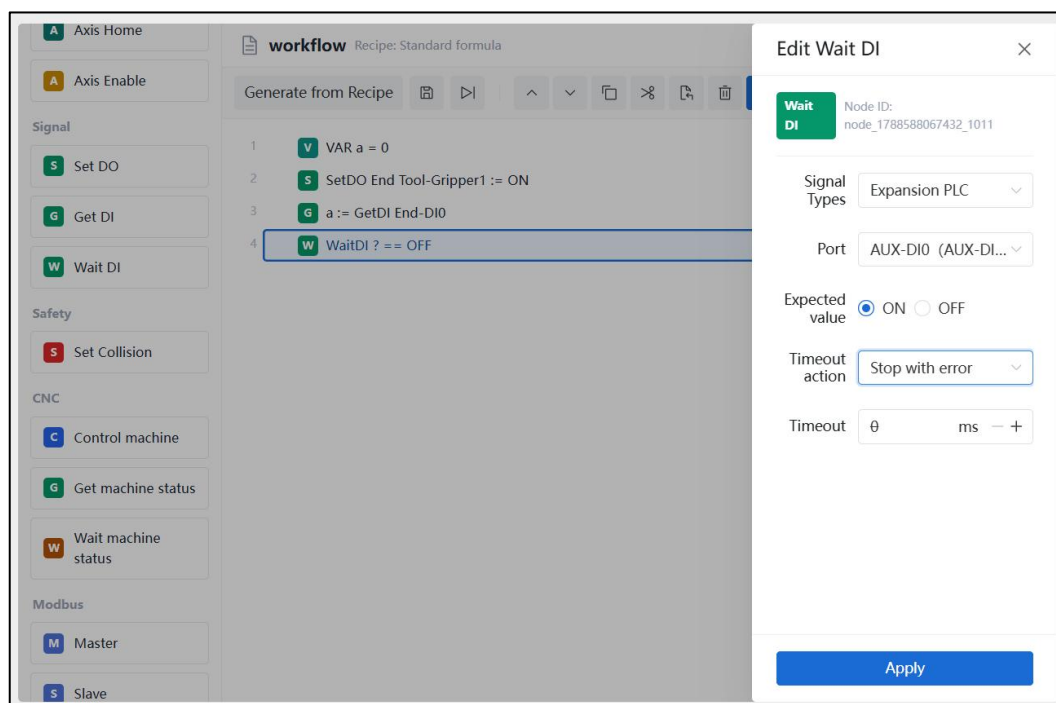


Figure 8.4.6.3-1 Waiting for the signal instruction

8.4.7 Safety

8.4.7.1 Set Collision

Click "Set Collision Level" to open the "Set Collision Level" interface on the right. This command allows you to adjust collision levels for each axis in real time during program execution, providing more flexible deployment options.

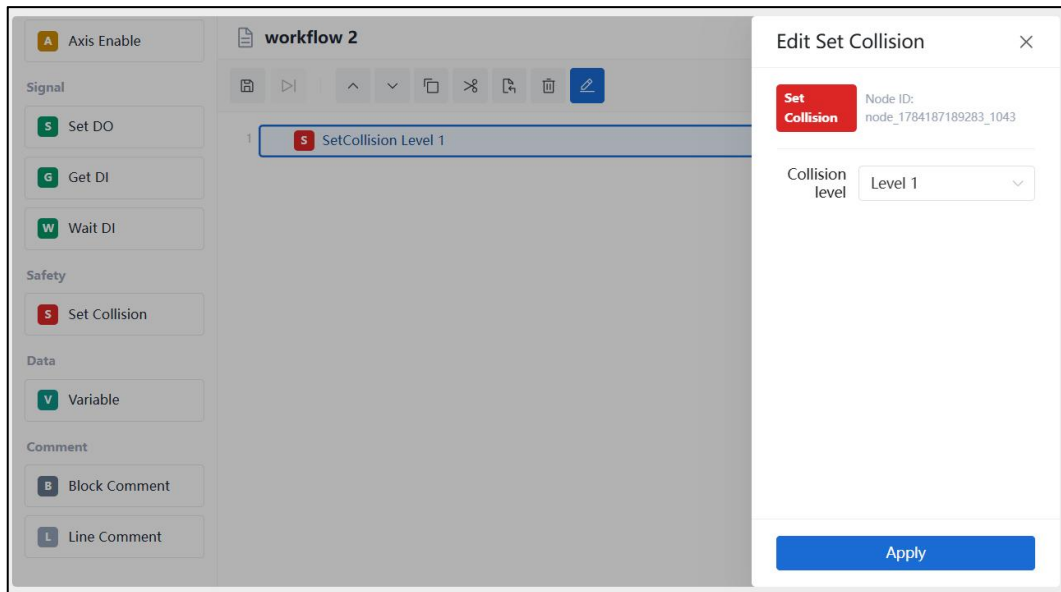


Figure 8.4.7.1-1 Set Collision Level Directive

8.4.8 Machine

8.4.8.1 Control Machine

Click "Control Machine Tool" to open the "Edit Control Machine Tool" interface on the right. Different execution action options will automatically match based on the selected machine tool communication method. The available actions are as follows:

- Processing Operation (Hide when the machine system is of the SYNTEC)
- Stop processing (Hide when the machine system is of the SYNTEC)
- Open the Door
- Close the door
- Clamp chuck tightened
- Release the chuck
- Emergency stop effective
- Stop and Exit
- Robot entry (Display when the machine system is of the SYNTEC)
- Shipment completed (Display when the machine system is of the SYNTEC)

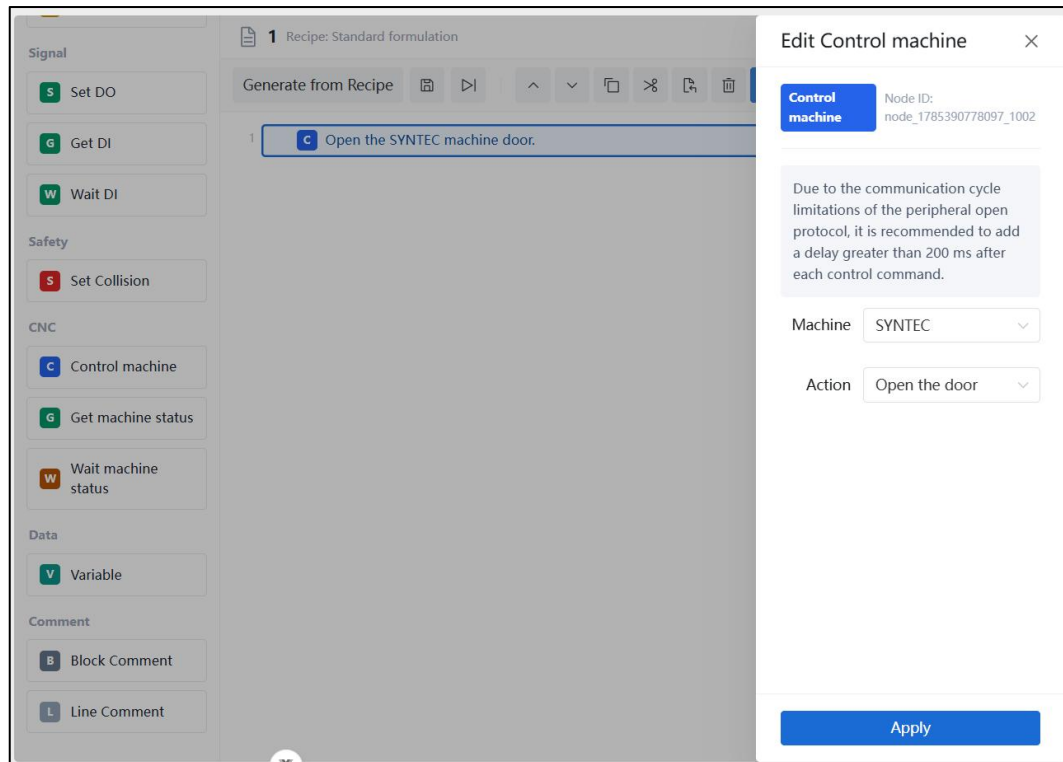


Figure 8.4.8.1-1 Machine – Controlling Machine Instructions

8.4.8.2 Get Machine Tool Status

Click "Get Machine Tool Status" to open the "Edit Get Machine Tool Status" interface on the right. Different action options will automatically appear based on your selected machine tool communication method. Enter the name of the variable to assign the retrieved machine tool status. The actions are as follows:

- Get the current status of the machine tool
- Get Operation Status
- Get Access Status
- Get Disk Status
- Get emergency stop status
- Get Alarm Status
- Get the delivery request status (displayed when the machine tool system is of the SYNTEC and uses a SYNTEC protocol)
- Get the material delivery positioning status (displayed when the machine tool system is of the SYNTEC and uses a SYNTEC protocol)

- Get the security screen status (displayed when the machine tool system is of the SYNTEC and uses a SYNTEC protocol)

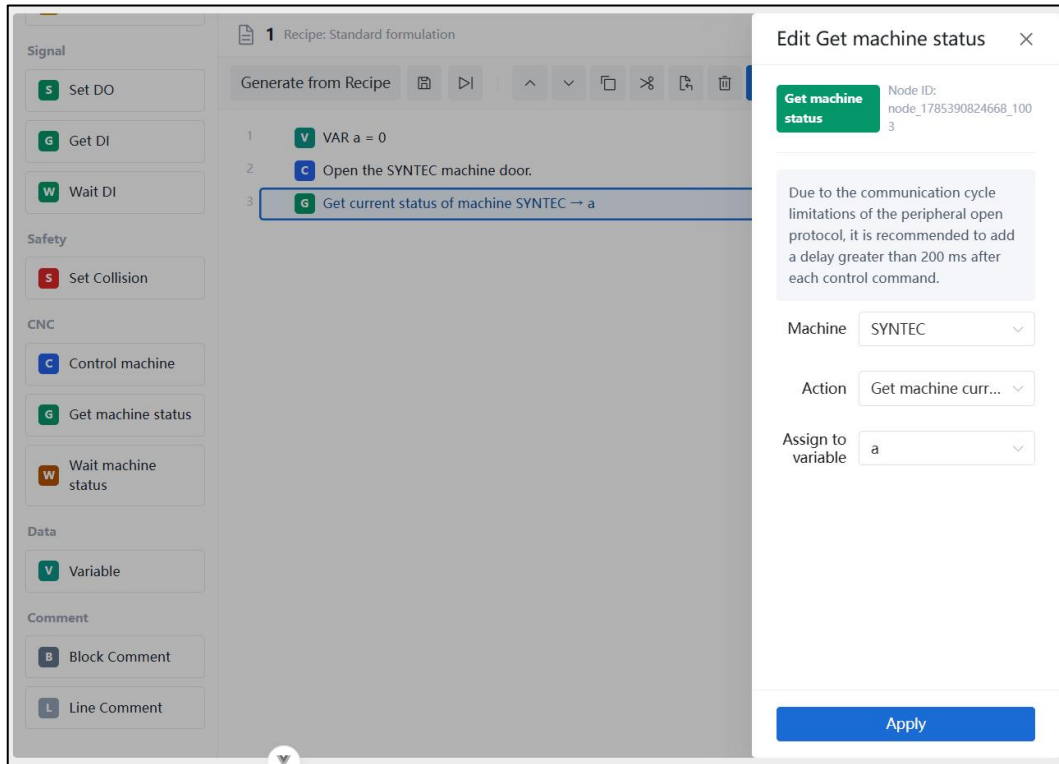


Figure 8.4.8.2-1 Machine – Obtain Machine Status Instructions

8.4.8.3 Waiting for machine status

Click "Wait for Machine Tool Status" to open the "Edit Wait for Machine Tool Status" interface on the right. Different execution action options will automatically appear based on the selected machine tool communication method. Select the target status for each execution action. For timeout scenarios, choose either "Wait Forever" or "Timeout Time"; selecting "Timeout Time" will display an input field. Timeout strategies include "Stop with Error" or "Continue Operation". The available execution actions are as follows:

- Machine operation status: Select stop/machining status.
- Machine tool chuck status: Select loose/tightened state.
- Wait for feeding request status (displayed when the machine tool system is of the SYNTEC and uses a SYNTEC protocol): Select failure/success status of the feeding request.

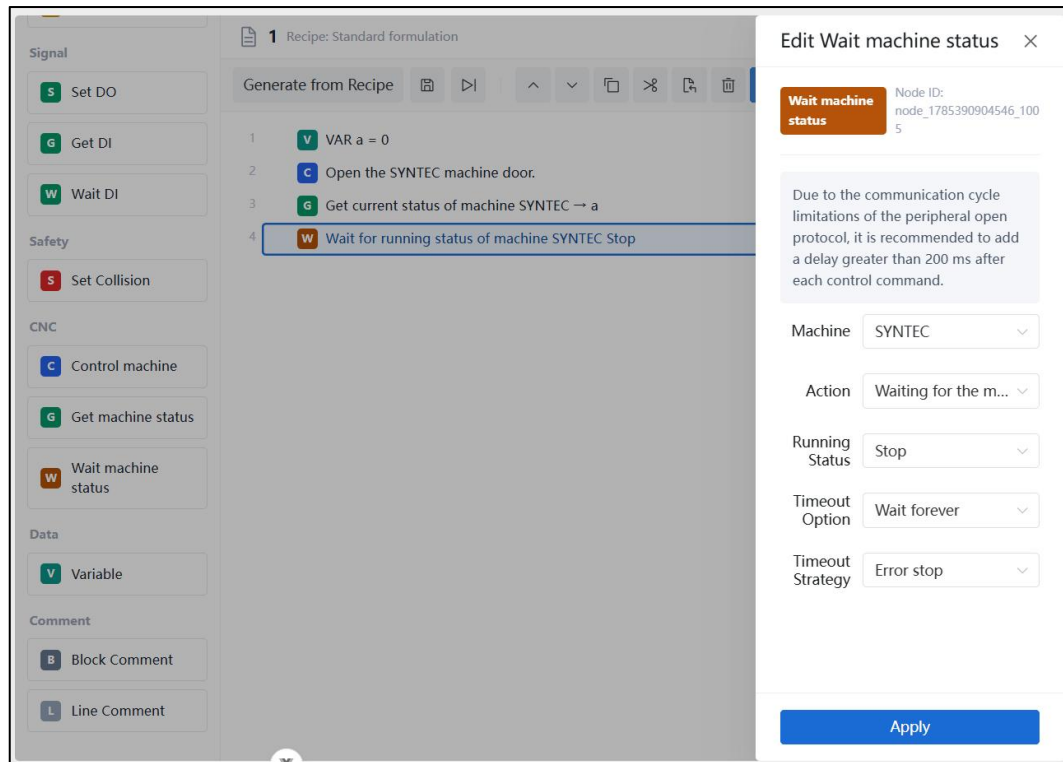


Figure 8.4.8.3-1 Machine – Waiting for a machine status command

8.4.9 Modbus

Click "Modbus" to open the "Edit Modbus" interface on the right side. This command function provides a bus-based functionality based on the Modbus TCP protocol; it uses relevant commands to control the communication between the robot and a Modbus TCP client or server (master – slave communication), enabling read/write operations on coils, discrete quantities, and registers. Detailed instructions for configuring Modbus TCP can be found under "System Settings – Advanced Parameters."

8.4.9.1 Master station

After selecting the main station and the execution action, the corresponding command content will be displayed. The main station's execution actions are as follows:

- Read digital output: Select a register name, specify the number of input registers, and enter the variable value.
- Write digital output: Select a register name, enter the number of registers, and specify the register values.

- Read digital input: Select a register name, enter the number of registers, and specify the variable to assign values to.
- Read analog output: Select a register name, specify the number of input registers, and enter the variable value.
- Write analog output: Select a register name, specify the number of input registers, and enter the register values.
- Read analog input: Select a register name, specify the number of input registers, and enter the variable value.
- Wait for digital input: Select the register name and wait state, then enter the timeout duration.
- Wait for digital output: Select the register name and wait state, then enter the register value and timeout duration.

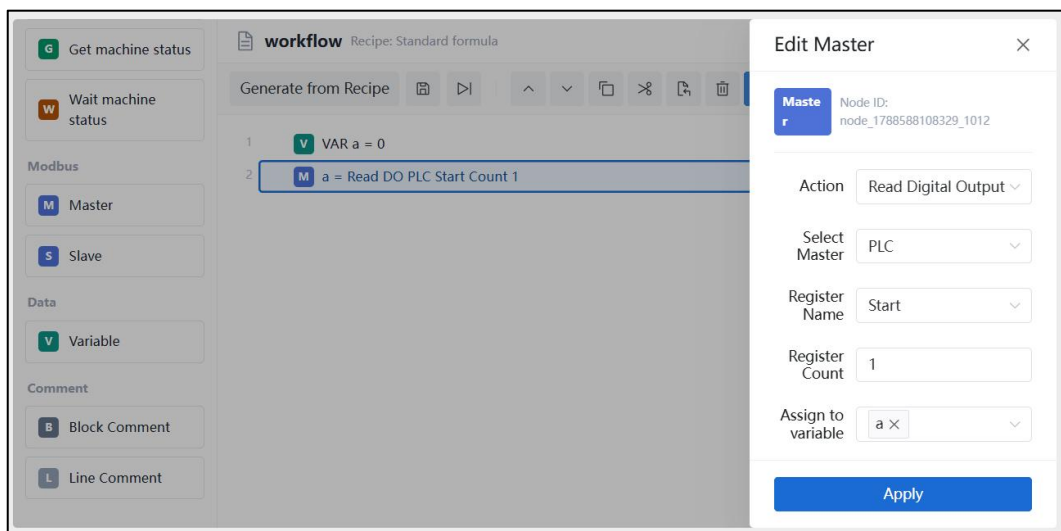


Figure 8.4.9.1-1 Modbus TCP Master Station Command

8.4.9.2 Slave station

After selecting an execution action, the corresponding instruction content is displayed. The slave station executes the following action:

- Read digital output: Select a register name, specify the number of input registers, and enter the variable value.
- Write digital output: Select a register name, enter the number of registers, and specify the register values.

- Read digital input: Select a register name, enter the number of registers, and specify the variable to assign values to.
- Read analog output: Select a register name, specify the number of input registers, and enter the variable value.
- Write analog output: Select a register name, specify the number of input registers, and enter the register values.
- Read analog input: Select a register name, specify the number of input registers, and enter the variable value.
- Wait for digital input: Select the register name and wait state, then enter the timeout duration.
- Wait for digital output: Select the register name and wait state, then enter the register value and timeout duration.

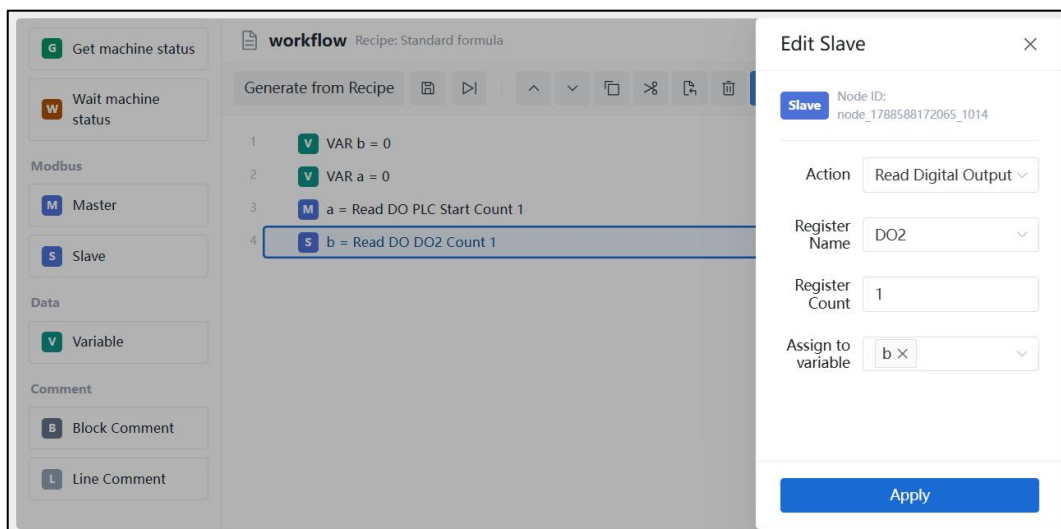


Figure 8.4.9.2-1 Modbus TCP Slave Device Command

8.4.10 Data

8.4.10.1 Variable

Click Variable to open the Edit Variable interface on the right.

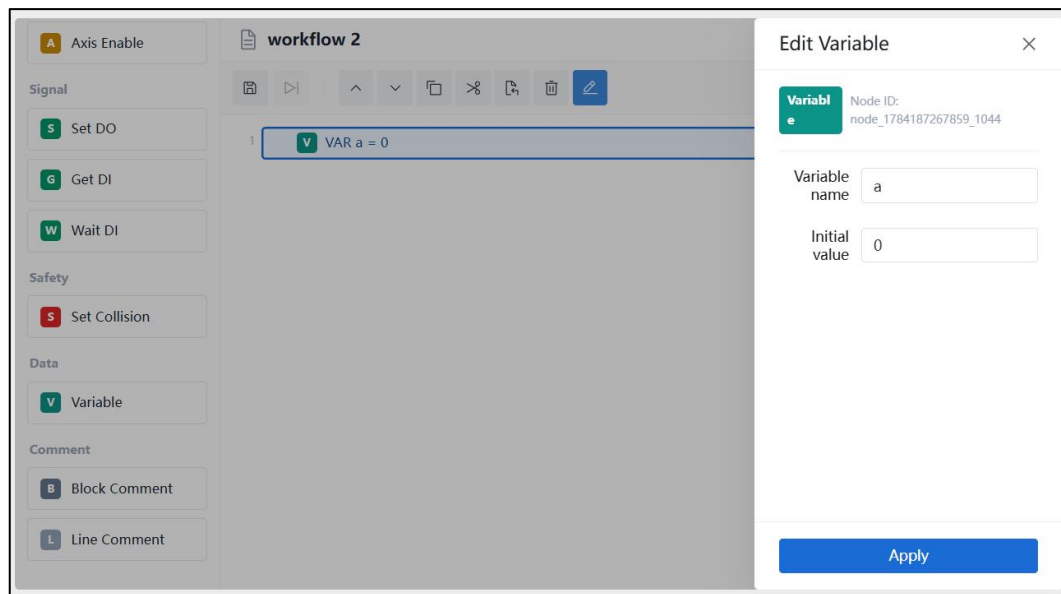


Figure 8.4.10.1-1 Variable Instruction

8.4.11 Comment

8.4.11.1 Block Comment

Click "Block Comment" to open the "Edit Block Comment" interface on the right. After entering the annotation content, add instructions below it to facilitate understanding of the workflow content later.

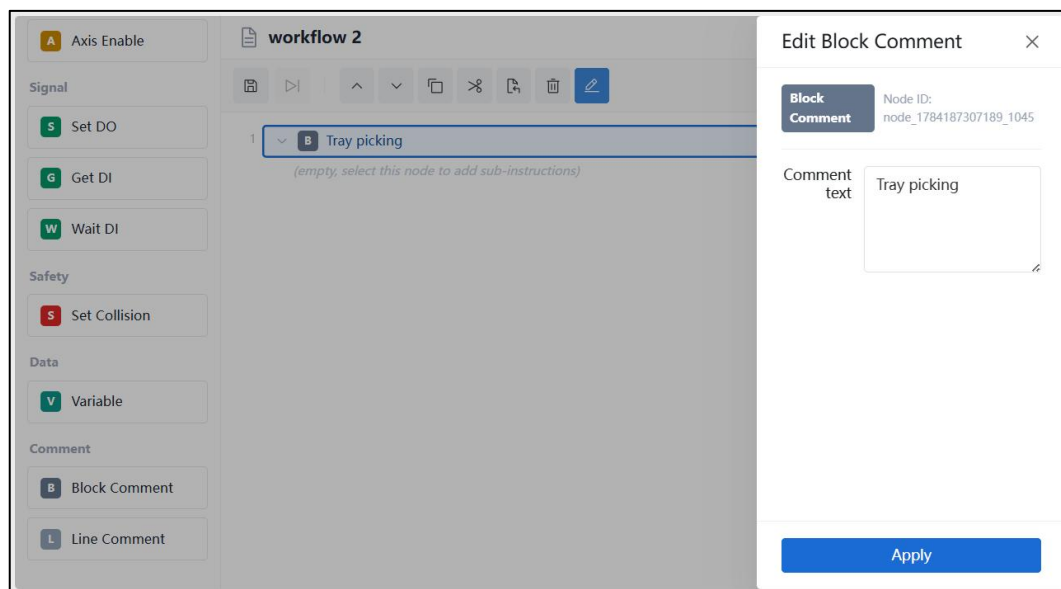


Figure 8.4.11.1-1 Block Comment

8.4.11.2 Line Comment

Click "Line Comment" to open the "Edit Line Comment" interface on the right.

After entering the commentary content, add instructions after this row commentary to facilitate understanding of the workflow content later.

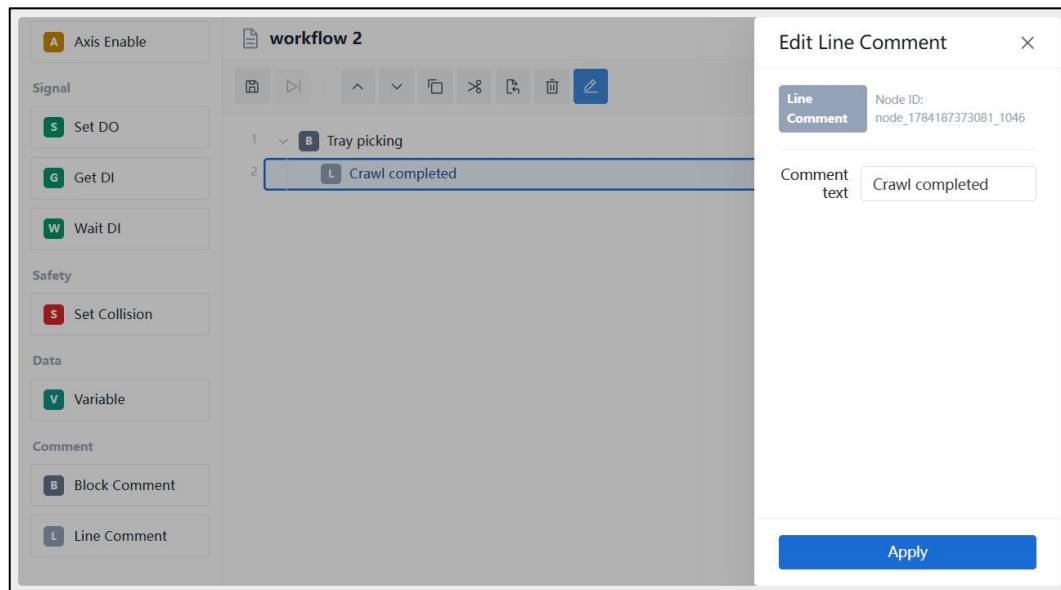


Figure 8.4.11.2-1 Line Comment

9 System

9.1 Data backup

Import and export backup packages containing loading/unloading formulas, points, workflows, and coordinate systems in System – Data Backup. Export is defaulted to.tcbk format, and imports only support this format.

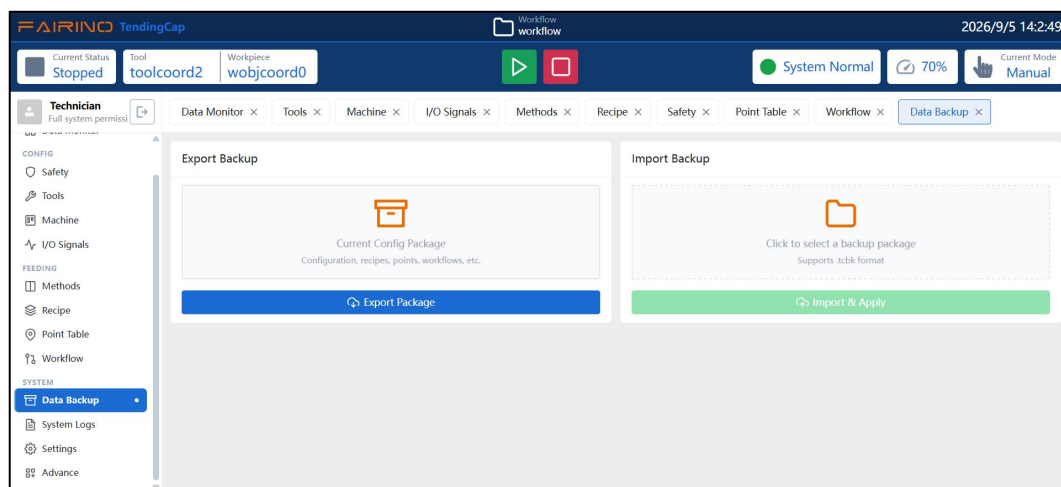


Figure 9.1-1 Data backup

9.2 System log

In "System – System Logs", view logs based on the returned date list; by default, all types are displayed. Current types include: Operations, Warnings, and Errors. Enter keywords in the search box on the right to filter results.

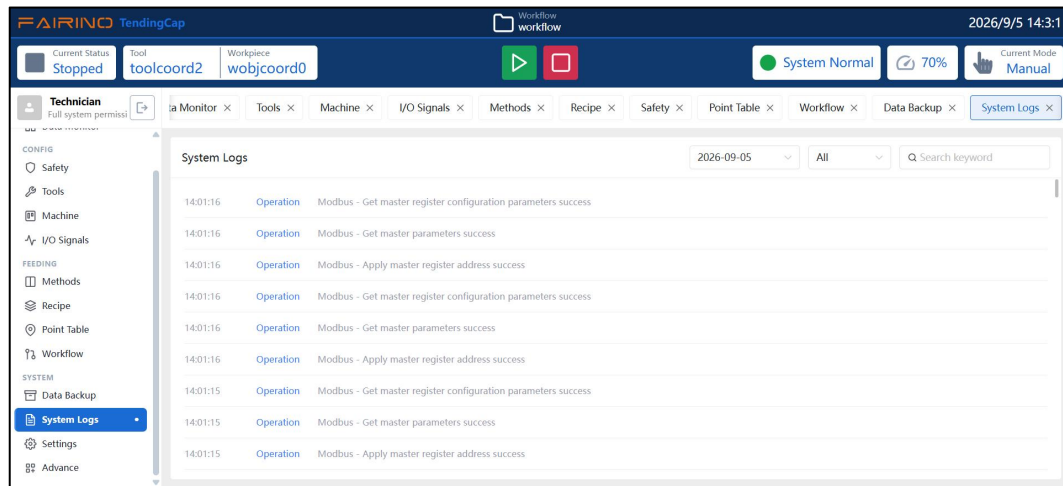


Figure 9.2-1 System log

9.3 System Settings

In "System – System Settings," you can find language switching (Chinese/English), access permission control, and one-click update of teaching points.

- Language Switch: Currently supports Simplified Chinese and English.
- Access control: Use the slider button to display and configure whether password protection is enabled for technicians accessing the page. When enabled, technicians can change their passwords and require two-factor authentication.
- One-click update of demonstration points: Update location data based on Joint/TCF.
- Extended-axis motion configuration: Extended-axis motion – either primarily driven by the robot or synchronized between the robot and the extended axis.

