HITBOT
Make it easy

HitBotStudio
USER MANUAL

Main Business:
Industrial Robot / Collaborative Robot / Electric Gripper / Intelligent Actuator / Automatic Solutions



Automation & Collaborative Robots
IMTEC
Kadnárova 19, 831 52 Bratislava
imtec@imtecrobots.com
www.imtecrobots.com

Huiling-tech Robotic Co.,Ltd.

Release Instructions:

Edition Number	Publish Time	Update Content	Note
Hi.19.1.0	2019-06-21	First edition released.	
Hi.19.1.1	2019-07-28	1. Modify issue of UI freeze. 2. Add the property of operation tray module. 3. Add a message pop-up.	
Hi.19.1.2	2019-07-05	Software crash caused by dragging & modifying module.	
Hi.19.1.3	2019-07-08	1. Add the function of modify process name. 2. Modify the problem of frequent file writes causing process delays. 3. Add and delete process data prompt. 4. Modify interface fonts.	
Hi.19.1.4	2019-07-10	1. Add the function to view the module properties when the mouse is hovered over. 2. Modify and delete rules.	
Hi.19.1.5	2019-07-18	1. Add "delay and assignment" module in process configuration. 2. Create trajectory modification motion function and add operation panel.	
Hi.19.1.6	2019-08-05	1. Remove quick start and track creation functions. 2. Optimize software performance.	
Hi.19.1.7	2019-08-13	Integrate and extend the I/O card communication module, increase the number of I/O from 12 groups to 28 groups.	
Hi.19.2.2	2019-11-01	1. Add Sub-process and reset modules. 2. Other related optimizations.	
Hi.19.2.3	2019-11-25	Optimized client communication and fix bugs of pallet.	
Hi.19.2.4	2019-12-12	1. When the client is debugging manually, the receiving delay is increased to 20s to facilitate debugging.	

		2. Add dynamic monitoring of variable values, in the last column of the variable table.	
Hi.20.0.5	2020-04-24	1. Add the judgment of coordinate, joint angle and hand system in the condition module. 2. 3D tray. 3. Add initialization system input . 4. UI colouring. 5. Increase output signal status judgment. 6. Increase the input signal of the initialization system 7. Improve error alarm of client message. 8. Increase the output of security zone limit and over-limit signal system 9. Add box selection of process copy, paste, cut and delete.	
Hi.20.0.6	2020-05-19	1. Add the tray array ID to specif the interface. After the tray is configured, the setting interface will appear if double-click the left mouse button. 2. Increase the output of the process state system. 3. Add the import and export of point table data (Excel) . 4. Add the function of security area, the initialization module can be seen by double-click the left mouse button. 5. Add the offset setting function with the second coordinate system, and the initialization module can be seen by double-clicking with the left mouse button . 6. Modules can automatically align and arrange. 7. Hide the condition module without condition content. When the mouse stops over the basic module, the condition module will be displayed. 8. Modify the incomplete display of the previous version, and the bug of calling blocking functions. 9. Increase xyz_r decomposition ways in point module decomposition mode.	

Hi.20.0.7	2020-08-06	1. New revision of UI to expand the area of design area . 2. Add script programming module. 3. In order to facilitate the design process, add the function of copy, paste and delete module. 4. After the module is selected, the lines connected with it will be highlighted. 5. Add the function of single step debugging . 6. Upgrade the mobile robot module. Step size greater than 2mm (including 2mm) is JOG mode. Press it to continuously move the robot; less than 2mm is inching for fine adjustment. 7. Optimize display and operation. 8. Add variable server to modify the global variable value according to the protocol. 9. Solve the problem of flickering when dragging . 10. Add ID configuration table to remove duplicate ID . 11. The connected line can be selected and removed to re connect the target module. 12. Add the function of saving sub-process as template. 13. Added pop-up prompts when dragging teaching on and off to prevent misoperation.	
Hi.20.0.8	2020-10-22	1. The arm ID list empty data abnormal pop-up box is shielded 2. The control panel uses letters. 3. Do not update global variables when saving. 4. Solve the problem that the runtime library is abnormal when multiple processes call scripts at the same time 5. Add process status indication. 6. Shield automatic arrangement. 7. Enable the bubble function of common modules, which can be displayed by placing the mouse over the module.	

Hi.20.2.8.24	2021-12-21	1. Solve the problem of abnormal I/O display when frequently operating the hardware button to trigger start, pause and other functions. 2. Solve the problem of abnormal I/O signal addition.	
Hi.22.2.11.1	2022-02-11	1.Replace the relevant library of the global exception code, and add the code. 2.Import replacement of new motion function new_move_xyz(). 3.Add the interpreter application of the script function. 4.Safe area judgment. 5.Save the value of the variable at runtime, restart the software to load and restore.	
Hi.22.8.13.1	2022-08-13	1.Added tool coordinate settings 2.Added pipette module 3.Joint shield detection 4.IO signals control single step Test	
Hi.22.9.19.1	2022-09-19	1.Optimized process loading and module dragging 2.Added anti-software flash back function 3.Added new modules for drag and copy deletion	

Table of Contents

I. Company Profile..... 1
II. Software Framework Description..... 2
2.1 Software framework structure diagram..... 2
III. Software Installation and Uninstallation..... 3
3.1 Software installation..... 3
3.2 Software uninstallation and repair..... 4
IV. Software Operation and Module Function Description..... 5
4.1 Software environment settings for the first operation..... 5
4.2 Software login..... 7
4.3 Software module description..... 8
4.3.1 Functional area..... 9
4.3.2 Basic programming module list area..... 10
4.3.3 Programming area..... 11
4.4 Robot operation..... 11
4.4.1 Initialization..... 11
4.4.2 Robot operation..... 12
4.5 Programming module..... 13
4.5.1 Condition..... 13
4.5.2 Position..... 15
4.5.3 Output signal..... 18
4.5.4 Pallet..... 19
4.5.5 Electric Gripper..... 22
4.5.6 Delay and Assignment..... 23
4.5.7 Sub-process..... 24
4.5.8 Reset..... 25
4.5.9 Client..... 26

4.5.10 Serial port..... 28
4.5.11 Script..... 29
4.5.12 Sample of script..... 30
4.5.12.1 Sample of script: client connection..... 30
4.5.12.2 Sample of script: draw circular trajectory..... 30
4.5.12.3 Sample of script: move to the set point..... 31
4.5.13 Pause..... 32
4.5.14 Pipette..... 32
4.6 Examples of programming structure..... 34
4.6.1 Single sequential execution structure..... 34
4.6.2 Cyclic structure..... 35
4.6.3 Branching structure..... 35
4.6.4 Circular judgment structure..... 36
4.6.5 Self circulation structure..... 36
4.7 Important function description..... 36
4.7.1 Safe area..... 36
4.7.2 Expand I/O board..... 37
4.7.3 System debugging..... 37
4.7.4 Determination of safe area..... 38
4.7.5 Tool coordinate function..... 39
4.7.6 Joint detection mask..... 40
4.7.7 I/O control single-step debugging function..... 41
V. Instruction of relevant configuration (the value is uniformly set to 0 or 1)..... 42
5.1 Instruction of key-value pair configuration in HitBotStudio.exe.config file..... 42
5.2 Instruction of software operating environment configuration..... 43
VI. Frequently Asked Questions and Solutions..... 44
6.1 The computer can not connect to the robot..... 44

6.2 Initialization error of missing axis.....44
6.3 The log reports over-current protection and needs to be powered off and restarted.....44
6.4 The log reports collision, please reinitialize or power off and restart.....44
6.5 Unable to use the drag teaching function.....45

I.Company Profile

Huiling-tech (HITBOT) Robotic Co., Ltd, one of Xiaomi ecological chain enterprises, is a leading manufacturer of the lightweight collaborative robot arm and electric gripper in the robot industry. Through nearly 10 years of R&D, Hibot has successfully lowered the threshold of automation transformation for SMEs in terms of cost and applicability, and has provided leading automation solution services in an efficient, cost-effective, and modular way.

Founded in 2015, Hitbot always takes automation redefining as its mission, aiming to create the most cost-effective products, reduce the cost and operating threshold of the intelligent manufacturing industry, and become a worldwide leading robot manufacturer and service provider. At present, Hitbot's direct sales and agents have spread all over the world, and the robotic arm Z-Arm series and electric gripper Z-EFG series have gained thousands of customers, including Huawei, Foxconn, P&G, BGI, and other industry-leading companies.

Special Notice: Please turn off your anti virus software and firewall before you unzip or run the HITBOT Studio Software.

II.Software Framework Description

2.1 Software framework structure diagram

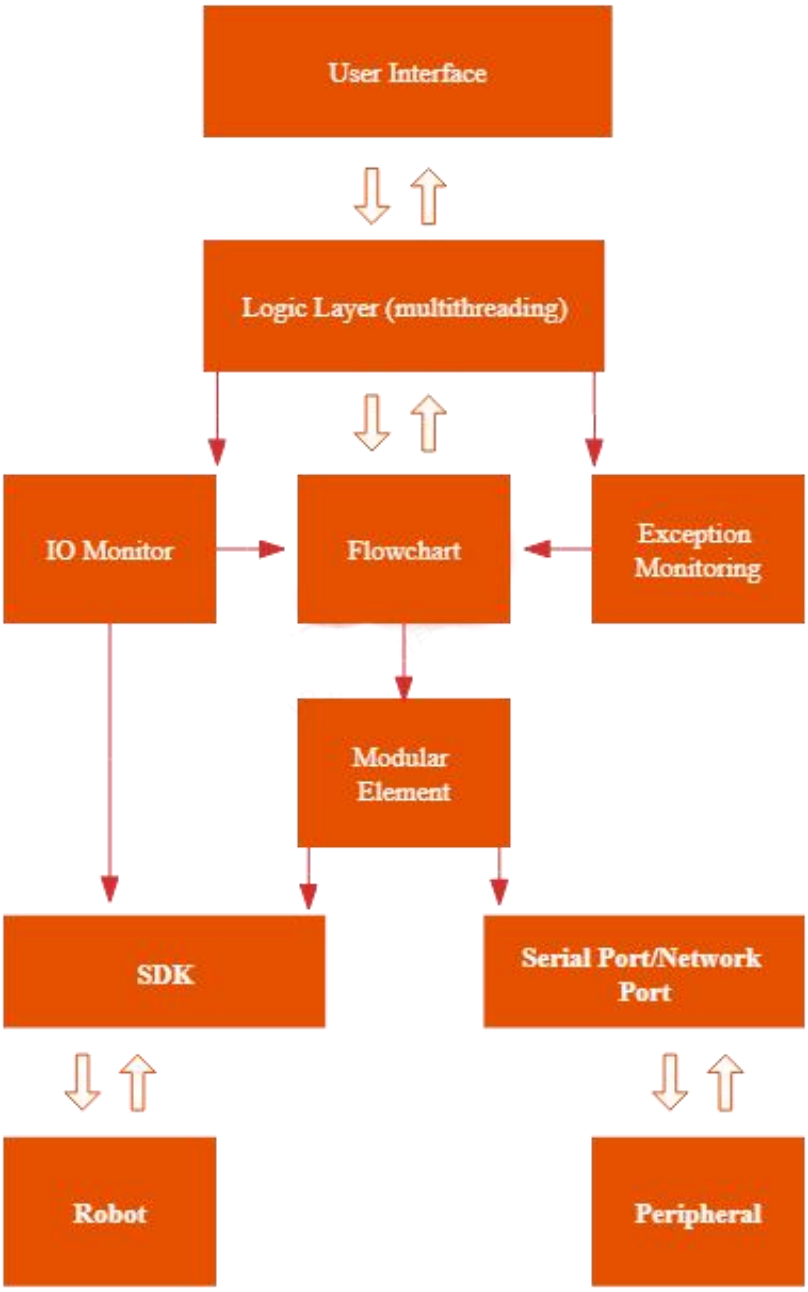


Figure 1. Software Framework Diagram

III.Software Installation and Uninstallation

3.1 Software installation

As shown in the figures, double-click the setup.exe file of the installation package to open the installation interface. Click the prompt to select the installation permission and directory, then click "Next" to complete the installation. Click "Close" to exit the installation interface after the installation is complete. Note that when upgrading the software version, you need to uninstall the old version before you can install the new version of the software correctly. Please refer to the next section "Software Uninstallation and Repair" for details.