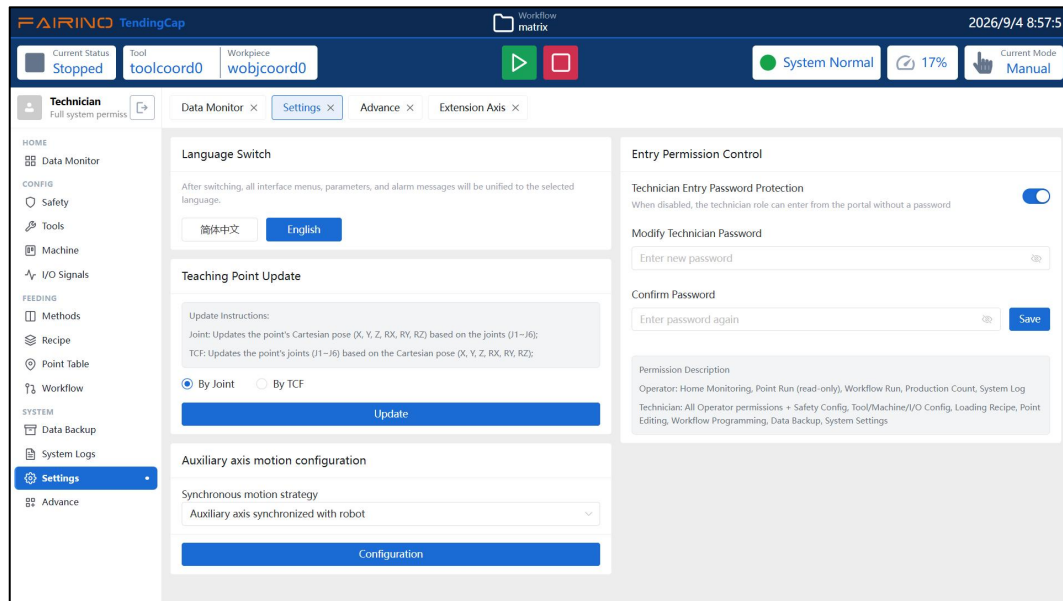


Figure 9.3-1 System Settings

9.4 Advanced Parameters

9.4.1 Configurable signal

In the "System – Advanced Parameters" section, click the "Enter" button at the bottom of the "Configurable Signals" tab. The left pane displays the "Input" signal content; use this section to configure the functions and active states of the control box's CI0 – CI7 pins, the active states of the control box's DI0 – DI7 pins, and the functions and active states of the end-point DI0 – DI1 pins. The right pane displays the "Output" signal content; use this section to configure the functions and active states of the control box's CO0 – CO7 pins, the active states of the control box's DO0 – DO7 pins, and the DO output settings for the control box during power-on.

9.4.1.1 Input

9.4.1.1.1 CI0 ~ CI7

Select the CI0 – CI7 function and click the "Apply" button at the bottom to apply the configurable function. For the list of configurable function options, refer to the table "Configurable Functions for Control Box Input CI0 – CI7." As an example, for configuring the effective state of CI0 as "Manual/Automatic Switching (Pulse Signal):" if you select "High-Level Valid," the "Manual/Automatic Switching (Pulse Signal)" function will be triggered whenever the CI0 level changes from low to high, causing

the robot to switch its operating mode once; if you select "Low-Level Valid" for the configurable input effective state, the "Manual/Automatic Switching (Pulse Signal)" function will be triggered whenever the CI0 level changes from high to low, causing the robot to switch its operating mode once.

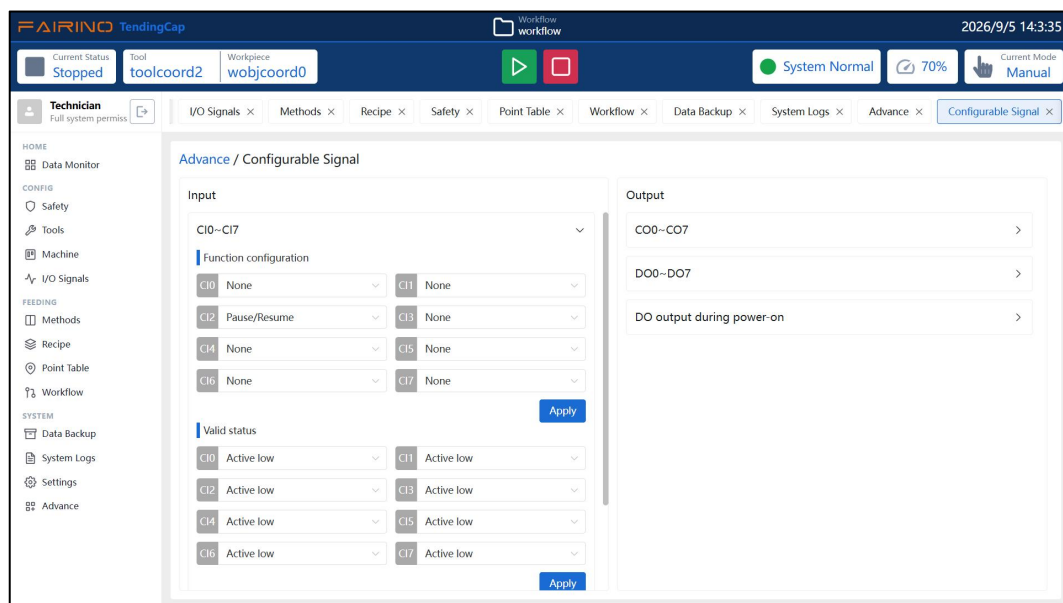


Figure 9.4.1.1-1 The control box's inputs CI0 – CI7 can be configured for various functions and valid status settings.

Table 9.4.1.1-1 Control box input CI0 – CI7 – Configurable Functions

Function Code	Function Name	Functional Description
0	Nothing	Nothing
4	Pause	Robot welding process – Motion Pause Signal
5	Recover	A welding interruption is triggered when the arc is unexpectedly interrupted during the robotic welding process or when the operator manually pauses the welding operation. Upon this interruption, when the external signal sent to the robot changes from "invalid" to "valid," the robot automatically resumes welding from its previous interrupted position.
6	Start-up	If the configurable input effective state is set to "High-Level Valid," then when the

		CI0 level changes from low to high, the "Start" function is triggered; this will initiate the program currently open in the teaching program interface, or, if no interface is open, it will execute the last saved program. If the configurable input effective state is set to "Low-Level Valid," then when the CI0 level changes from high to low, the "Start" function is triggered; this will initiate the program currently open in the teaching program interface, or, if no interface is open, it will execute the last saved program.
7	Stop	Robot welding process stop motion signal
8	Pause/Resume	After the robot completes its movement, the cycle triggers a pause/continue movement signal.
9	Start/Stop	After the robot completes its movement, the cycle triggers a start/stop motion signal.
11	Move to origin	Using the current robot pose as the operation origin, the robot moves to the operation origin signal.
12	Manual/Automatic Switching (Pulse Signal)	If the input effective state is configured as "High-Level Valid," then when the CI0 level changes from low to high, the "Manual/Automatic Switching (Pulse Signal)" function is triggered, causing the robot to switch its operating mode once; if the input effective state is configured as "Low-Level Valid," then when the CI0 level changes from high to low, the

		"Manual/Automatic Switching (Pulse Signal)" function is triggered, causing the robot to switch its operating mode once.
14	Exercise interruption	Robot Motion Program Interrupt Signal
15	Run the main program	Start Robot Main Program Signal
30	Clear all errors	Clear all error signals triggered by the cleaning robot.
32	Enabled	Control Robot Enablement
33	Disable	Enable robotic control

9.4.1.1.2 DI0 ~ DI7

Select the desired state for DI0 – DI7, then click the "Apply" button at the bottom to complete the configuration.

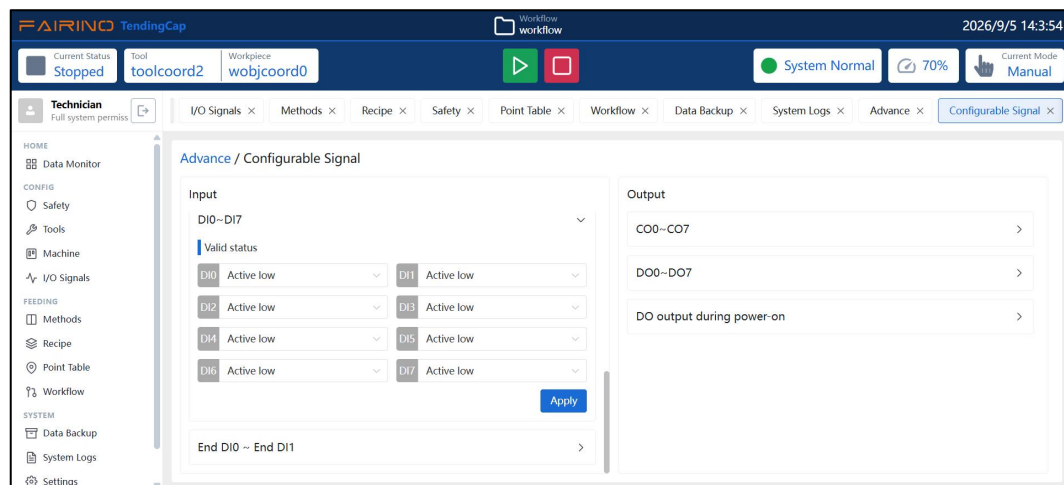


Figure 9.4.1.1-2 Control box input DI0 – DI7 (active state)

9.4.1.1.3 End DI0 ~ End DI1

Synchronizing all functions of the Robot Control Box (CI) with the End-Point CI involves establishing a control system in which the logical functions are equivalent and the physical locations are complementary. The two interfaces are fully equivalent in terms of logical functionality and can be used either in parallel or selectively, enabling

the robot control system to intelligently allocate signal paths based on the task scenario, the physical layout of the equipment, and reliability requirements.

Select the "End DI0 - End DI1" function and click the "Apply" button at the bottom to apply the configurable function. For the list of configurable function options, refer to the table titled "Configurable Functions for End DI0 - End DI1 Input on Control Box." As an example, when configuring the effective state for End DI0 as "Manual/Automatic Switching (Pulse Signal)," selecting "High-Level Valid" will trigger the "Manual/Automatic Switching (Pulse Signal)" function whenever the End DI0 level changes from low to high, causing the robot to switch its operating mode once; if the configurable input effective state is set to "Low-Level Valid," the "Manual/Automatic Switching (Pulse Signal)" function will be triggered whenever the End DI0 level changes from high to low, causing the robot to switch its operating mode once.

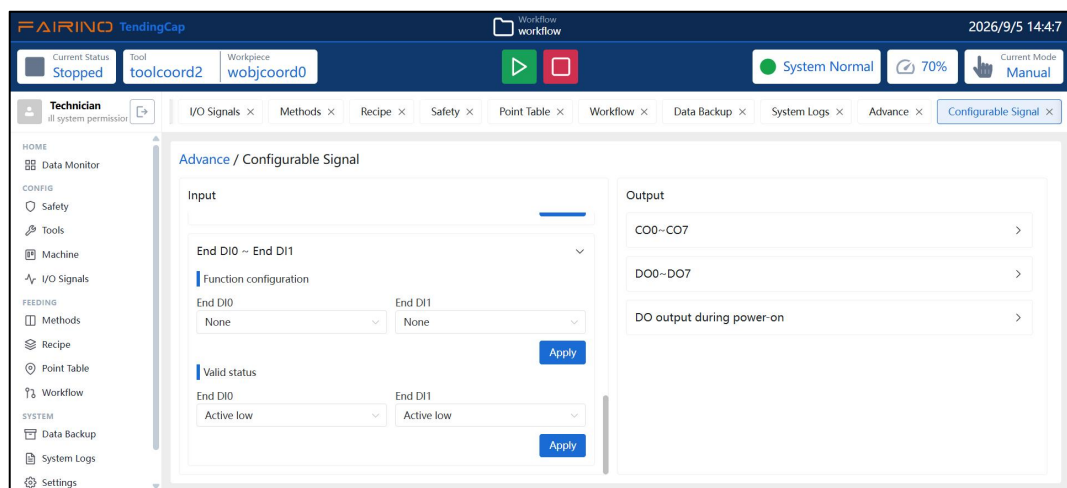


Figure 9.4.1.1-3 End DI0 - End DI1: Configurable functions and valid status

Table 9.4.1.1-2 End DI0 - End DI1: Configurable Function

Function Code	Function Name	Functional Description
0	Nothing	Nothing
3	Manual/Automatic Switching (Pulse Signal)	If the input effective state is configured as "High-Level Valid," then when the End DI0 level changes from low to high, the "Manual/Automatic Switching (Pulse Signal)" function is triggered, causing the robot to switch its operating mode once; if

		the input effective state is configured as "Low-Level Valid," then when the End DI0 level changes from high to low, the "Manual/Automatic Switching (Pulse Signal)" function is triggered, causing the robot to switch its operating mode once.
5	Pause	Robot welding process – Motion Pause Signal
6	Recover	A welding interruption is triggered when the arc is unexpectedly interrupted during the robotic welding process or when the operator manually pauses the welding operation. Upon this interruption, when the external signal sent to the robot changes from "invalid" to "valid," the robot automatically resumes welding from its previous interrupted position.
7	Start-up	If the configurable input effective state is set to "High-Level Valid," then when the End DI0 level changes from low to high, the "Start" function is triggered; this will initiate the program currently open in the teaching programming interface, or, if no interface is open, it will execute the last saved program. If the configurable input effective state is set to "Low-Level Valid," then when the End DI0 level changes from high to low, the "Start" function is triggered; this will initiate the program currently open in the teaching programming interface, or, if no interface is open, it will execute the last saved program.

8	Stop	Robot welding process stop motion signal
9	Pause/Resume	After the robot completes its movement, the cycle triggers a pause/continue movement signal.
10	Start/Stop	After the robot completes its movement, the cycle triggers a start/stop motion signal.
16	Move to origin	Using the current robot pose as the operation origin, the robot moves to the operation origin signal.
17	Exercise interruption	Robot Motion Program Interrupt Signal
18	Run the main program	Start Robot Main Program Signal
23	Clear all errors	Clear all error signals triggered by the cleaning robot.
25	Enable	Control Robot Enablement
26	Disable	Enable robotic control

9.4.1.2 Output

9.4.1.2.1 CO0 ~ CO7

Select the CO0 – CO7 function and click the "Apply" button at the bottom to apply the configurable function. For the list of configurable function options, refer to the table "Control Box – Configurable CO0 – CO7 Functions." As an example of configuring the effective state, set CO0 to "Manual/Automatic Switching": if you select "High-Level Valid," the "Manual/Automatic Switching" function will be triggered whenever the CO0 level changes from low to high, causing the robot to switch its operating mode once; if you select "Low-Level Valid" for the configurable input effective state, the "Manual/Automatic Switching" function will be triggered whenever the CO0 level changes from high to low, causing the robot to switch its operating mode once.

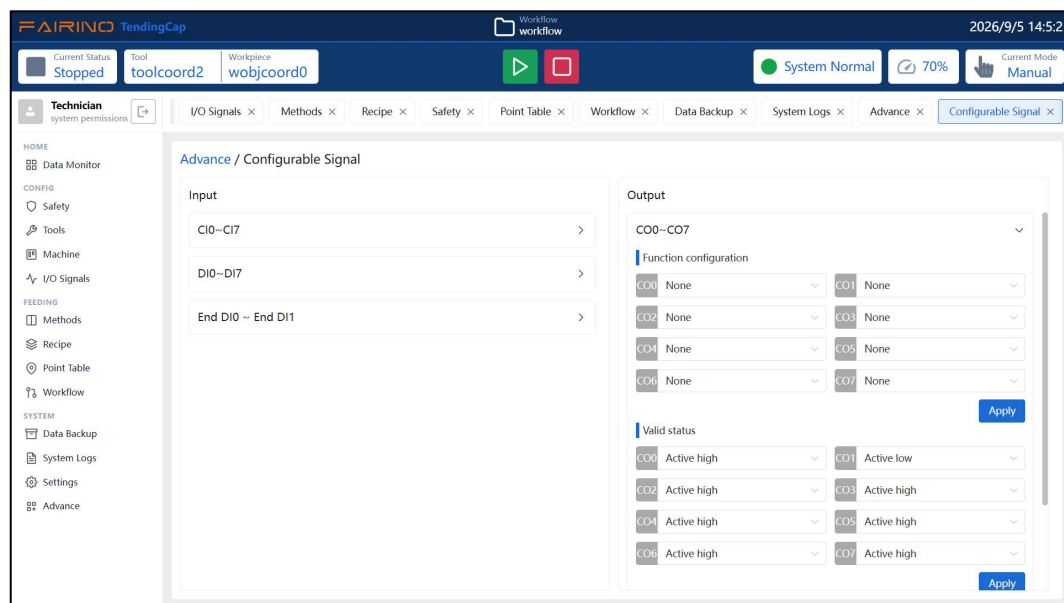


Figure 9.4.1.2-1 The Control Box's outputs CO0 – CO7 can be configured for various functions and valid status settings.

Table 9.4.1.2-1 The Control Box's Output CO0 – CO7 supports configurable functions.

Function Code	Function Name	Functional Description
0	Nothing	Nothing
1	Report an error	Output error signal
2	Movement	Robot Motion Signal
13	Pause	Robot Movement Pause Signal
14	Reach the origin point	Robot movement to operation origin signal
17	Robot startup completed	Robot Startup Complete Signal
18	Program Start/Stop	Robot Motion Program Start/Stop Signal
19	Automatic/Manual Mode	Robot Hand Automatic/Manual Mode Switching Signal
47	Robotics Enablement	Robot Enable State

9.4.1.2.2 DO0 ~ DO7

Select the desired state (DO0 – DO7) and click the "Apply" button at the bottom to complete the configuration.

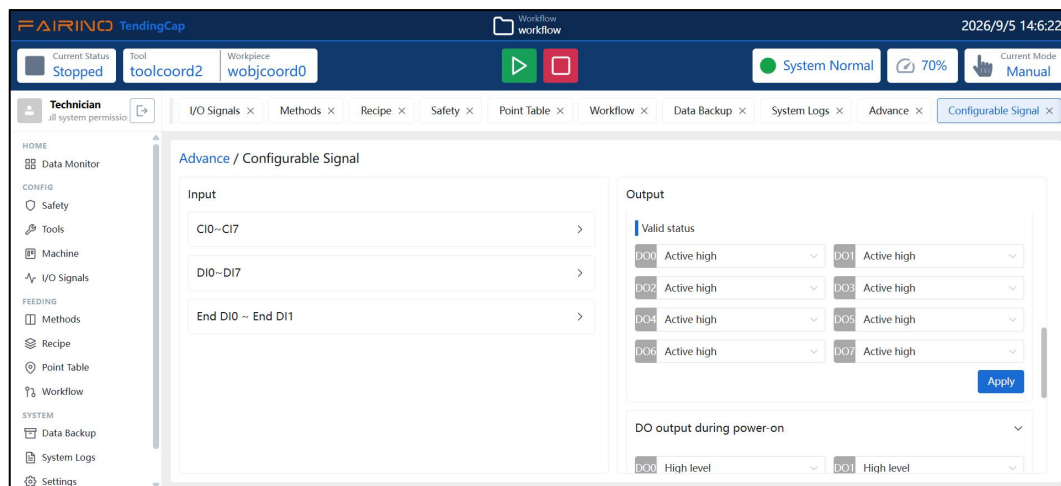


Figure 9.4.1.2-2 Control box output DO0 – DO7 (active state)

9.4.1.2.3 Power-on: Control Box DO Output

During the entire process—from powering on the control box until the robot is enabled—the DO can be configured to the desired output state based on the specific application scenario, offering greater flexibility and convenience. During power-on, select the required high/low-level output for the control box's DO; then simply click the "Apply" button.

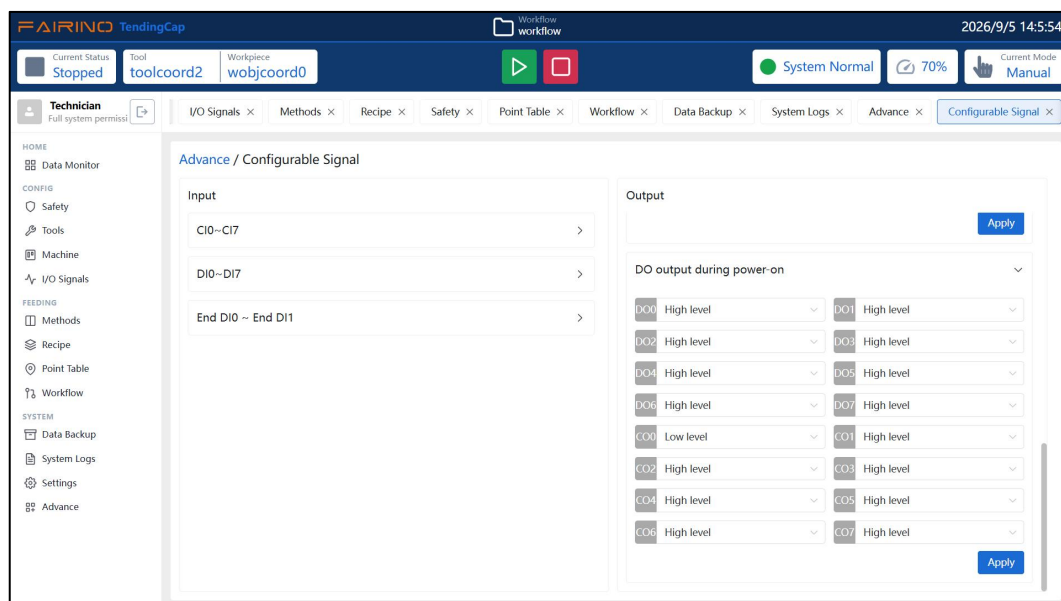


Figure 9.4.1.2-3 Power-on: Control Box DO Output

9.4.2 Extended shaft

In the Advanced Parameters section, click the "Enter" button at the bottom of the "Extended Axis" tab to open the Extended Axis Function Configuration page. Currently, this feature supports connections using only the UDP communication protocol; by default, Extended Axis No.1 is selected. The setup process consists of three steps: Extended Axis Coordinate System, UDP Communication Configuration, and Extended Axis Configuration & Testing.

9.4.2.1 Extended Axis Coordinate System

Switching the selection in the extended axis coordinate system applies the current extended axis coordinate system data; when selecting a coordinate system other than "exaxis0", click the "Calibrate" button at the bottom to calibrate the extended axis coordinate system, rename the extended axis coordinate system, or modify the extended axis numbering.

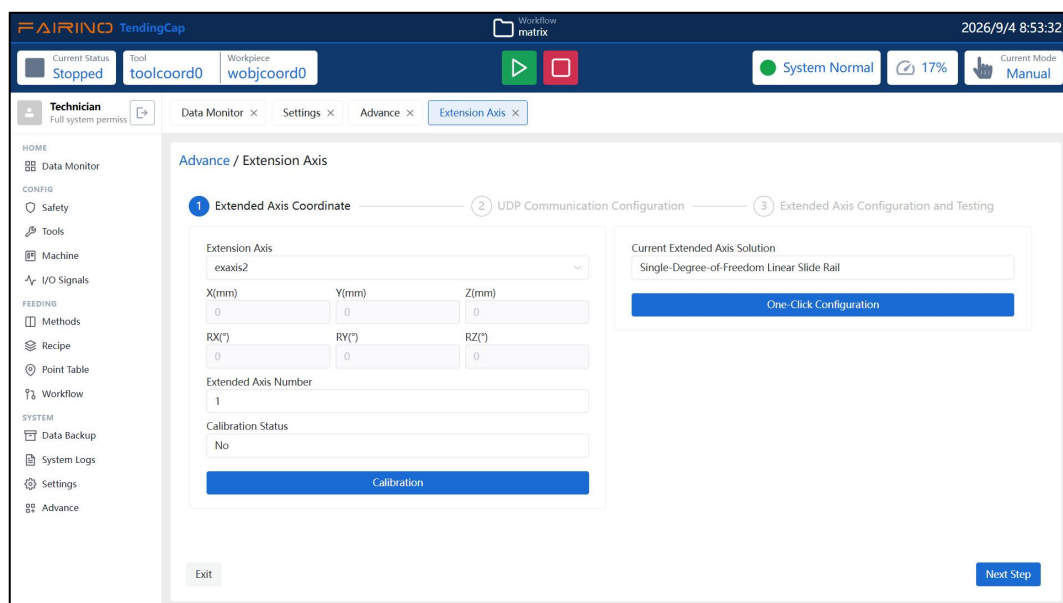


Figure 9.4.2.1-1 Extended Axis Coordinate System Interface

When the current extended axis configuration is not a "Single-Degree-of-Freedom Linear Rail," the system will display the message: "The current extended axis configuration is not a single-degree-of-freedom linear rail; click the 'One-Click Configuration' button below to configure it as a single-degree-of-freedom linear rail. After successful one-click configuration, ensure that the currently selected extended axis coordinate system ID is 1; if it is not, please navigate to the Extended Axis Coordinate System Calibration interface to modify it." At this point, clicking the "One-

Click Configuration" button will allow you to quickly configure the current extended axis configuration.

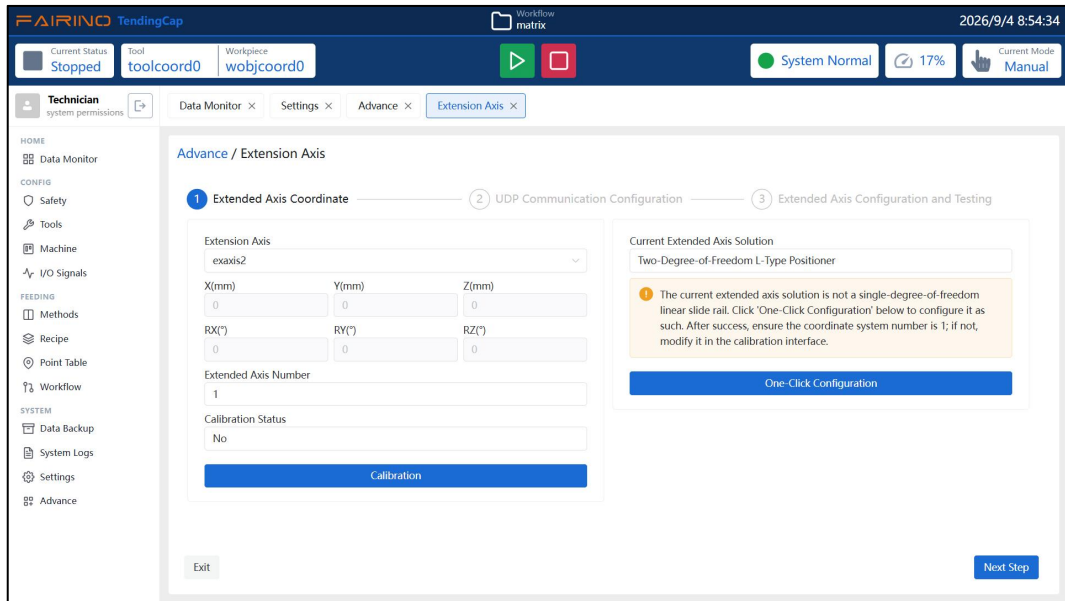


Figure 9.4.2.1-2 The current extended axis solution does not support the "Single-Degree-of-Freedom Linear Guide Rail" interface.

If the selected extended axis coordinate system ID is not 1, clicking the "Next" button will display the message: "Please configure the extended axis ID for the current extended axis coordinate system to 1."

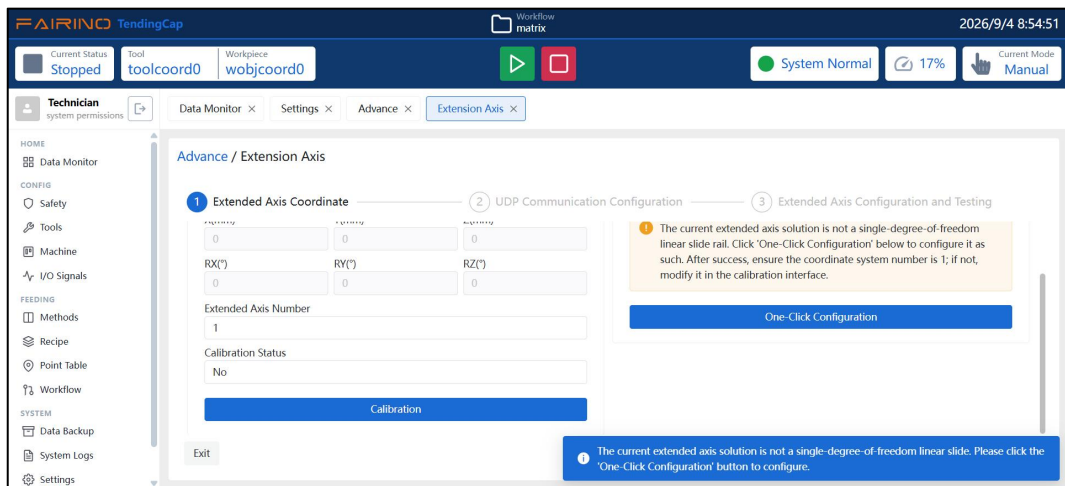


Figure 9.4.2.1-3 The extended axis coordinate system number is not 1 – Warning

Click the "Calibrate" button to access the Extended Axis Coordinate System Calibration interface. Then, click the Edit button next to the axis name to display the rename text box; enter the new name and click the Edit button again to complete the renaming. Next, click the "Extended Axis Number" multiple-selection menu, select the desired axis, and then click the "Apply" button at the bottom to modify the extended

axis number.

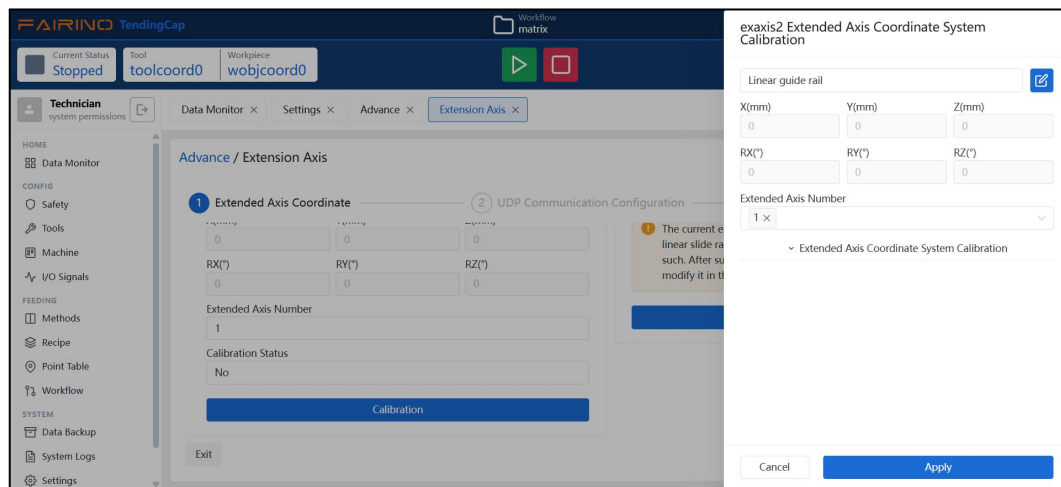


Figure 9.4.2.1-4 Extended Axis Coordinate System Renaming

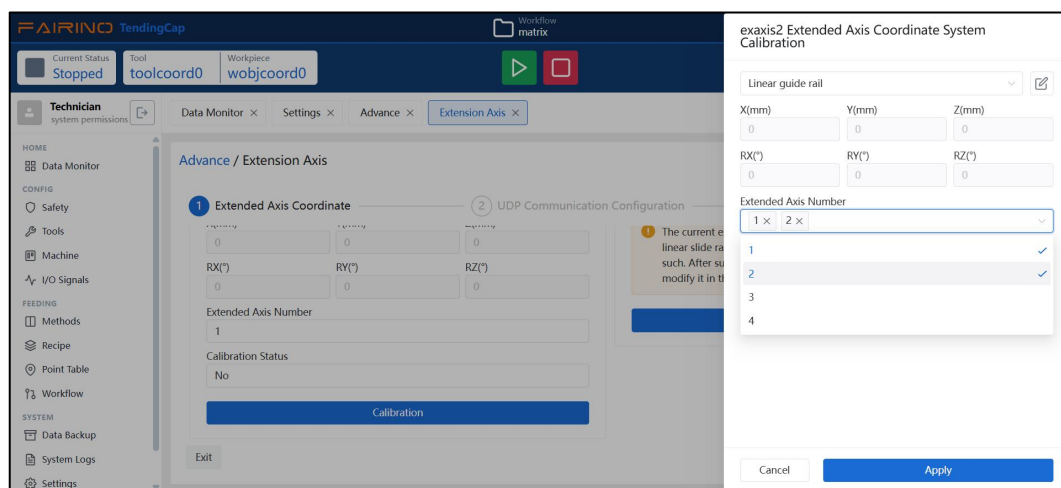


Figure 9.4.2.1-5 Extended Axis Coordinate System Number Modification

Before calibrating the extended-axis coordinate system, configure the extended-axis motion strategy in "System – System Settings" to "Extended Axis Synchronized with Robot". At the extended-axis zero point, align the robot's end-effector center (or the tool endpoint in the Application Tool Coordinate System) with a fixed point on the extended axis in two different poses; designate these as Point 1 and Point 2, respectively.

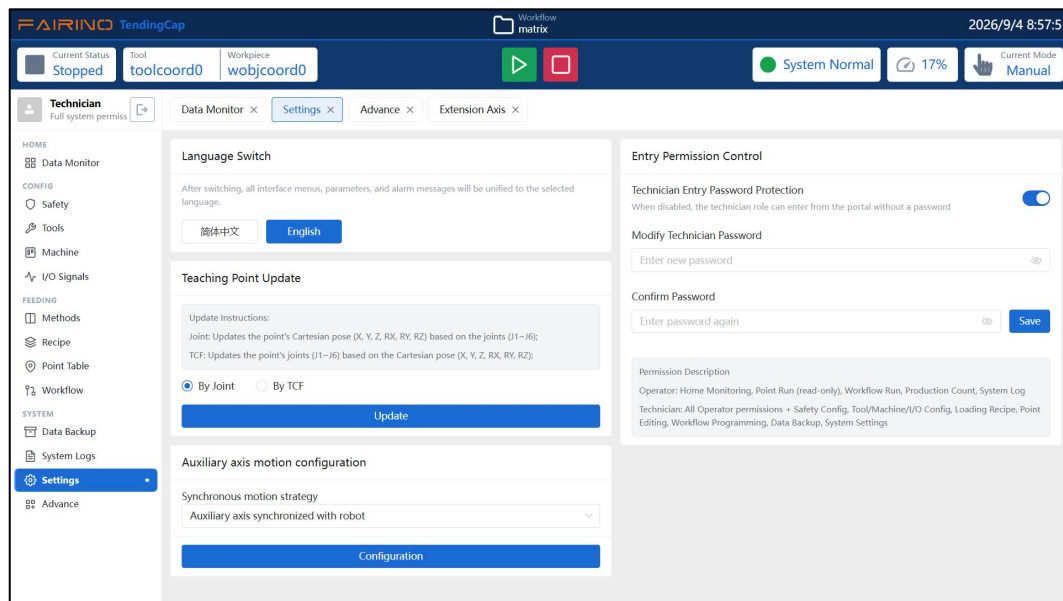
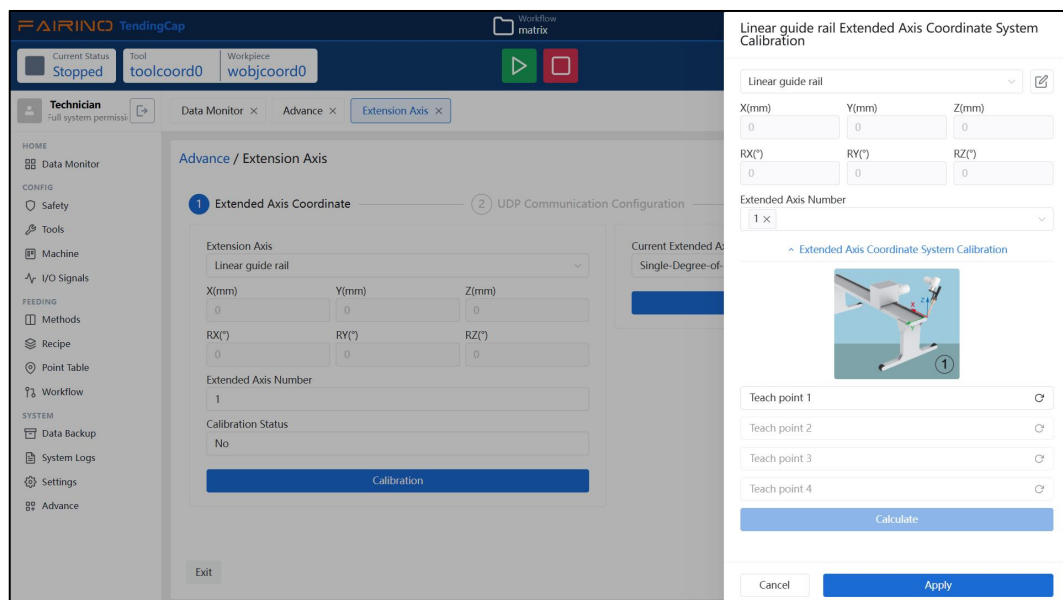


Figure 9.4.2.1-6 The extended axis motion strategy is configured as "Extended Axis Synchronized with Robot".



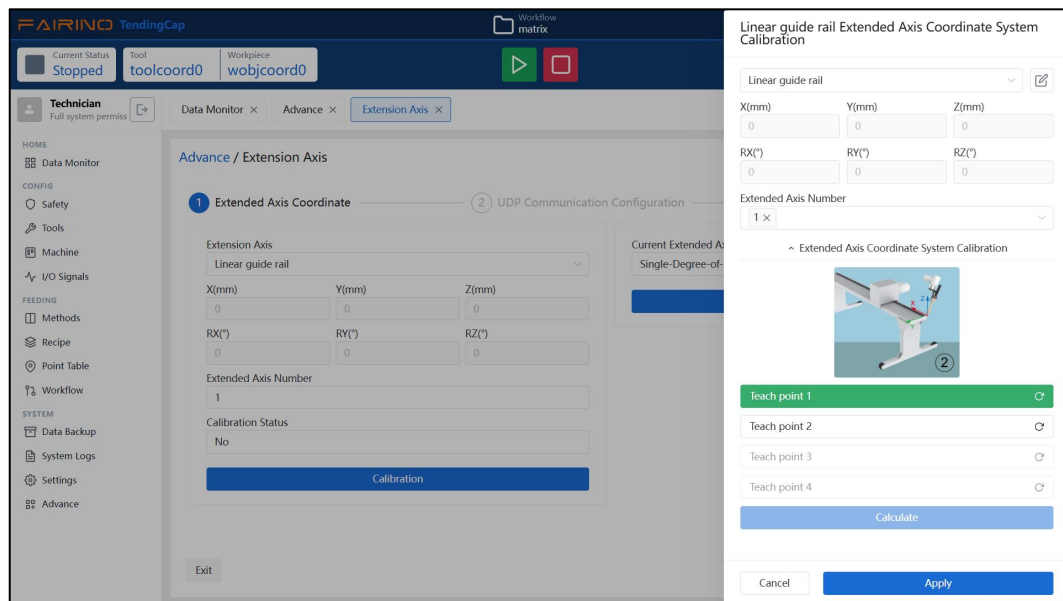
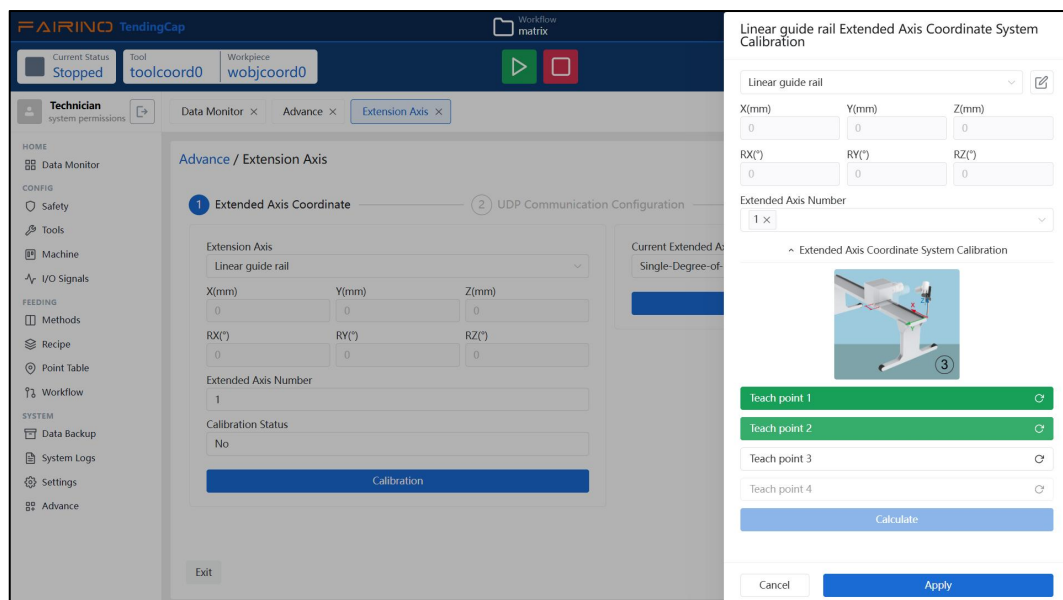


Figure 9.4.2.1-7 Extended axis coordinate system reference points 1 and 2

Disengage the enablement, then move the extended axis by a certain distance; once enabled, align the robot's end-effector center with the previously defined fixed point (Setpoint 3). Disengage the enablement, then move the extended axis to its zero position and re-enable it. Move the robot's end-effector center to a point located a certain distance vertically above the fixed point (Setpoint 4). Then, click the "Calculate" button to obtain the extended-axis coordinate system results; finally, click the "Apply" button at the bottom to apply the extended-axis coordinate system data.



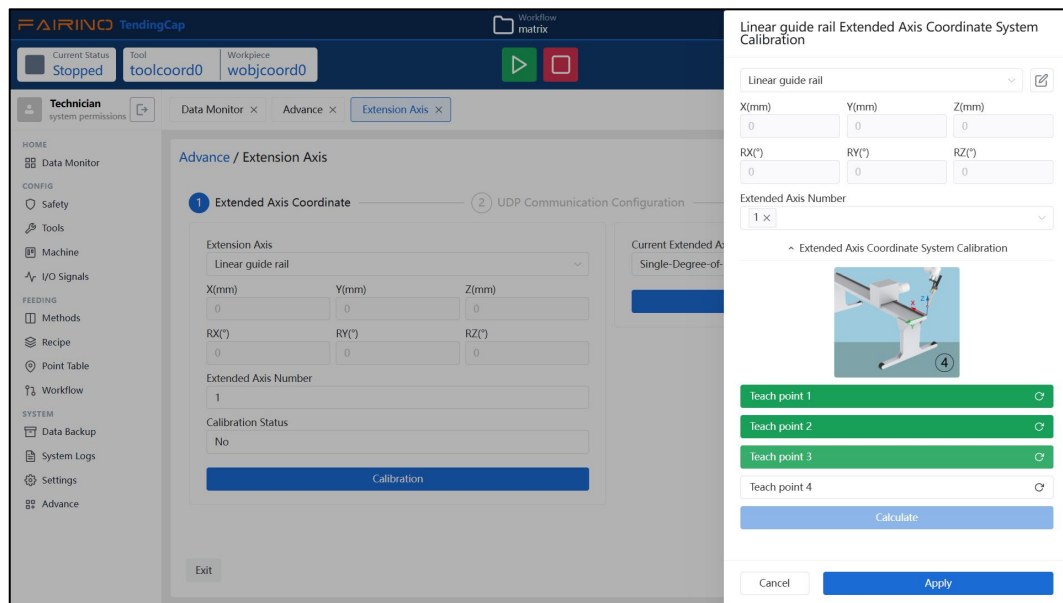


Figure 9.4.2.1-8 Extended axial coordinate system reference points 3 and 4

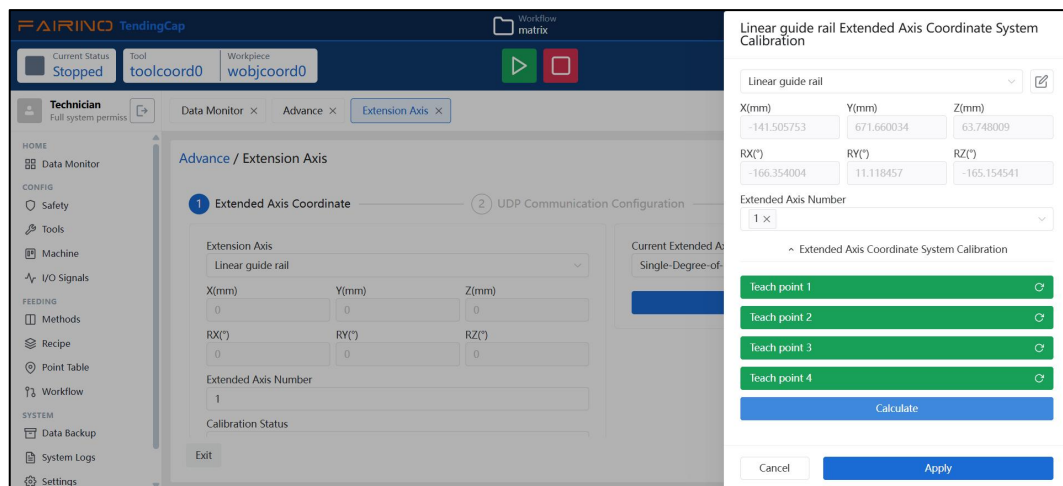


Figure 9.4.2.1-9 Extended axis coordinate system calibration results

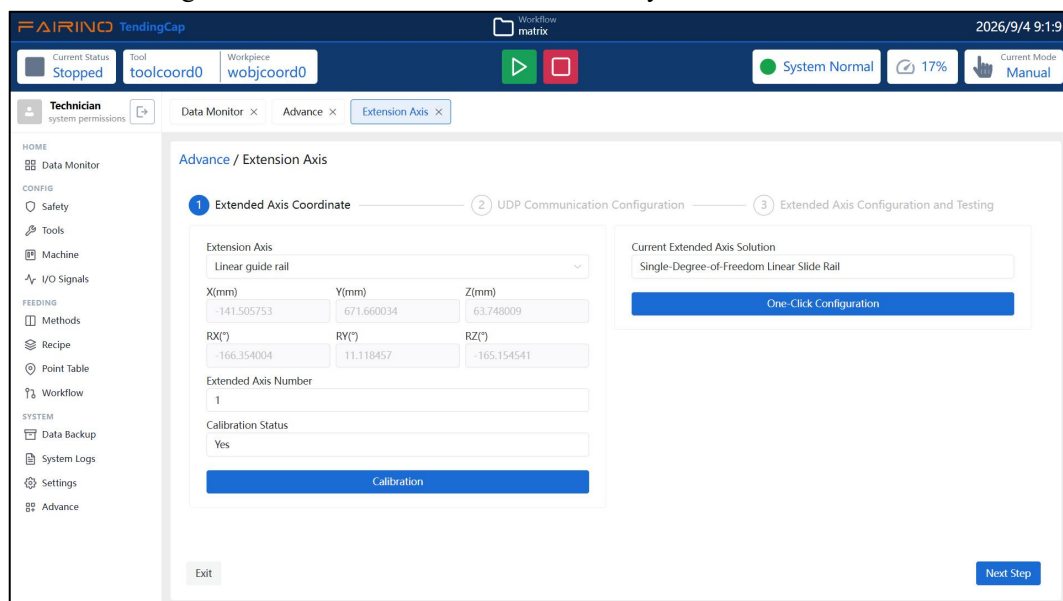


Figure 9.4.2.1-10 Application of extended axis coordinate system calibration results

9.4.2.2 UDP Communication Configuration



Name: Edit Button; Function: UDP Communication Parameter

Configuration.



Name: Load Button; Function: UDP Communication Loading.

Click the Edit button to configure the extended-axis UDP communication parameters: set the IP address, port number, communication cycle, packet loss detection cycle, number of packet losses, and other parameters. Note that the Reconnect Cycle and Reconnect Count can only be configured after the "Auto-Reconnect on Communication Disruption" switch is enabled.

- IP Address: Custom IP Address;
- Port number: Define according to actual requirements;
- Communication period: Defined based on actual requirements; unit: ms;
- Packet loss detection communication period: 10–1000 ms;
- Packet loss count: 1 ~ 100;
- Communication Interruption Confirmation Duration: 0 ~ 500 ms;
- Automatic reconnection upon communication interruption: On/Off;
- Reconnection Cycle: 1 ~ 1000 ms;
- Reconnection attempts: 1 ~ 100;

Important:

- After configuring the communication disconnection confirmation duration, the system will only confirm a communication disconnection and raise an error if the communication anomaly exceeds this duration.

- When UDP communication is terminated, a UDP connection termination error is triggered (this can be reset); you can click the "Clear Warning Message" button to re-establish the UDP communication.

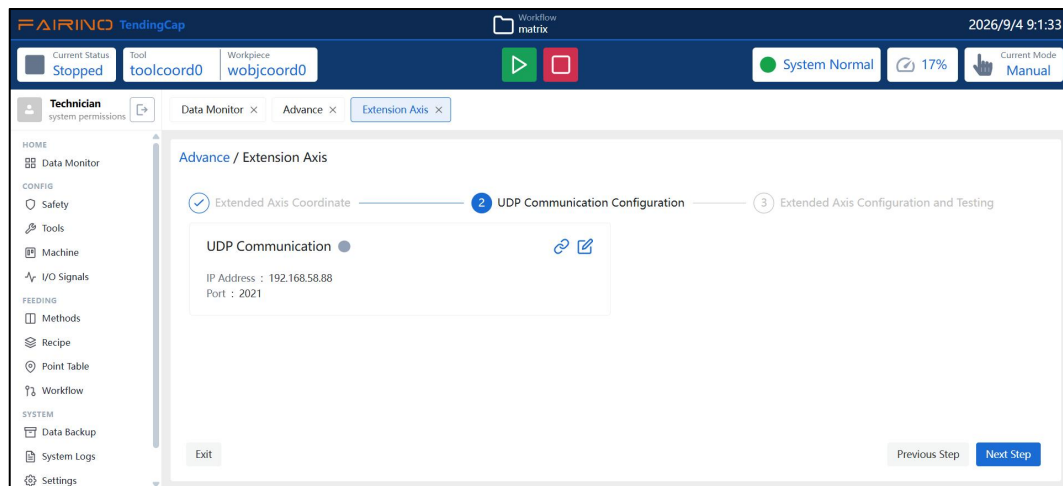


Figure 9.4.2.2-1 UDP communication is not connected.

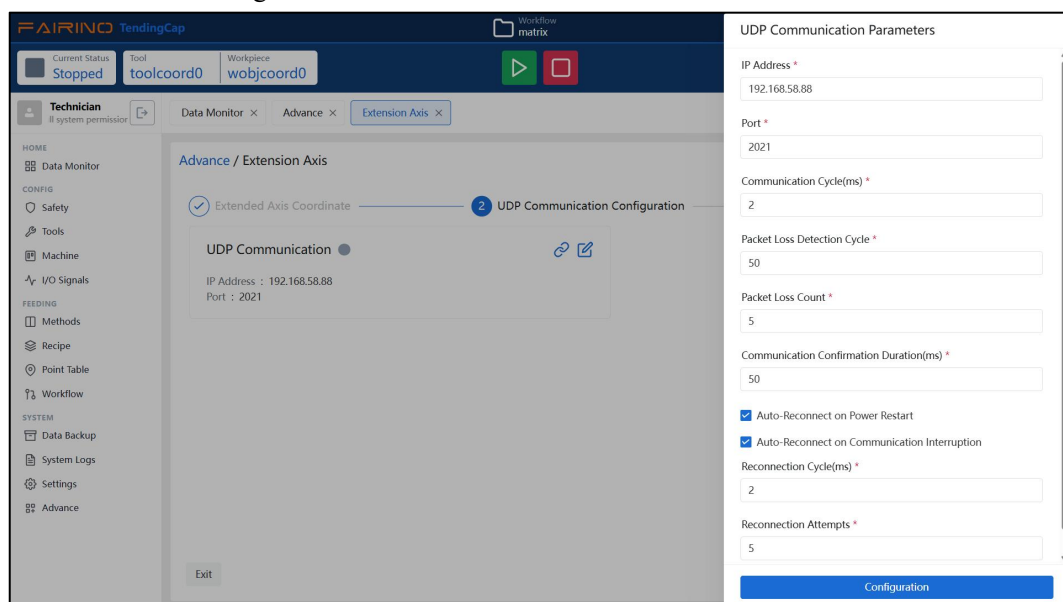


Figure 9.4.2.2-2 UDP communication parameter configuration

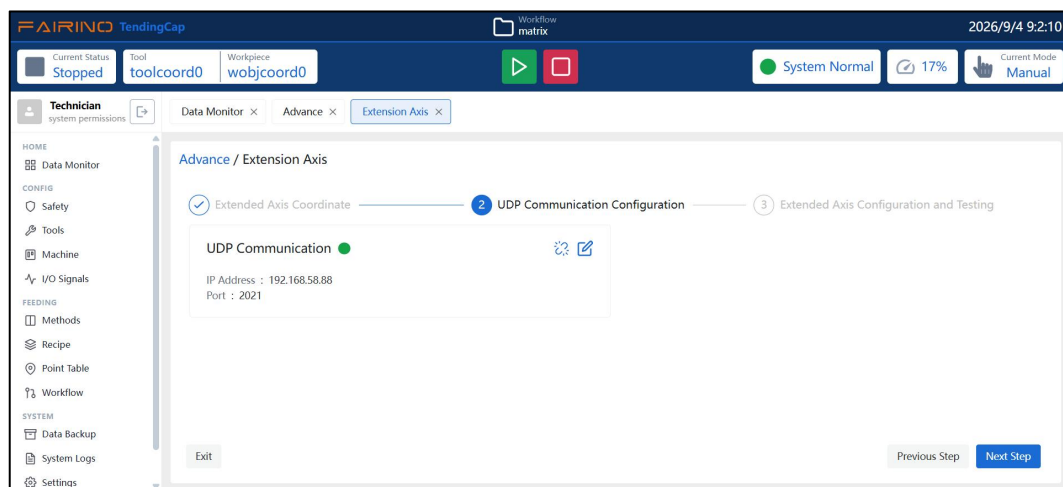


Figure 9.4.2.2-3 UDPcommunication junction

9.4.2.3 Extended shaft configuration and testing



Name: Edit Button; Function: Extend Axis Parameter Configuration;



Name: Enable Button; Function: Activate the extension axis – clicking this button will enable the extension axis.



Name: Disabling Button; Function: Set the extension shaft to disabled state; Click the button to enable the extension shaft.



Name: Zero Return Button; Function: Configure extended axis zero return method.



Name: Test Button; Function: Extended Shaft Function Test.

Click the "Edit" button to open the Extended Axis Parameter Configuration interface. Configure the axis type, axis direction, operating speed, acceleration, forward limit, reverse limit, lead pitch, encoder resolution, starting point offset, manufacturer, model, and mode; then click the "Configure" button at the bottom to complete the configuration.

- Axis type: Linear guide rails, rotating axis, and infinitely rotating axis;
- Axis direction: positive/negative;
- Running speed: 0~2000mm/s;
- Acceleration: 0 ~ 2000 mm/s² ;
- Positive direction limit switch: 0 ~ 50000;
- Reverse Direction Limit Stop: -50000 ~ 0;
- Lead: 0~1000;
- Encoder Resolution: 0 ~ 10000000;
- Starting offset: 0 ~ 10000mm;
- Manufacturers: Hechuan, Huichuan, and Panasonic.
- Model: Automatically matched from the manufacturer's model list;
- Modes: Incremental System and Absolute Positioning System;

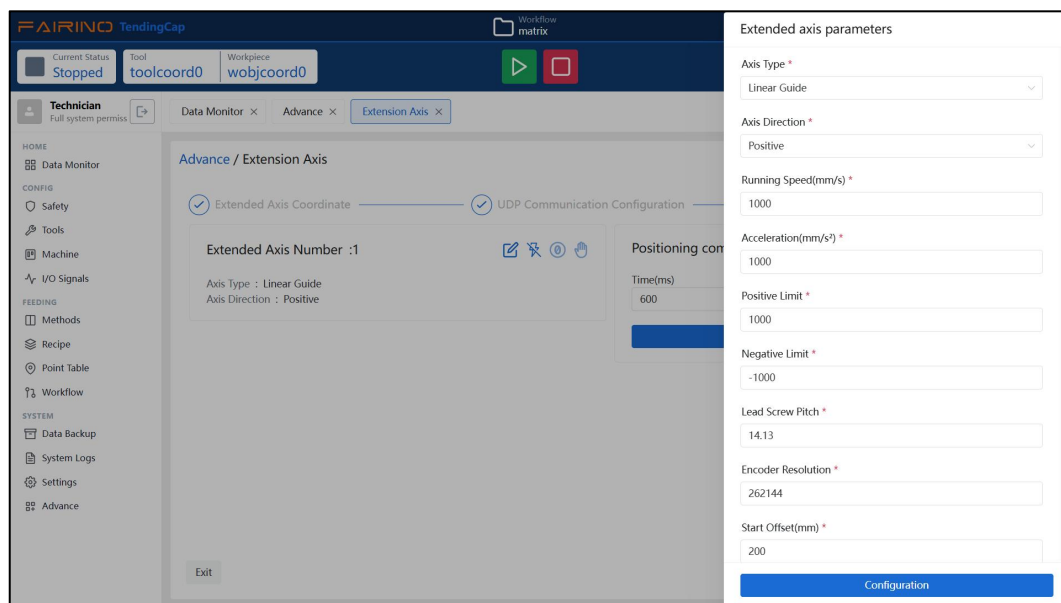


Figure 9.4.2.3-1 Extended Axis Parameter Configuration

After completing the extended axis parameter configuration, click the "Enable" button to enable Extended Axis 1. Once enabled, you can configure the home position setting and perform the extended axis test; however, the home position setting and extended axis test cannot be performed when the extended axis is not enabled.

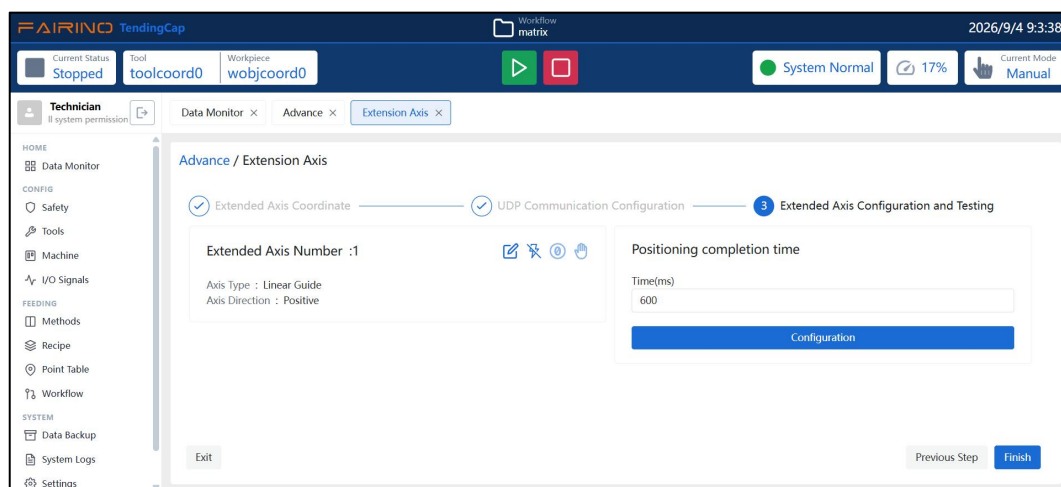


Figure 9.4.2.3-2 Extended axis is not enabled

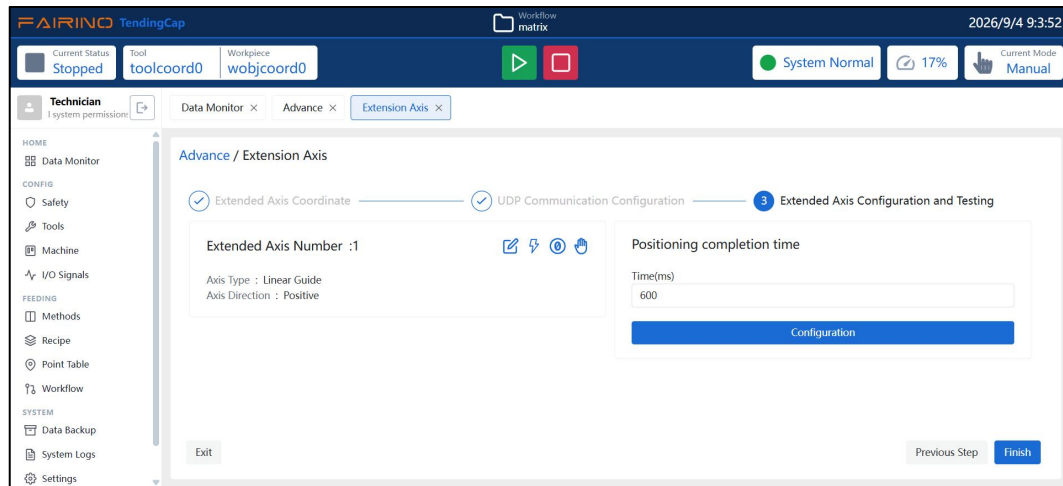


Figure 9.4.2.3-3 Extended Axis Enable

If the extended axis fails to enable successfully, the user cannot access the configuration interface, and the button will remain disabled; once the extended axis has been successfully enabled, clicking the "Return to Zero" button will open the Return to Zero Method Configuration Interface. Configure the return-to-zero method, zero-seeking speed, and zero-point positioning speed; then click the "Set" button. The extended axis will then begin returning to zero – the "Return to Zero" status will be displayed in the blank area below the axis indicator; when the message "Return to Zero Complete" appears, it indicates that the extended axis zero-point setting has been successful.

- Zero-return mode: Return to zero from the current position, return to zero from the negative limit switch, and return to zero from the positive limit switch;
- Find zero speed: 0~2000mm/s;
- Zero-point clamping velocity: 0~2000mm/s;

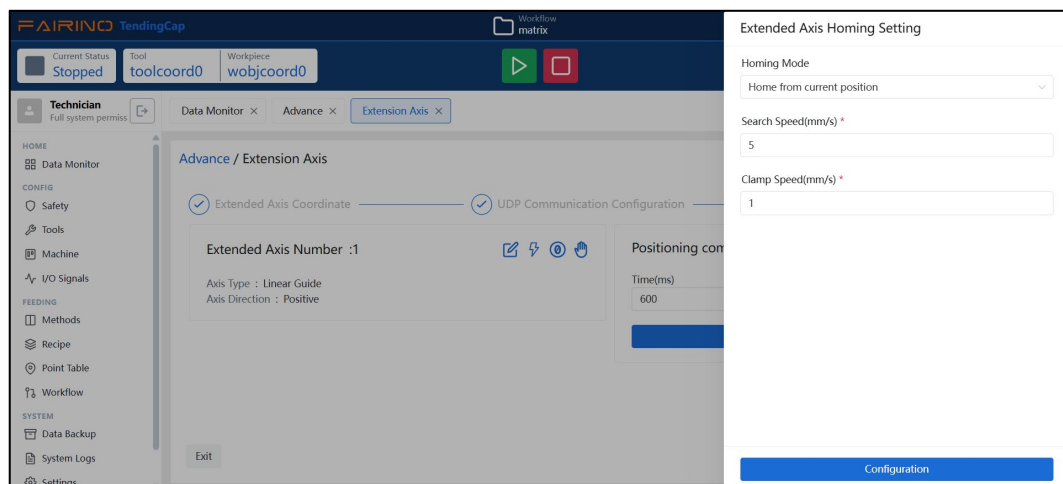


Figure 9.4.2.3-4 Extended Axis Zero Return Setting

If the extension shaft fails to enable successfully, the user cannot access the setup interface, and the "Test" button will remain disabled; once the extension shaft has been successfully enabled and the home position setting is complete, clicking the "Test" button will display the Extension Shaft Test Interface. Set the operating speed, acceleration, and maximum distance, then perform forward and reverse rotation tests on the extension shaft; during the rotation process, you may click the "Stop" button to verify whether the extension shaft stops properly.

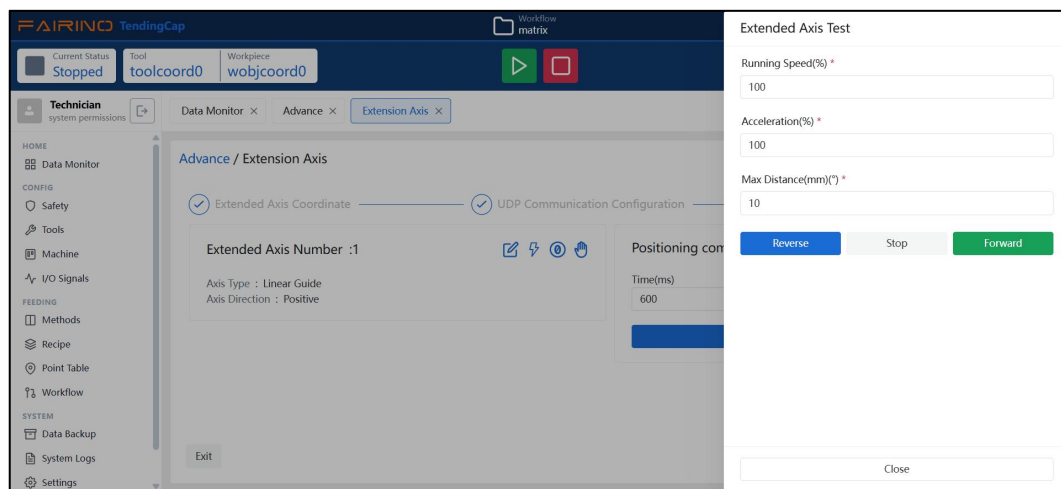


Figure 9.4.2.3-5 Extended Shaft Test

Once the extended axis has established UDP communication, enter the time value and click the "Configure" button to complete the setup. This configuration option is used to monitor the time elapsed until the extended axis stops moving.