Figure 2. Installation package file directory

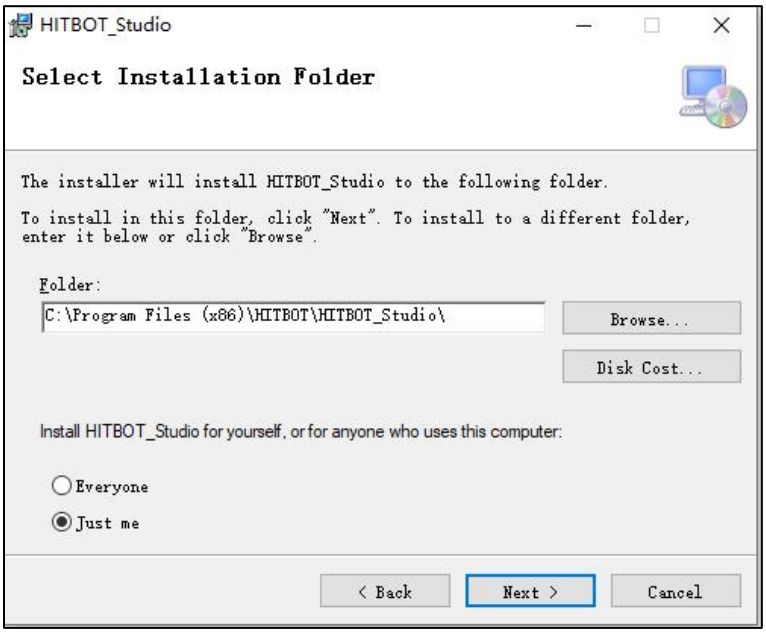


Figure 3. Start the installation interface

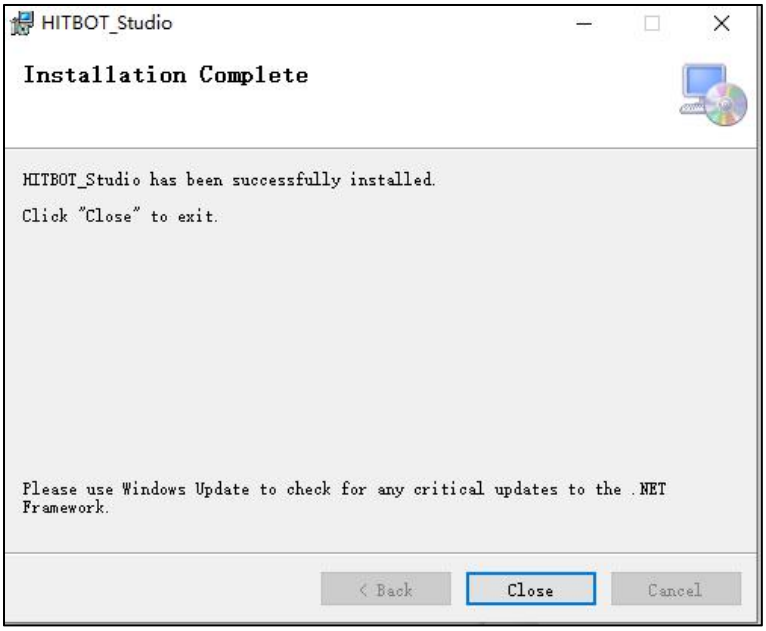


Figure 4. Installation completion interface

3.2 Software uninstallation and repair

This operation is required when a file is lost or the software needs to be upgraded. As shown in the figures, double-click the "RemoveHitBotStudio.msi" file of the installation package to open the repair and uninstall interface. After selecting the repair or uninstall operation as required, click "Finish" to finish the repair or uninstall operation.

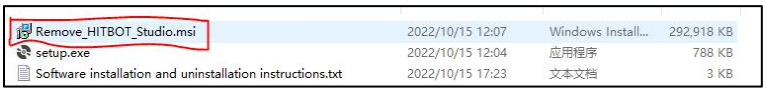


Figure 5. Start uninstall and repair file directory

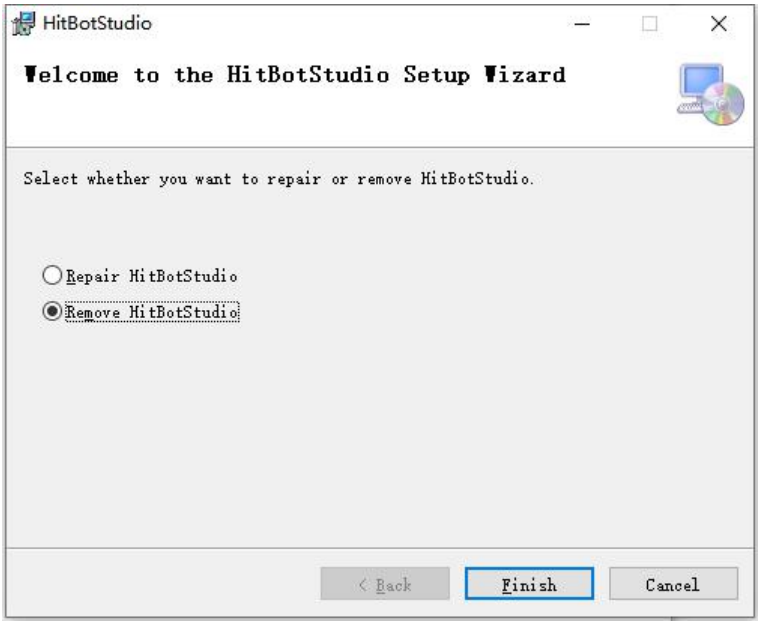


Figure 6. Uninstallation interface

IV.Software Operation and Module Function Description

4.1 Software environment settings for the first operation

When opening the software for the first time to connect the robotic arm, you need to set up the relevant environment, as shown in the figures.

Step 1: The network port connected to the mechanical arm is set as follows: IP address is set to "192.169. 0.100" and the sub-net mask is set to "255.255. 255.0".

Step 2: Right-click the software icon, select "Property", "Compatibility" in turn, and check "Run this program as administrator".

Step 3: Turn off the firewall or add the application by "Allow the application to pass through the firewall."

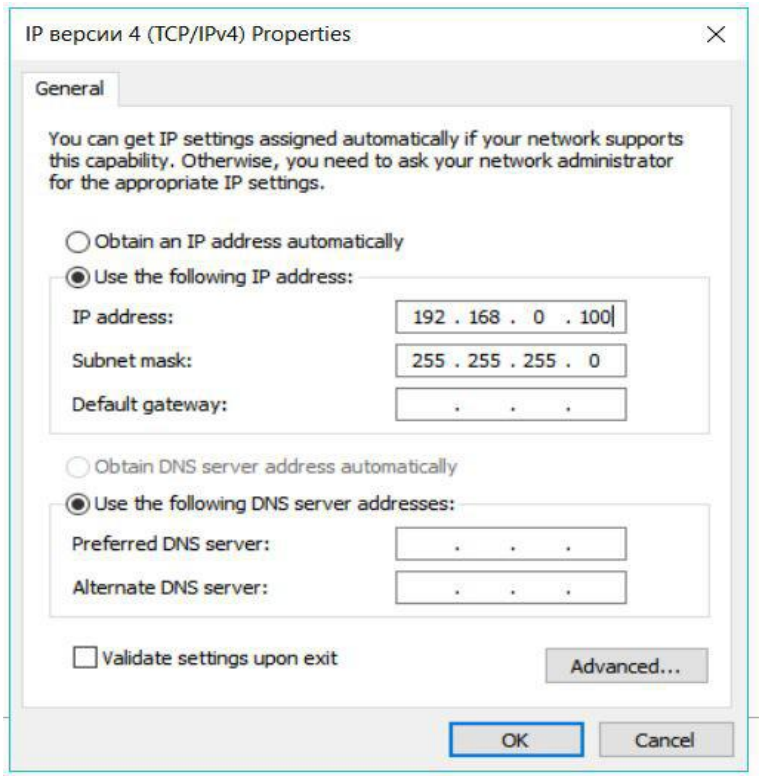


Figure 7. Static host IP settings

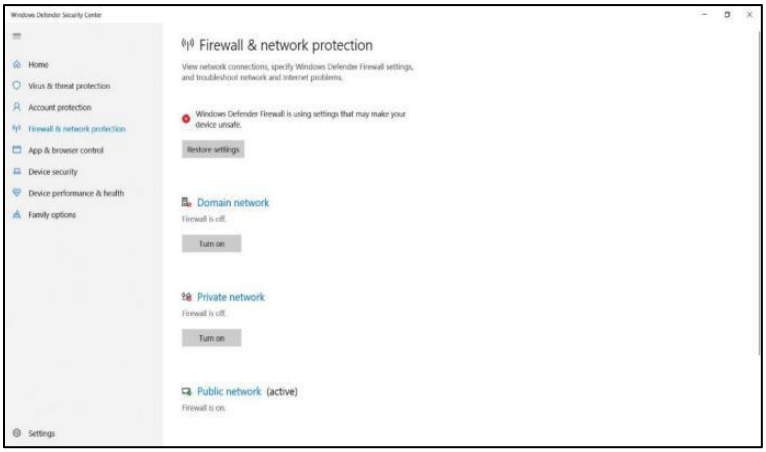


Figure 8. Firewall shutdown settings

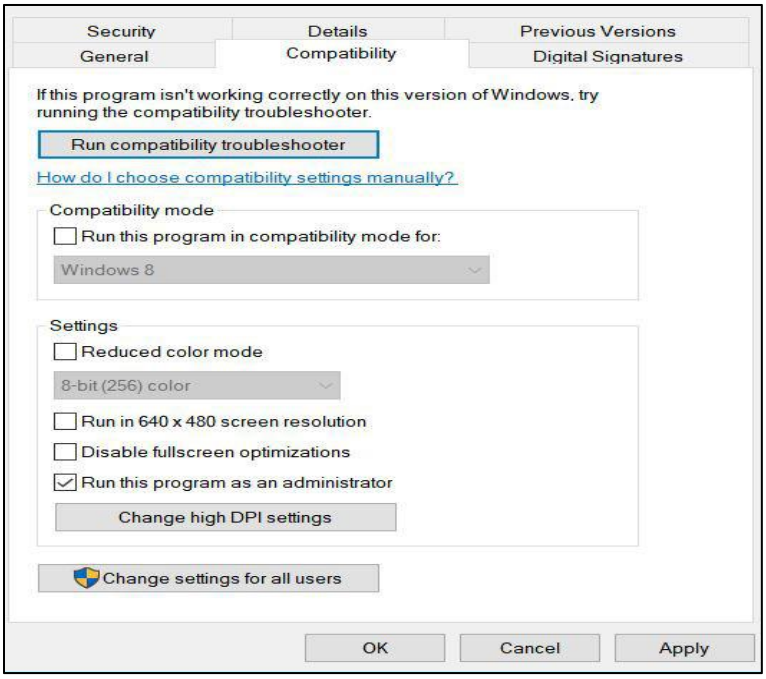


Figure 9. Software compatibility settings

4.2 Software login

Double-click the software icon to open the software and enter the main login interface. Click the "background debugging" button and enter the default login password "hitbot" to enter the main interface of the software to perform relevant operations (the password can be modified. After entering the correct password, the password modification button will appear in the lower right corner of the pop-up box).