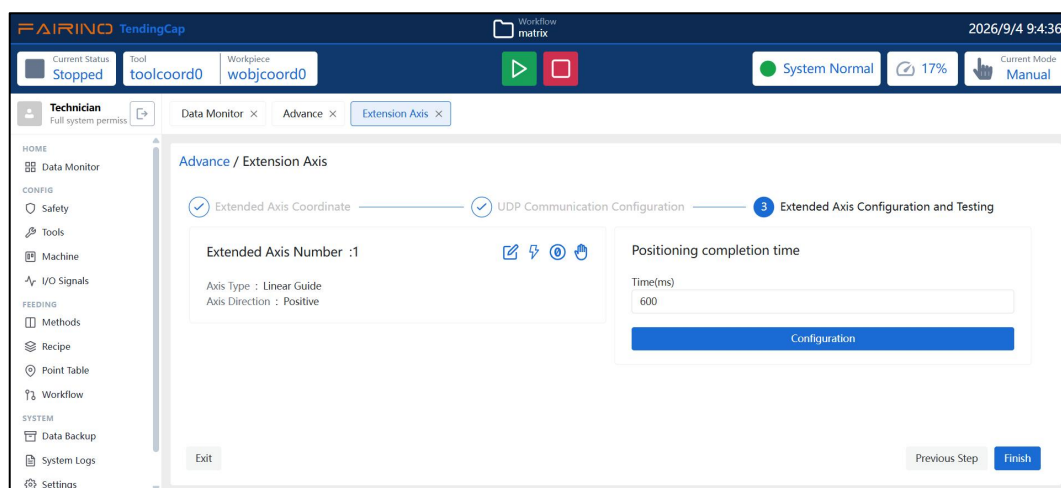


Figure 9.4.2.3-6 Extended Axis Positioning Completion Time

9.4.3 Modbus TCP

Modbus TCP is a widely used communication protocol in industrial manufacturing; the Fao collaborative robot supports two communication modes with your devices: Modbus TCP master and Modbus TCP slave.

The collaborative robot supports up to 8 Modbus TCP master stations communicating with external devices simultaneously; each master station can support up to 128 registers. The collaborative robot's Modbus TCP slave station features 128 coils, 128 discrete inputs, 64 holding registers, and 64 input registers (the data types for holding and input registers include unsigned, signed, and floating-point types). Additionally, certain input register addresses on the collaborative robot's Modbus TCP slave station are dedicated to providing feedback on the robot's current joint position, motion speed, and other parameters; other coil register addresses are dedicated to controlling the robot's start/stop programs and configuring the control box's DO functions.

The Robot Modbus TCP slave device can only establish a connection with one master device; however, the robot can simultaneously act as both a master and a slave to communicate with different devices. Below is a detailed usage guide.

9.4.3.1 Master station

Before using a collaborative robot to communicate with your device as a Modbus TCP master station, please verify the network connection between your device and the robot, and ensure that their network interfaces are on the same subnet.

To use a robotic Modbus TCP master station, follow these steps:

- Add main site;
- Add register;
- Communication Testing;
- Write user programs;
- Execute user program.

9.4.3.1.1 Add Modbus TCP Master Station

In the "Loading/Unloading – Workflow" section, create a new workflow named

"testModbusMaster".

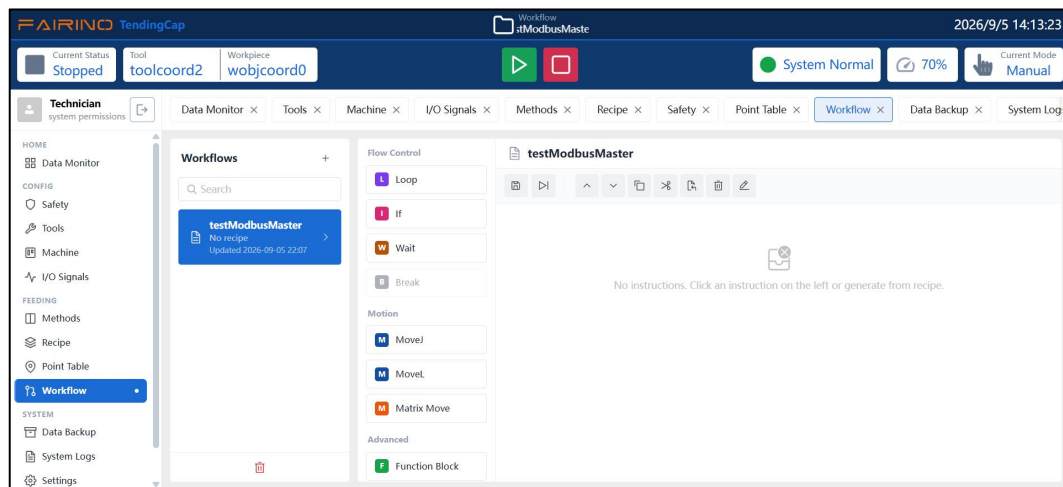
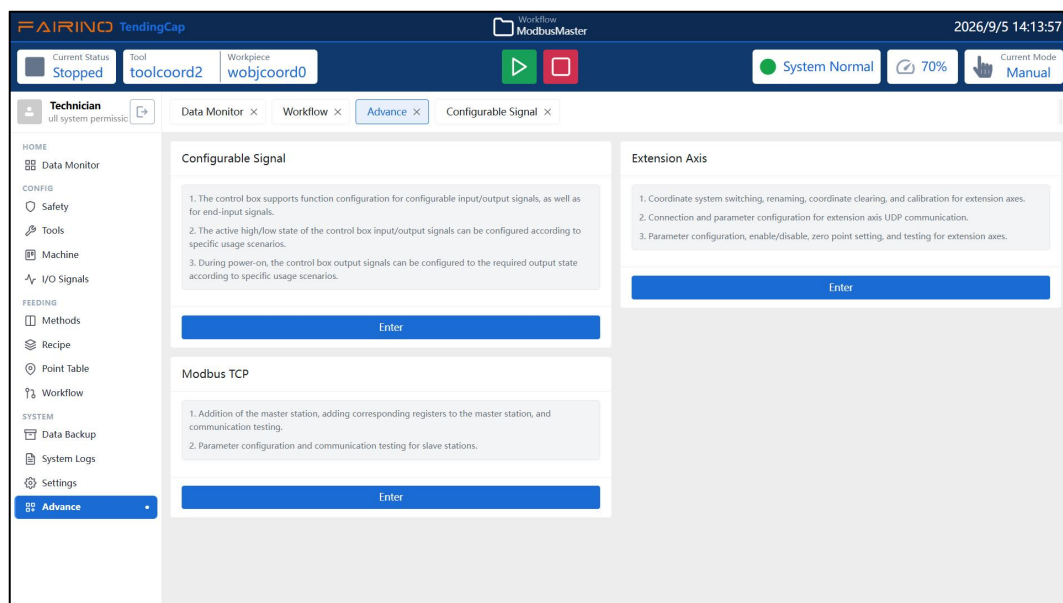


Figure 9.4.3.1-1 Create a ModbusTCP master station workflow

Click the "Enter" button at the bottom of the "Modbus TCP" section in the Advanced Parameters to open the Modbus TCP Function Configuration page.



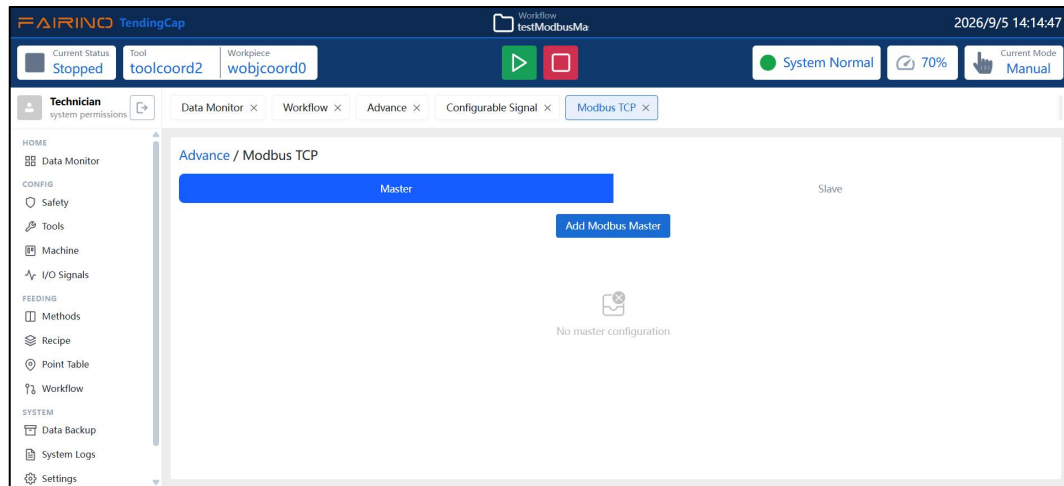


Figure 9.4.3.1-2 Open Modbus TCP Configuration

Click "Main Station" and then "Add Modbus Main Station" in sequence to add a Modbus TCP main station.

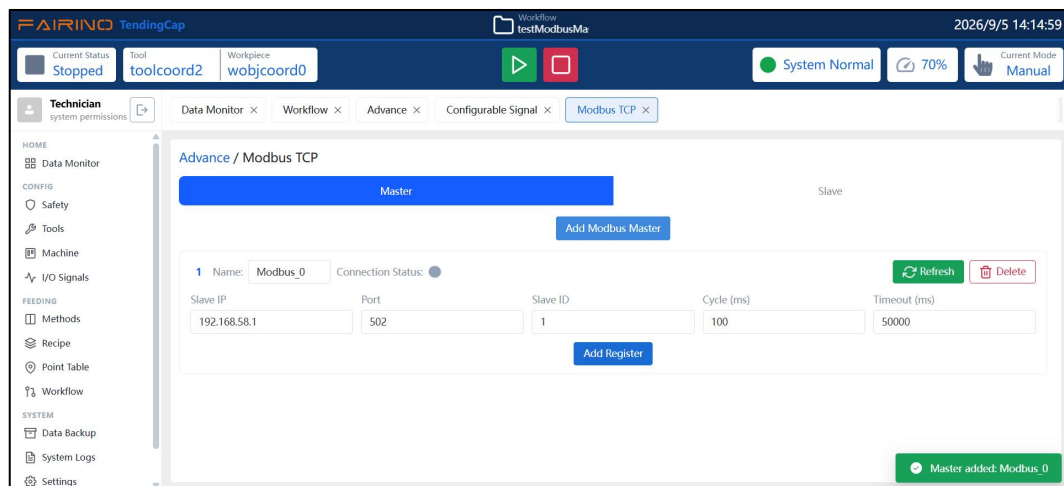


Figure 9.4.3.1-3 Add "Modbus TCP Master Station"

Depending on your device configuration, enter the following parameters in order: "Name", "Slave IP Address", "Port Number", "Slave ID", "Communication Cycle", and "Timeout Duration". The specific meaning of these parameters is as follows:

- **Name:** Robot Modbus TCP Master Station Name: The robot supports up to 8 master stations; each master station can be assigned a unique name to distinguish it from others (e.g., "PLC", "Camera", "Data Acquisition Card", "FRRobot1", etc.).

- **Slave IP Address:** The IP address of the slave device to be connected to the Modbus TCP master robot (Note: First, connect the robot to the slave device using an Ethernet cable, and ensure that both the robot and the slave device reside on the same network segment).

- **Port number:** Modbus TCP slave device port number to connect to.
- **Substation Number:** Modbus TCP slave device ID to be connected.
- **Communication cycle:** The polling interval (in ms) for the robot's Modbus TCP master station to query the slave station status; this interval only affects the update rate when viewing slave station register data on the "Modbus TCP Settings" page, but does not affect the speed at which values in Modbus TCP slave station registers are read from or written to within the user's Lua program.
- **Timeout duration:** When the Modbus TCP read/write interface is invoked and the system fails to establish a connection within the specified timeout period, a "Modbus connection failed" error will be reported. Units: ms; valid range: 100–60000.

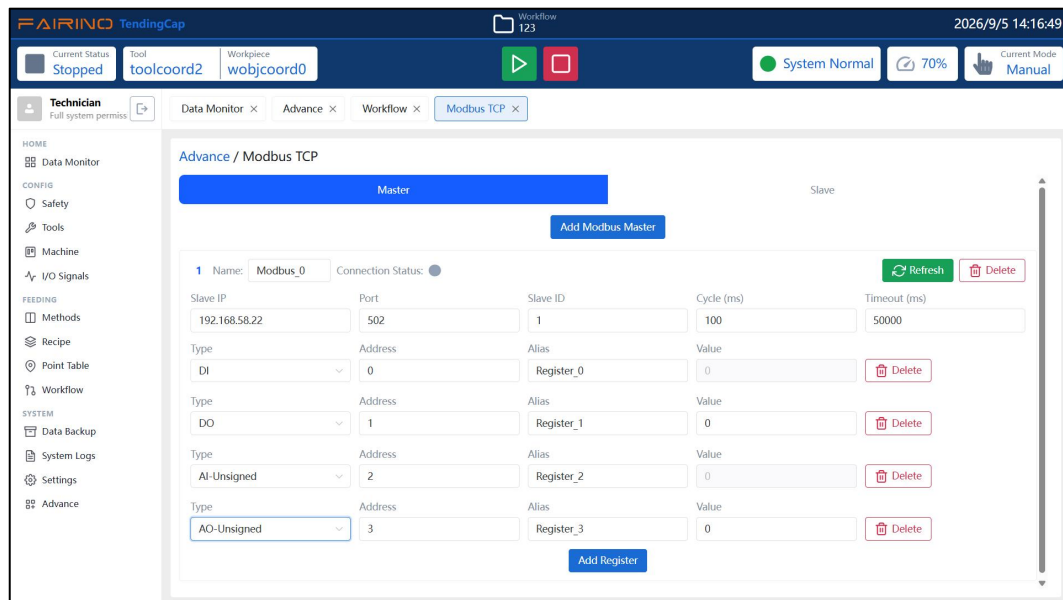


Figure 9.4.3.1-4 Configure Modbus TCP Master Station Parameters

After correctly entering the above parameters, the robot's Modbus TCP master station will automatically establish a connection with the configured slave station; upon successful connection, the "Connection Status" indicator light on the page will illuminate.

Note: If you have confirmed that the relevant parameters for the ModbusTCP master station have been correctly configured, but the robot still fails to connect to your device, please check the following configurations.

- ① Physical network connection between the robot and slave devices.
- ② The IP addresses of the robotic teach pendant and the control box's two network I/O ports are different; please verify whether they are connected to the correct network

ports.

③Please verify whether the robot's network port and the slave device's network port are on the same network segment. For example, if the robot's IP address is 192.168.58.2, then the slave device's IP address must be between 192.168.58.0 and 192.168.58.255, and it must not be the same as the robot's IP address.

④Verify whether the port number of the remote station equipment matches the configured port number. If the connection status indicator light is flashing, this indicates an incorrect register address in the main station; please check whether the register type and address are correct.

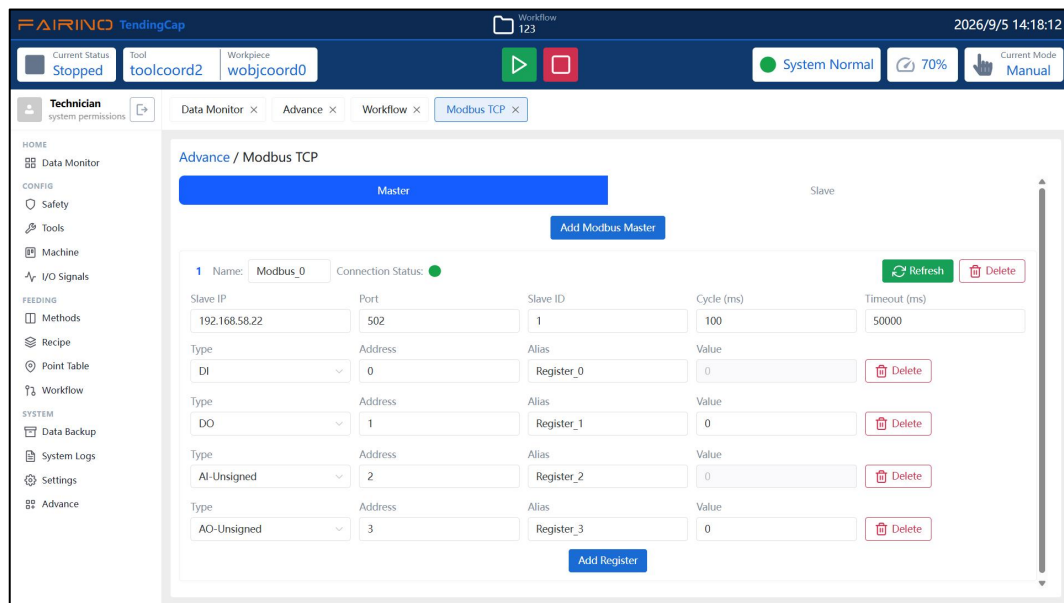


Figure 9.4.3.1-5 Modbus TCP Master Station Connection Status

You have now successfully created a Robot Modbus TCP master station. If you click "Add Modbus Master Station" again, you can create another new Modbus TCP master station. The robot supports up to 8 master stations communicating with external devices simultaneously. To delete a Modbus master station, click the "Delete" button in the upper-right corner of the Modbus Master Station window, and follow the pop-up confirmation prompt to remove it.

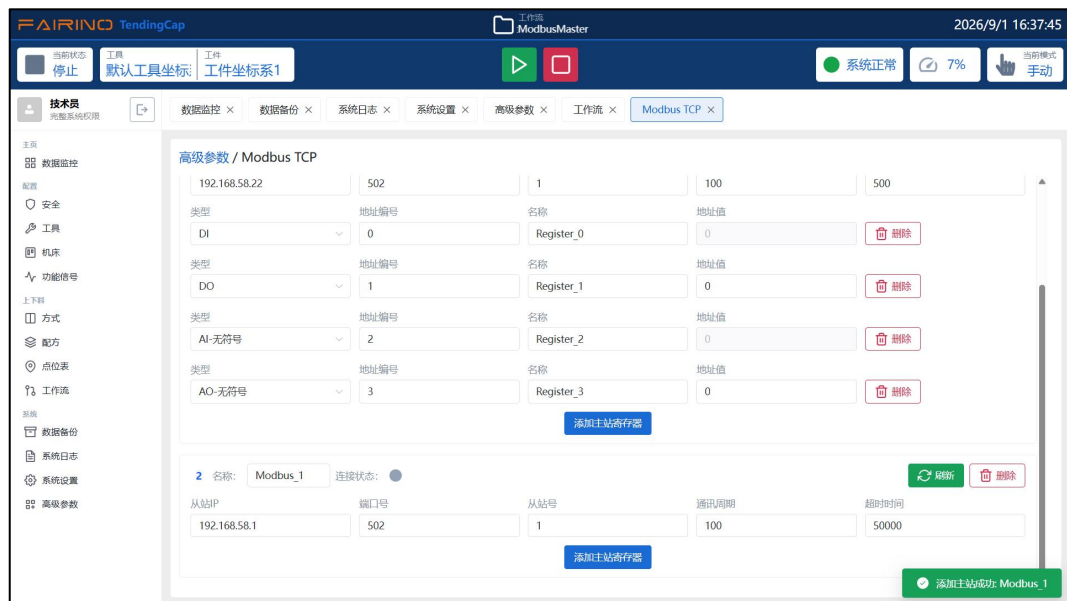


Figure 9.4.3.1-6 Add another Modbus TCP master station

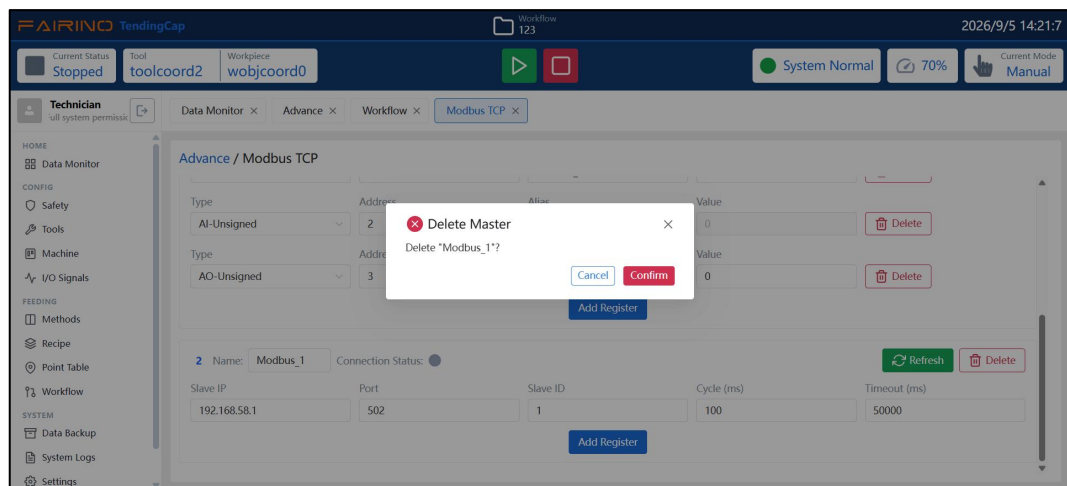


Figure 9.4.3.1-7 Delete Modbus TCP Master Station

9.4.3.1.2 Add Register to Modbus TCP Master Station

Click the "Add Main Site Register" button to add a register to this main site.

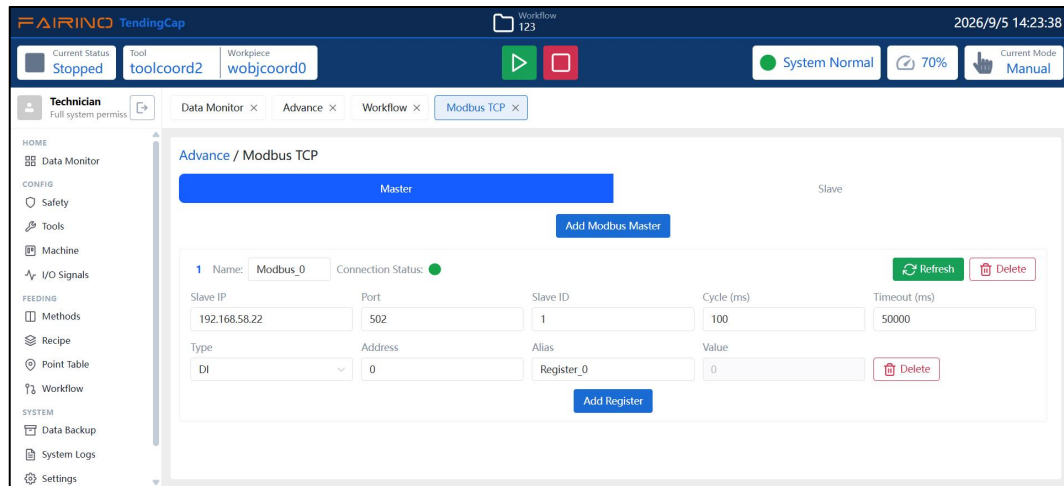


Figure 9.4.3.1-8 Add Modbus TCP Master Station Register

Select, in sequence, the main station register type, the input address number, and the input address name; the meaning of each parameter is as follows:

- **Type:** Register types: DI – Discrete Input; DO – Coil; AI (Unsigned) – Unsigned input register (0 – 65535); AI (Signed) – signed input register (-32768 – 32767); AI (Floating-Point) – floating-point input register (the floating-point register has a data length of 32 bits and occupies two signed or unsigned registers); AO (Unsigned) – unsigned hold register (0 – 65535); AO (Signed) – signed hold register (-32768 – 32767); AO (Floating-Point) – floating-point hold register (the floating-point register has a data length of 32 bits and occupies two signed or unsigned registers); Note: The floating-point registers for AI and AO utilize big-endian display format.

- **Address Code:** The Modbus TCP slave register address to read from or write to.

- **Name:** Alias for registers: The robotic Modbus TCP master station can configure up to 128 different registers; each register can be assigned a distinct name based on its actual function — for example, "Start," "Servo Position," or "Liquid Level."

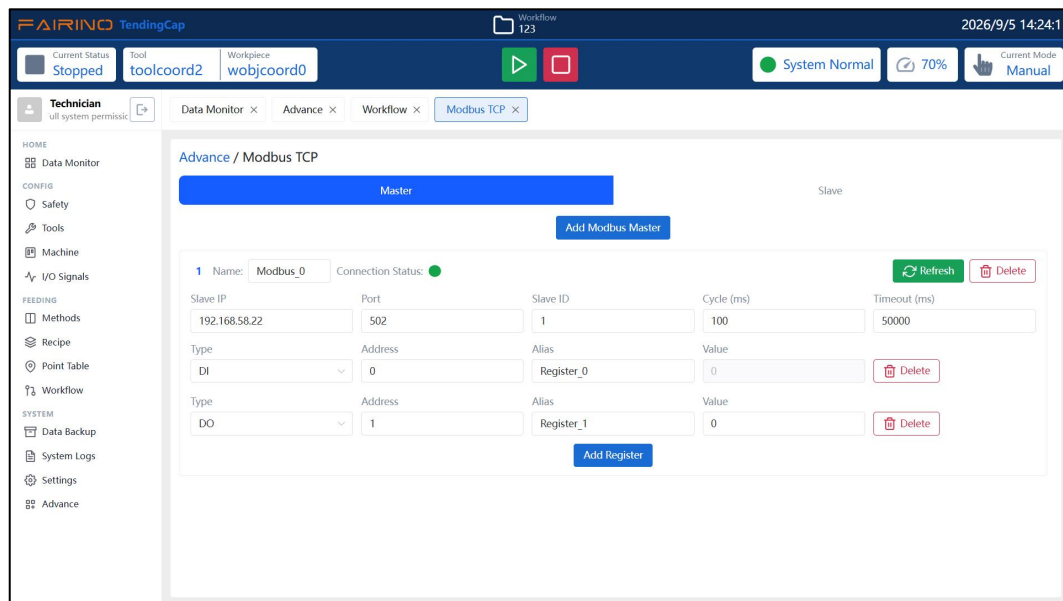


Figure 9.4.3.1-9 Configure Modbus TCP Master Station Register Parameters

Click the "Add Primary Site Registry" button again to add another primary site registry; then click the "Delete" button on the right side of the registry, and follow the pop-up confirmation prompt to delete that registry. As shown in the figure below, a registry has been added for each registry type.

Note: If the main station's connection status indicator light flashes after the main station register is added, it indicates that the specified main station register address cannot be read; please verify that the register type and address are correct.

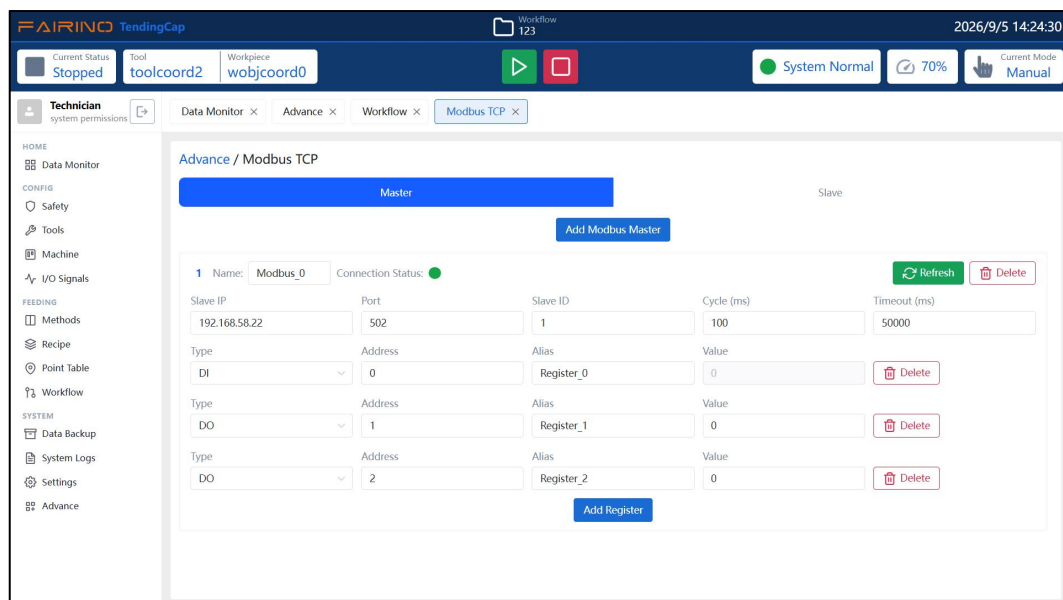


Figure 9.4.3.1-10 Add multiple main station registers

9.4.3.1.3 ModbusTCP Master Station Communication Test

Before conducting communication tests, please check whether the "Connection Status" indicator light on the ModbusTCP master station is constantly illuminated; if the indicator light is constantly lit, it indicates that the current connection was established successfully.

The Robot Modbus Master Station register interface includes an "Address Value" text box for displaying the current register value; however, registers of type DI (Discrete Input) and AI (Input Register) are read-only, and their corresponding address values are displayed as gray, non-editable text boxes.

When the value at the corresponding slave device address changes, the current value is displayed synchronously at the corresponding register address on the robot's master station. Since the DO (coil) and AO (holding register) are read-write registers, their addresses appear as white editable numerical fields – allowing users to both read the value from the corresponding ModbusTCP slave device register and modify that value directly on the robot's Modbus master station configuration page.

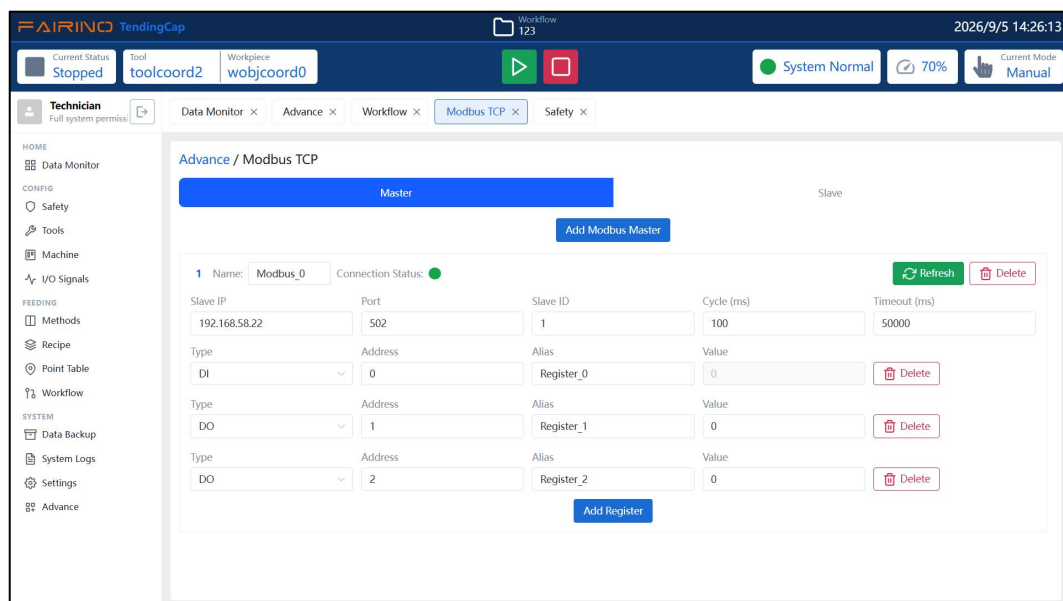


Figure 9.4.3.1-11 Modbus master station address value

- **Main Station DI/AI Type Register Value Monitoring:** On an external Modbus TCP slave device, set the value at address 255 of the DI (Discrete Input) register to 1; set the value at address 257 of the AI (Analog Input) register to 123; set the value at address 258 of the AI register to-123; and set the value at address 259 of the AI register to 123.3. At this point, the corresponding register values on the robot's Modbus main station configuration page will be displayed accordingly. (Note:

Since the register at address 259 is a floating-point register, it actually occupies both 16-bit registers at addresses 259 and 260 to store a single floating-point number; therefore, you cannot configure a separate register to manipulate the AI register at address 260 – doing so will result in a numerical error.)