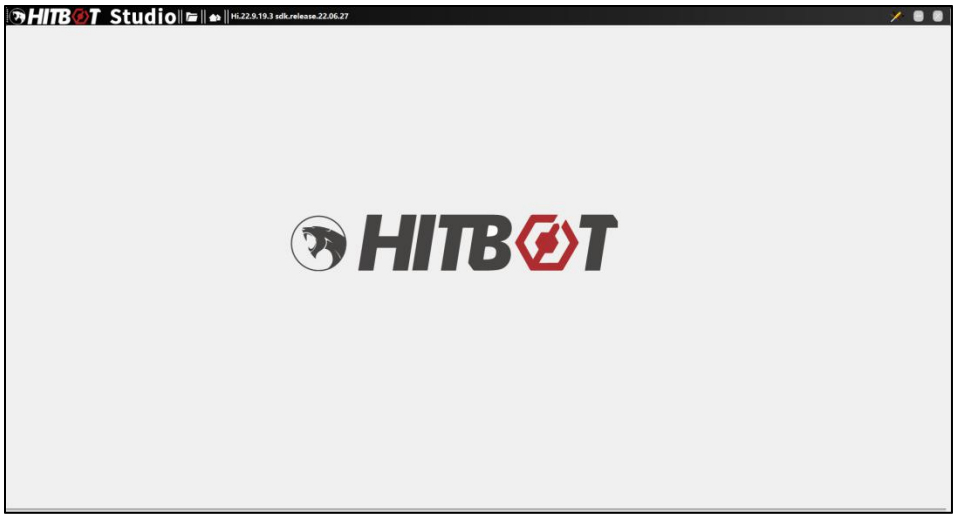


Figure 10. Software main interface

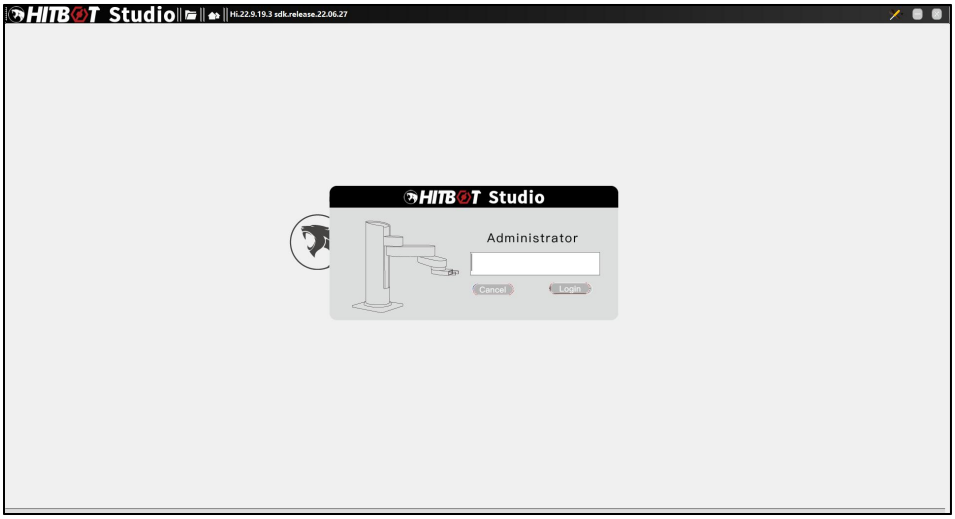


Figure 11. Software login interface

4.3 Software module description

After successful login, the main interface of the software will be entered, which is divided into three main parts: function area, basic programming module list area and programming area.

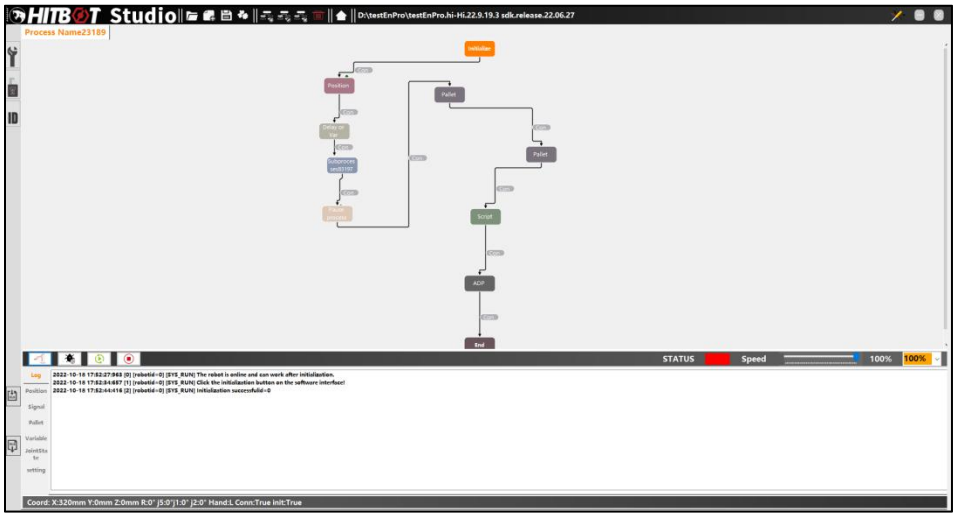


Figure 12. Flow example diagram

4.3.1 Functional area

As shown in the figure below, the function area is mainly composed of project management, process management, process control and convenient tools.

1. Project management and process management



Figure 13. Menu bar

Project management includes opening, creating, saving, combining and splitting engineering files (used to split or combine the processes in project files and save them as new project files).

Process management includes process creation, process name modification, flowchart clearing and process removal.

2. Process control



Figure 14. Process control area

The buttons are initialized, single-step debugging, continuous operation, and stop of the robot arm from left to right.

3. Additional function



Figure 15.Additional control area

From left to right are global speed regulation (module set speed x percentage), automatic alignment, scaling options (100%, 75%, 50%).

4. Convenience tools

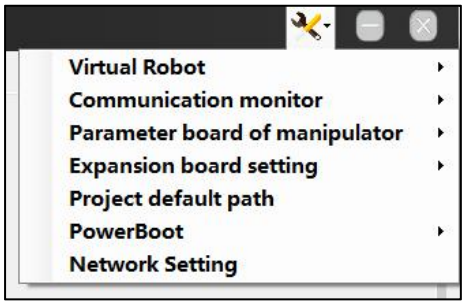


Figure 16. Convenient tools and settings

Convenient tools include virtual robot (id = 0), monitoring and communication tool, parameter setting tool of robot (not for non professionals), I/O expansion board tool, default project file path saving, one key boot. Among them, virtual robot is the most commonly used, which is convenient for debugging process, and does not need to connect the solid robot, including the simulation of input signal.

4.3.2 Basic programming module list area

Based on graphical programming, it provides users with basic function blocks such as point position, output signal, electric gripper, pallet, etc.



Figure 17. Tool module

4.3.3 Programming area

Based on graphical programming, users can drag the module in the programming area (right-click in the blank area to select the module). According to the operation logic, the user can realize the logic programming of the relevant process control of the robot with the arrow. The mouse operation is simple and convenient, and it is very friendly to the novice.

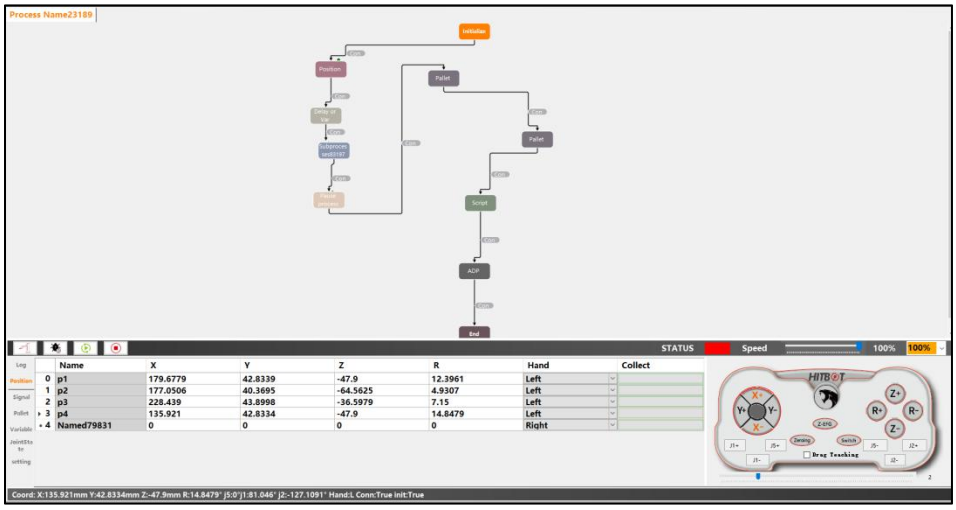


Figure 18. Programming area

4.4 Robot operation

4.4.1 Initialization

Create or open an existing project, as shown in the figure below. After clicking the ID icon on the left, configure the ID according to the online ID list. Note that the ID should be in the interval [0255], Except 0 can be repeated, other ID cannot be reused. Click the "manipulator icon" button and wait for the log to print out the "initialization success" message, which indicates that the connection between the computer and the manipulator is established successfully, and subsequent control operation can be carried out.

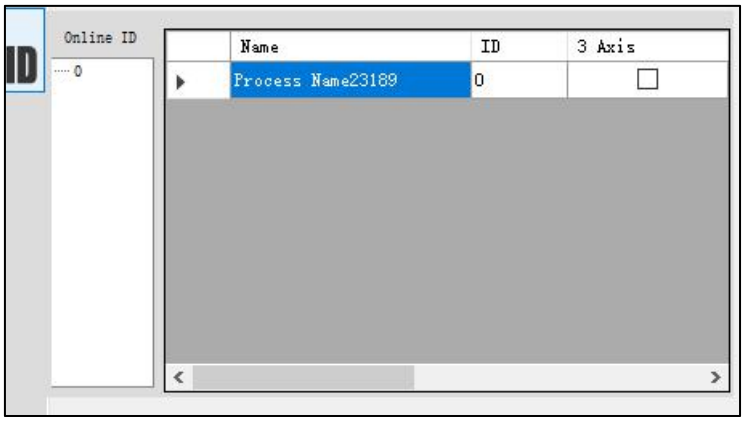


Figure 19. Config Robot ID

4.4.2 Robot operation

After initialization, select the "point position" table to open the operation interface of the robot for relevant operation. The movement of the mechanical arm can be realized by the combination of JOG and inching mode. When the step size is greater than or equal to 2mm, it is JOG mode, and long press the button can move all the way along the axis direction. Otherwise, it will automatically switch to the inching mode for fine position adjustment. As shown in the left side of the figure below, press up, down, left and right respectively for X +, X -,Y +, Y - adjustment. The right side is to controlled Z-axis and R-axis, and the up, down, left and right are used for controlling Z +, Z -, R+, R -, respectively. The sliding bar can adjust the step length, and the default value is 2mm.



Figure 20. Robot (left) and coordinate system (right)

1. **Z-EFG：** press the button to control the clamping and opening of EFG-8 (NK) and EFG-20 (nm).



Figure 21. EFG control panel

2. **Zeroing:** input the robot arm ID to return each joint to its initial zero position.