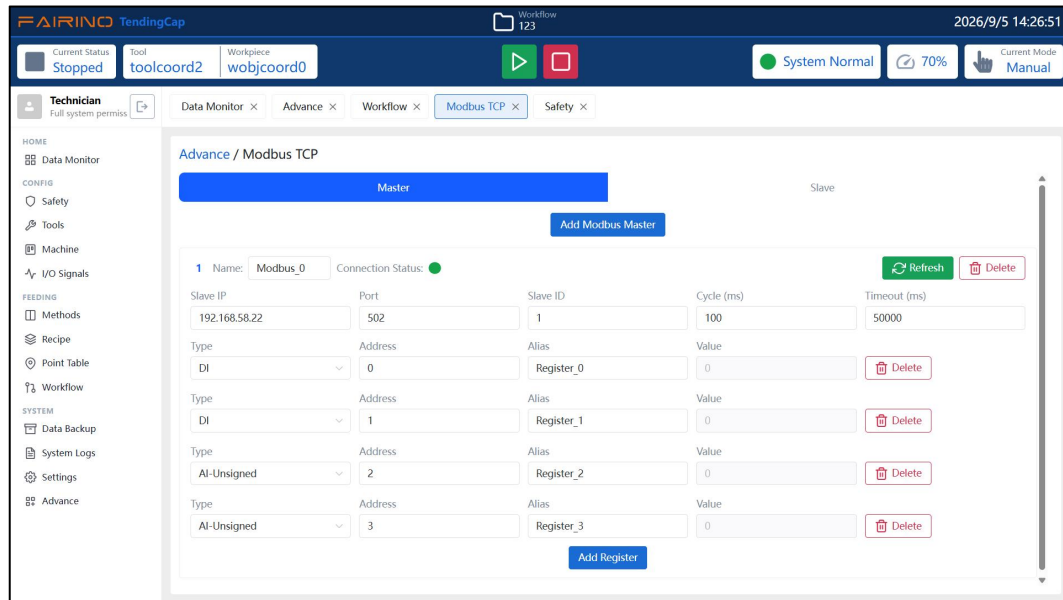


Figure 9.4.3.1-12 Modbus master station displays the values of DI and AI registers.

- Writing values into the DO and AO type register values on the main station: On the Robot Modbus Main Station Configuration Page, enter 1 into the value entry field for address 255 of the DO (coil) type register named "Start"; then enter 65535, -32768, and 128.78, respectively, into the value entry fields for addresses 260, 261, and 262 of the AO (holding register) types named "Target Position A", "Target Position B", and "Target Position C". At this point, the corresponding register addresses on the Modbus slave station have been written with the respective values.

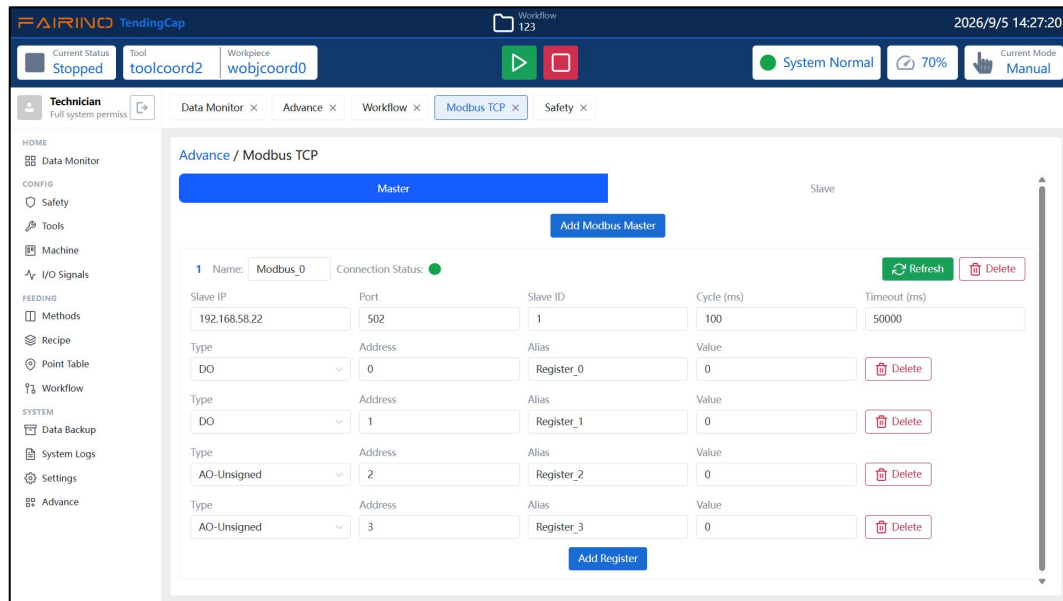


Figure 9.4.3.1-13 Modbus master station writes to DO and AO registers

- **Main Station DO/AO Type Register Value Monitoring:** When you modify the values of DO (Coils) or AO (Hold Registers) on a ModbusTCP slave device, the register address values displayed on the ModbusTCP Main Station Configuration Page will not update immediately. You must click the "Refresh" button located in the upper-right corner of the Main Station Configuration page for the DO and AO register address values on the page to update.

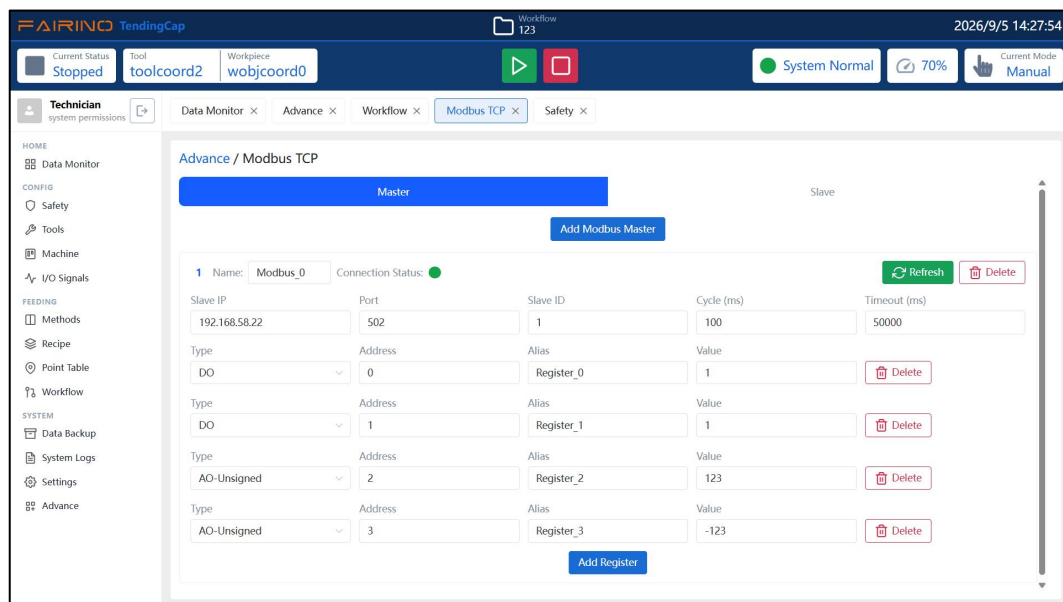


Figure 9.4.3.1-14 Refresh ModbusTCP master station DO and AO address values

9.4.3.2 Write a Modbus TCP master station program

In the "Loading/Unloading - Workflow" section, open the workflow

"testModbusMaster", click on the "Modbus" master station, and then access the Modbus TCP Master Station Command Addition page.

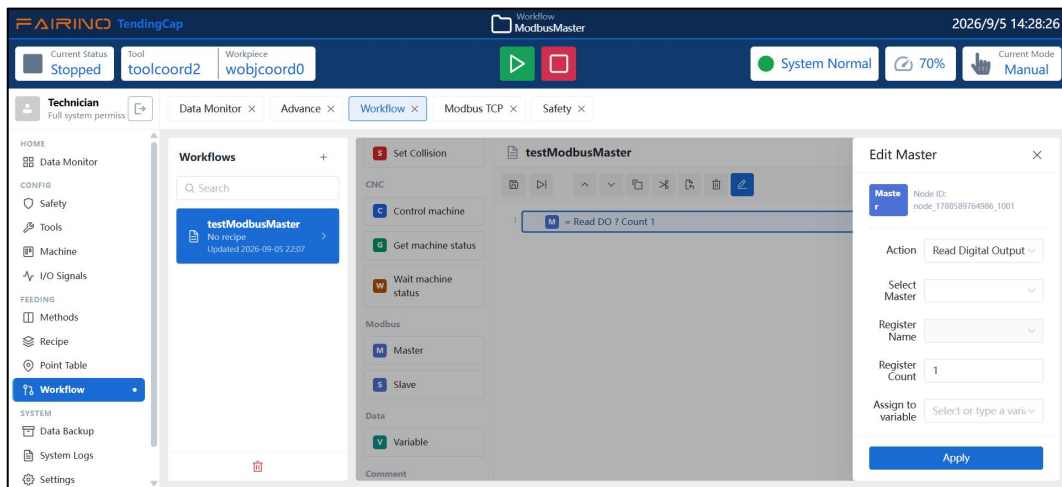


Figure 9.4.3.2-1 Modbus TCP Master Station Command Addition Page

9.4.3.2.1 Write a single digit to output DO (Coil).

Select "Modbus Master Station Name" as the master station "PLC" that was previously added on the Modbus Master Station Configuration page; set the DO name to "Start"; choose the action as "Write Digital Output"; specify 1 register; set the register value to 1. Then click the "Apply" button at the bottom.

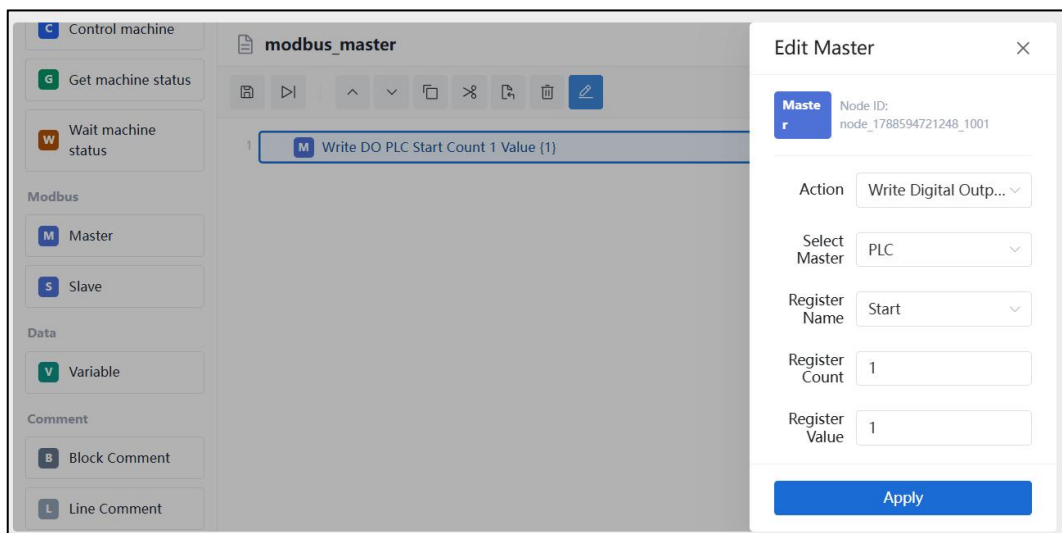


Figure 9.4.3.2-2 Add single-digit output

The workflow "testModbusMaster" already includes a command for the robot's Modbus master station to write a single digital output value; by switching the robot to automatic mode and clicking the "Start" button, the robot will set the address value of the "Start" coil register (corresponding to the "PLC" master station) to 1.

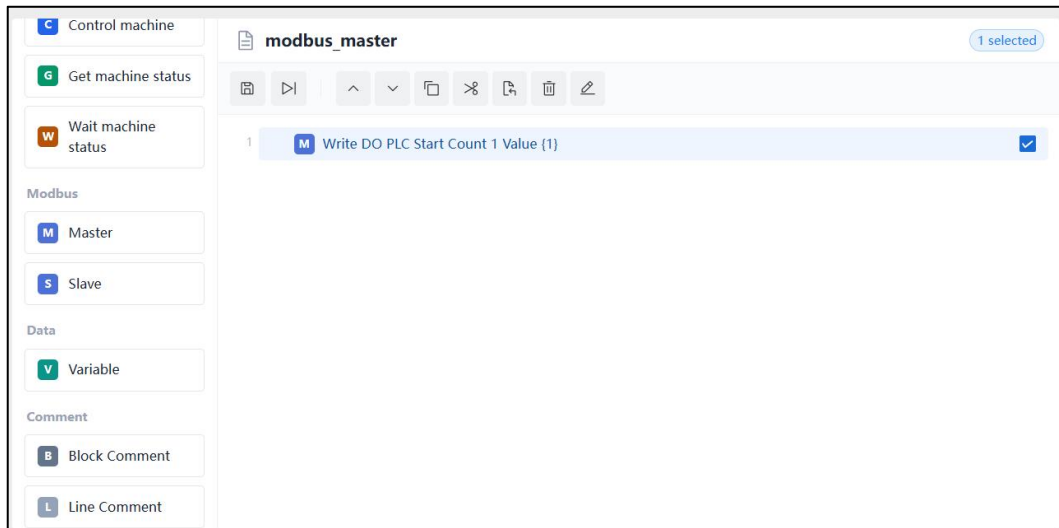


Figure 9.4.3.2-3 Write single-digit output workflow

9.4.3.2.2 Output multiple digital values to a DO (coil).

Open the Modbus TCP Master Station Command Addition Page, select "Modbus Master Station Name" as the master station "PLC" that was previously added on the Modbus Master Station Configuration Page, set the DO name to "Start", choose the action as "Write Digital Output", specify the number of registers as 5, and set the register values to 1, 0, 1, 0, 1. The number of register values must match the number of registers specified, and multiple register values should be separated by English commas. Click the "Apply" button at the bottom.

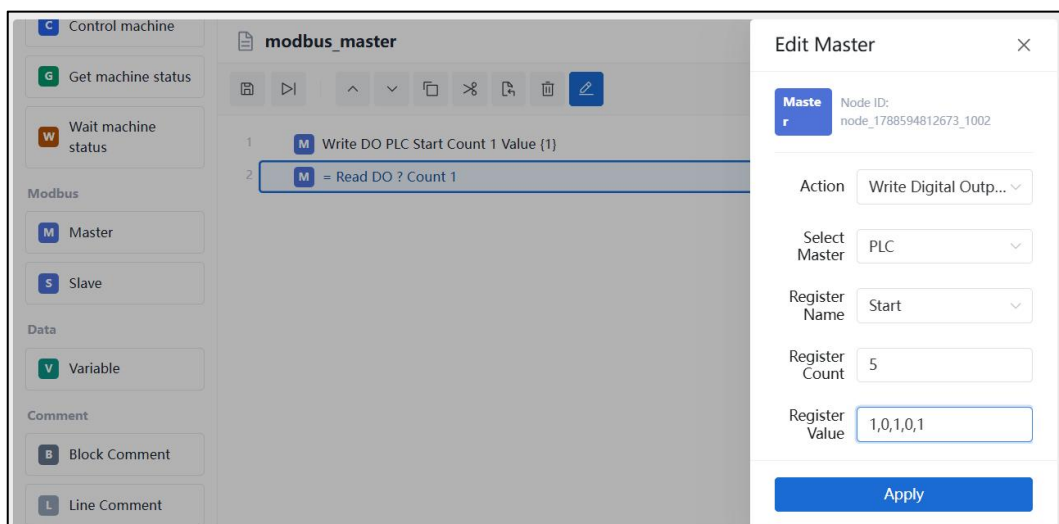


Figure 9.4.3.2-4 Add multiple-digit output

At this point, the workflow "testModbusMaster" already contains a command instructing the robot's Modbus master station to write values to multiple digital outputs.

Switch the robot to automatic mode and click the "Start" button; the robot will then write the value 1 to the "Start" coil register corresponding to the master station ("PLC") and the values 0, 1, 0, and 1 to the next four coils, respectively.

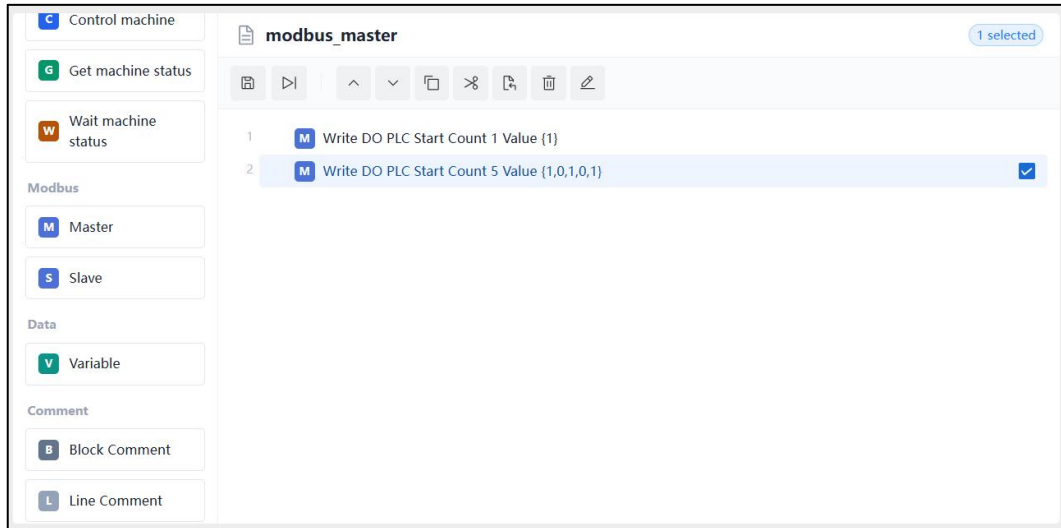


Figure 9.4.3.2-5 Write multiple numbers to output workflow

9.4.3.2.3 Read a single digit DO (Coil)

Open the Modbus TCP Master Station Command Addition Page, select "Modbus Master Station Name" as the master station "PLC" that was previously added on the Modbus Master Station Configuration Page, set the DO name to "Start", choose the action "Read Digital Output", specify the number of registers as 1, and set the variable input to "a". Typically, after reading a Modbus register, the retrieved value is stored in a variable; therefore, define a variable "a" to store this value. After executing the program, the retrieved value will be stored in "a". Click the "Apply" button at the bottom.

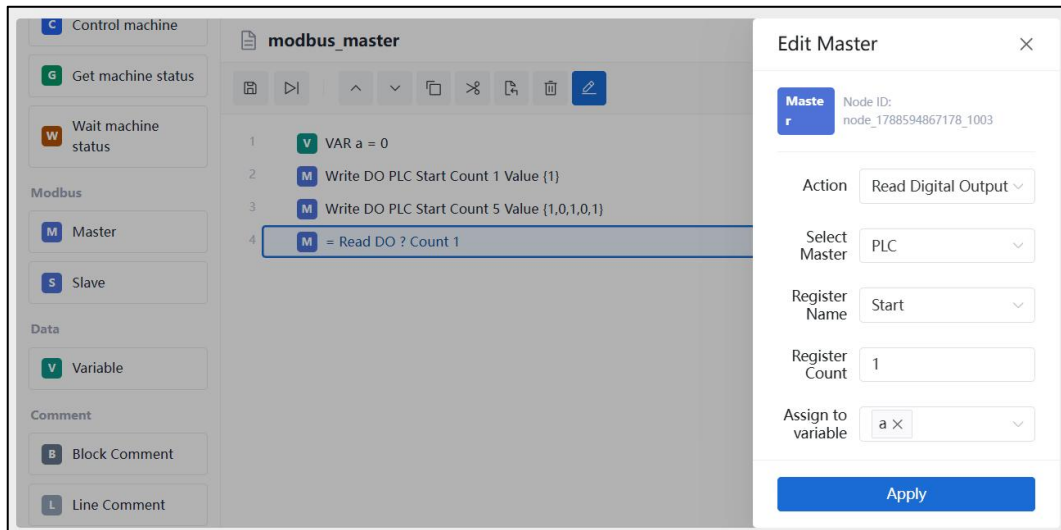


Figure 9.4.3.2-6 Configure for single-digit output

At this point, the workflow "testModbusMaster" already contains a command for the robot's Modbus master station to read a single digital output.

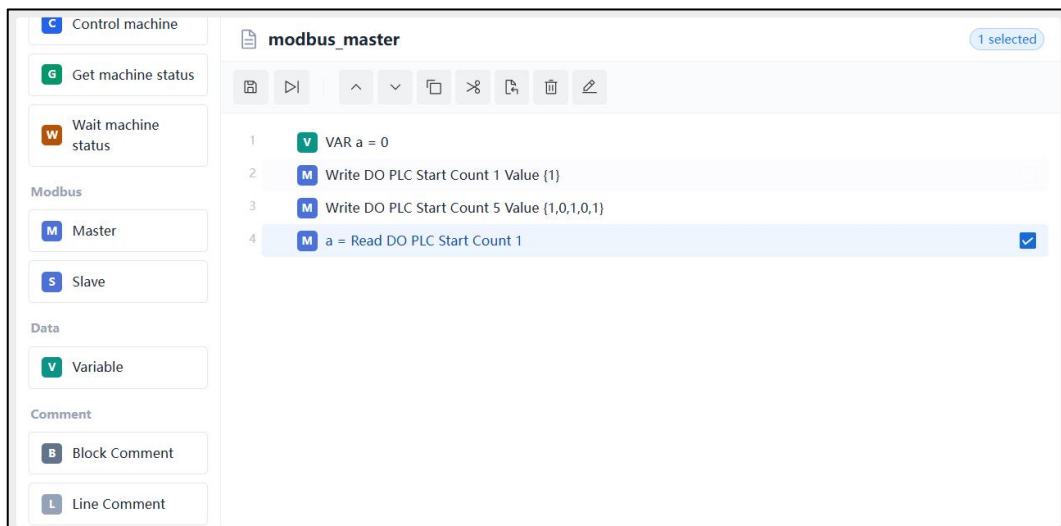


Figure 9.4.3.2-7 Read single-digit output workflow

The coil-based register value can only take two possible values: 0 or 1; in a robot program, different operations can be executed by checking whether the register value is different. Click "Judge", select "Variable Comparison", and add an "Otherwise" condition. Then, add two joint movement commands to move the robot to two different positions: "P1" and "P2".

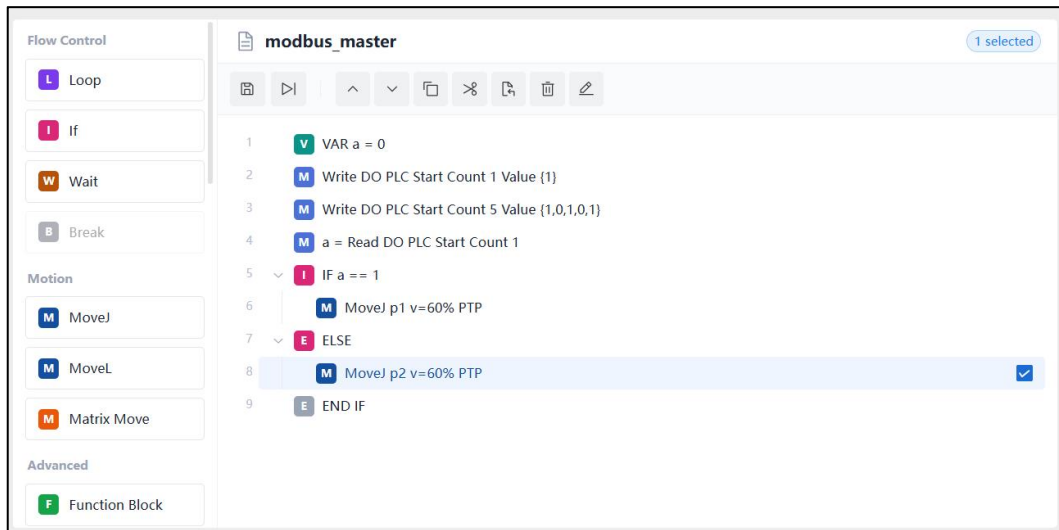


Figure 9.4.3.2-8 Read a single-digit output and add decimal points for operation

9.4.3.2.4 Read multiple numerical values (Coil)

Open the Modbus TCP Master Station Command Addition Page, select "Modbus Master Station Name" as the master station "PLC" that was previously added on the Modbus Master Station Configuration Page, set the DO name to "Start", choose "Read Digital Output" as the action, specify 5 registers, and select the variables "b, c, d, e, f" for assignment. After executing the program, the values read from these 5 registers will be stored in the respective variables "b", "c", "d", "e", and "f". You can then use these values to control the robot to perform different actions. Click the "Apply" button.

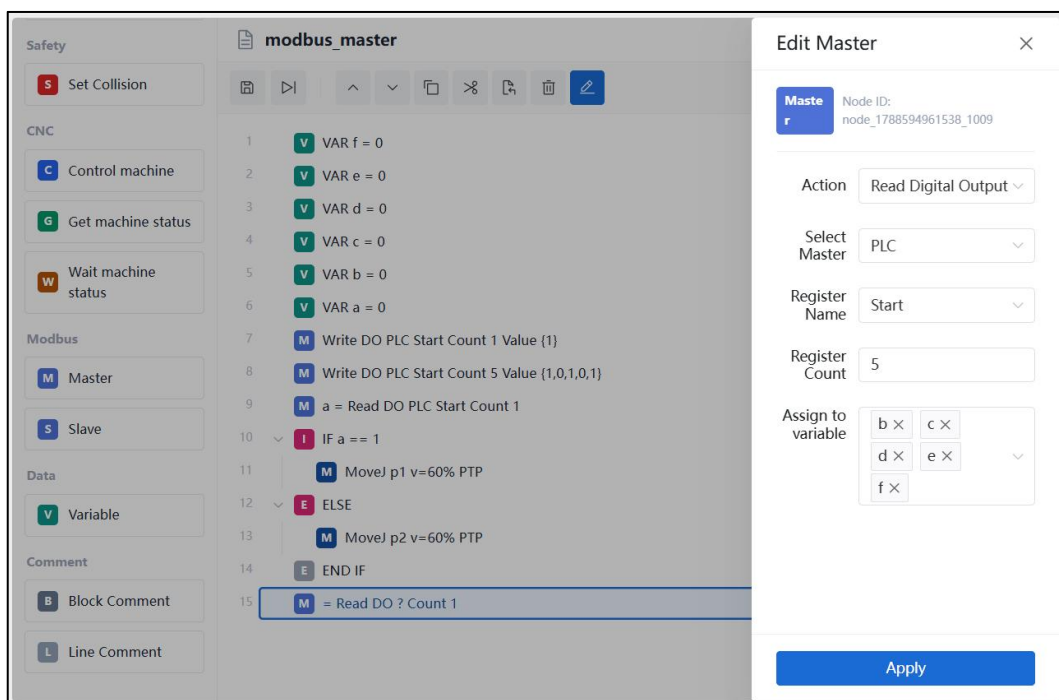


Figure 9.4.3.2-9 Configure multiple digital outputs

At this point, the workflow "testModbusMaster" already contains an instruction for the robotic Modbus master station to read multiple digital outputs.

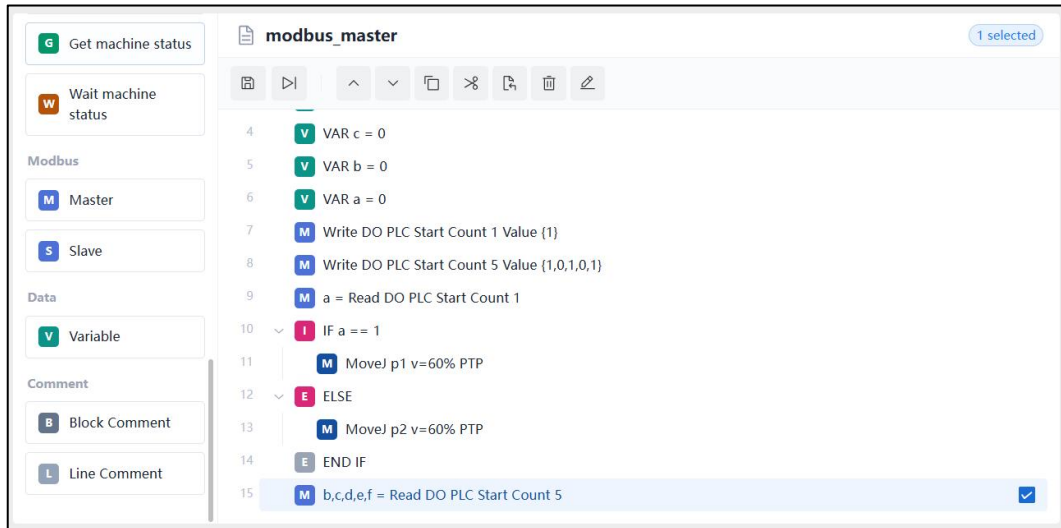


Figure 9.4.3.2-10 Read multiple numerical output workflow

9.4.3.2.5 Read digital input DI (Discrete Input)

Open the Modbus TCP Master Station Command Addition page, select "Modbus Master Station Name" as the master station "PLC" that was previously added on the Modbus Master Station Configuration page, set the DI name to "Servo Position", choose "Read Digital Input" as the action, specify 2 registers, and select "g, h" as the variables to assign values to. After executing the program, the two digital input values read from the device will be stored in variables "g" and "h" respectively; you can then use these variable values to control the robot to perform different operations. Click the "Apply" button at the bottom.

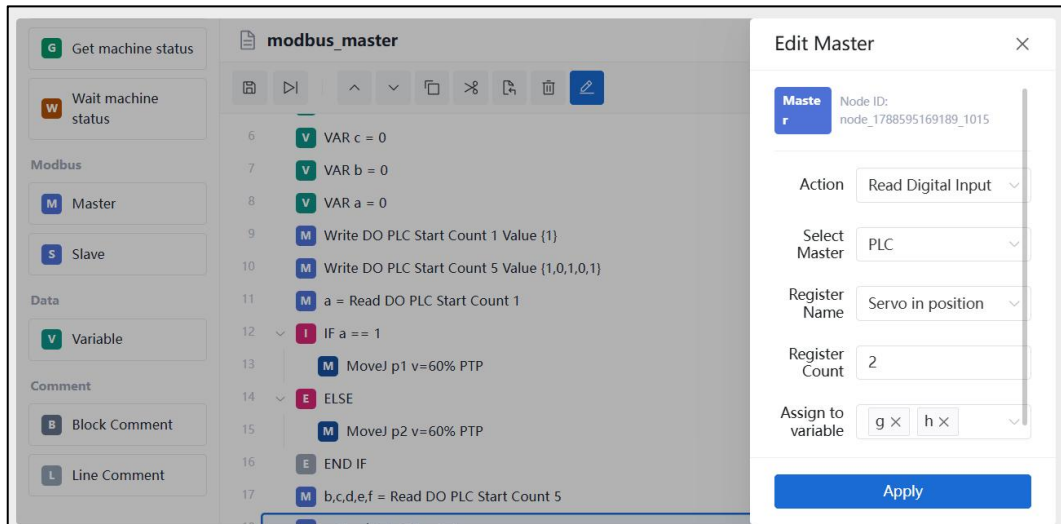


Figure 9.4.3.2-11 Configure digital input

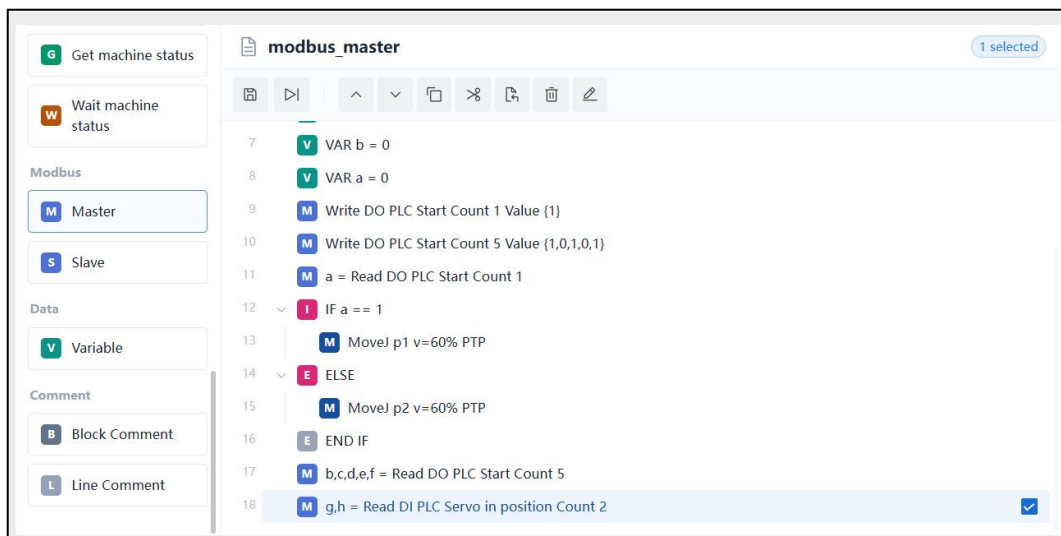


Figure 9.4.3.2-12 Read a numeric value and store it in a variable.