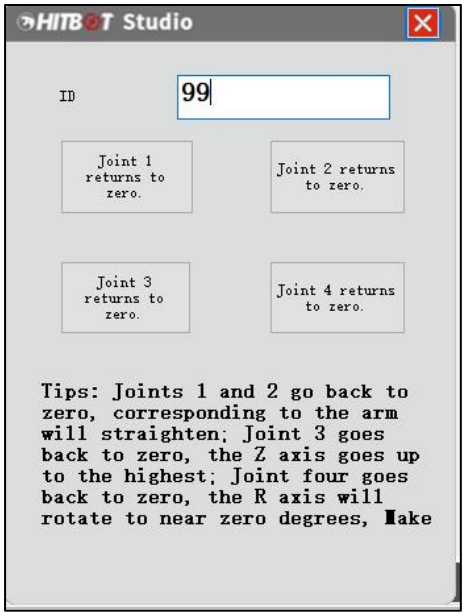


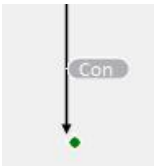
Figure 22. Joint homing control panel

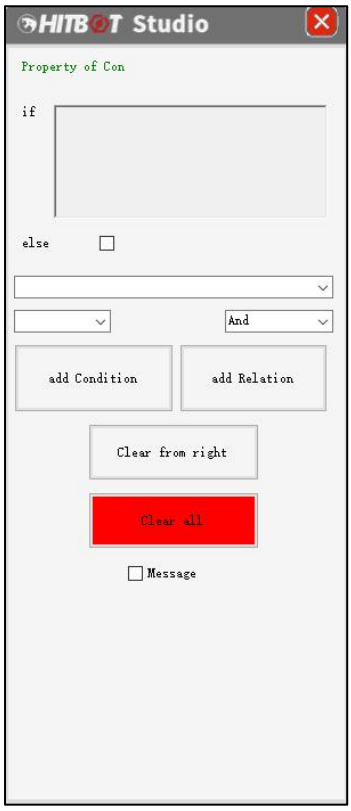
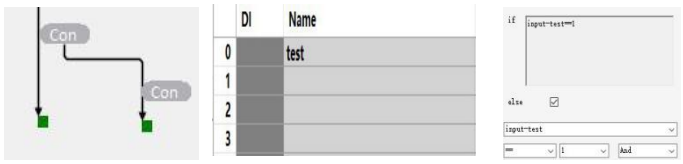
3. **Switch:** the left and right hand systems of the current robot can be switched.

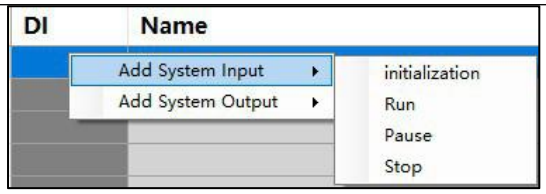
4. **Drag Teaching:** after checking, users can directly drag the robotic arm in the horizontal direction to complete the point teaching (Note: do not check frequently, it takes a period of time to open and close).

4.5 Programming module

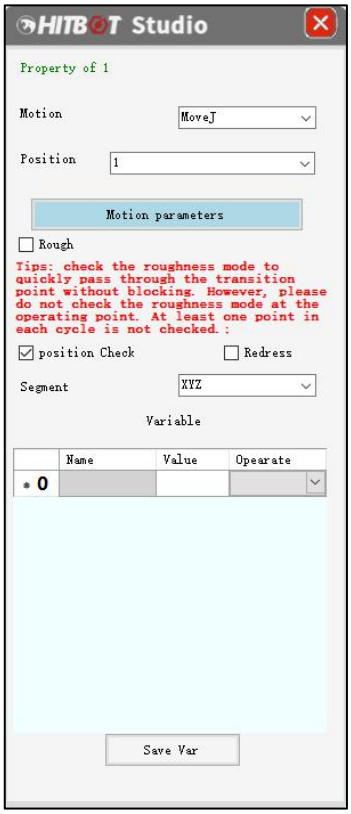
4.5.1 Condition

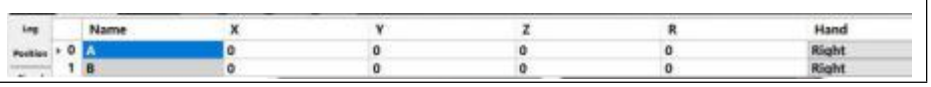
Name	Condition
Function	1. Condition judgment 2. Process branch 3. Attach the arrow line to connect the front and rear modules
Icon	

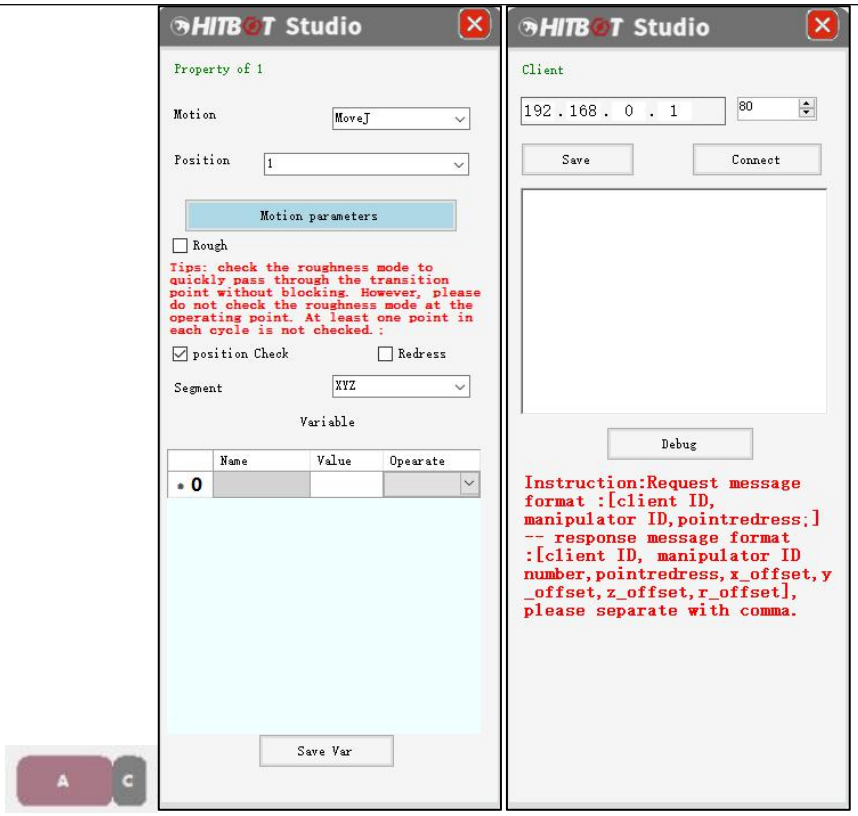
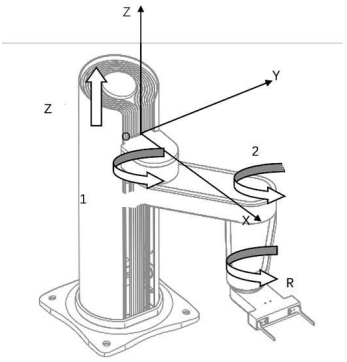
Property	
Configuration	1. Click the corresponding module in the programming area with the left mouse button to generate the rear connection line of the module. 2. Click the right mouse button and the operation menu will appear, you can view the property edit box or delete the condition. 3. As shown in the figure above, select the required condition and click "add condition". If there are multiple conditions juxtaposed, you can select the condition "and / or" and then click "add parallel condition". After checking "else", further branch of conditions can be realized in the programming area. At present, conditional operation only supports the condition judgment of input signal, self adding variable, robot coordinate value and output state. 4. When the process runs here, the configured conditions will be judged, and different logic will be executed according to the true or false of the result. 5. Use the left mouse button to drag the green module at the end of the arrow to the top of the module to be connected, and the arrow will automatically connect the module. 6. Select the condition module, the line and end green modules are highlighted, users can remove the arrow from the connected arrow and reconnect the target module.
Example	
Note	Input signals can be added and changed by filling in or changing information in the "Name" of the input signal list, and the red color of the "input" cell indicates that the corresponding input signal is detected.

	 The user can select a row in the signal table to add system input signals, such as initialization, start, pause and stop. The system automatically detects the rising edge signal and triggers the corresponding function to initialize the operation of the robot or control process. Red represents high level.

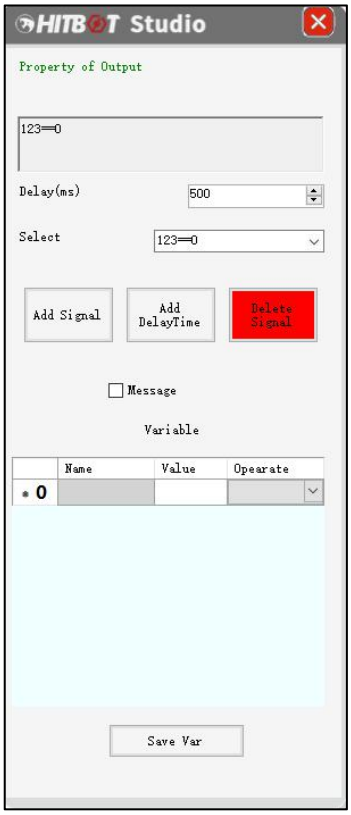
4.5.2 Position

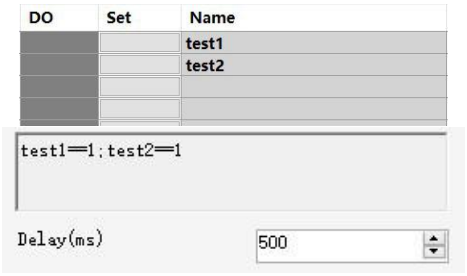
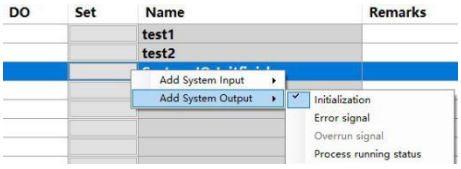
Name	Position module
Function	Control the robot to reach the target point
Icon	
Property	

Configuration	1. Click the corresponding point module with the right mouse button to open the point property edit box or delete the module. 2. In the property box, users can select the motion mode, motion parameters, target point, continuous mode, point check, point compensation, decomposition action and variable assignment and other properties to reach the target point. 3. There are two motion modes: non-linear (MoveJ) and linear mode (MoveL). If there is no requirement for the motion trajectory, MoveJ is recommended. 4. The point is the target point that the robot arm will reach. Before specifying the "point" property of the point module, the corresponding point must be added to the point list. The user can successfully add a point by changing the point name in the point list, and click the corresponding "teach point" to pick up the current coordinates of the robotic arm as the coordinates of the point. 5. The settings of motion parameters include motion speed and acceleration of each axis. The acceleration is a percentage. Users can adjust it according to their needs. Pay attention to the size of the motion parameters, which has a certain relationship with the motion distance, payload, and centroid position. The stability of the robotic arm must be guaranteed. Increase the speed and acceleration in advance, otherwise the mechanical arm will be damaged and normal use will be affected. 6. When the continuous mode (smooth mode) and fast-passing mode are not checked, move to the previous target point first, and the speed at the current target point is 0, and then move to the new target point. If checked, when there is a new target point, it will pass at the maximum speed near the current target point, but the passing point is farther from the current target point. 7. The function of point verification is to perform closed-loop verification of the reaching position according to the value of the encoder when the robotic arm reaches the target point, so as to ensure that the arm has indeed reached the target point under the allowable slight deviation. If it is not in place, a log alarm prompt will be printed and the process will be stopped. 8. The function of dynamic compensation of point coordinate value. Check point compensation, click the "C" module that appears on the right side of the point module with right button of the mouse, set the IP address and port number of the server to be connected, then save and try to connect. After connection, the communication can be performed normally according to the protocol, and the coordinate compensation value can be obtained from the server. Get compensation value message format: Request: head (external identification of client module) + "," + robot ID number + "," + pointredress + ";" Response: head (external identification of client module) + "," + robot ID number + "," + pointredress + "," + x_ offset+“,”+y_ offset+“,”+z_ offset+“,”+r_ offset 9. The variable table can assign assignments, auto-increment and auto-decrement operations to all the variables, and thees variables can be used for conditional judgment.
Example	

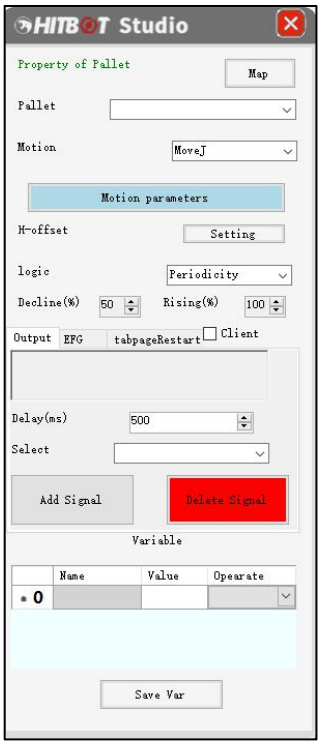
	
Note	The coordinate system of the robot is Cartesian coordinate system, which is divided into left hand (angle2 < 0) and right hand (angle2 > 0) according to the relative posture of the forearm and the big arm. 

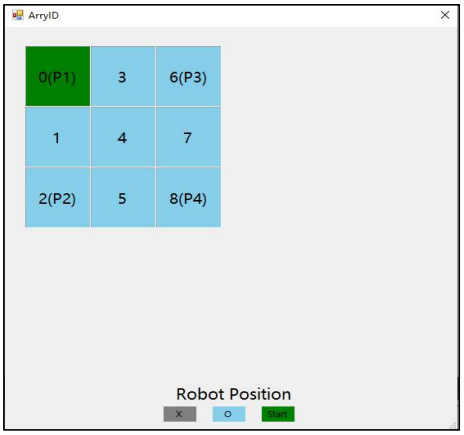
4.5.3 Output signal

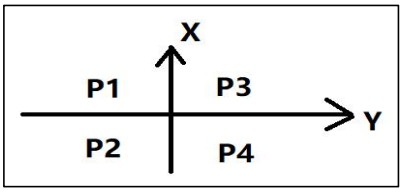
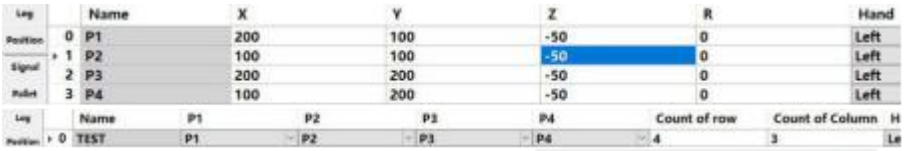
Name	Output signal module
Function	Control the output port of robot to output high or low level signals
Icon	
Property	
Configuration	1. Click the corresponding output signal module with the right mouse button to open the property editing box or delete the module. 2. Here, users can choose to add one or more different level states of the output signal. It should be noted that currently only related signals in the output signal list are supported. Users can add and change output signals by filling in or changing the information in the "Output Name" of the output signal list. 3. The function of delay is to stay in this module for a period of time and then continue to execute the next module after setting the output, so as to ensure that the equipment or tool controlled by the signal are executed. (In industrial application scenarios, tools such as output control grippers, cylinders, suction cups, etc. are commonly used, and their actions need a certain time to complete.) 4. The variable table can assign assignments, auto-increment and auto-decrement operations to all the variables, and thees variables can be used for conditional judgment.

Example	
Note	 Users can select the row in the signal table to add system output signals, such as initialization completion, fault, overrun signal, process operation status and other system output signals. The system automatically detects and sets the corresponding output signal to high level (red).

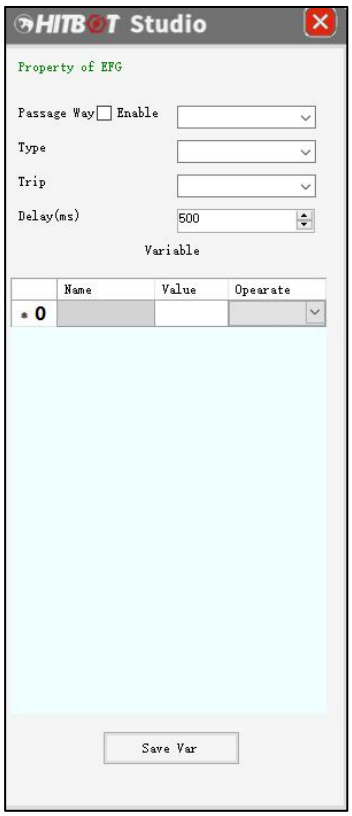
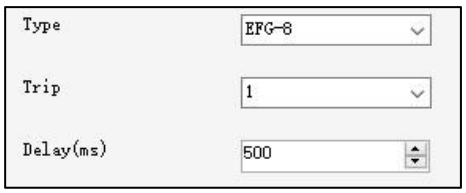
4.5.4 Pallet

Name	Pallet
Function	Control the robotic arm to execute horizontal rectangular array logic,vertical single-layer or multi-layer array logic
Icon	
Property	

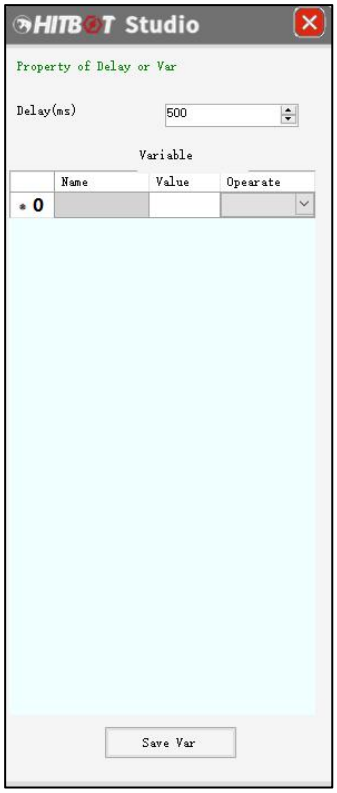
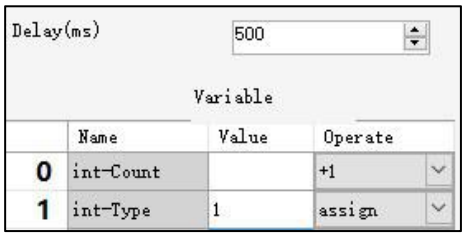
Configuration	1.Click the corresponding pallet module with the right mouse button to open the property editing box or delete the module. 2.The pallet module is actually a point array, which can be a horizontal single-layer or multi-layer rectangular array or an array in the vertical direction. It can be achieved through attribute settings, including starting point designation (Map), array point executable designation (Map), pallet, motion mode, motion parameters, multi-layer settings, execution logic, descent rate, output signal, electric gripper, customer terminal and variable settings. 3.The Map pop-up box lists all the array points, users can select the starting point and the execution permission of the array point (to be executed or not to be executed).  4.The pallet selection box is to select the configured pallet object from the pallet table. 5.The setting of motion mode and motion parameters is the same as that of the point module property setting. 6.In the multi-layer setting, users can set the offset height, number of layers, starting layer, direction and whether to shield the offset. 7.The execution logic is divided into periodic increment and one-time execution, which can be selected according to requirements. When periodic increment is selected, each time the pallet module is passed, it will be automatically shifted to the next point of the array to execute until all points are executed and then restarted; when one-time execution is selected, each time the pallet module is passed, all points in this pallet will be executed once, and then the next module will be executed. 8.The descent rate is to set the speed from the top of the pallet array point (offset height) to the array point (the pallet setting speed multiplied by the descent rate) to achieve the effect of deceleration and descending. 9.The setting of output signal and electric gripper is the same as that of its independent module, which will be executed when it reaches the pallet array point. 10.The variable table can perform assignment, auto-increment or auto-decrement operations to all the variables, and then use them for conditional judgment. 11.Check the client, the point coordinate value to compensate and the function specified by the tray array ID, right-click the "C" module that appears on the right side of the pallet module, set the IP address and port number of the server to be connected, save and try to connect. After connecting, users can communicate normally according to the protocol, and obtain the coordinate

	compensation value and tray array ID from the server. Format of message for obtaining coordinate compensation value: Request: head (external identification of client module) + "," + robot ID number + "," + palletredress + ";" Response: head (external identification of client module) + "," + robot ID number + "," + palletredess + "," + x_offset + "," + y_offset + "," + z_ offset + "," + r_ offset Format of message for getting pallet ID value: Request: head (external identification of client module) + "," + robot ID number + "," + palletarrayID + ";" Response: head (external identification of client module) + "," + robot ID number + "," + palletarrayID + "," + Layer + "," + ID
Example	  
Note	

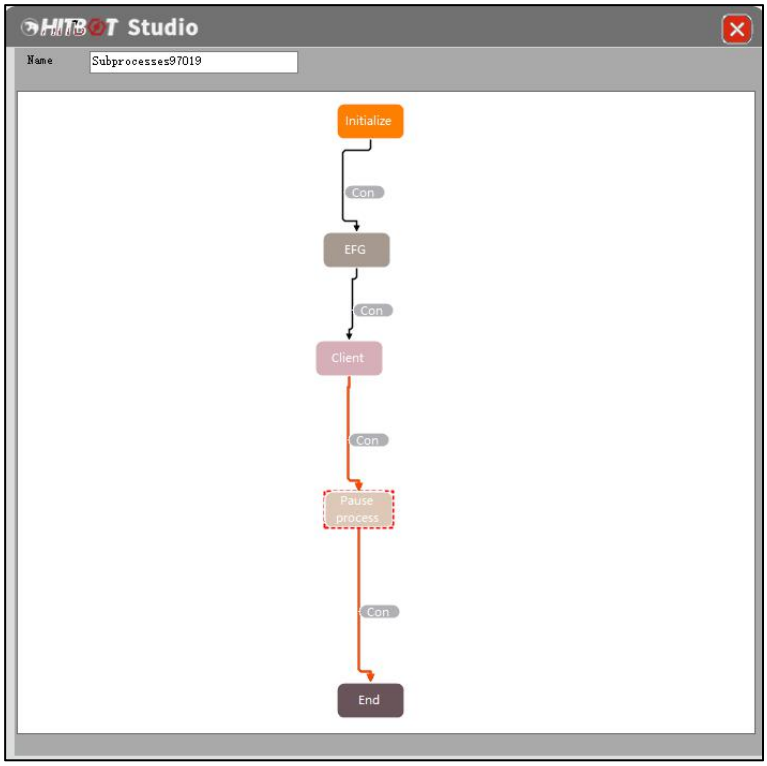
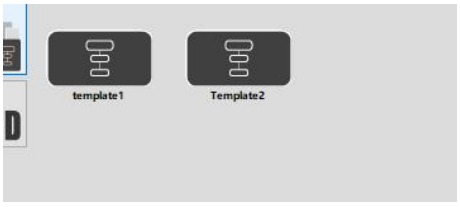
4.5.5 Electric Gripper

Name	Electric gripper
Function	Control the opening and clamping of the Hitbot standard products EFG-8NK and EFG-20NM
Icon	
Property	
Configuration	1. Right-click the corresponding output electric gripper module to open the property edit box or delete the module. 2. Here, users can select the type of gripper (EFG-8NK and EFG-20NM) and stroke setting. 3. It takes time to clamp and loosen the electric gripper. Users can set the appropriate delay time according to the actual effect. 4. The variable table can perform assignment, auto-increment or auto-decrement operations to all the variables, and then use them for conditional judgment.
Example	
Note	