9.4.3.2.6 Read/write operations for the analog input AI (Input Register) and analog output AO (Hold Register)

The read/write operations for the analog input (AI – Input Register) and analog output (AO – Holding Register) are essentially identical to those for the digital input (DI – Discrete Input) and digital output (DO – Coil); the key difference lies in the fact that the data range for the latter is limited to only 0 or 1, whereas the data range for the former is much broader. For specific implementation details, please refer to the programming guidelines for digital input and digital output functions; here, we provide only example programs demonstrating the read operation for the analog input (AI) and the read/write operations for the analog output (AO).

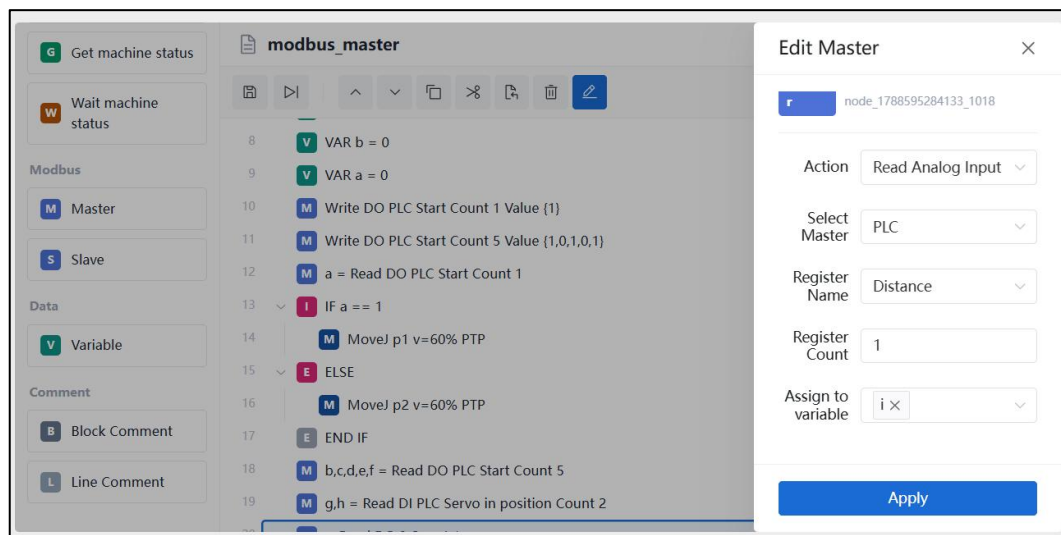


Figure 9.4.3.2-13 Read analog input (AI)

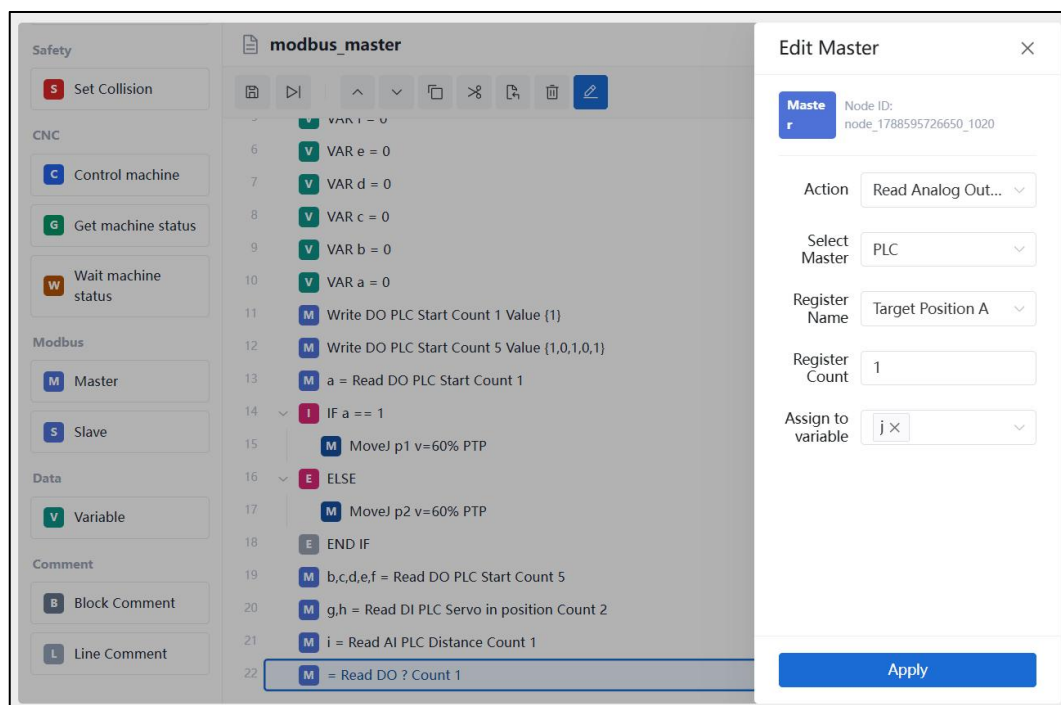


Figure 9.4.3.2-14 Read analog output (AO)

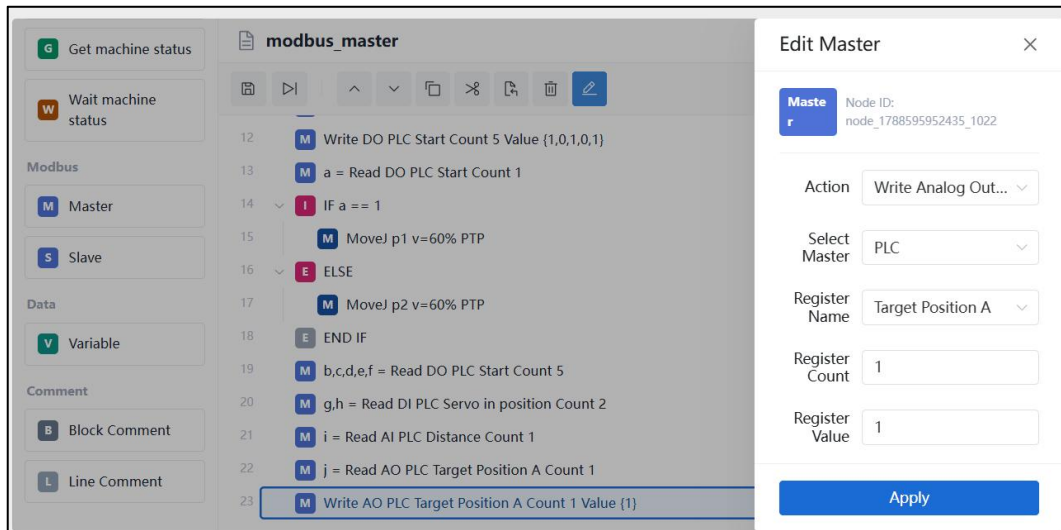


Figure 9.4.3.2-15 Write analog output (AO)

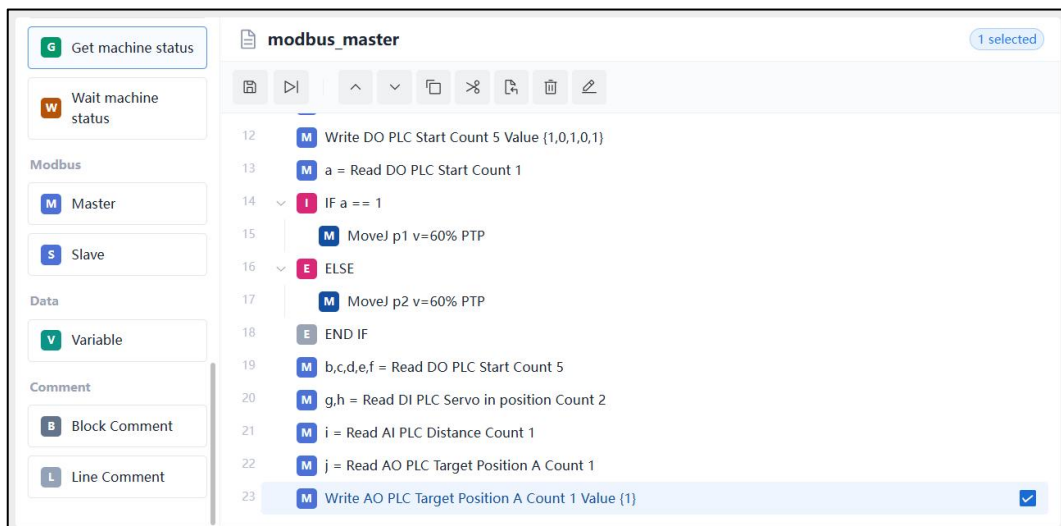


Figure 9.4.3.2-16 Read/Write workflow for the analog input AI (Input Register) and analog output AO (Hold Register)

9.4.3.2.7 Waiting for digital input

Open the Modbus TCP Master Station Command Addition page, select the "Wait for Digital Input" action (which corresponds to waiting for the DI discrete input configuration), choose the "Servo Position" register as the DI name, set the wait status to "True" and the timeout duration to 5000 ms, then click the "Apply" button at the bottom.

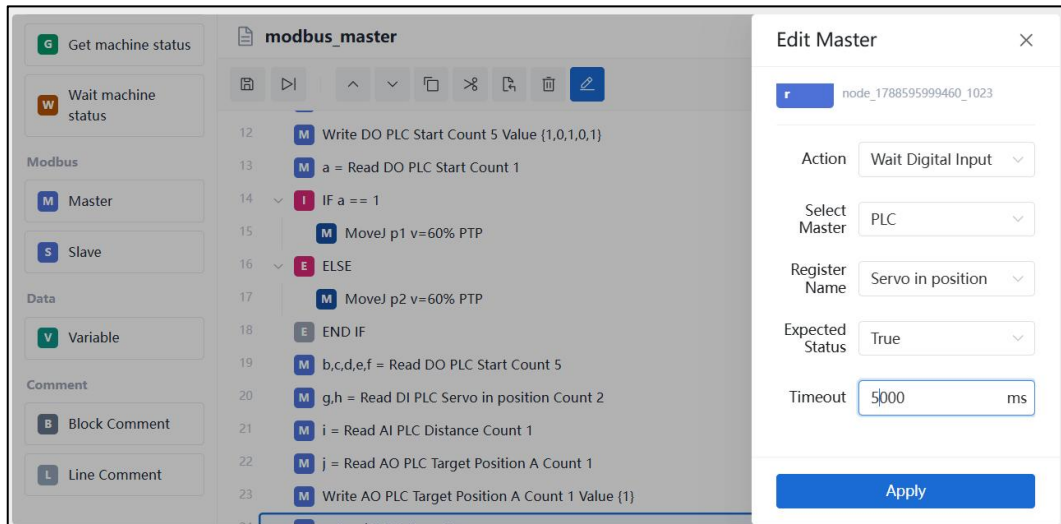


Figure 9.4.3.2-17 Add Wait DI Input Command

At this point, the workflow "testModbusMaster" already contains an instruction instructing the robot's Modbus master station to wait for a digital input (DI) signal. Upon starting the program, the robot will continuously wait until the value of the "Servo Position" register on the "PLC" master station changes to `true` – i.e., the value 1. Since the configured timeout duration is 5 seconds, if the "Servo Position" signal remains at `0` after 5 seconds have elapsed, the robot's program will raise a timeout error and the program will automatically stop running.

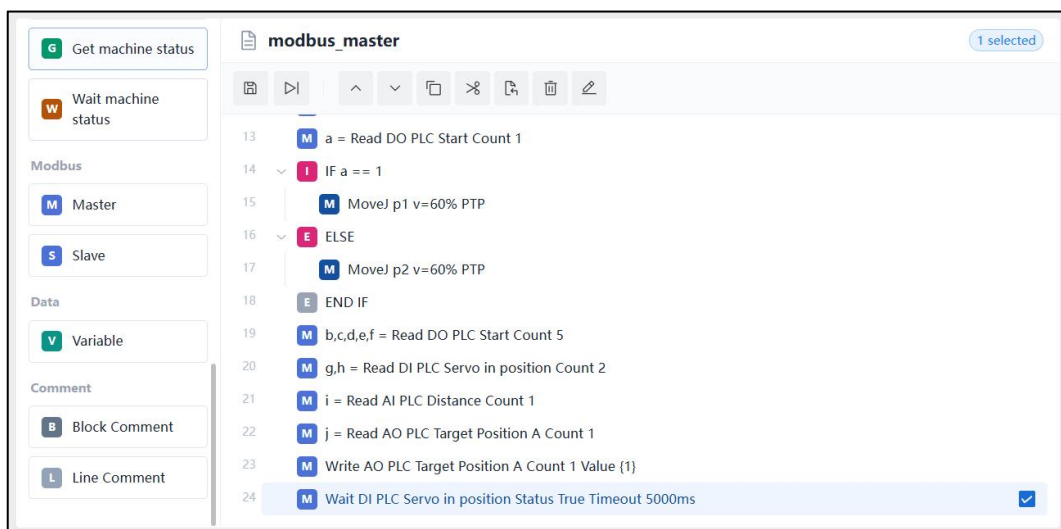


Figure 9.4.3.2-18 Waiting for Digital Input DI Workflow

9.4.3.2.8 Waiting for analog input

Open the Modbus TCP Master Station Command Addition Page, locate the "Wait for Analog Input" setting (i.e., the Wait for AI Input Register setting), select the "Liquid

Level" register (with the AI name set to "Liquid Level"), set the Wait Status to ">", the Register Value to 255, and the Timeout Duration to 5000 ms. Then, click the "Apply" button at the bottom.

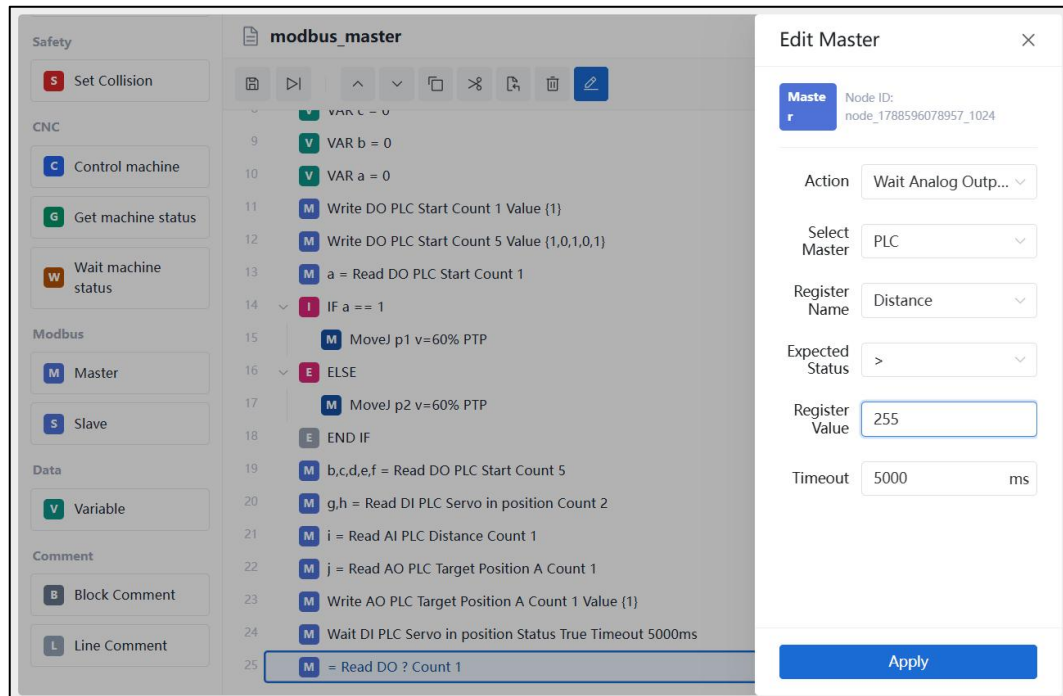


Figure 9.4.3.2-20 Add a waiting-simulated input wait state greater than

At this point, the workflow "testModbusMaster" already contains an instruction instructing the robotic Modbus master station to wait for the value of an AI input register. Upon starting the program, the robot will continuously wait for the value of the "Liquid Level" register on the "PLC" master station to exceed 255. Since the configured timeout duration is 5 seconds, if the "Liquid Level" signal remains ≤ 255 after 5 seconds have elapsed, the robot program will raise a timeout error and the program will automatically terminate.



Figure 9.4.3.2-21 Wait until the AI Input Register exceeds the workflow limit.

Open the Modbus TCP Master Station Command Addition page, select the "Wait for Analog Input" action (i.e., wait for the AI Input Register configuration), choose the "Liquid Level" register (with the AI name set to "Liquid Level"), set the wait status to "=", the register value to 255, and the timeout duration to 5000 ms. Then click the "Apply" button at the bottom.

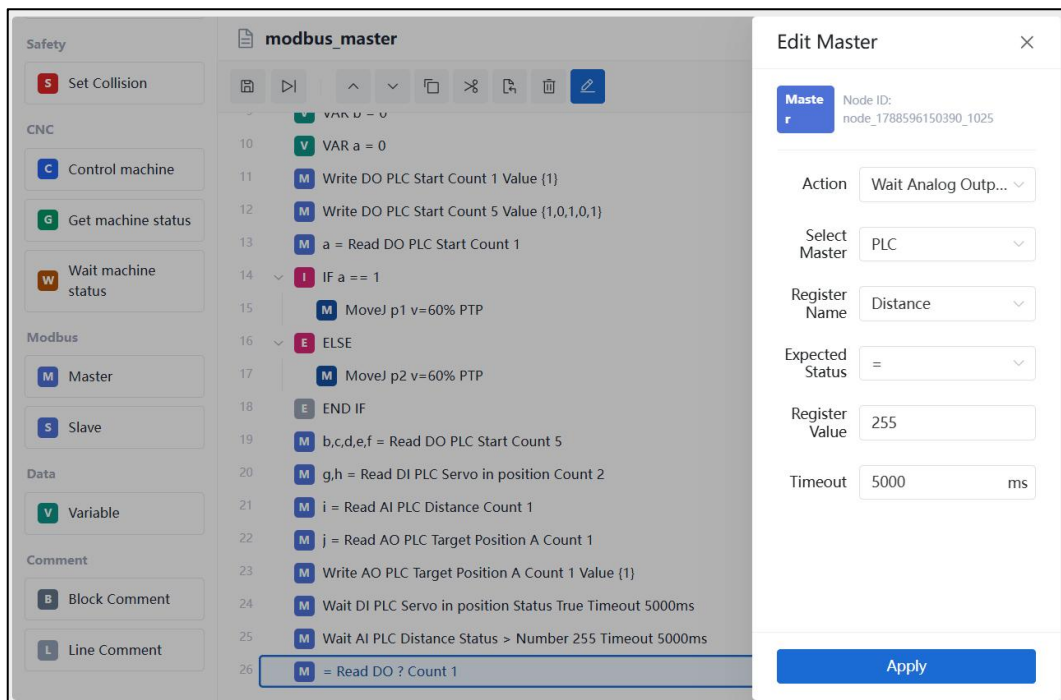


Figure 9.4.3.2-22 Adding the waiting-simulation input waiting state is equal to

At this point, the workflow "testModbusMaster" already contains an instruction instructing the robotic Modbus master station to wait for the value of the AI input register. Upon starting the program, the robot will continuously wait until the value of

the "Liquid Level" register on the "PLC" master station equals 255. Since the configured timeout duration is 5 seconds, if the "Liquid Level" signal does not equal 255 after the robot has been waiting for 5 seconds, the robot program will raise a timeout error and the program will automatically stop running.

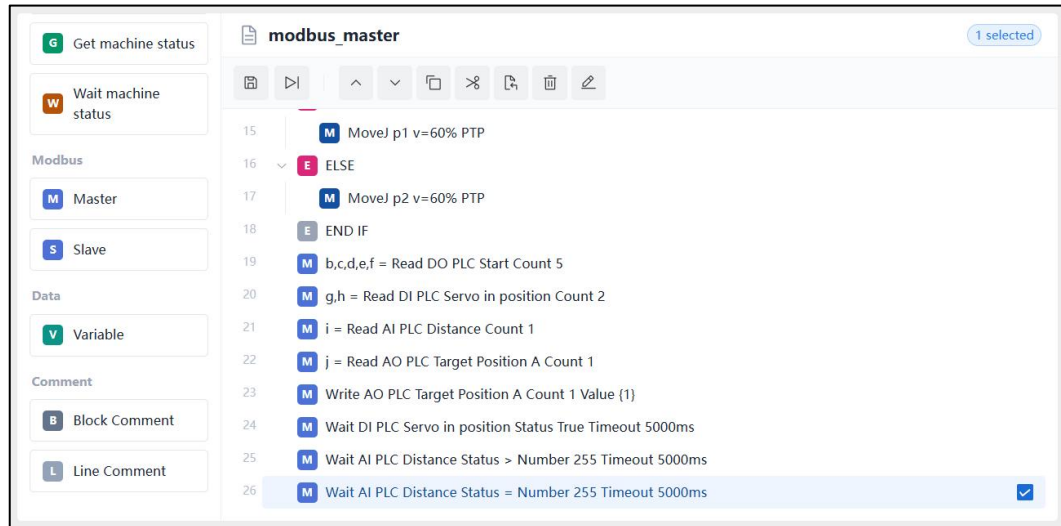


Figure 9.4.3.2-23 Adding a waiting-simulation input waiting state is equivalent to a workflow.

9.4.3.3 Slave station

The robotic Modbus TCP slave device supports four types of registers: General Digital Input (Coils), General Digital Output (Discrete Inputs), General Analog Input (Holding Registers), and General Analog Output (Input Registers). The General Digital Input and Analog Input registers are primarily used by the robot to read data from an external Modbus TCP master device to control its operations; whereas the General Digital Output and Analog Output registers are used by the robot to send data signals to an external Modbus TCP master device, allowing the external master device to read the values from these registers and thereby control the operation of its connected equipment. In addition to the aforementioned general input and output registers, the robot also provides certain "Functional Digital Input (Coils)" registers for external master devices to use when controlling the robot's start-up or stop procedures, as well as specific input registers for displaying the current status information of the robot – including the robot's current Cartesian position and current operating status (for detailed definitions, please refer to the attached document: Modbus TCP Slave Device Address Mapping Table). The typical usage process for the robotic Modbus TCP slave device includes: ① Parameter configuration; ② Communication testing; ③ Program development.

9.4.3.3.1 Modbus TCP Slave Device Communication Parameter Configuration

In the "Loading/Unloading - Workflow" section, create a new workflow named "testModbusSlave".

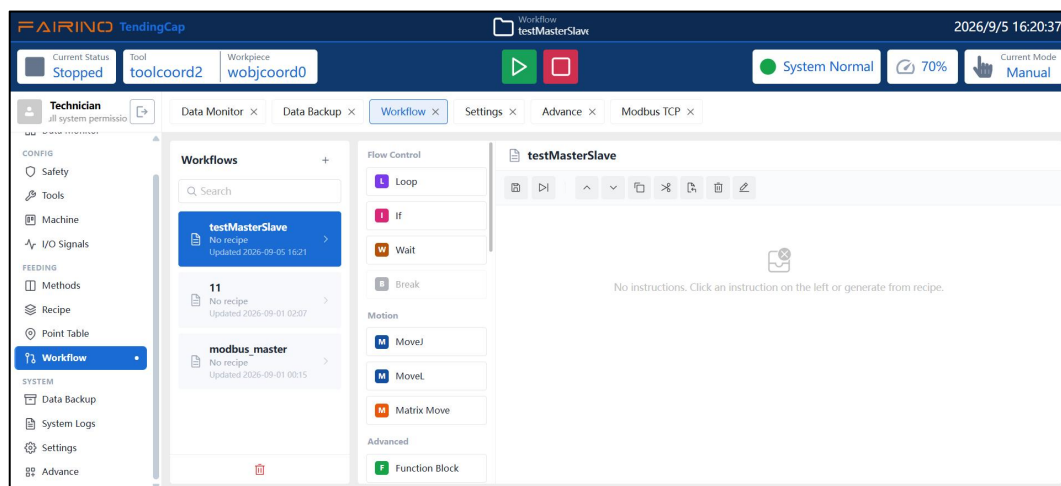
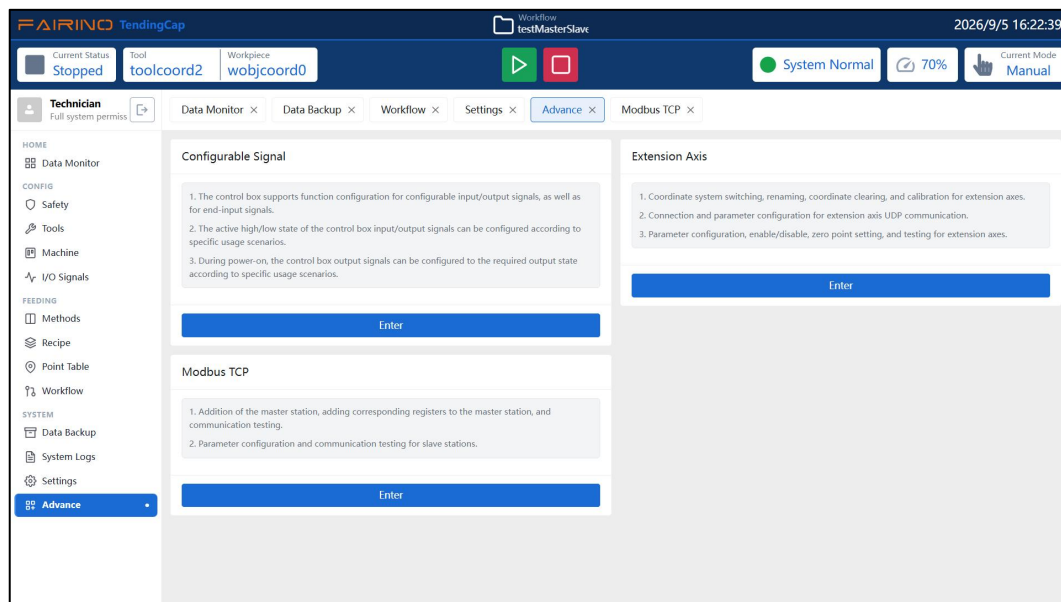


Figure 9.4.3.3-1 Create a Modbus TCP slave device workflow

Click the "Enter" button at the bottom of the "Modbus TCP" section in the Advanced Parameters to open the Modbus TCP Function Configuration page.



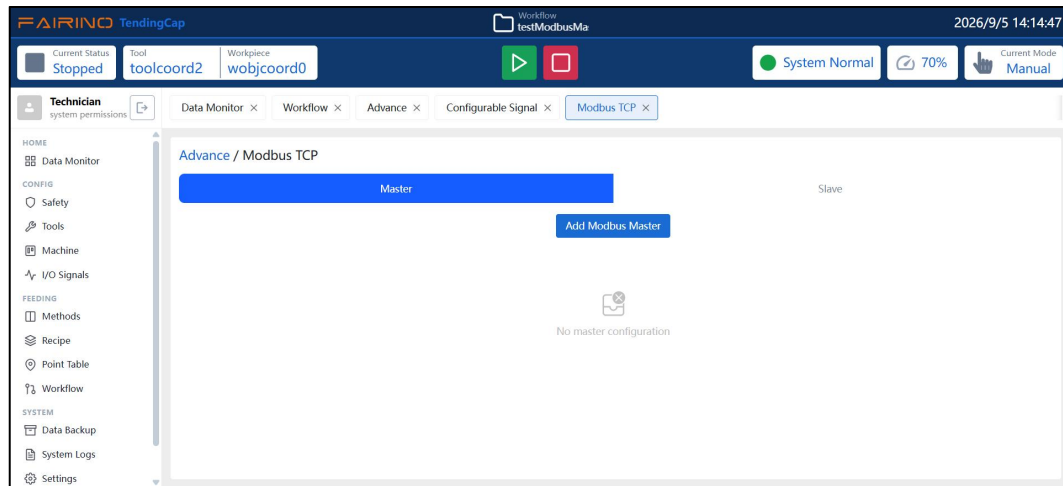


Figure 9.4.3.3-2 Open Modbus TCP Configuration

Click "Slave Station," then enter the slave robot's IP address, port number, and slave station ID. Here, "IP" refers to the IP address of the slave robot. The Faao collaborative robot has two network ports – one for the teach pendant and one for the control box – each with a different IP address. Enter the correct IP address according to the network port used to connect the external device to the slave robot (it is recommended to use the network port on the control box). After modifying the slave robot's Modbus TCP IP address, port number, or slave station ID, restart the robot for the changes to take effect.

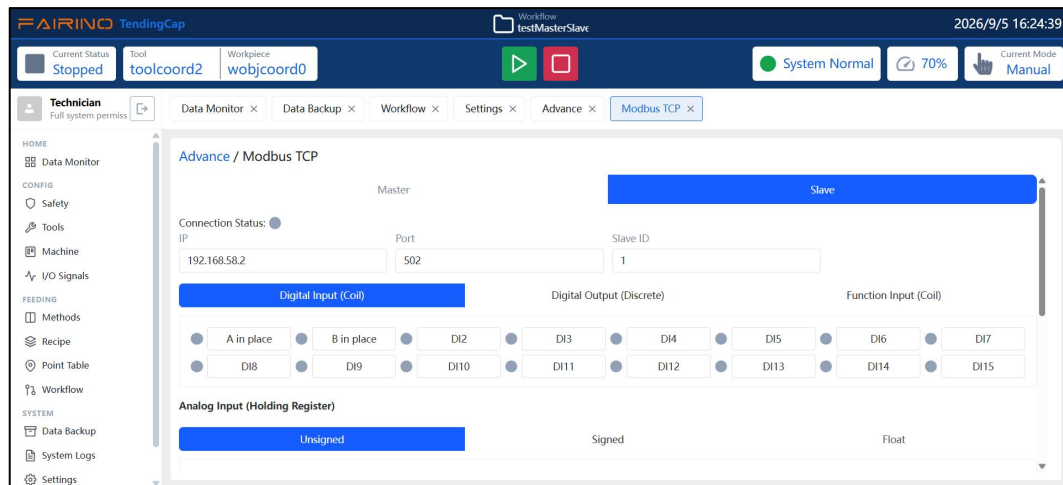


Figure 9.4.3.3-3 Modbus TCP Slave Device Configuration

After completing the Modbus TCP slave device parameter configuration and restarting the robot, the external master station device can establish a connection with the robot slave device using the configured parameters. Upon successful connection, the "Connection Status" indicator light on the robot slave device's configuration page will illuminate.

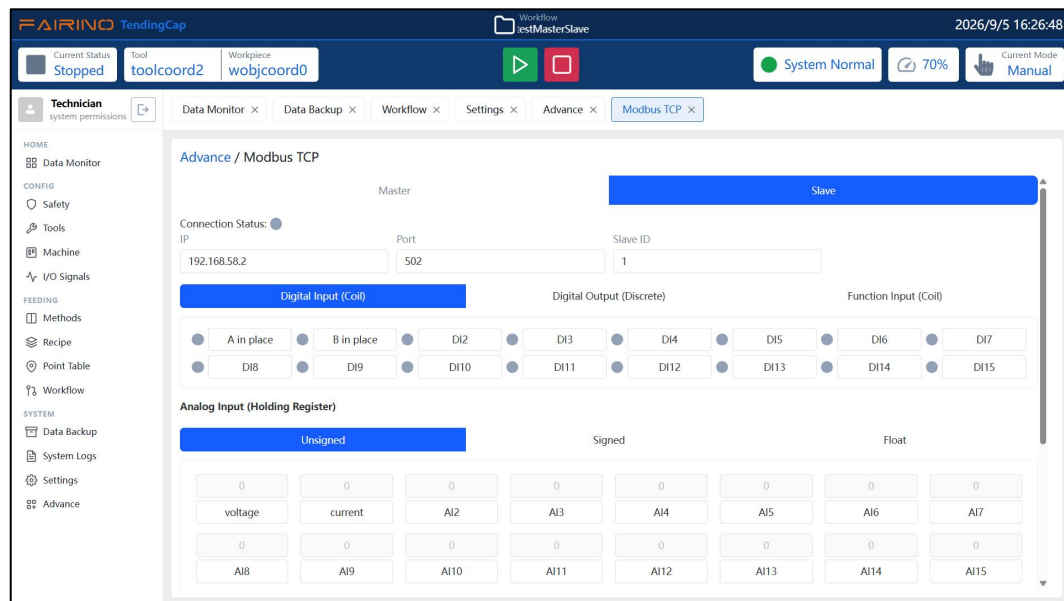


Figure 9.4.3.3-4 Slave Station Connection Status Indicator Light

9.4.3.3.2 ModbusTCP slave device communication test

● General-purpose digital input (coil): The robot Modbus TCP slave device provides 128 coil registers, with their register addresses ranging from 100 to 127 (Note: For the specific definitions, please refer to the Modbus TCP Slave Device Address Mapping Table). All general-purpose registers on the robot Modbus TCP slave device can be assigned aliases; for example, the name of the robot slave device's coil register DI0 can be changed to "A-Position", and the name of DI1 can be changed to "B-Position". According to the address mapping table, the Modbus coil addresses for "A-Position" and "B-Position" are 100 and 101, respectively. On an external Modbus TCP master device, set both the robot slave device's coil register addresses 100 and 101 to the value 1; at this point, the indicator lights for these two registers will illuminate on the robot Modbus TCP slave device monitoring page.