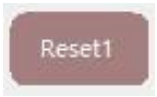
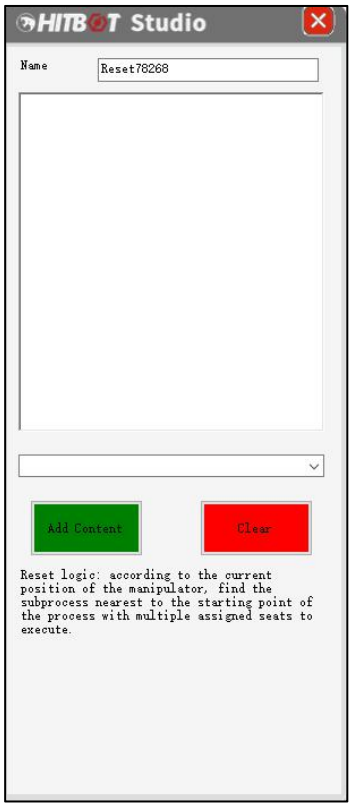
4.5.6 Delay and Assignment

Name	Delay and assignment
Function	1. Process delay; 2. Process variable assignment; 3. Connection node
Icon	
Property	
Configuration	1. Right-click the corresponding delay and assignment module to open the property edit box or delete the module. 2. The delay unit is MS, the default is 500 ms, the upper limit is 10s. 3. The variable table can perform assignment, auto-increment or auto-decrement operations to all the variables, and then use them for conditional judgment. 4. The delay can be set to 0 to use this module as a connection point.
Example	
Note	

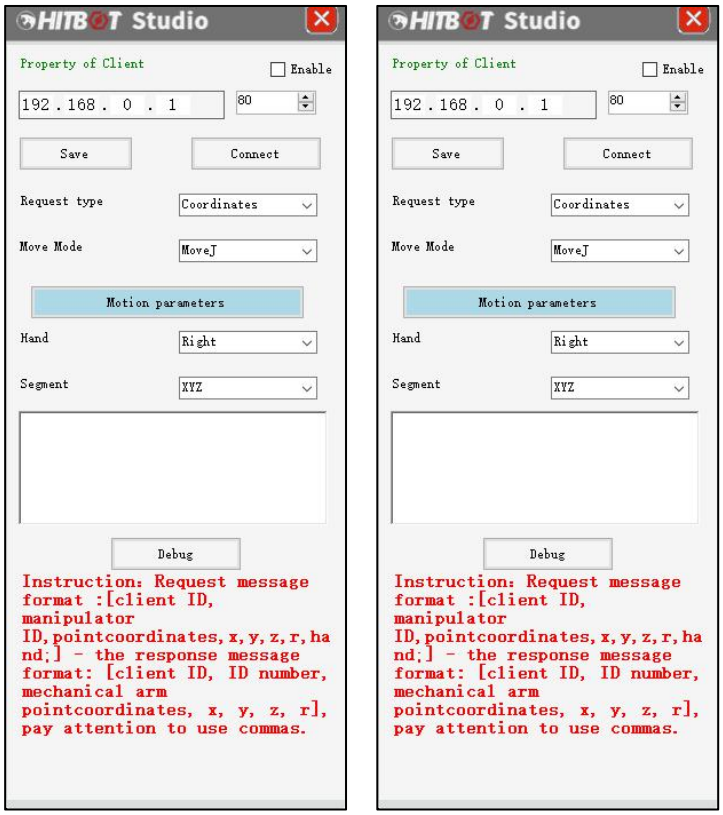
4.5.7 Sub-process

Name	Sub-process
Function	Encapsulate the basic modules of the flowchart to make the flowchart more concise
Icon	
Property	
Configuration	1. Right-click the corresponding sub-process module in the programming area to open the property edit box or delete the module. 2. The sub-process module is mainly used to simplify the main process logic, and to provide an operator for the reset function. 3. The operation of process configuration is consistent with that of the main process, which will not be described in detail. 4. Sub-process frameworks can be saved as templates for use by other processes.
Note	 Place the mouse on the template icon on the left sidebar, drag the template into the design area, modify the name or delete the template.

4.5.8 Reset

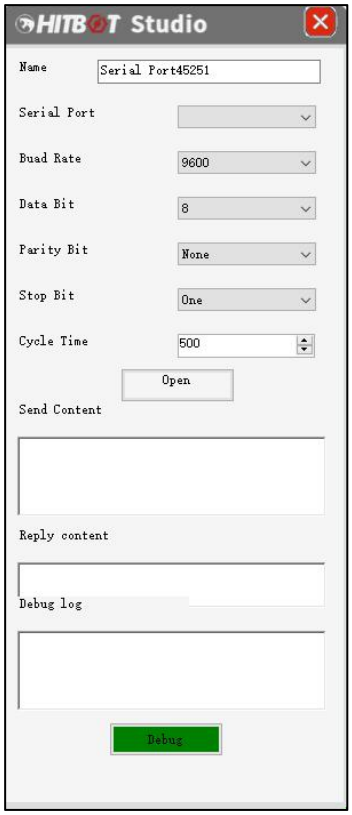
Name	Reset
Function	When the last stop position of the robot is random, multiple reset sub-processes can be customized after restart operation. After adding the reset module, the module will compare the current stop position and the starting point of each sub-process module (the coordinates of the first point module) to find the nearest starting point and execute the current sub-process.
Icon	
Property	
Configuration	1. Right-click the corresponding reset module in the programming area to open the property edit box or delete the module. 2. Select the corresponding sub-process, click add content or clear the content.
Note	The system input signal has a reset signal, which can trigger the reset module independently, and each process has at most one reset module.

4.5.9 Client

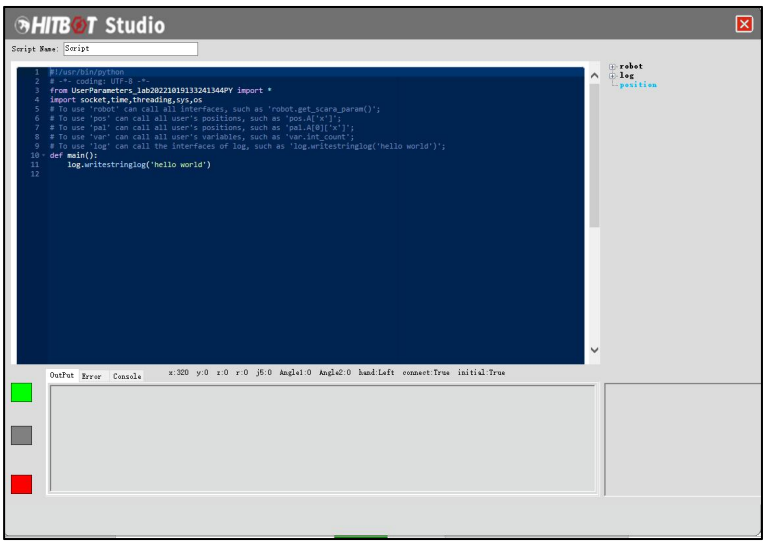
Name	Client
Function	Based on TCP / IP protocol, it is used to request the value of coordinates or global variables from the server.
Icon	
Property	
Configuration	1. Right-click the corresponding client module in the programming area to open the property edit box or delete the module. 2. Set the IP address and port number of the server to be connected, click save, and connect to the server manually. If the connection is successful, communication can be carried out. At present, the timeout setting of the client receiving data is set to 20 seconds, so as to facilitate manual adjustment of communication. When sending instructions manually, the communication data can be displayed. The server must reply the data according to the specified message format, otherwise it can not be parsed normally. 3. There are two types of point coordinates and global variables. After selecting the point coordinates, it is necessary to set the motion parameters, hand system and action decomposition; to select the global variables, you need to check to request the variables and save the settings, and the variables come from the variable table. 4. The communication data can be debugged by manual request, and it can run automatically after receiving and sending correctly. 5. For the coordinates, select the appropriate motion mode, motion parameters

	and hand system to execute the transfer coordinates. The message format is as follows: Request: head (external identification of client module) + "," + robot ID + "," + pointcoordinates + ","+X + "," + y + "," + Z + ", + R +", + hand + "; (Note: hand = 1 is right-handed, hand = - 1 is left-handed) Response: head (external identification of client module) + "," + robot ID + "," + pointcoordinates + "," + X + "," + y + "," + Z + "," + R 6. For global variables, create the corresponding variable in the basic variable table, fill in the initialization value, and you can see the corresponding variable in the client property. If "request" is checked, the request for the corresponding variable will be sent. It supports the simultaneous assignment of multiple variables, and manual debugging. When the process is running automatically, you can view the change of the variable value in the variable table monitoring column (note that "#" connects the variable name and its value). The message format is as follows: Request: header (external identification of client module) + "," + robot ID + "," + globalvars + "," + variable name current value +...+ ";" Response: header (external identification of client module) + "," + robot ID+ "," + globalvars + "," + variable name value +... "
Note	

4.5.10 Serial port

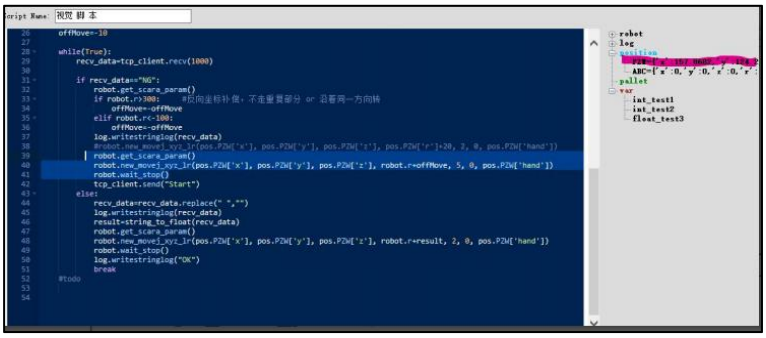
Name	Serial port
Function	Conventional serial communication
Icon	
Property	
Configuration	1. Right-click the corresponding serial port module to open the property edit box or delete the module. 2. Select the serial port and set the required baud rate, data bit, check bit, stop bit and cycle. 3. Customize the data content to be sent. The string is encoded by UTF-8 and converted to byte array and sent out. 4. Customize the content of the reply. The received data is converted into a string and compared with itself. If it does not match, the log alarm prompts to stop the process. 5. Manual debugging can be carried out to verify the correctness of data sending and receiving.
Note	

4.5.11 Script

Name	Script
Function	Based on embedded Python, embedded the module in the control interface of the robot arm, the logic can be realized by programming.
Icon	
Editing	
Configuration	1. Right-click the corresponding script module to open the property edit box or delete the module. 2. After the initialization of the robot is completed, click Edit to enter the editor editing. After programming, you can confirm to exit and click save.
Instruction	1. Compile and run, without break-point debugging. 2. Program errors will be printed in "Error" when compiling. 3. Additional console window is provided. 4. Support Python 2.7 programming. 5. Provide log printing function,which can print the string in "Output". 6. Support to obtain and modify global variables of the process. 7. Support the use of point data of process point table. 8. Support to call the point data of the pallet.
Note	After selecting the API function in the upper right text box, the function description and application example will be displayed in the lower right text box.

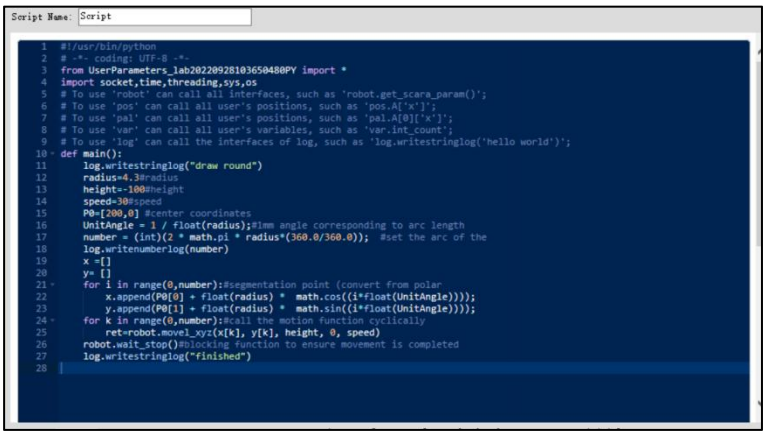
4.5.12 Sample of script

4.5.12.1 Sample of script: client connection

Name	Sample of script: client connection
Editing	

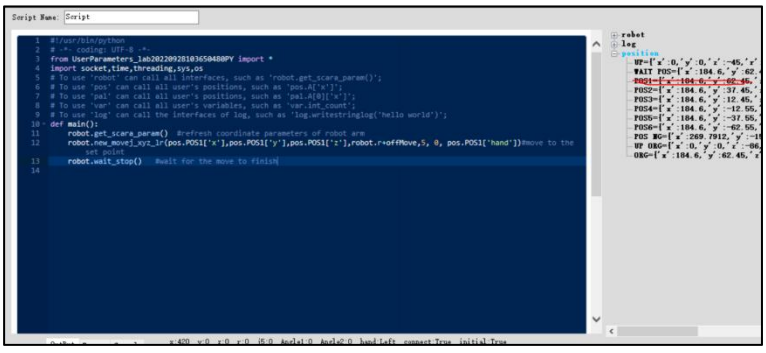
Instruction	from socket import * #import namespace
	def main():
	log.writestringlog('Start Main') #print log
	tcp_client=socket(AF_INET,SOCK_STREAM) #create a client
	tcp_client.connect(('127.0.0.1',6000)) #connect to server
	tcp_client.send("OK0") #send data
	recv_data=tcp_client.recv(6000) #receive data
	recv_data=recv_data.replace(" ","") #remove spaces

4.5.12.2 Sample of script: draw circular trajectory

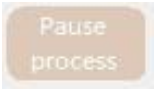
Name	Sample of script: draw circular trajectory
Editing	
Instruction	import socket,time,threading,sys,os,math def main(): log.writestringlog("draw round")

	#In polar coordinate system, the center of the circle is at the pole and the equation of the circle with radius radius p=radius
	radius=4.3#radius
	height=-100#height
	speed=30#speed
	P0=[200,0] #center coordinates
	UnitAngle = 1 / float(radius);#1mm angle corresponding to arc length
	number = (int)(2 * math.pi * radius*(360.0/360.0)); #set the arc of the circle to be drawn
	log.writenumberlog(number)
	x =[]
	y= []
	for i in range(0,number):#segmentation point (convert from polar coordinate system to Cartesian coordinate system)
	x.append(P0[0] + float(radius) * math.cos((i*float(UnitAngle))));
	y.append(P0[1] + float(radius) * math.sin((i*float(UnitAngle))));
	for k in range(0,number):#call the motion function cyclically
	ret=robot.movel_xyz(x[k], y[k], height, 0, speed)
	robot.wait_stop()#blocking function to ensure movement is completed
	log.writestringlog("finished")

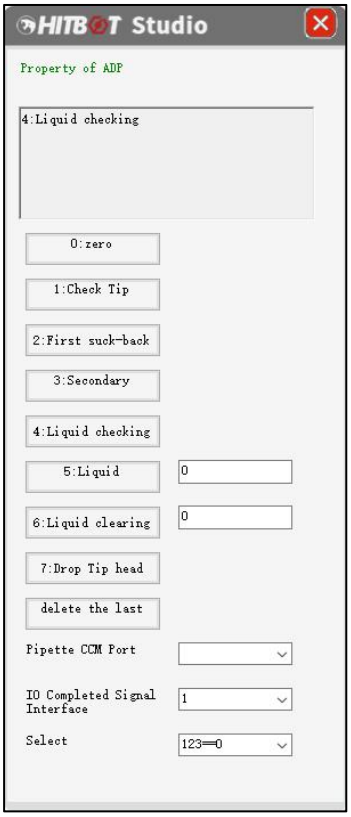
4.5.12.3 Sample of script: move to the set point

Name	Sample of script: move to the set point
Editing	
Instruction	robot.get_scara_param() #refresh coordinate parameters of robot arm robot.new_movej_xyz_lr(pos.PZW['x'], pos.PZW['y'], pos.PZW['z'], robot.r+offMove, 5, 0, pos.PZW['hand']) #move to the set point robot.wait_stop() #wait for the move to finish

4.5.13 Pause

Name	Pause
Function	Pause the process
Icon	
Operation	Right-click the corresponding script module to delete the module.
Note	After the process is suspended, you can click the run button or the input signal of the operation system to resume operation.

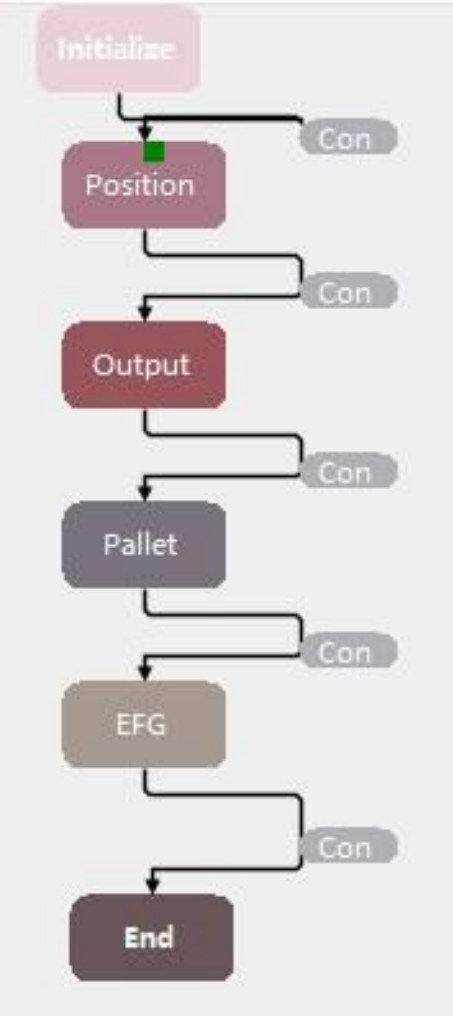
4.5.14 Pipette

Name	Pipette
Function	Execute single or multiple pipette commands
Icon	
Property	
Configuration	1. Correctly select the COM port of the pipette gun (the port number connecting the pipette gun and the USB to 485 communication line of the computer), and the I/O

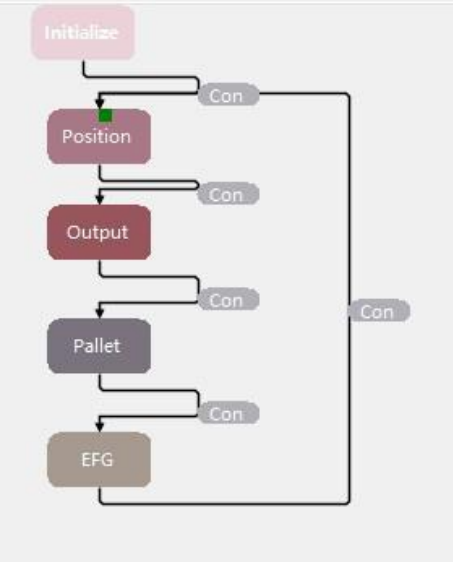
	completion signal interface (the pipette gun completion signal port connecting to the I/O port number of the robotic arm). Click the 0-7 button to add the corresponding execution item. 2. Click the Delete Last Item button to clear the last command. 3. Single command function: (1) 0 : Initialization: set the robot arm to the initial state. (This command will empty the liquid and return the Tip head) (2) 1: Detect whether there is a Tip head: detect whether there is a Tip head on the pipette. The results are output in log form. (3) 2: First suction: before the first suction, execute this command. (4) 3: Secondary suction: when there are large droplets on the tip of the pipette after suction, execute this command. (5) 4: Liquid detection (capacitive): detect whether there are droplets at the tip of the gun. (6) 5: Aspiration: aspirate the amount of liquid set in the text box. The unit is uL. (7) 6: Discharge: discharge the amount of liquid set in the text box. The unit is uL. (8) 7: Return the Tip head: Return the Tip head.
Example	
Note	

4.6 Examples of programming structure

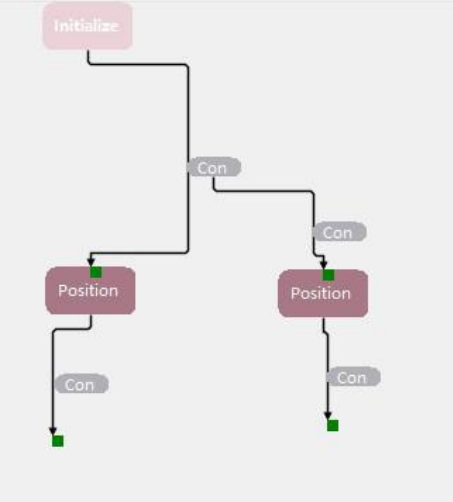
4.6.1 Single sequential execution structure



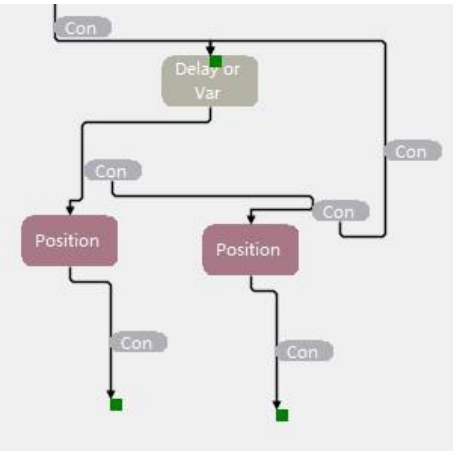
4.6.2 Cyclic structure



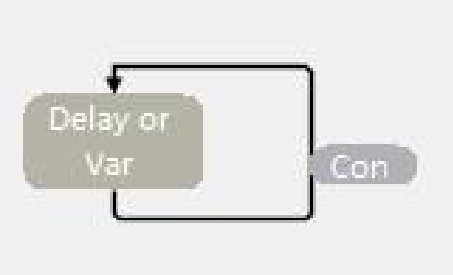
4.6.3 Branching structure



4.6.4 Circular judgment structure



4.6.5 Self circulation structure



4.7 Important function description

4.7.1 Safe area

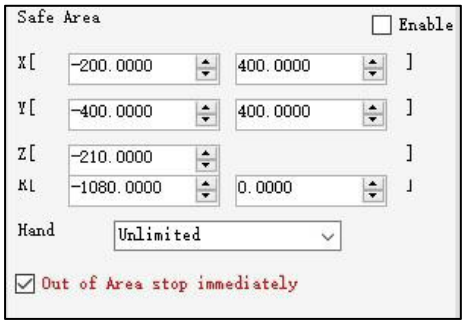


Figure 23. Setting panel of safe area

The setting of safety area includes x, y, z, r coordinate range and the current hand system of the robot arm. The system will independently monitor whether it is within the limited area. If this function is enabled, it will give an alarm, and there will be corresponding system output signal

(overrun signal) to inform other equipments. If "Out of Area stop immediately" is checked, the process and robot will be stopped immediately.

4.7.2 Expand I/O board

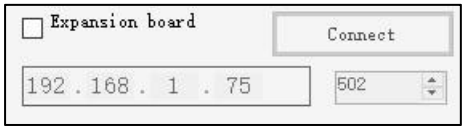


Figure 24. I/O expansion board setting panel

In order to satisfy users who use multiple signals, the system is compatible with a mature I/O ex pansion board (GECON) on the market. Through Ethernet communication control, Modbus TCP prot ocol, the expansion board is a server. The default binding IP address is 192.168.1.75, port 502, a total of 16 digital inputs and 16 transistor outputs. After connecting the power supply and communication network cable, check "Expansion board". If the server is not connected successfully, you can try to c onnect to the server manually. The use method is the same as the I/O of the mechanical arm body, w hich can be configured in the signal table. Taobao link (note that just purchase the boards with Ether net port configuration): https://item.taobao.com/item.htm?spm=a1z10.5-c.w4002-11584738786.13.6 1406eda7Jt2Ha&id=529001943100

4.7.3 System debugging

After the program is written, in order to verify the logic correctness of the program, the online virtual robot can be used for logic simulation. As shown in the figure below, the virtual robot can be opened in the menu bar, and appropriate virtual robot can be selected according to the model. When the current status of the virtual robot is updated to "connected successfully", it means that the virtual robot is successfully opened. In the start page, the ID of the robot is selected as 0. After the initialization is successful, click "start running" to verify the program logic.