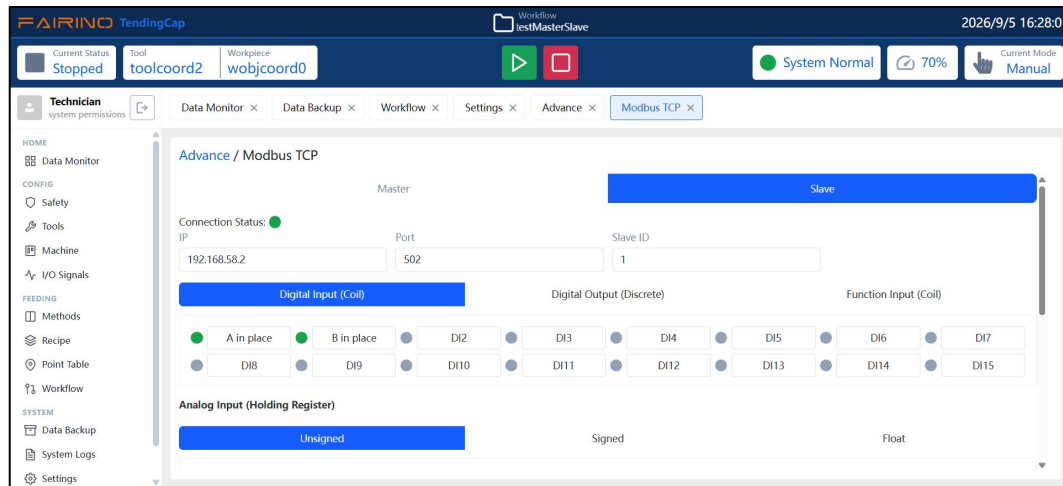


Figure 9.4.3.3-5 Modbus TCP Slave Coil Status Monitoring

- **General Digital Output (Discrete Input):** The robot Modbus TCP slave device provides 128 discrete input registers, with their register addresses ranging from 100 to 127 (Note: For specific definitions, please refer to the Modbus TCP Slave Device Address Mapping Table). The discrete input registers of the robot Modbus TCP slave device can also be assigned aliases; click "General Digital Output (Discrete Input)" to rename the robot slave device's discrete input register DO0 as "A Start" and DO1 as "B Start". According to the address mapping table, the Modbus discrete input addresses for "A Start" and "B Start" are 100 and 101, respectively. Click the discrete input indicator corresponding to "A Start"; this indicator will illuminate, and the value at register address 100 will change to 1 – this value can be read from an external Modbus TCP master device.

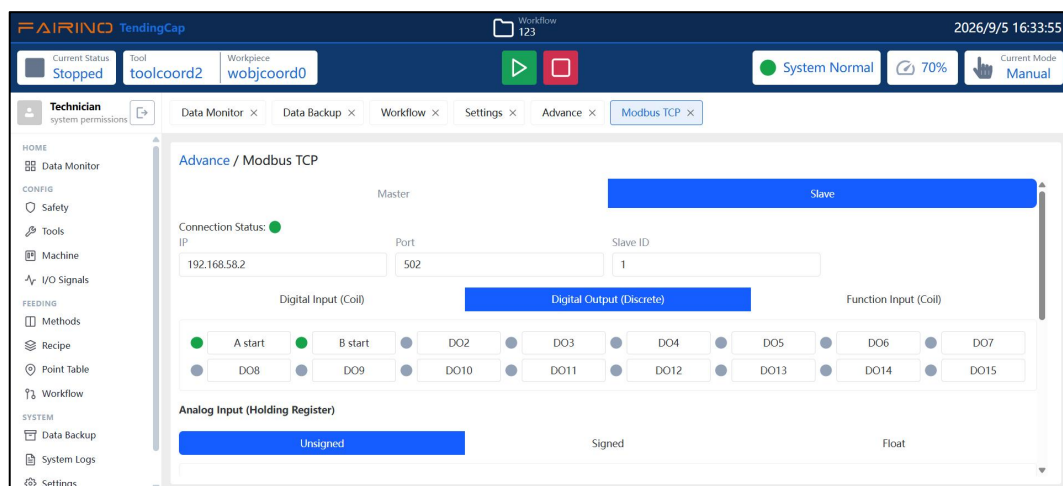


Figure 9.4.3.3-6 Modbus TCP Slave Coil Status Monitoring

- **Analog Input (Holding Registers):** The robot provides a total of 64 holding registers across three data types—unsigned, signed, and floating-point. The addresses

for AI0 - AI63 range from 100 to 195 (for detailed definitions, please refer to the Modbus TCP Slave Address Mapping Table). The data range for unsigned registers is 0 - 65535; for signed registers, it is - 32768 - 32767; and for floating-point registers, the data is displayed in big-endian format. Change the names of AI0 and AI1 to "Voltage" and "Current" respectively; according to the Modbus TCP Slave Address Mapping Table, the addresses for these two registers are 100 and 101, respectively. Therefore, when the connected master station device modifies the values stored in holding registers at addresses 100 and 101, the corresponding values displayed under the "Voltage" and "Current" registers on the robot's Modbus TCP slave station monitoring page will be synchronously updated. The robot's analog input is primarily used to read value signals from external master station devices.

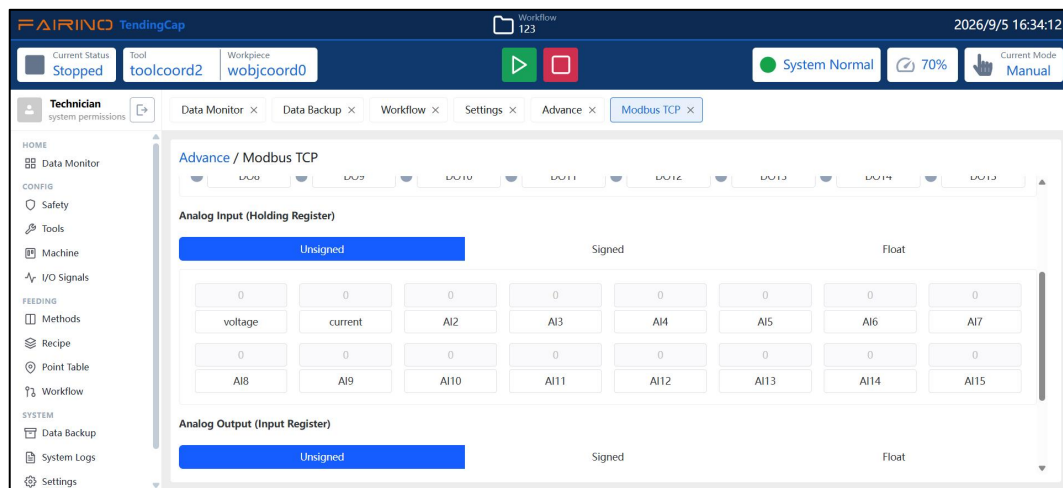


Figure 9.4.3.3-7 Modbus TCP slave device analog input monitoring

- **Analog Output (Input Registers):** The robot provides a total of 64 input registers spanning three data types — unsigned, signed, and floating-point. The addresses for AO0 - AO63 range from 100 to 195 (Note: For detailed definitions, please refer to the Modbus TCP Slave Address Mapping Table; the data range for unsigned registers is 0 - 65535, for signed registers it is - 32768 - 32767, and floating-point registers use big-endian format for display). Change the names of AO0 and AO1 to "Target Position A" and "Target Position B" respectively; set the input register values to 2000 and 1500, respectively. According to the Modbus TCP Slave Address Mapping Table, the addresses for these two registers are 100 and 101, respectively. Therefore, when the connected master station device reads the values from the input registers at addresses 100 and 101, it will obtain the configured values. The robot's analog output function is primarily used to transmit numerical signals from the

robot to external master station devices.

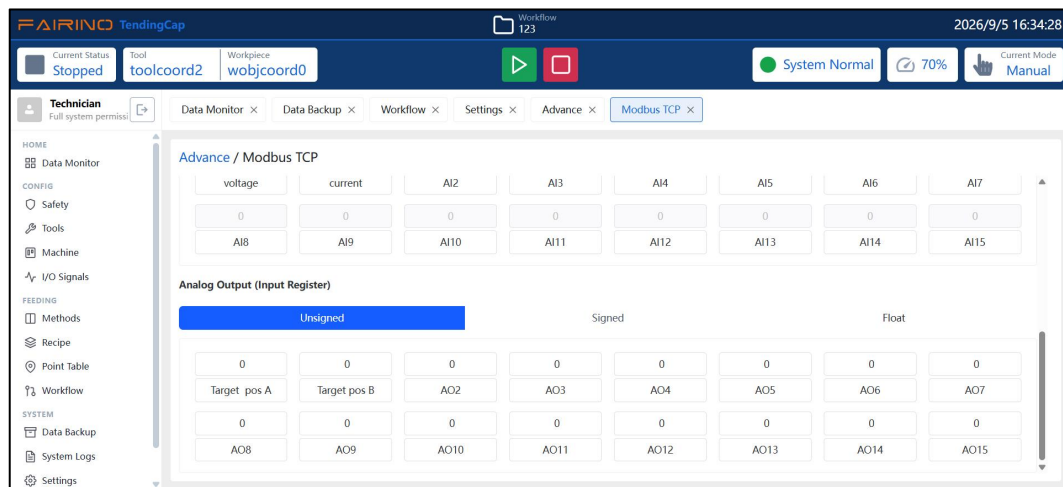


Figure 9.4.3.3-8 Modbus TCP slave device – Modify analog input

9.4.3.4 Modbus TCP Slave Device Programming

In the "Loading/Unloading – Workflow" section, open the workflow "testModbusSlave", click on the "Modbus" slave device, and then access the Modbus TCP slave device command addition page.

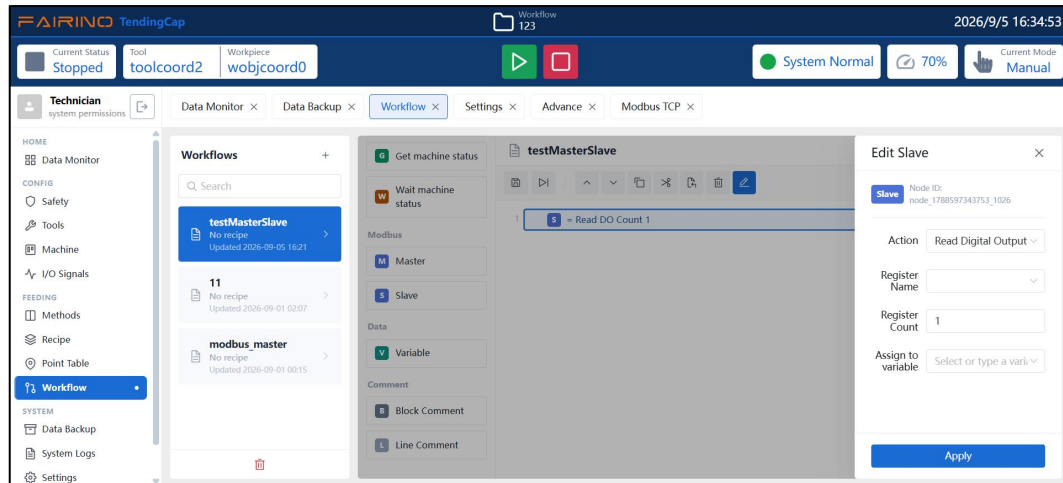


Figure 9.4.3.3-9 Modbus TCP slave device – Modify analog input

9.4.3.4.1 Write a single digit to output DO (Discrete Input).

Select the action "Write Digital Output", set the DO name to "A Start", specify 1 register, and set the register value to 0. Then click the "Apply" button at the bottom.

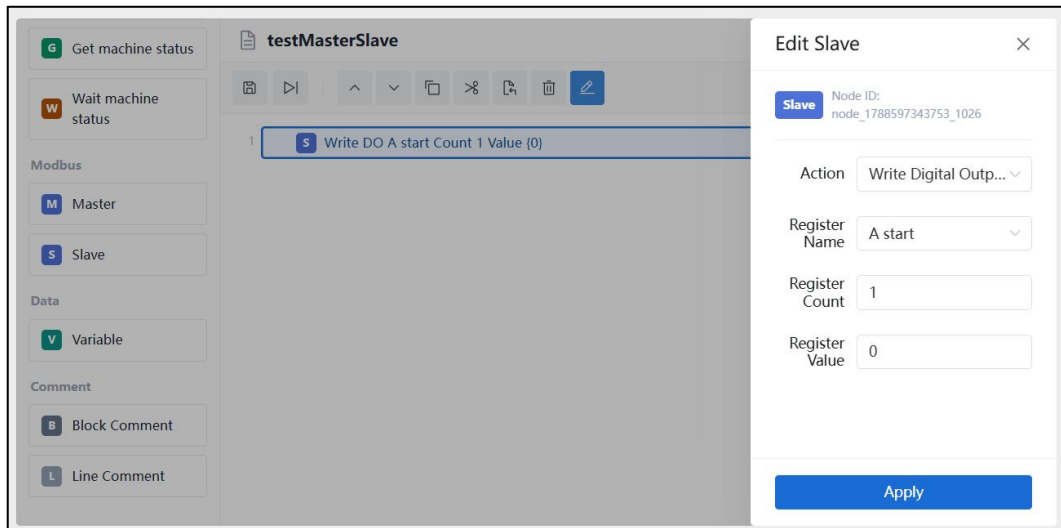


Figure 9.4.3.3-10 Add a single-digit output instruction

At this point, the workflow "testModbusSlave" already contains a command instructing the robotic Modbus slave device to write a single digital output value; switch the robot to automatic mode, then click the "Start" button – the robot will set the address value of the digital output corresponding to the "A Start" label to 0.

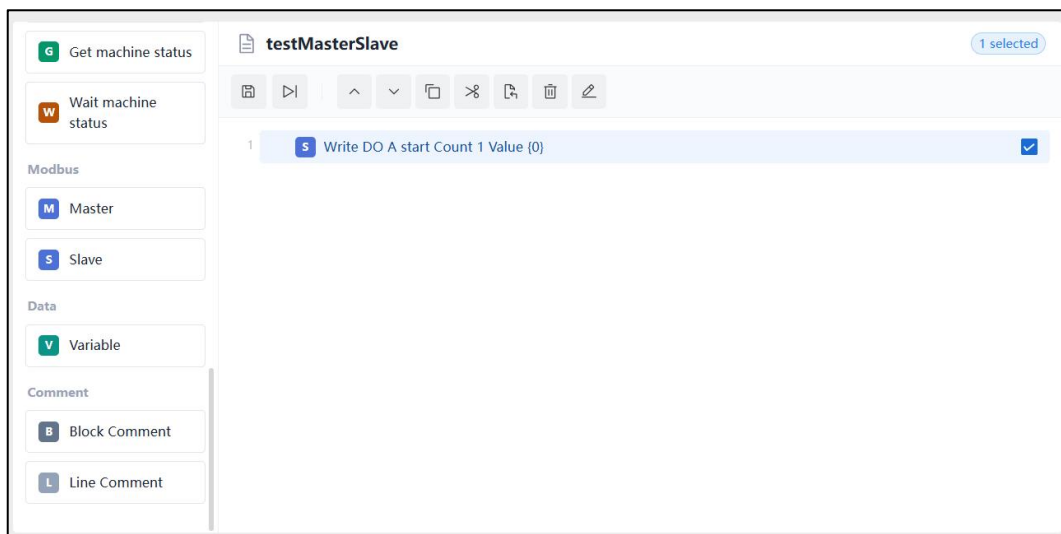


Figure 9.4.3.3-11 Write single-digit output workflow

9.4.3.4.2 Output multiple numerical values as DO (Discrete Input).

Open the Modbus TCP Slave Instruction Addition page, select the "Write Digital Output" action, choose the DO name as "A Start", set the number of registers to 5, and set the register values to 1, 0, 1, 0, 1. The number of register values must match the number of registers specified, and multiple register values should be separated by English commas. Click the "Apply" button at the bottom.

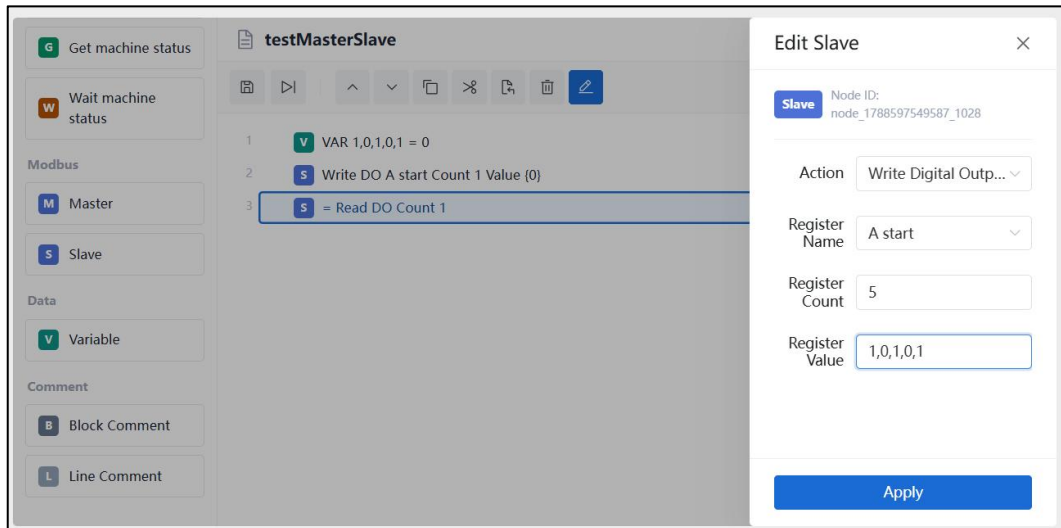


Figure 9.4.3.3-12 Configure multiple digital outputs

At this point, the workflow "testModbusSlave" already contains a command instructing the robotic Modbus slave device to write multiple digital output values; switch the robot to automatic mode and click the "Start" button – the robot will then write the values 1, 0, 1, 0, and 1, respectively, into the next four discrete input registers of slave device "A".

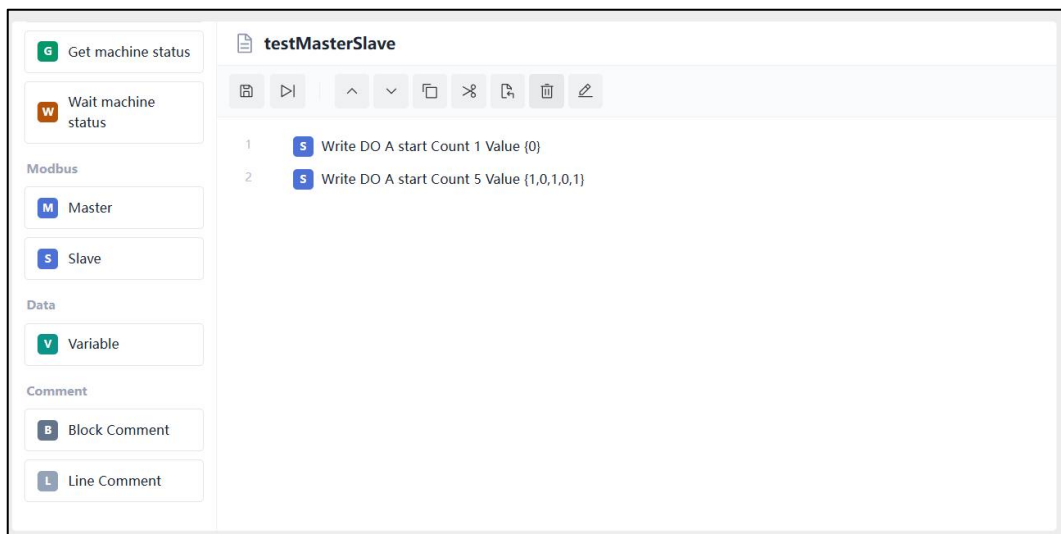


Figure 9.4.3.3-13 Write multiple numbers to output workflow

9.4.3.4.3 Read a single-digit output DO (Discrete Input)

Open the Modbus TCP Master Station Command Addition page, select the action "Read Digital Output", set the DO name to "A Start", specify the number of registers as 1, enter the variable value "a", and after executing the program, the read value will be stored in "a". Click the "Apply" button at the bottom.

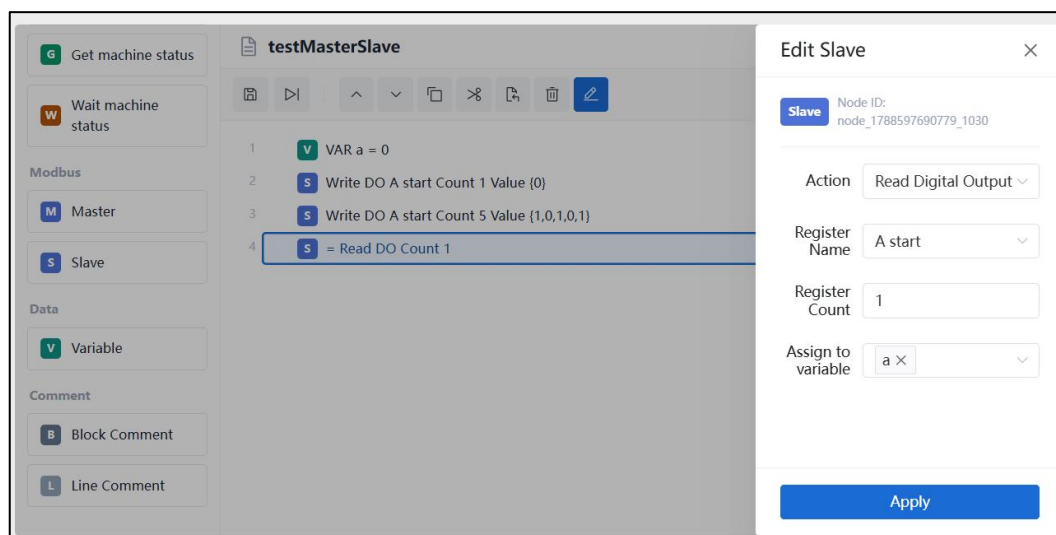


Figure 9.4.3.3-14 Configure for single-digit output

At this point, the workflow "testModbusSlave" already contains a command for a robotic Modbus slave device to read a single digital output.

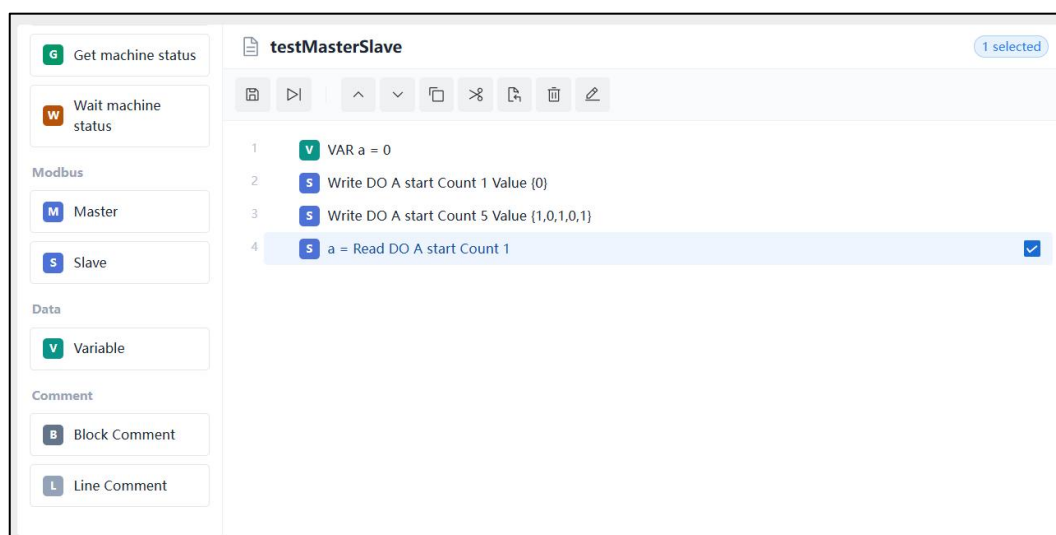


Figure 9.4.3.3-15 Read single-digit output workflow

The coil-based register value can only take two possible values: 0 or 1; in a robot program, different operations can be executed by checking whether the register value is different. Click on the "Judge" command and select "Otherwise Judge," then add two joint movement commands to move the robot to two distinct positions – "P1" and "P2." When the "a" value is 1, the robot moves to position "P1"; otherwise, the robot moves to position "P2."

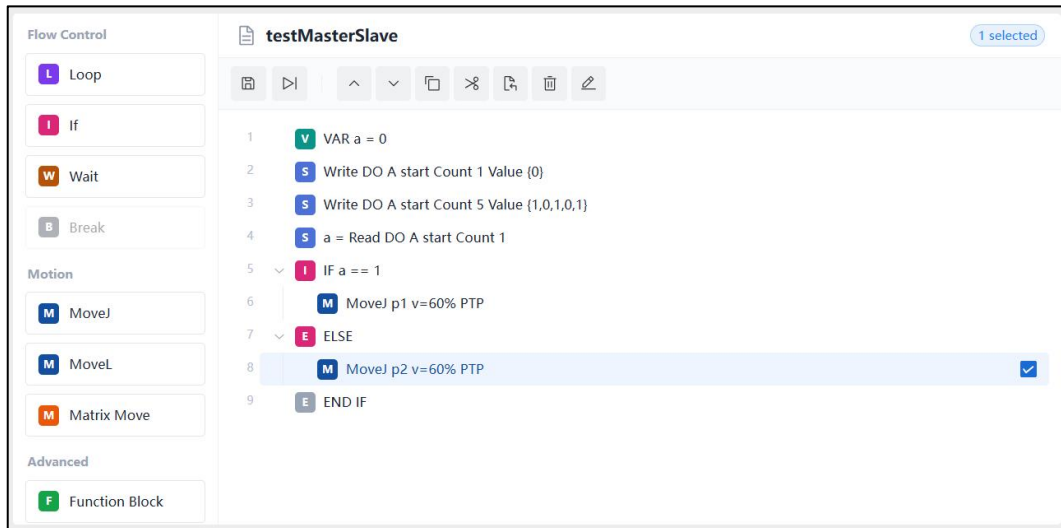


Figure 9.4.3.3-16 Read a single-digit output workflow – determines the output value based on the numerical position during movement.

9.4.3.4.4 Read multiple numerical outputs (DO – Discrete Input)

Open the Modbus TCP Master Station Command Addition page, select "Write Digital Output" as the operation, specify the DO name as "A Start", set the number of registers to 2, and choose "b" and "c" as the variable assignment options. After executing the program, the values read from the two digital output registers will be stored in the aforementioned variables "b" and "c" respectively; you can then use these values to control the robot to perform different actions. Click the "Apply" button at the bottom.

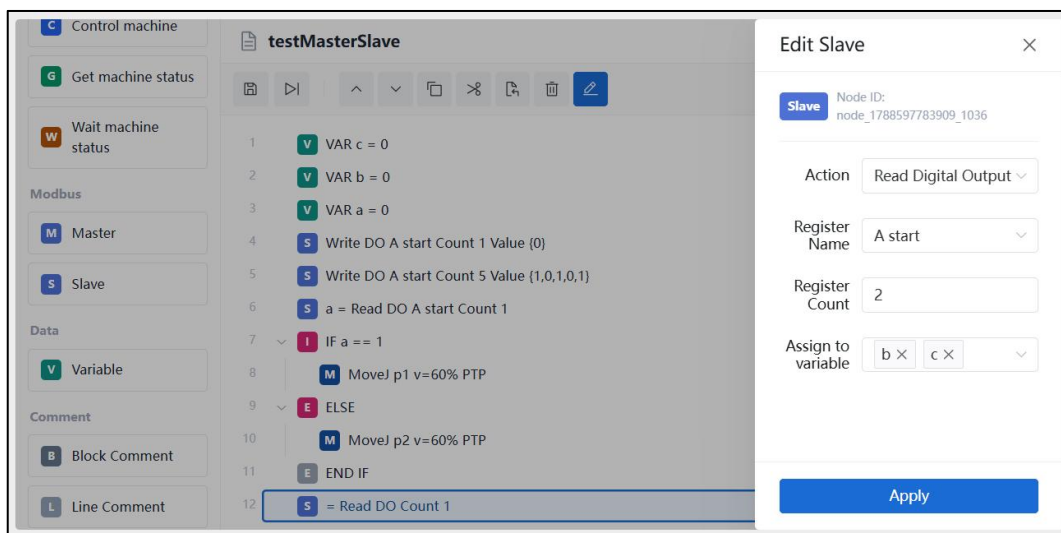


Figure 9.4.3.3-17 Configure multiple digital outputs

At this point, the workflow "testModbusSlave" already contains a command for

the robotic Modbus slave device to read multiple digital outputs.

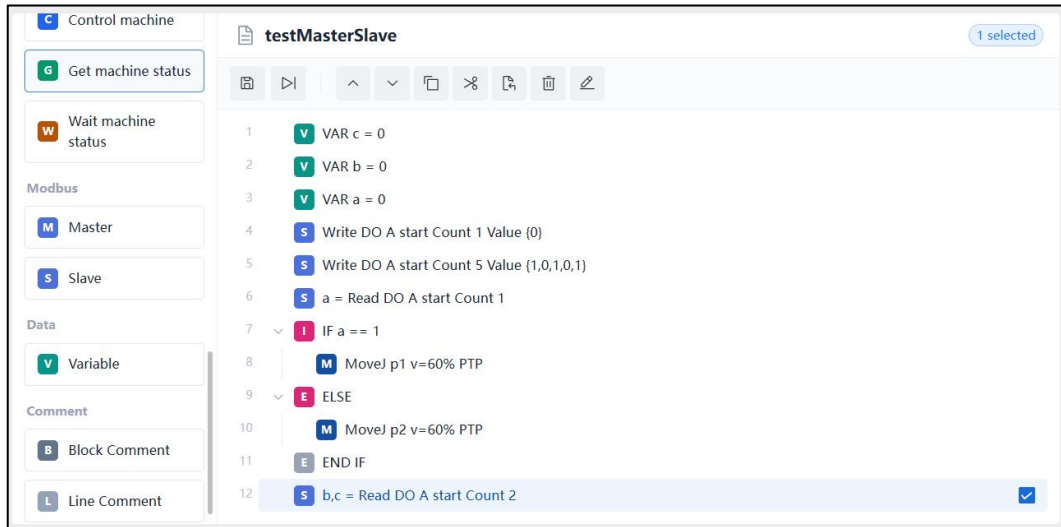


Figure 9.4.3.3-18 Read multiple numerical output workflow

9.4.3.4.5 Read digital input DI (coil)

Open the Modbus TCP Slave Instruction Addition page, select "Read Digital Input" as the operation, choose the DI name as "A-Position", set the number of registers to 2, and select "d, e" as the variable assignment options. After executing the program, the two digital input values read from the device will be stored in variables "d" and "e" respectively; you can then use these variable values to control the robot to perform different operations. Click the "Apply" button at the bottom.