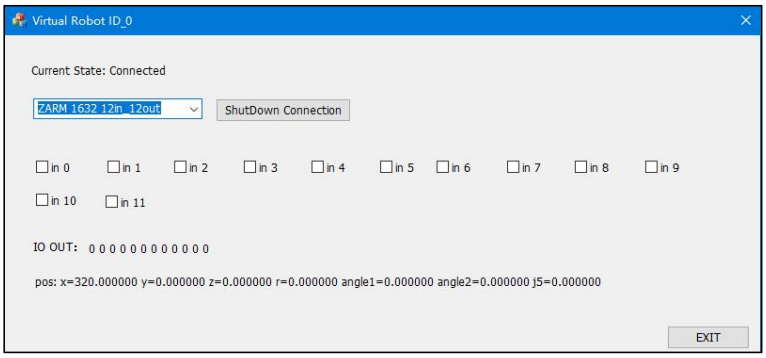


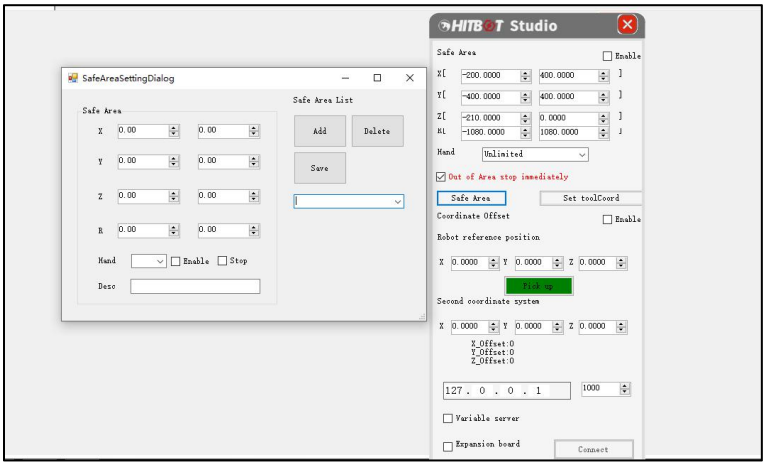
Figure 25. Virtual robot

The virtual robot provides four different types of arms. Except for the function of analog drag teaching, other interfaces can be realized. There is a check box for analog digital input. Check the input high level. with the movement of the control arm, the coordinates will change in real time, and the output signal is also monitored and displayed.

4.7.4 Determination of safe area

Function： Limit the range of motion of the robotic arm.

Instruction:



1. Open the initialization module, click Safe Area, and the safe area setting box will pop up on the left.

2. Click Add, set the values corresponding to X, Y, etc., click Save to save.

3. Click Add, multiple safe areas can be added. Select the number of the corresponding security area in the drop-down box to set. Click Delete to delete the safe area.

4. When multiple safety areas are set, if the robot arm is within any safety area, it is regarded as a safe state and the robot arm works normally. Otherwise the robot arm stops running, and the log displays an error message.

4.7.5 Tool coordinate function

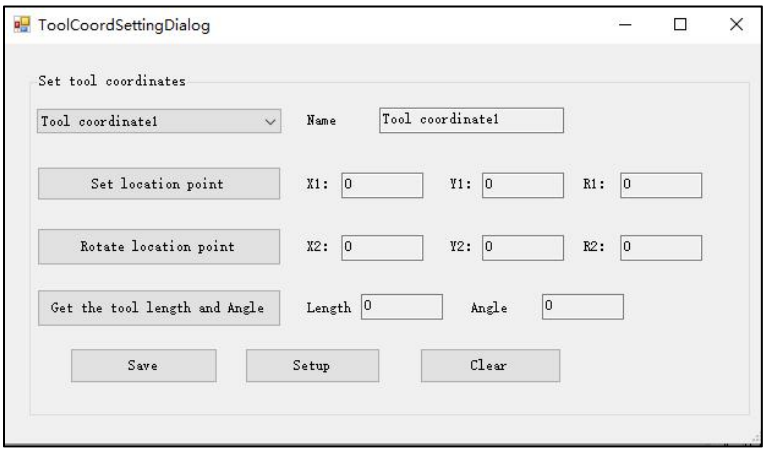
Function： When adding tools at the end of the robot arm, set the tool coordinates and use the end of the tool coordinates as the reference point to control the movement of the robot arm.

Instruction:

1. Double-click the initialization control, find the Set Tool Coordinates button on the interface,



, click the button, and the following interface will pop up.



(1) Select the tool coordinate number.

(2) Move the end of the robotic arm to the desired position, and click to set the positioning point.

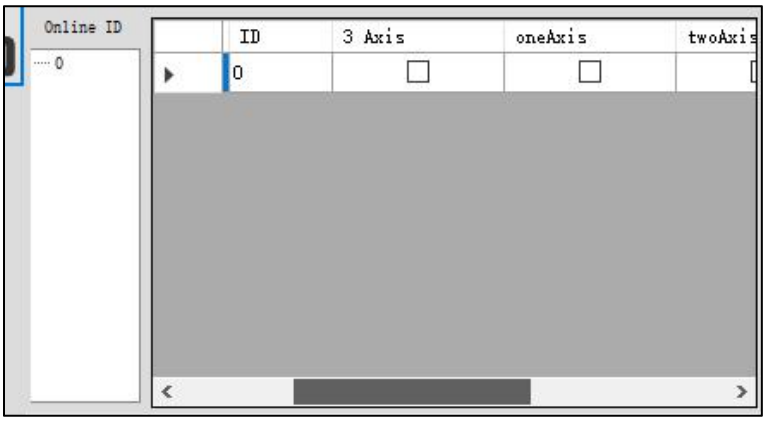
(3) After rotating the R-axis of the robotic arm by a certain angle, move the end of the robotic arm to the position of the tool coordinate point 1 again through the X-axis and Y-axis, and click the rotated positioning point.

(4) Click the Get Tool Length and Angle button. Confirm the deviation between the obtained data and the actual data, which should be in line with expectation. Click the Save button to record the data.

(5) Click the setting button to take effect. Click the Clear button to cancel the tool coordinate setting.

4.7.6 Joint detection mask

Function: When the robotic arm software is started, the specified joint state is not detected. When used to cancel some joints, the robotic arm can still be used (commonly used to cancel R-axis detection).



Instruction:

After placing the mouse on the ID label on the upper left side, the above pop-up window will pop up for setting.

(1) 3 Axis: Shield R-axis detection. (For compatible use, fourAxis must be checked to take effect)

(2) oneAxis: When checked, the detection of joint 1 is shielded.

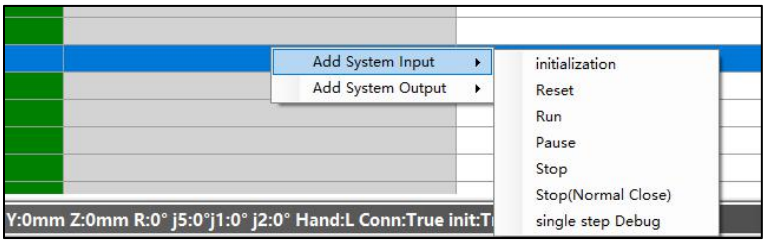
(3) twoAxis: When checked, the detection of joint 2 is shielded.

(4) threeAxis: When checked, the detection of joint 3 is shielded.

(5) fourAxis: When checked, the detection of joint 4 is shielded.

4.7.7 I/O control single-step debugging function

Function: Single-step running process through I/O signal.



Instruction:

Add an input signal in the signal cell and click single-step debugging to add it. Add the corresponding I/O signal to execute the single-step debugging function.

V.Instruction of relevant configuration (the value is uniformly set to 0 or 1)

5.1 Instruction of key-value pair configuration in HitBotStudio.exe.config file

File path: \Release\HitBotStudio.exe.config (open the file with Notepad)

(1) < add key="disable_robot_when_stop_io_up" value="0" />

When the value is set to 1 or empty, when the stop button or the stop (normally closed) button is triggered, the robot arm will clear the initialization status flag, and it needs to be re-initialized for use.

When the value is set to 0 or other values, the enable is normal, and the robotic arm does not need to be re-initialized.

(2) <add key="update_var_in_script" value="0"/>

When value is set to 1, the robot will update the value of the global variable while running the script.

When the value is 0 or other values, the robot arm will not update the value of the global variable until the end of running the script.

(When the key is 0, the value has changed inside the script, but when accessed externally, it is still the value before running the script.)

(3) < add key = "retain_var_at_start" value = "1" />

When the value is set to 1, the robot will not change the value of the global variable when it starts.

When the value is set to 0 or other values or empty, the robot arm will reset the value of the global variable to the initial value when click to run.

(4) <add key="LoadSaveData" value="0"/>

When value is set to 0, the value of the variable will be the initial default value when the robot starts.

When the value is set to 1 or other values, the value of the variable at the start of the robot armis the value of the variable at the end of the last run of the software (used to restore software runtime variables in case of power failure).

5.2 Instruction of software operating environment configuration

File Path: \Release\Dependency

(1) Unzip the Dependency file and run the two unzipped files.

(2) Unzip the IE11_update file and run the exe file in the unzip file directory.

(3) Decompress the Microsoft Runtime Collection, run the decompressed exe file, and click Next to install.

VI.Frequently Asked Questions and Solutions

6.1 The computer can not connect to the robot

1. Set the static IP of local Ethernet to "192.168.0.100" (robot host IP), subnet mask: 255.255.255.0.

2. Make sure the firewall is turned off.

3. Confirm that the software running mode is administrator mode.

4. Make sure the green light of the computer network port is always on. If not, please check the network cable and network card.

5. Re-power the robot and try again.

6. If you still cannot connect to the robotic arm using the above methods, please contact us in time.

6.2 Initialization error of missing axis

Open the communication monitoring tool, select the network card flow, pull down to select the corresponding network card, and check the receiving rate. The rate of a single arm should always be around 390kbits / s. If it is significantly lower than this value, it indicates that there is a problem with the network card or switch, which leads to abnormal data transmission and reception. You can restart the computer or replace the computer.

6.3 The log reports over-current protection and needs to be powered off and restarted

Check the wiring and working conditions to see if there is greater resistance to the movement of the robot, which causes the robot to work under the condition of overcoming large resistance, resulting in an increase of current.

6.4 The log reports collision, please reinitialize or power off and restart

1. Check whether the arm body is impacted during the movement of the arm, resulting in collision protection and joint loosening.

2. Whether the payload exceeds the maximum payload, causing the machine to run at a higher

speed. This increases the inertia of the machine, causing the assist function to be triggered when it stops, resulting in a false collision.

6.5 Unable to use the drag teaching function

During the process of turning on and off the drag teaching, the robot servo has a process of power-off and power-on, please do not switch the teaching state frequently. In addition, after turning on the drag teaching and completing the operation, if want to turn off the drag teaching, please wait about 5s before performing the relevant operations. If this problem occurs, please try to turn off the drag teaching and re-initialize the robot to see if it can be solved. Note that the drag teaching cannot be enabled for the Z axis.





Huiling-tech Robotic Co.,Ltd.

Tel: 0755-36382405 Hotline: 0086-15220259858
Email: Marketing@hitbot.cc
Website: https://en.hitbot.cc
Address: 2nd Floor, Building E, Huafeng International Robot Industrial Park, Hangcheng Ave,
Xixiang St, Baoan District, Shenzhen City, Guangdong Province, China