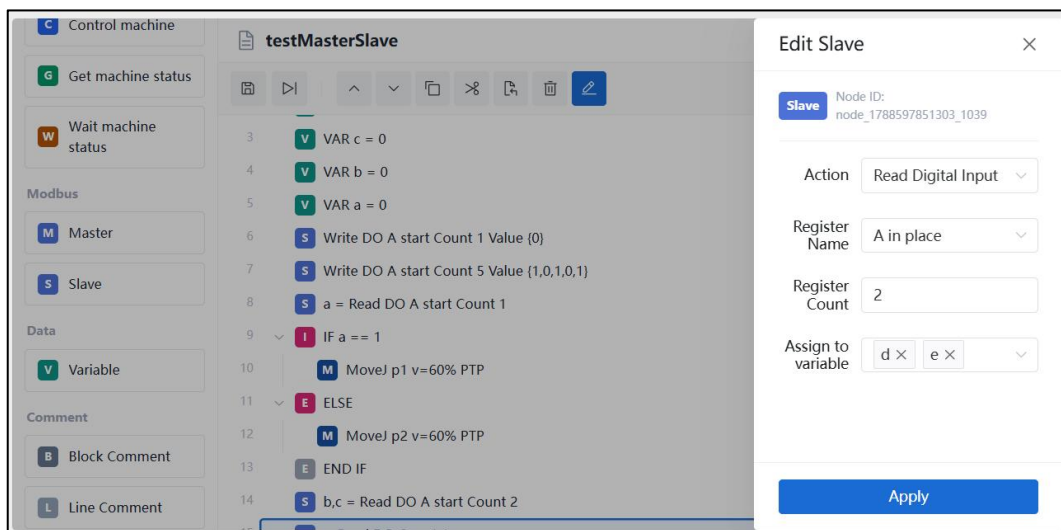


Figure 9.4.3.3-18 Configure digital input

At this point, the workflow "testModbusSlave" already contains a command for the robotic Modbus slave device to read digital inputs.

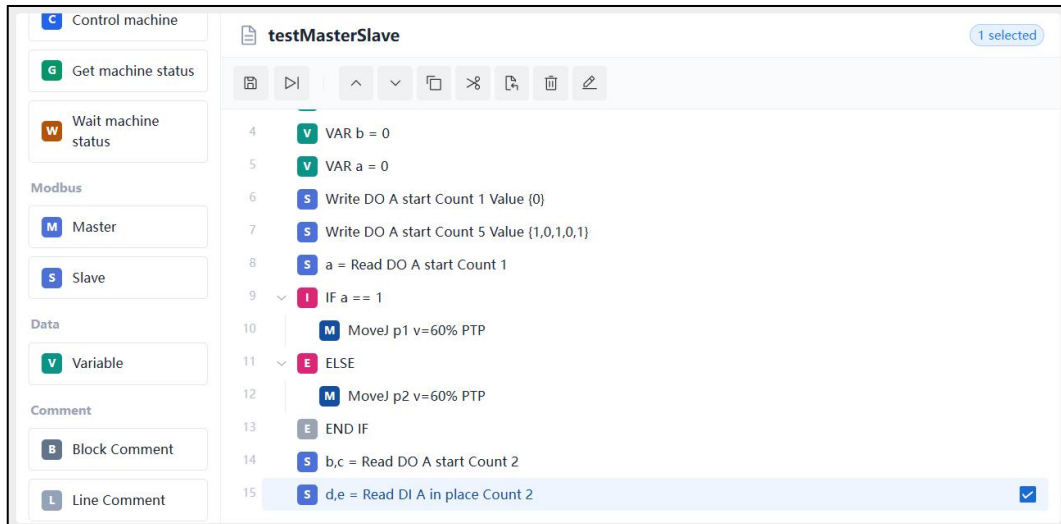


Figure 9.4.3.3-19 Read Digital Input Workflow

9.4.3.4.6 Read/write operations for the analog output (AO – Input Register) and analog input (AI – Hold Register)

The read/write operations for analog output (input register) and analog input (hold register) are essentially identical to those for digital output (discrete input) and digital input (coil); the key difference lies in the fact that the data range for the latter is limited to only 0 or 1, whereas the data range for the former is much broader. Therefore, for specific implementation details, please refer to the programming examples for digital output and digital input; here, we provide only sample code snippets for reading analog input and for reading and writing analog output.

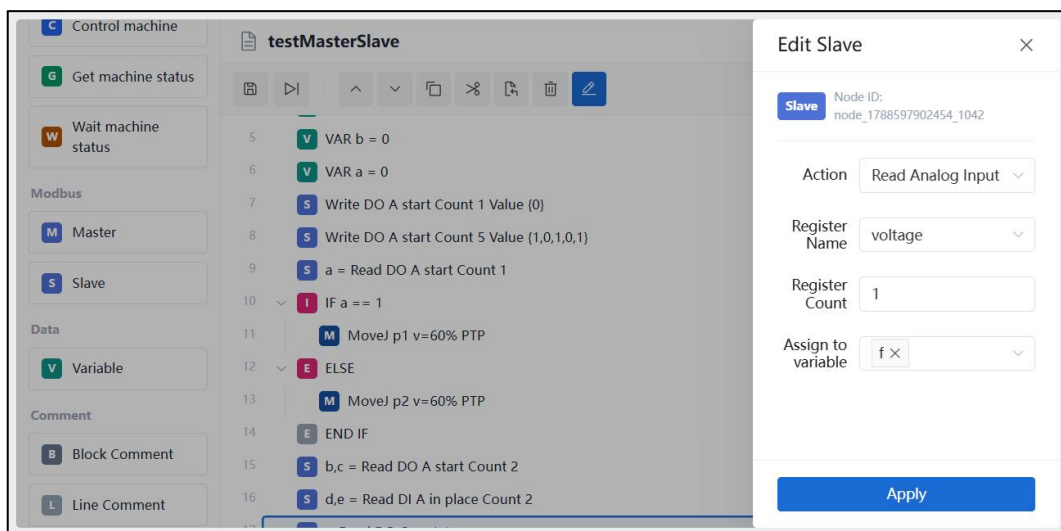


Figure 9.4.3.3-20 Read analog input

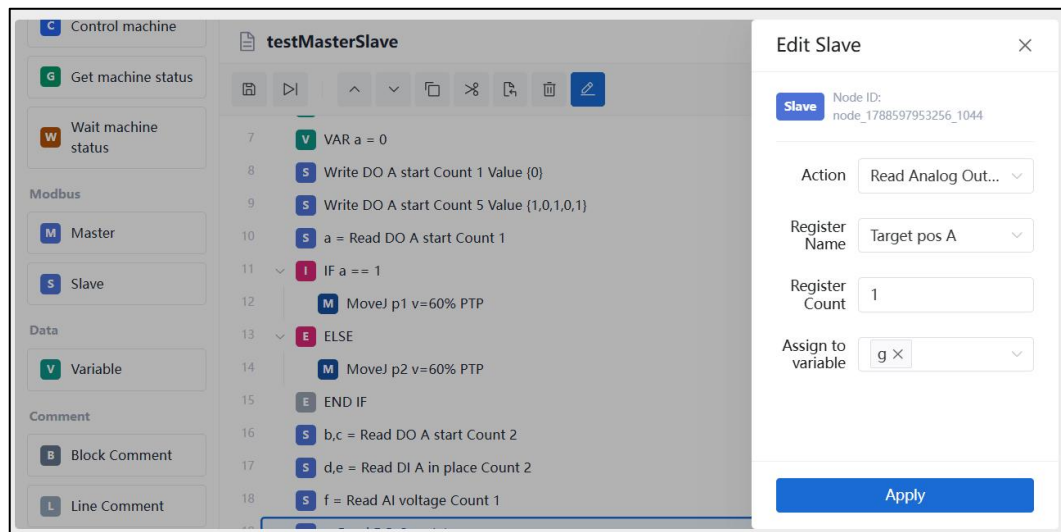


Figure 9.4.3.3-20 Read analog output

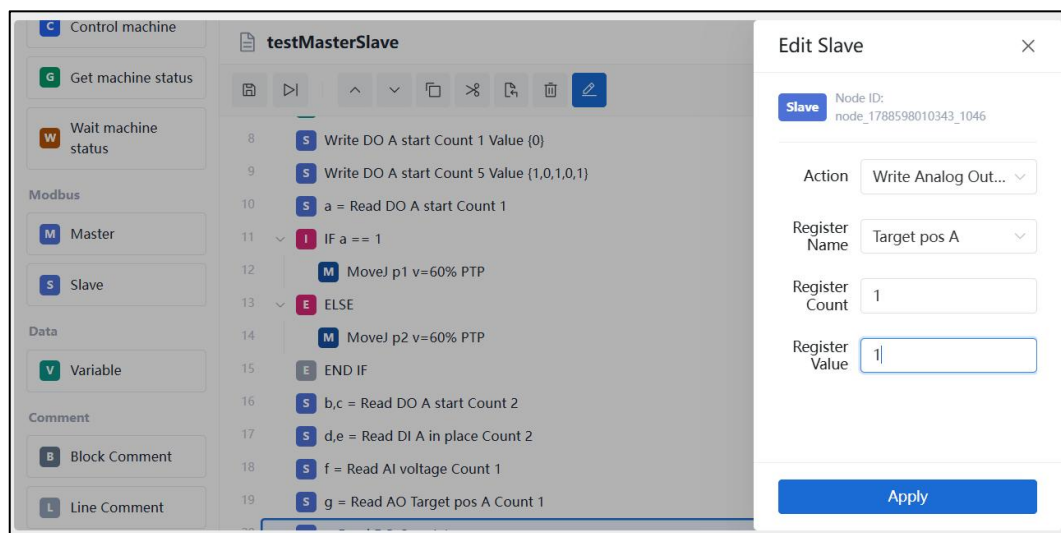


Figure 9.4.3.3-21 Write analog output

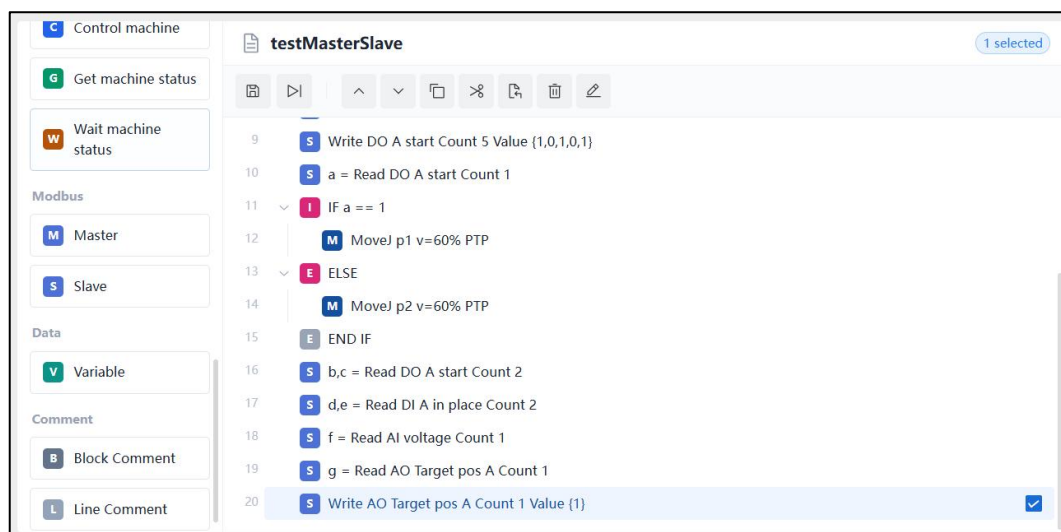


Figure 9.4.3.3-22 Analog Output (AO – Input Register) and Analog Input (AI – Hold

Register) workflow

9.4.3.4.7 Waiting for digital input

Open the Modbus TCP Slave Instruction Addition page, select the action "Wait for Digital Input", choose the DI register named "A-Position" (configured as the "A-Position" register), set the wait status to "True" and the timeout duration to 5000 ms. Then click the "Apply" button at the bottom.

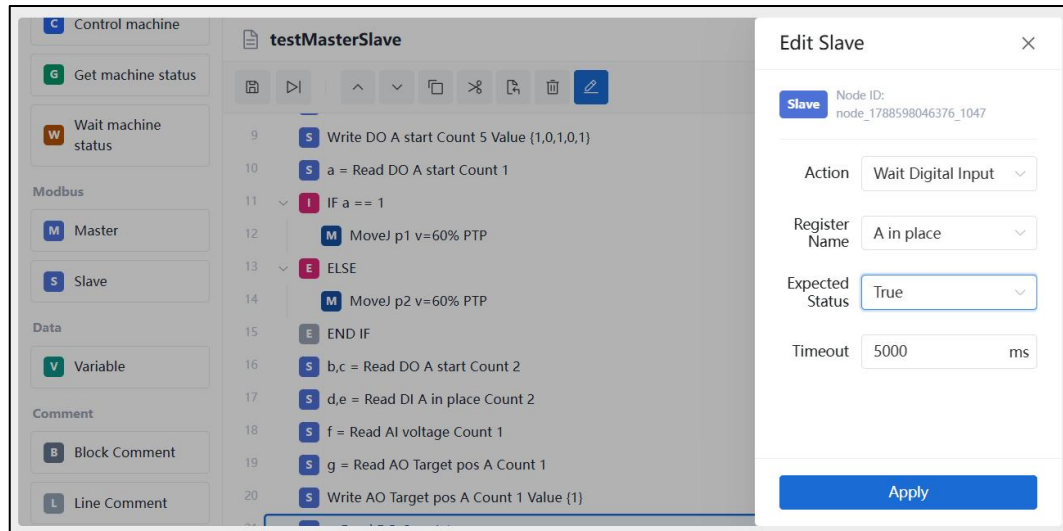


Figure 9.4.3.3-23 Waiting for a digital input command

At this point, the robot program "testModbusSlave.lua" has already been populated with an instruction instructing the robot's Modbus slave device to wait for a digital input. Upon launching the program, the robot will continuously wait until the value of the "A-Position" coil register on the slave device changes to 'true' – i.e., the value 1. Since the configured timeout duration is 5 seconds, if the "A-Position" signal remains at '0' after 5 seconds have elapsed, the robot program will raise a timeout error and the program will automatically terminate execution.

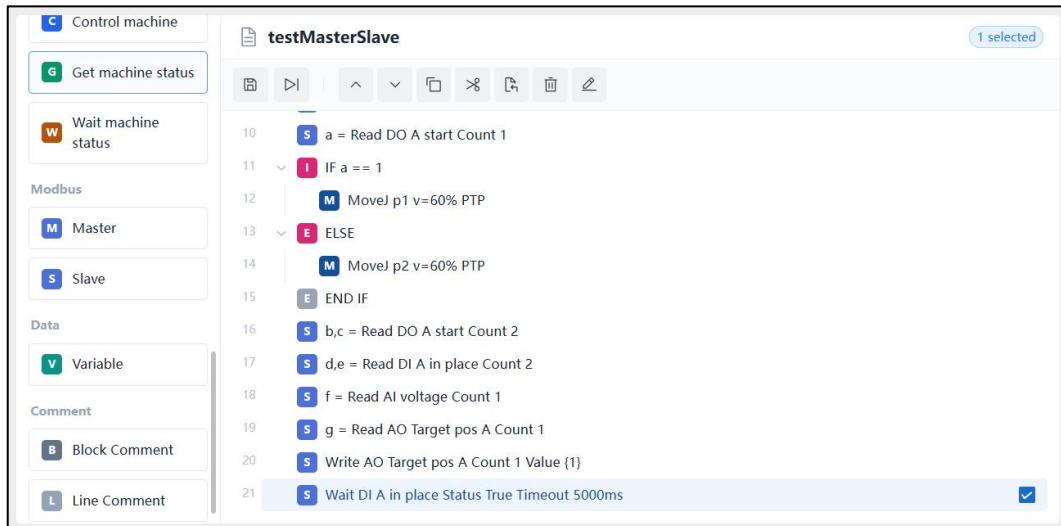


Figure 9.4.3.3-24 Waiting for Digital Input Workflow

9.4.3.4.8 Waiting for analog input

Open the Modbus TCP Slave Instruction Addition page, select the action "Wait for Analog Input Configuration", choose the "Voltage" register (with the AI name set to "Voltage"), set the wait status to ">", the register value to 255, and the timeout duration to 5000 ms. Then click the "Apply" button at the bottom.

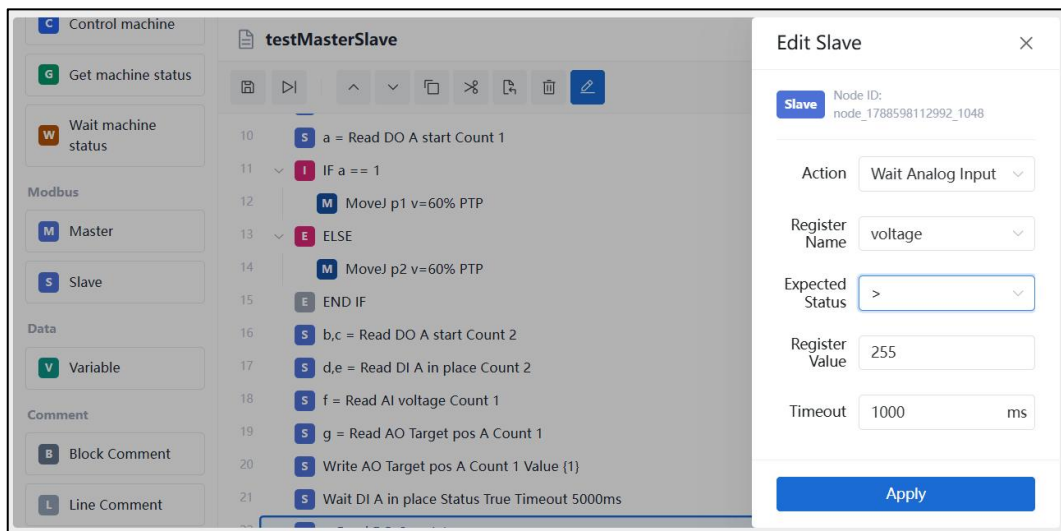


Figure 9.4.3.3-25 Adding a Wait Simulation Input Instruction when the system is not in the Wait State

At this point, the workflow "testModbusSlave" already contains an instruction instructing the robotic Modbus slave device to wait for a simulated input value. Upon starting the program, the robot will continuously wait until the value of the slave device's "Voltage" register exceeds 255. Since the configured timeout duration is 5

seconds, if the "Voltage" signal remains below 255 after 5 seconds have elapsed, the robot's program will raise a timeout error and the program will automatically stop running.

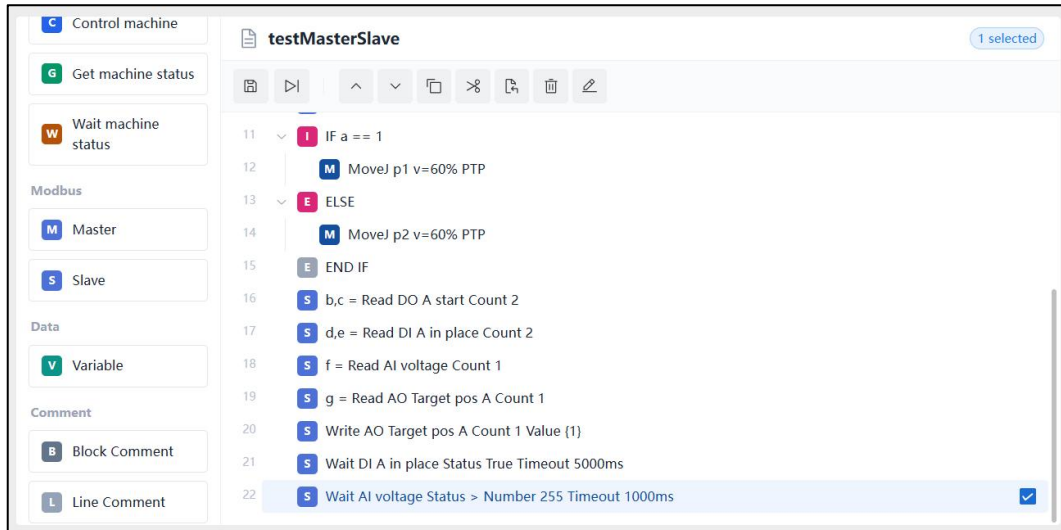


Figure 9.4.3.3-26 Wait for analog input to exceed the threshold – Wait State Workflow

Open the Modbus TCP Slave Instruction Addition page, locate the "Wait Analog Input Configuration" option (i.e., the Wait AI Input Register Configuration), select the "Liquid Level" register with the AI name set to "configured", the wait status set to "=", the register value set to 255, and the timeout duration set to 5000 ms. Click the "Add" button, and finally click the "Apply" button.

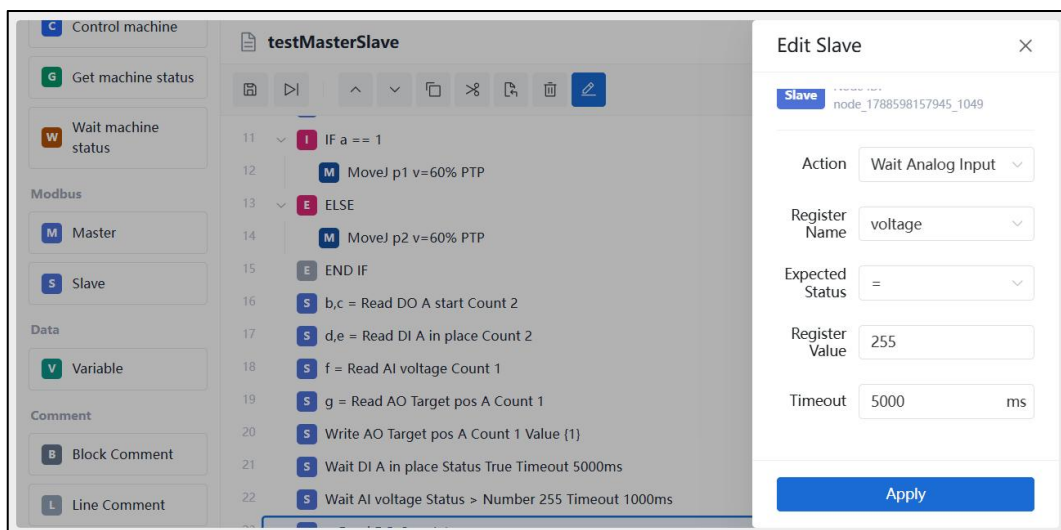


Figure 9.4.3.3-27 Adding a Wait Simulation Input Instruction equates to the Wait State.

At this point, the workflow "testModbusSlave" already contains an instruction instructing the robotic Modbus slave device to wait for the value of an AI input register. Upon starting the program, the robot will continuously wait until the value of the

"Liquid Level" register equals 255. Since the configured timeout duration is 5 seconds, if the "Liquid Level" signal does not equal 255 after 5 seconds have elapsed, the robot's program will raise a timeout error and the program will automatically stop running.

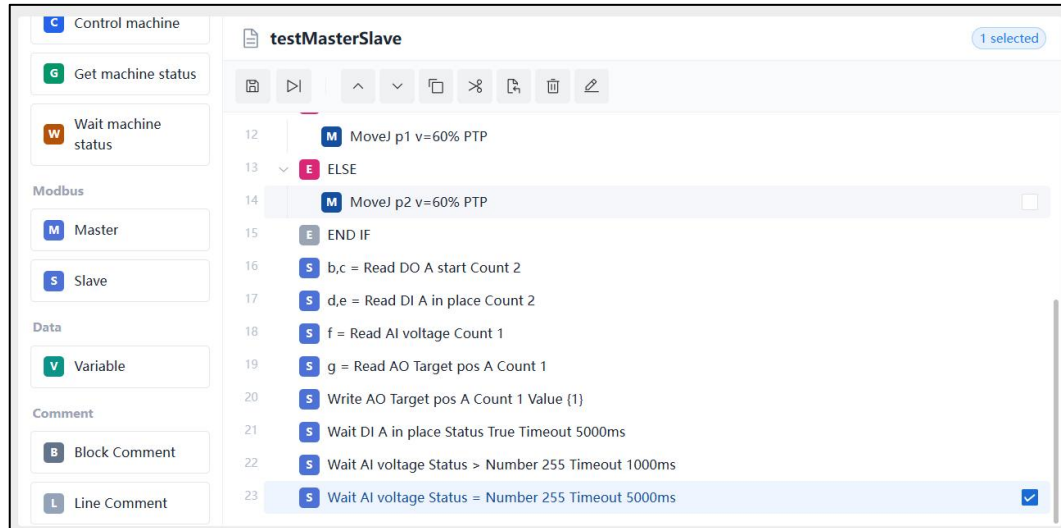


Figure 9.4.3.3-28 Waiting for analog input equals the Waiting State Workflow

9.4.3.5 Modbus TCP slave device – Robot state feedback and control

The Modbus TCP slave device's input register addresses (310–473) are used to provide feedback on the robot's real-time status (for detailed address definitions, refer to Appendix I: Modbus TCP Slave Address Mapping Table); simply read the value from the corresponding register using the master device to obtain the corresponding real-time robot status data.

The Modbus TCP slave device coil register addresses (300–599) are used by the master station device to control the robot (for specific address definitions, refer to Appendix I: Modbus TCP Slave Address Mapping Table). Taking coil address 502 as an example, this address corresponds to the "Start Program" function.

When the robot is in automatic mode and the master station device sets the value of address 502 from 0 to 1, the robot automatically starts executing the currently configured program; taking coil address 300 as an example – this address is used to control the output of the robot's control box DO0 – when the external master station sets coil address 300 from 0 to 1, the output from control box DO0 becomes active; similarly, when the external master station sets coil address 300 from 1 to 0, the output from control box DO0 becomes inactive. By clicking "Functional Digital Input (Coil)" on the Modbus TCP slave device configuration page, you can monitor the status of all

current functional digital inputs.

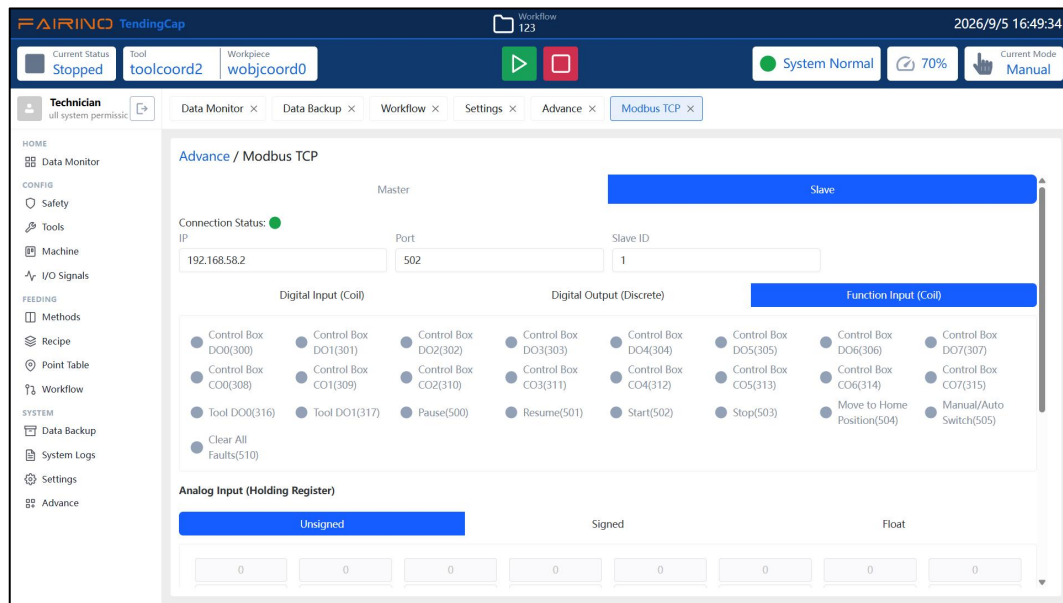


Figure 9.4.3.5-1 Robot End-Station Function - Digital Input

FAIRINC WOODSTOCK Slave station address mapping table									
address	type	name	data type	function	R/W	remarks			
100	General digital input (coils)	B10	BOOL	0x02	R/W				
101		B11							
102		B12							
103		B13							
...		...							
227	General digital output (Discrete input)	D1127	BOOL	0x01 0x05 0x15	R				
100		B00							
101		B01							
102		B02							
103		B03							
...	...								
227	Analog input (Holding register)	D0127	UINT16	0x04	R				
100		A10							
101		A11							
102		A12							
...		...							
115		A115	INT16						
116		A116							
117		A117							
118		A118							
...		...							
131		A131	FLOAT32 (Big Endian)						
132		A132							
133		A133							
134		A134							
135		A135							
...		...							
194		A160							
195		A160							
100	Analog output (Input register)	A00	UINT16	0x03 0x06 0x16	R/W				
101		A01							
102		A02							
...		...							
115		A015	INT16						
116		A016							
117		A017							
118		A018							
...		...							
131		A031	FLOAT32 (Big Endian)						
132		A032							
133		A033							
134		A034							
135		A035							
...		...							
194		A063							
195		A063							
Robot state feedback									
310	Robot state (Input register)	Enabled state: 0-not enabled, 1-enabled	UINT16						
311		Robot mode: 1-manual mode, 0-automatic mode							
312		Robot running state: 1-Stop, 2-Running, 3-Pause, 4-Down							
313		Tool number							
314		Emergency stop state: 0-Normal, 1-Emergency stop							
315		Soft limit exceeded							
316		Main Error code							
317		Sub Error code							
318		Collision detection: 1-collision, 0-no collision							
319		Motion arrive							
320		Safety stop S10	FLOAT32 (Big Endian)	0x04	R				
321		Safety stop S11							
322		TCP speed							
330		Joint 1 position							
342		Joint 2 position							
344		Joint 3 position							
346		Joint 4 position							
348		Joint 5 position							
350		Joint 6 position							
352		Joint 1 velocity							
354		Joint 2 velocity							
356		Joint 3 velocity							
358		Joint 4 velocity							
360		Joint 5 velocity							
362		Joint 6 velocity							
364		Joint 1 current	UINT16						
366		Joint 2 current							
368		Joint 3 current							
370		Joint 4 current							
372		Joint 5 current							
374		Joint 6 current							
376		Joint 1 torque							
378		Joint 2 torque							
380		Joint 3 torque							
382		Joint 4 torque							
384		Joint 5 torque							
386		Joint 6 torque							
388		TCP pos X	UINT16						
390		TCP pos Y							
392		TCP pos Z							
394		TCP pos RX							
396		TCP pos RY							
398		TCP pos RZ							
400		TCP velocity X							
402		TCP velocity Y							
404		TCP velocity Z							
406		TCP velocity RX							
408		TCP velocity RY							
410		TCP velocity RZ							
430	Digital inputs B10-B17 of the control box correspond to B10-B17 corresponds to C10-C17	Control box analog input A10	UINT16						
432		Control box analog input A11							
434		Tool analog input A10							
436		Control box analog output A00							
438		Control box analog output A01							
440		Tool analog output A00							
442		Digital inputs B18-B17 of the control box correspond to B18-B17 corresponds to C18-C17							
444		Digital outputs B18-B17 of the control box correspond to B18-B17 corresponds to C18-C17							
446		Digital inputs B18-B17 of the control box correspond to B18-B17 corresponds to C18-C17							
448		Digital outputs B18-B17 of the control box correspond to B18-B17 corresponds to C18-C17							
470		Robot control					Control box DO0	BOOL	0x02
500	Control box DO1								
501	Control box DO2								
502	Control box DO3								
503	Control box DO4								
504	Control box DO5								
505	Control box DO6								
506	Control box DO7								
507	Control box CO0								
508	Control box CO1								
509	Control box CO2								
510	Digital input (coils)	Control box CO3	BOOL	0x02	R/W				
511		Control box CO4							
512		Control box CO5							
513		Control box CO6							
514		Control box CO7							
515		Tool DO0							
516		Tool DO1							
517		Tool DO2							
518		Tool DO3							
519		Tool DO4							
520		Tool DO5							
521	Digital input (coils)	Pause	BOOL	0x02	R/W				
522		Resume							
523		Start							
524		Stop							
525		Move to Job origin							
526		Manual automatic switching							
527		Start the main program							
528		Level 1 reduction mode							
529		Level 2 reduction mode							
530		Level 3 reduction mode (stop)							
531		Digital input (coils)					Reset all error	BOOL	0x02
532	Reset								
533	Reset								
534	Reset								
535	Reset								
536	Reset								
537	Reset								
538	Reset								
539	Reset								
540	Reset								
541	Reset								

Figure 9.4.3.5-2 Modbus Slave Address Mapping Table



ModbusTCP
Slave station adc

Appendix: