CS Series Quick Start Guide

Suzhou Elite Robots

V2.6

Preamble

Please carefully read the guide before use. The guide gives a brief summary of the robot operations. For more details, please refer to the user manual. Before use, check and make sure that the version information of the product is correct. The guide will be periodically checked and revised, and the updates will be included in the new version. The content or information herein is subject to change without prior notice.

Read the guide carefully before the installation and use of the product and keep it for your easy reference. The pictures included in the guide are used for reference only, which is subject to the product received.

CONTENTS

1. Ready to Use

How to connect the CS series robot to the hardware (e.g. controller)?

1.1 Robot Installation

Mount the robot in a flat and firm surface, which is sufficient to carry at least 5 times the weight of the robot. See Appendix 1-1 for the installation dimensions of all CS series robots. Please refer to the dimensions when installing.

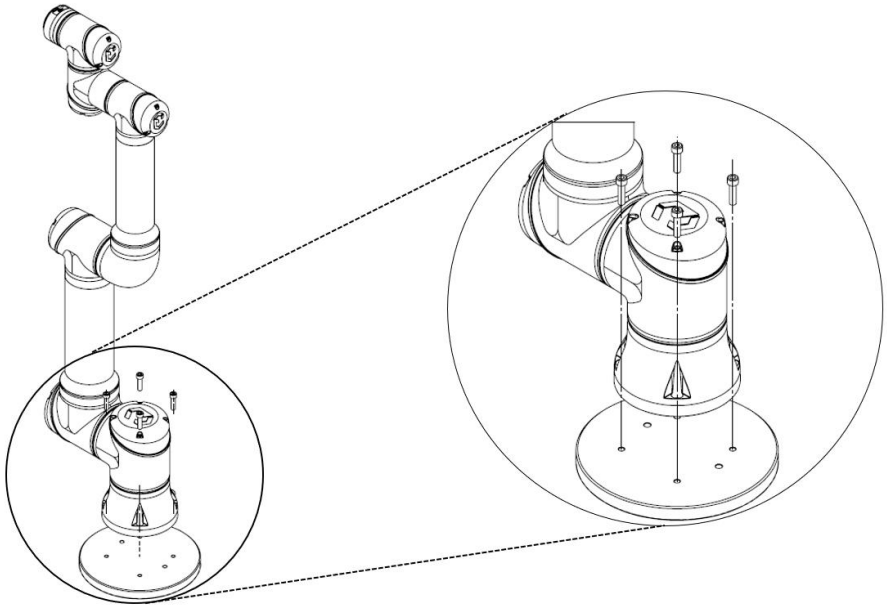


Figure 1-1

↓

1.2 Robot & Controller Connection

Insert the plug to the connector at the bottom of the controller and turn the locking ring on the plug clockwise to lock the connection. If it is difficult to rotate, it indicates that the plug is not inserted into the innermost part. (Note that the controller in the following figure is a standard equipment. If a customized or a MINI controller is connected, please contact our technical support department.)

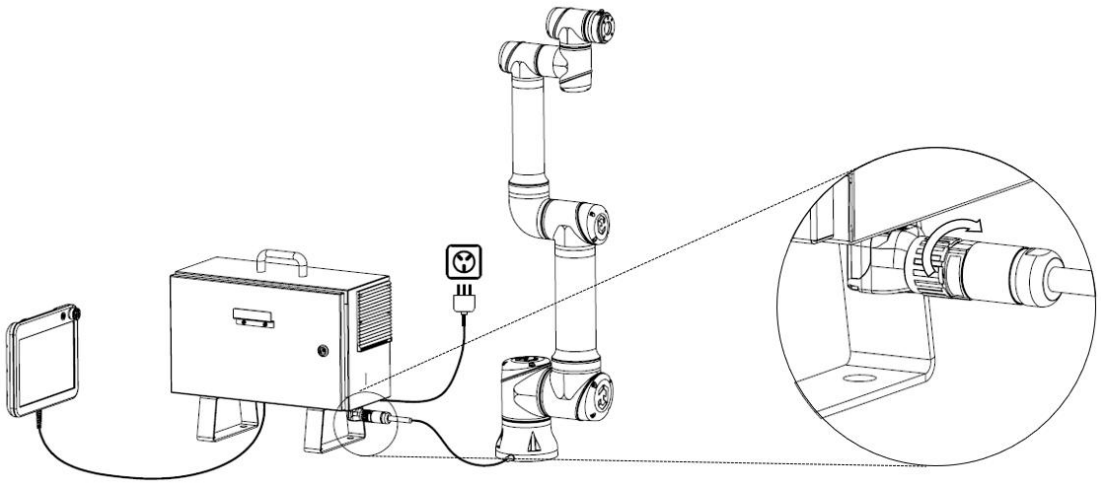


Figure 1-2

1.3 Power On

After connecting the controller, the teach pendant, the robot and the power supply, insert the power plug to the socket and push the red switch button at the bottom of the controller, as shown in the following figure.

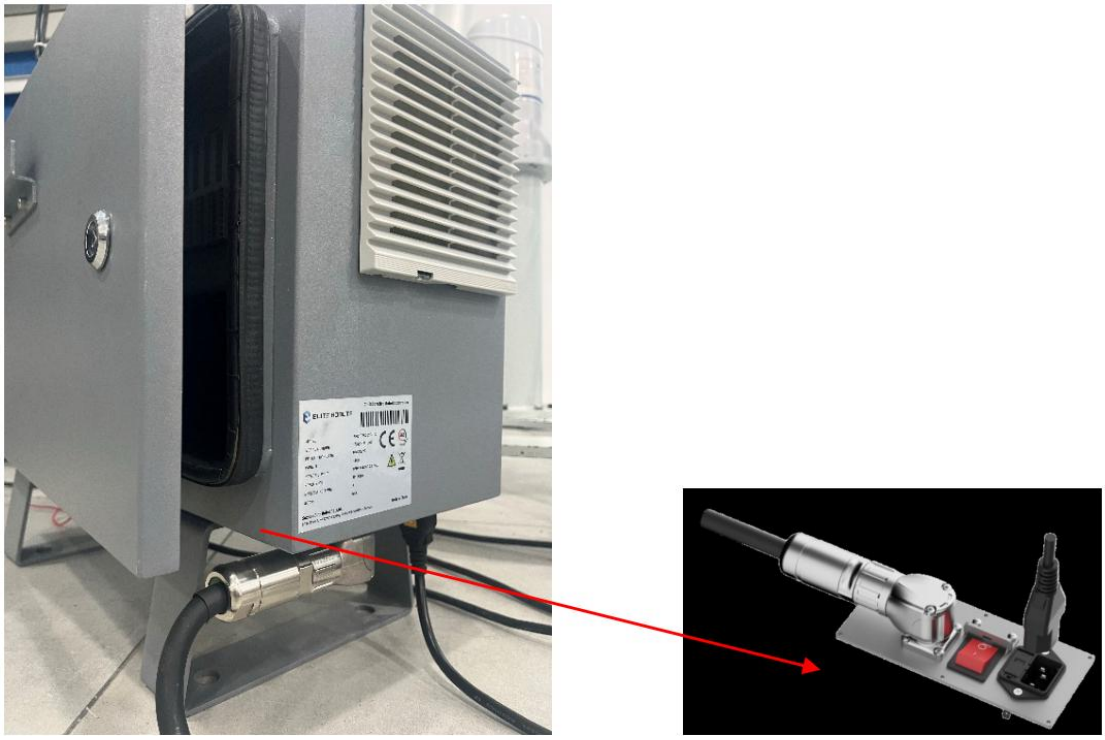


Figure 1-3

2. Startup

Power on the robot before the startup.

Step:

① Press the power button on the teach pendant to start it up.

Figure 2-1

② Select the interface languange. When it is the first time to start up the robot, a dialog box of selecting a languange will pop up. Select ”Chinese” and click ”OK”.

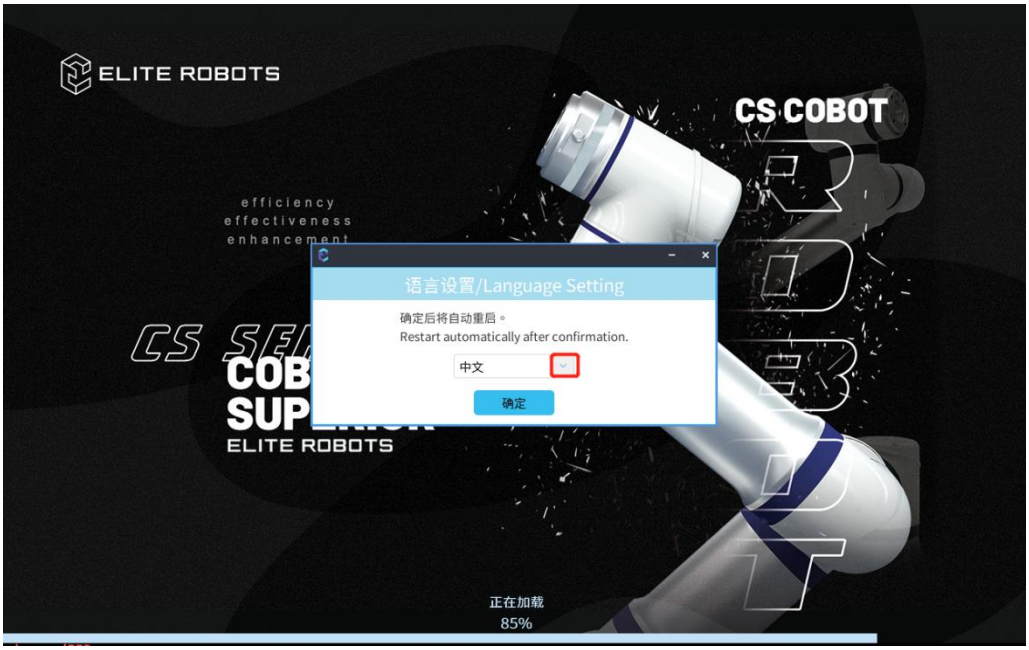


Figure 2-2

③ Click ”Power On” and ”Brakes Release” in sequence.

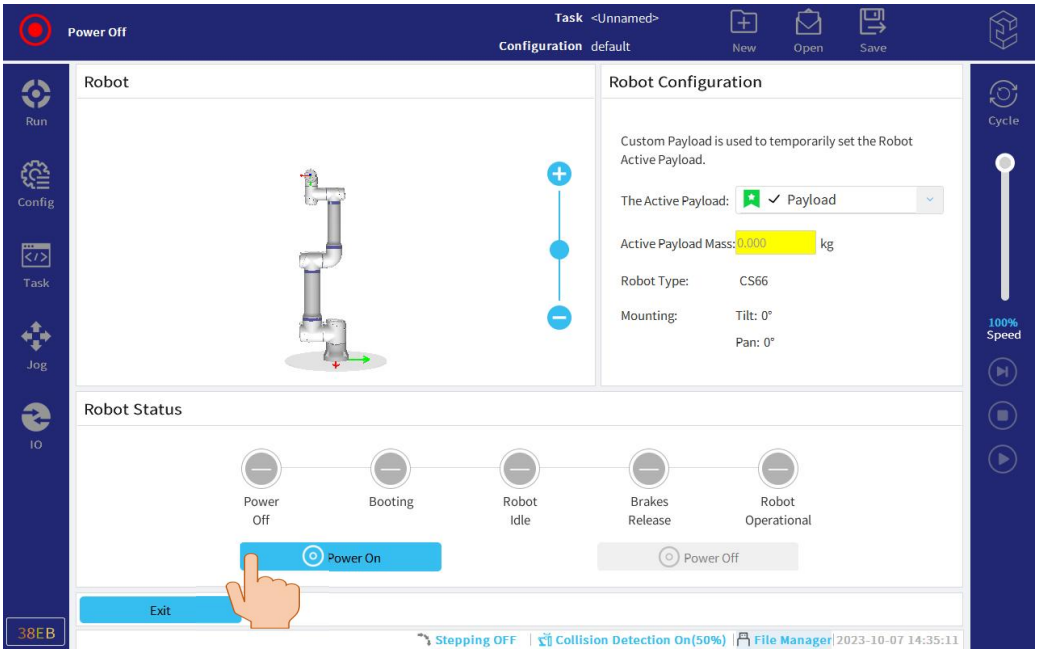


Figure 2-3

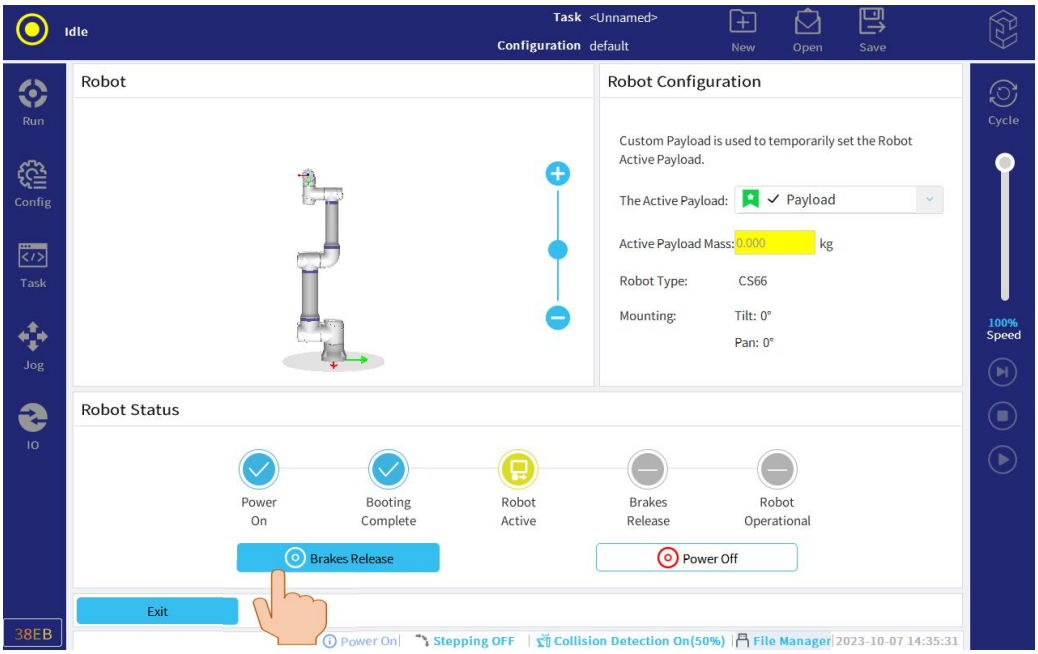


Figure 2-4

④ To exit the current initialization interface, click ”Exit”. At the time, the green terminal indicator in the robot flashes.

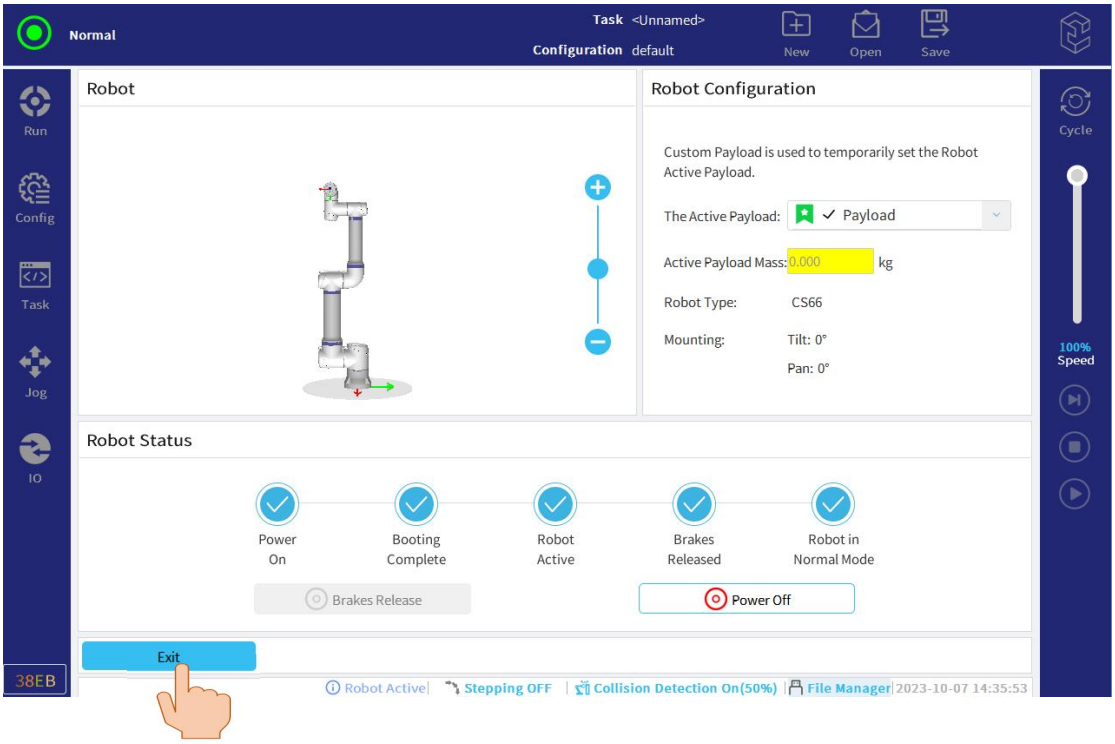


Figure 2-5

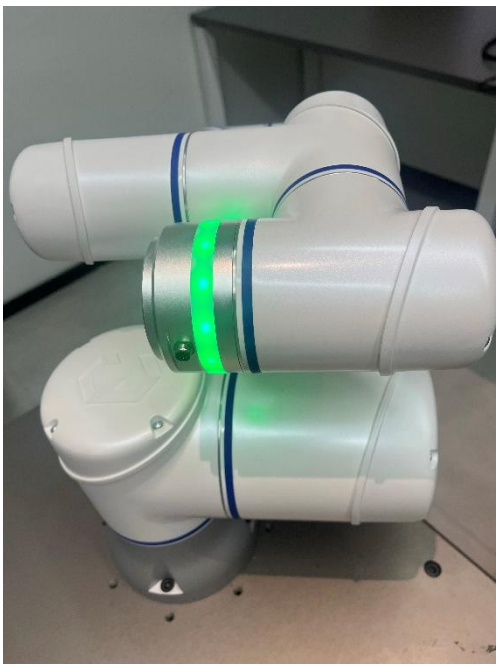


Figure 2-6

3. Preparations

Before creating a new task, please check the mechanical home of the robot and properly set all the basic parameters, incl. the mounting angle, tool TCP, payload and others.

3.1 Set Mounting Angle

Step: Click **Config -> Mounting**. Please set the mounting angle of the robot upon the actual mounting situation.

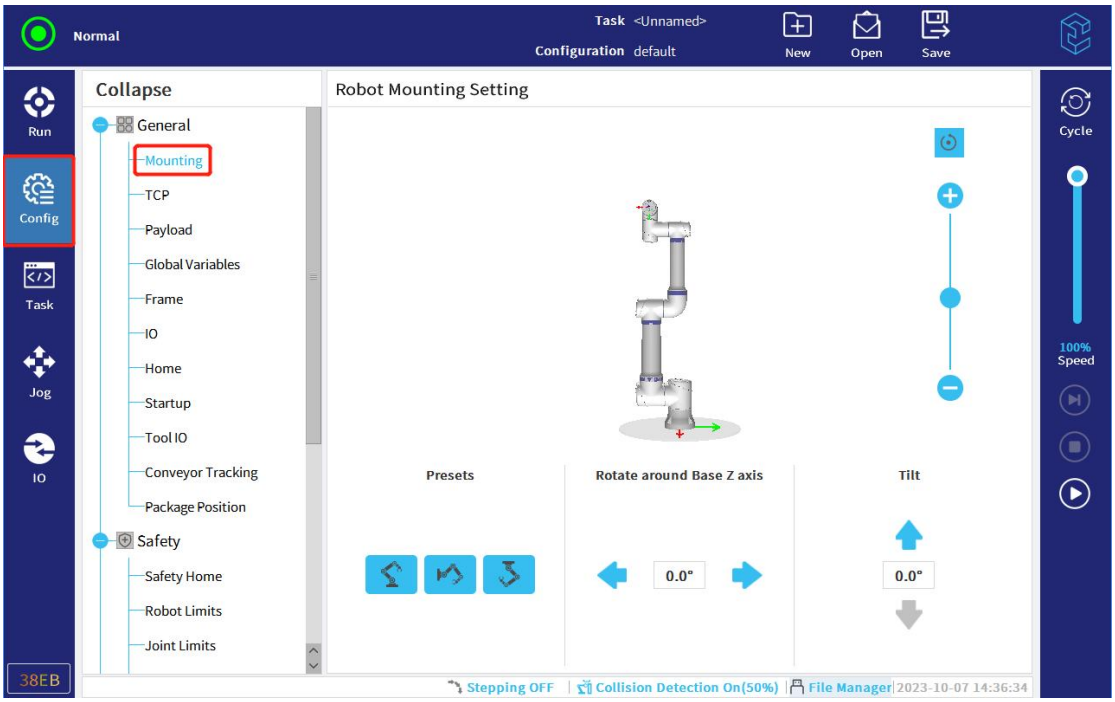


Figure 3-1

3.2 Check the Robot Pose at the Home

Check if the robot pose at the default home is consistent with that is shown in the teach pendant.

Step: Click **Jog -> Home -> Auto Move** (press and hold the ”Auto Move” button, and the robot will turn back to the mechnical home by default. Release the button when the robot stops running.). Then click ”continue”. After the robot goes back to the home, the outlet of the power cord points towards the center of the base (i.e. positive

direction of the X axis). Please check the direction.

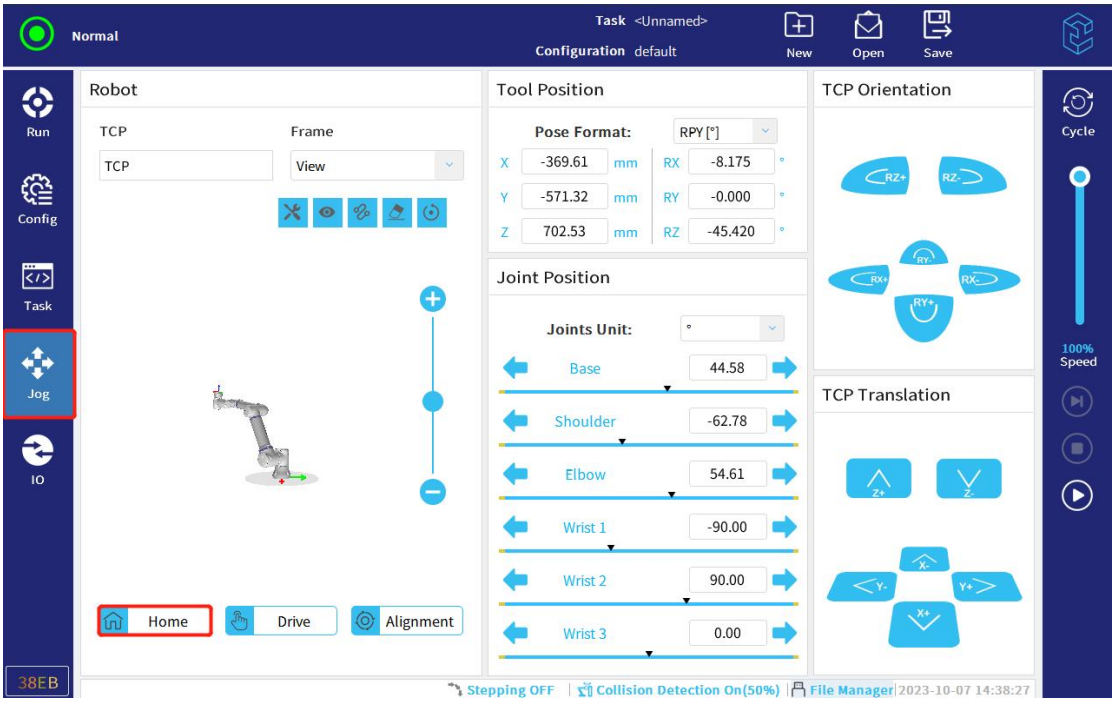


Figure3-2

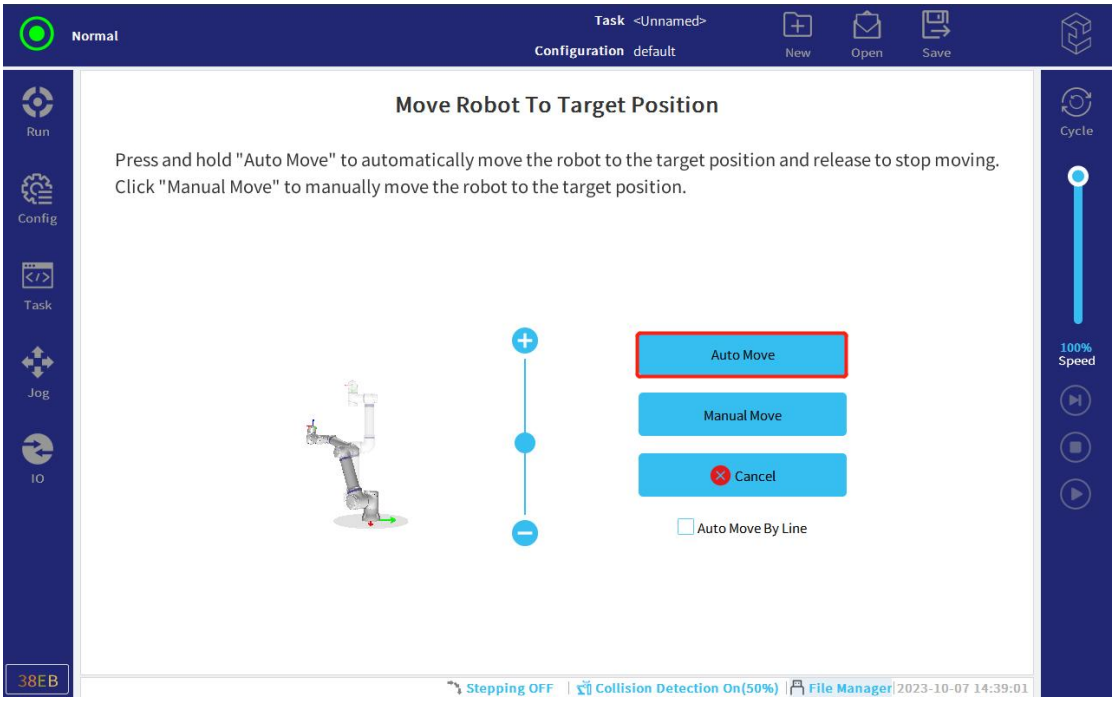


Figure 3-3

3.3 Set TCP

TCP is the abbreviation for Tool Center Point. In the robot system, TCP of the terminal tool (gripper or suction cup) refers to a point in contact with the workpiece.

The position and direction that the point moves are the trajectory and the direction of the robo. Here is an example.

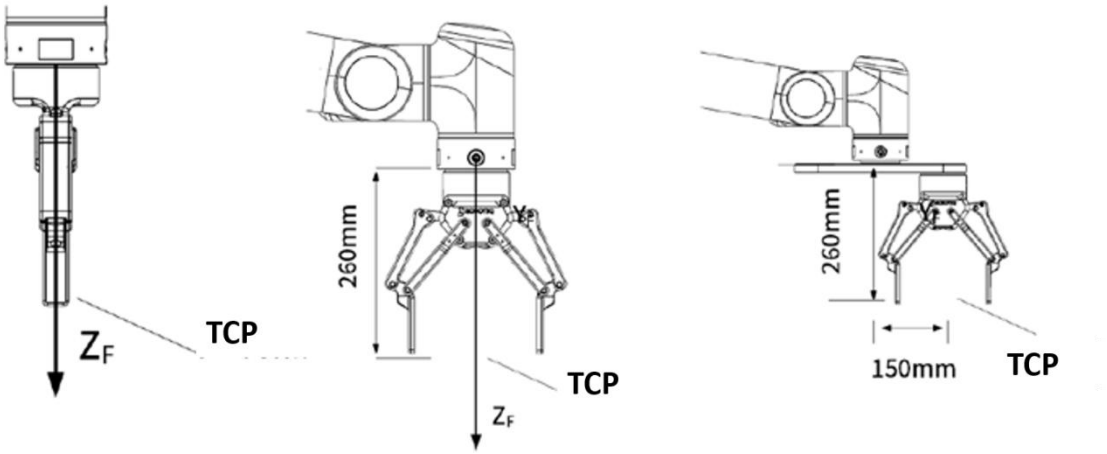


Figure 3-4

The coordinate data on the pictures in the left and middle is as follows:
X, Y, Rx, Ry and Rz are 0, and Z is 260.
The coordinate data on the right picture is as follows:
X, Rx, Ry and Rz are 0, Y is 150, and Z is 260.

3.3.1 Set TCP by Entering the Values

If the coordinate data of the TCP in the mathmatic model is known, please click the input box marked by a red square and enter the values of x, y, z, rx, ry and rz.

Step: Click **Config -> TCP**. To set the position and direction of the TCP, click the input box marked by a red square and enter the values of x, y, z, rx, ry and rz.

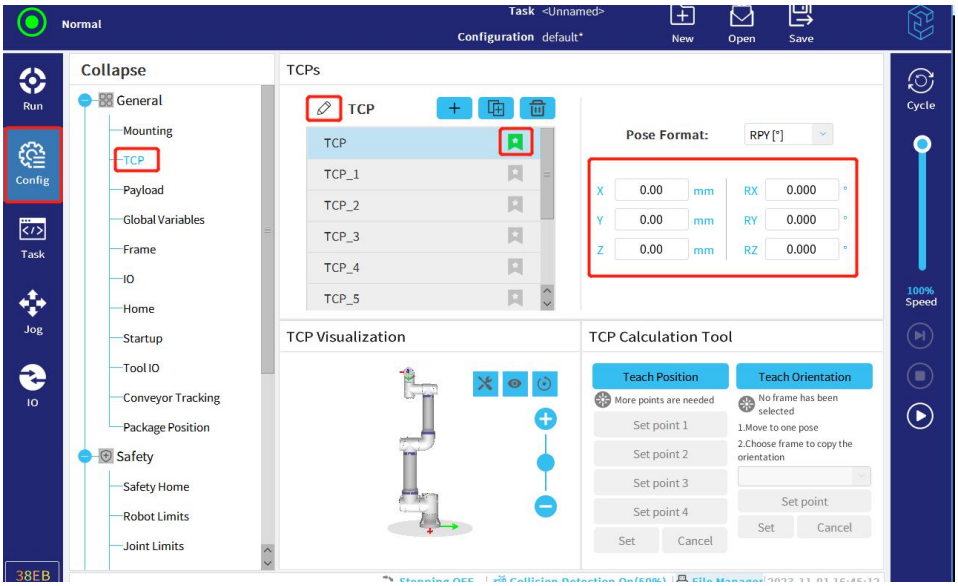


Figure 3-5

Note:

: Rename the TCP

: Create a new TCP

: Copy a TCP

: Delete a TCP

: The TCP is currently activated

3.3.2 Set TCP by Executing the Script Commands

The **coordinate and pose of the TCP** can be set by the script command set_tcp(pose).

Step: Click **Task -> Advanced -> Script**, select ”Line” from the Source dropdown list and click the code editing area. Then select ”set_tcp” from the function dropdown list and enter the values of {x,y,z,rx,ry,rz}. Press the Enter button, as shown in the following figure.

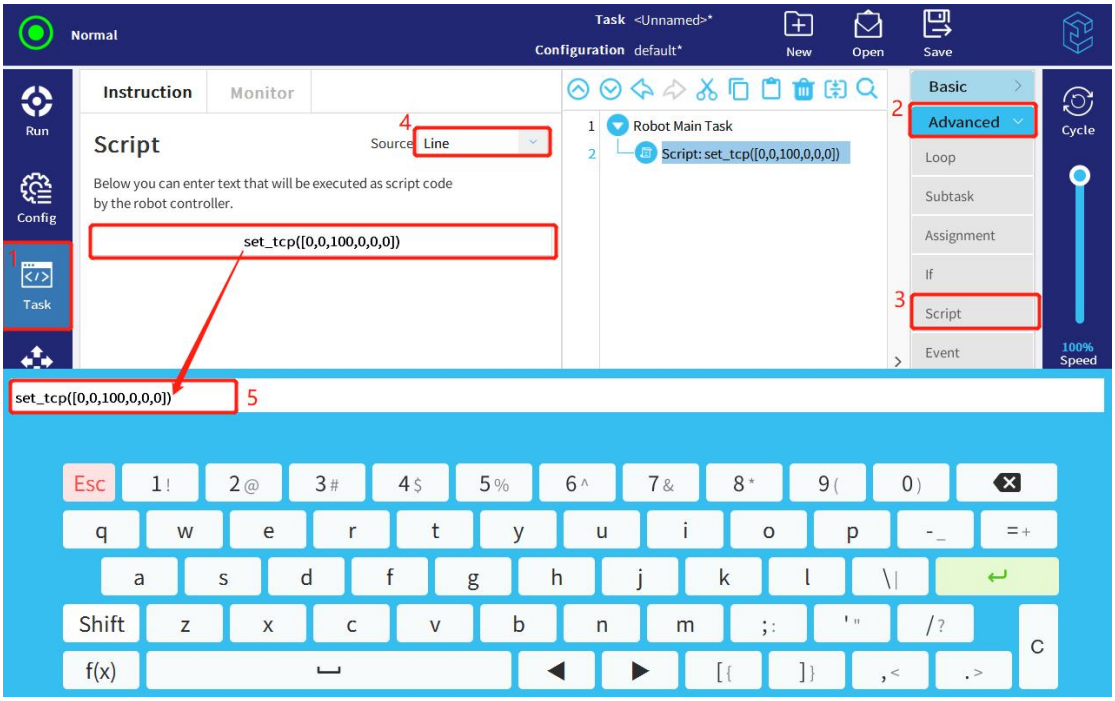


Figure 3-6

3.3.3 Set TCP with 4-Point Teaching Method

Move the robot to contact with the fixed reference point in 4 different poses and calculate the deviation of the tool TCP from the center (TOOL0) of the robot flange in the X, Y and Z direction. The direction of TCP can be set by one of the points.

Step:

1. Select a fixed reference point in the working space of the robot. Move the TCP to contact with the point and click ”Set point 1”. Change the robot pose and move the TCP to contact with the point again. Click ”Set point 2”. Repeat the steps above until the 4 points are set. Then click ”Set”. The system will automatically work out the position of the TCP.

2. Click ”Teach Pose” and select the reference frame. Change the TCP pose to make the TCP direction parallel with the x/y/z direction of the frame, respectively. Then click ”Set point” and ”Set”. After all poses are taught, the system will automatically calculate the direction of the TCP.

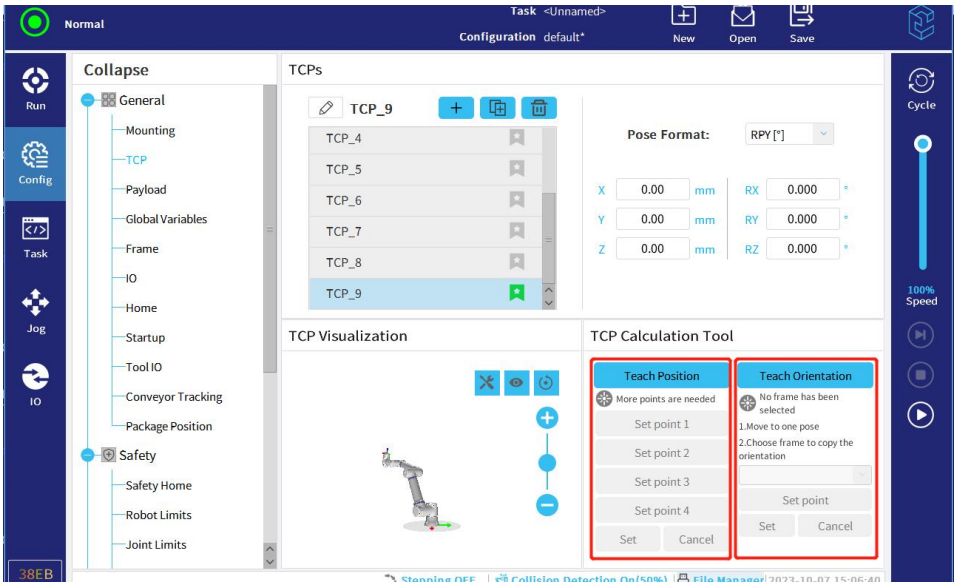


Figure 3-7

Here gives an example of the point teaching in 4 diffferent poses. In the example, we take the red point pinched by the DH e-gripper as the TCP to contact the tip of the calibration tools.

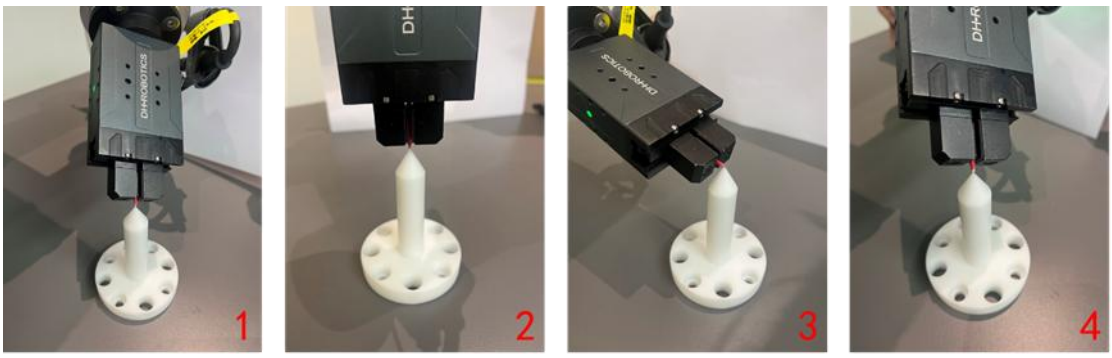


Figure 3-8

The reference frame in the following figure is the robot coordinate system. The orientation going straight up is the positive direction of the Z axis (Z+). The outlet of the power cord points towards the center of the base (i.e. positive direction of the X axis (X+)). The home of the coordinate system is the center of the base.

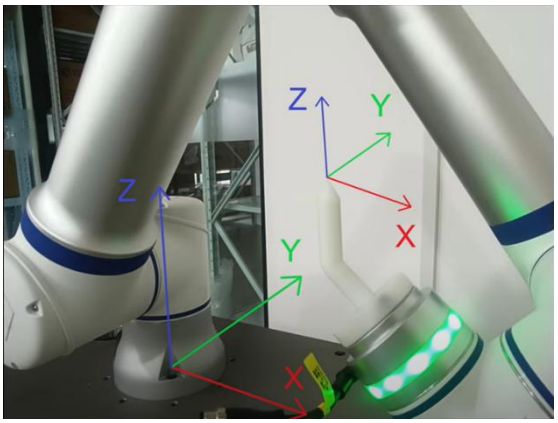


Figure 3-9

Note: After the teaching is complete, the user can select the tool coordinate system as the Kinetic Coordinate System. To test whether the TCP is correctly set, please move the robot along the Rx, Ry and Rz direction. To verify if the direction of the TCP is the pre-set one, please move the robot along the X, Y and Z direction.

3.4 Tool Installation

When installing the tool, use the M6 screws with a torque of 8Nm and a grade of 8.8 strength. It is suggested to use the pins to locate the installation position of the tool. Do not adjust the screws too tight. The depth shall be no more than 8mm. See Appendix 1-2 for the tool installation dimensions of all CS series robots. Please refer to the dimensions when installing.

3.5 Payload Setting

After the terminal tool is installed, the robot will carry the load. It is necessary to set the payload. The payload refers to the weight and inertia that the terminal end can bear, including the weight of the tool itself and the workpiece pinched by the tool. Note that it excludes the weight of the robot.

Step: Click **Config -> Payload**. The steps for setting the payload are the same as those of setting TCP, so are the pictures. Enter the values of ”Mass” and ”CoG” in the input box marked by a red square. The system will automatically work out the moment of inertia.

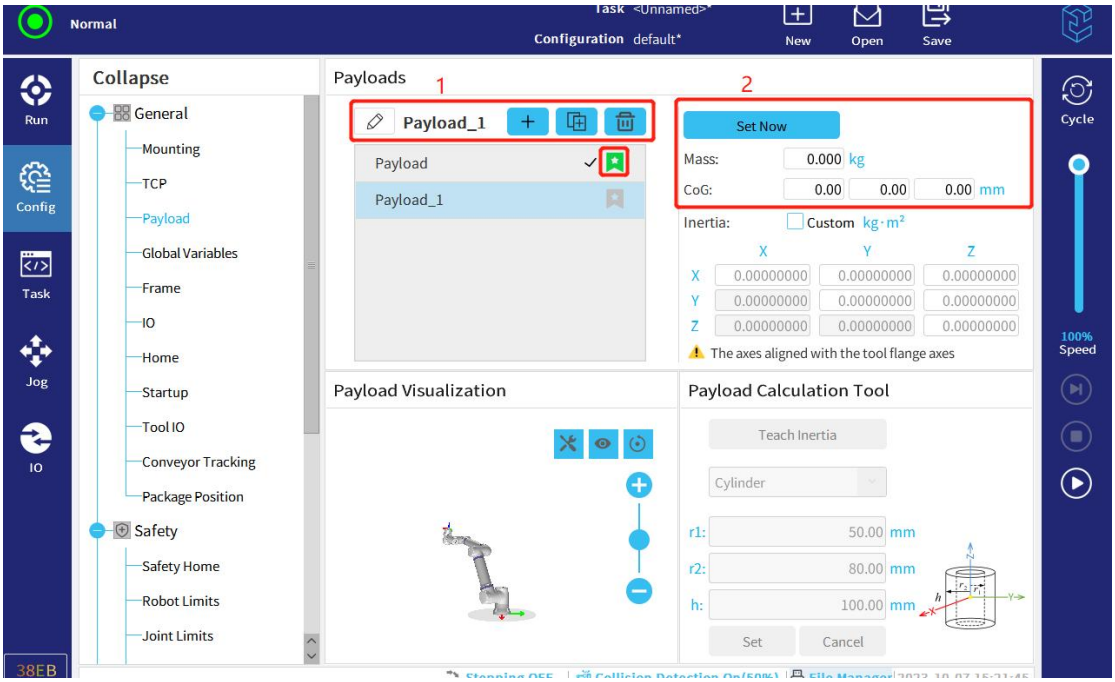


Figure 3-10

3.5.1 Calculate the Inertia with the Model

If the tool is in the special shape, please select the corresponding model and correct the inertia data.

Step: ① Select ”Custom” in the right side of ”Inertia”

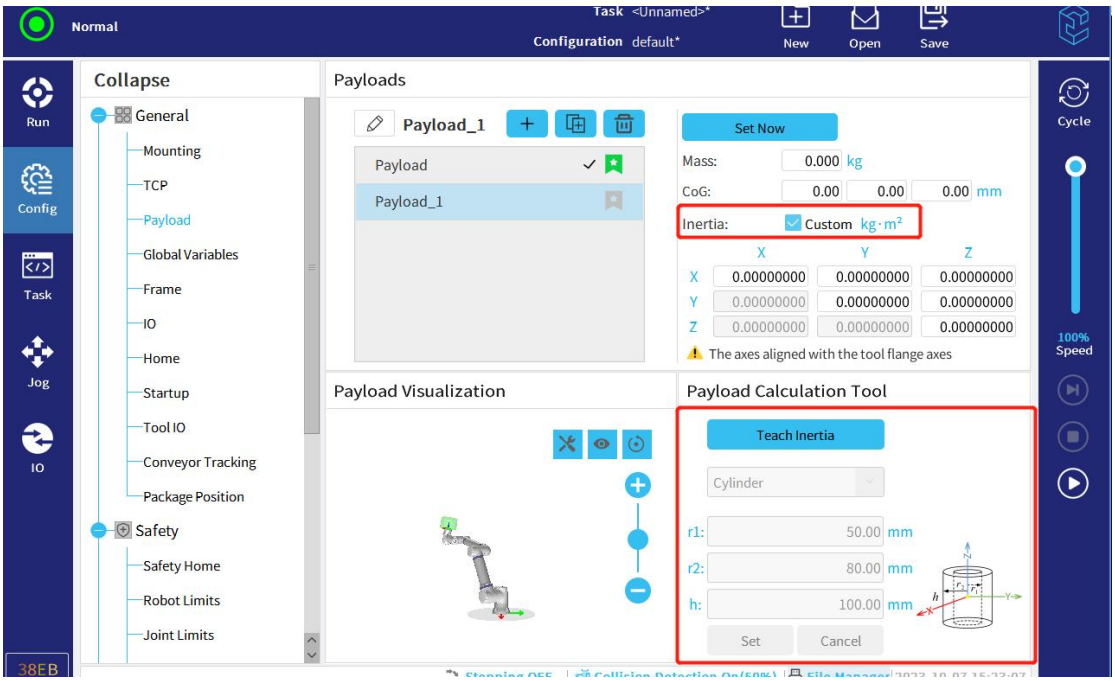


Figure 3-11

② Use the corresponding model in accordance with the tool installed

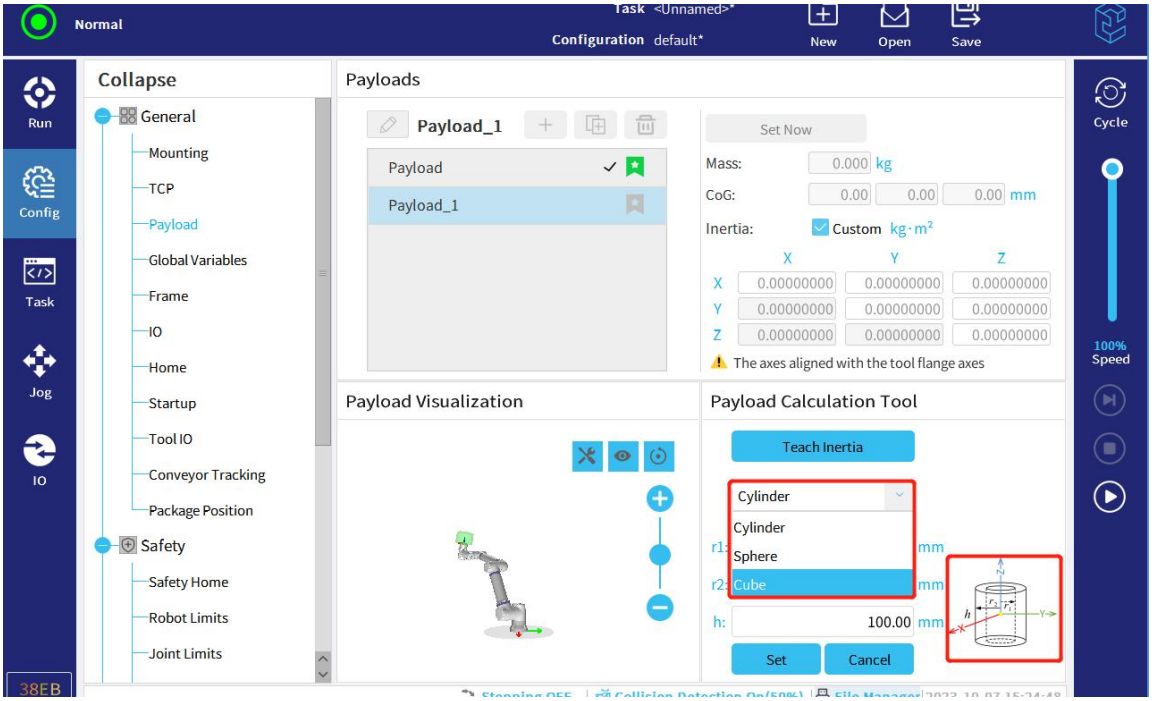


Figure 3-12

③ Set the parameters, e.g. use the Cylinder, enter the radius r1, the radius r2 and the height of the internal circle. The system will automatically work out the inertia and the value will display in the page.

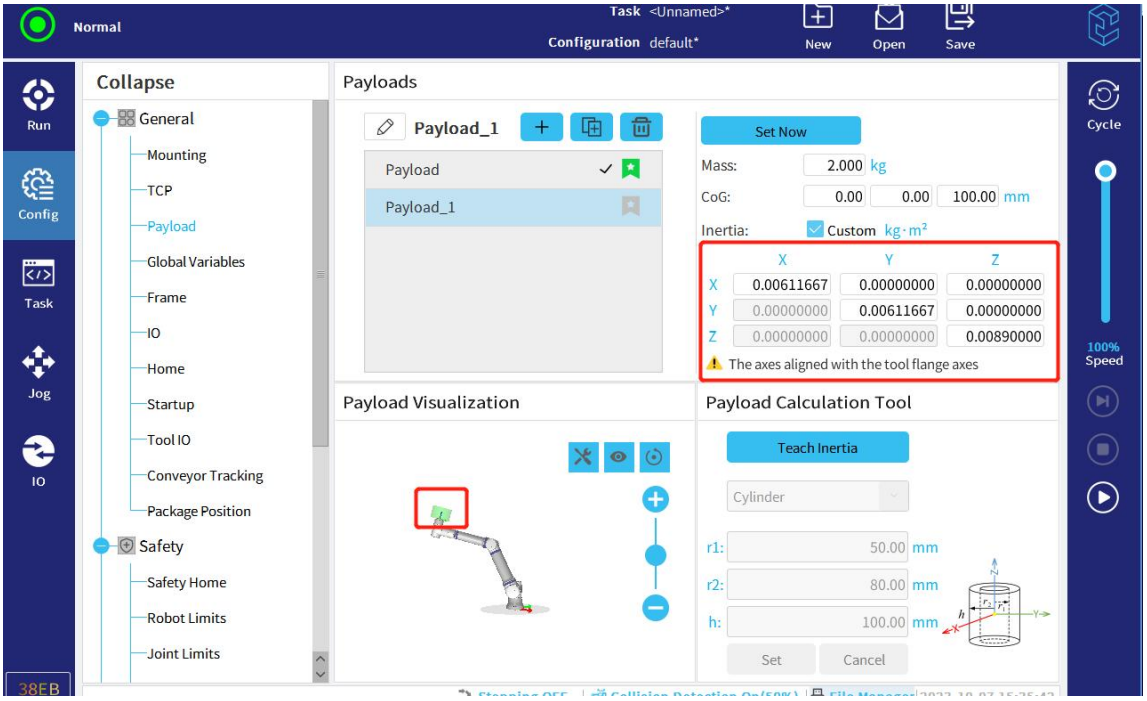


Figure 3-13

3.5.2 Set the Payload in the Task Node

Set the payload in the task node.

Step:

Method 1: Click **Task -> Basic -> Set Payload**. Select ”payload” or ”payload_1” from the dropdown list (see the part marked with the number ”3”). Then click ”Set Now”.

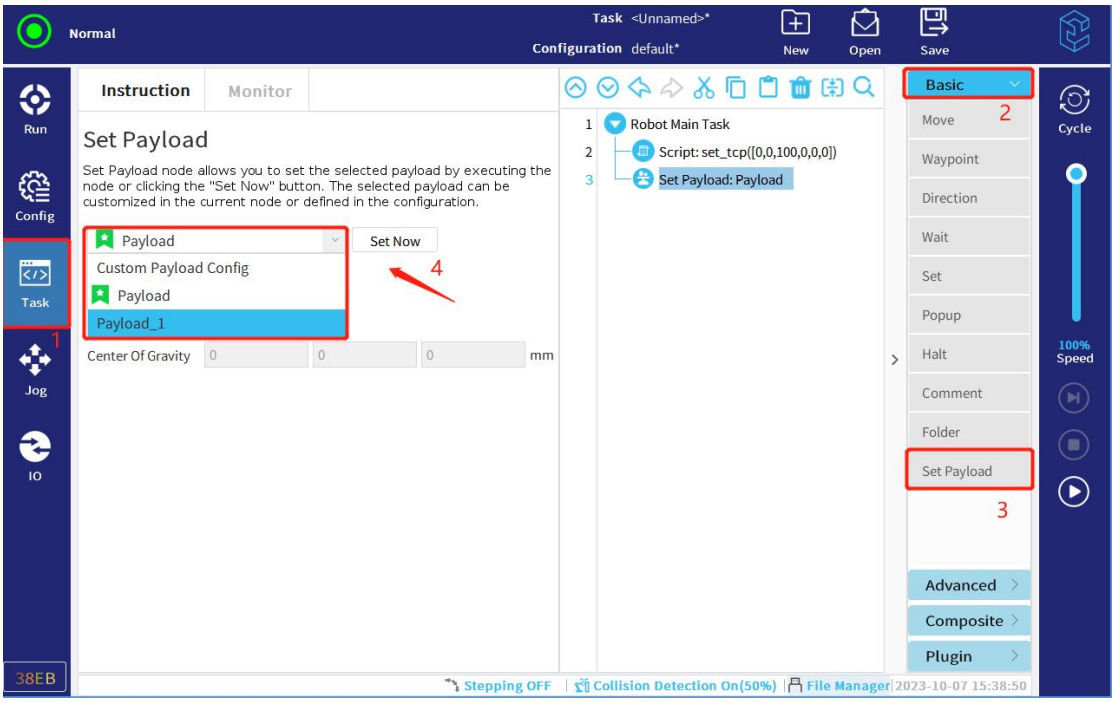


Figure 3-14

Method 2: Repeat the method 1, select ”Custom Payload Config” and enter the values in the input box of Mass and Center of Gravity.

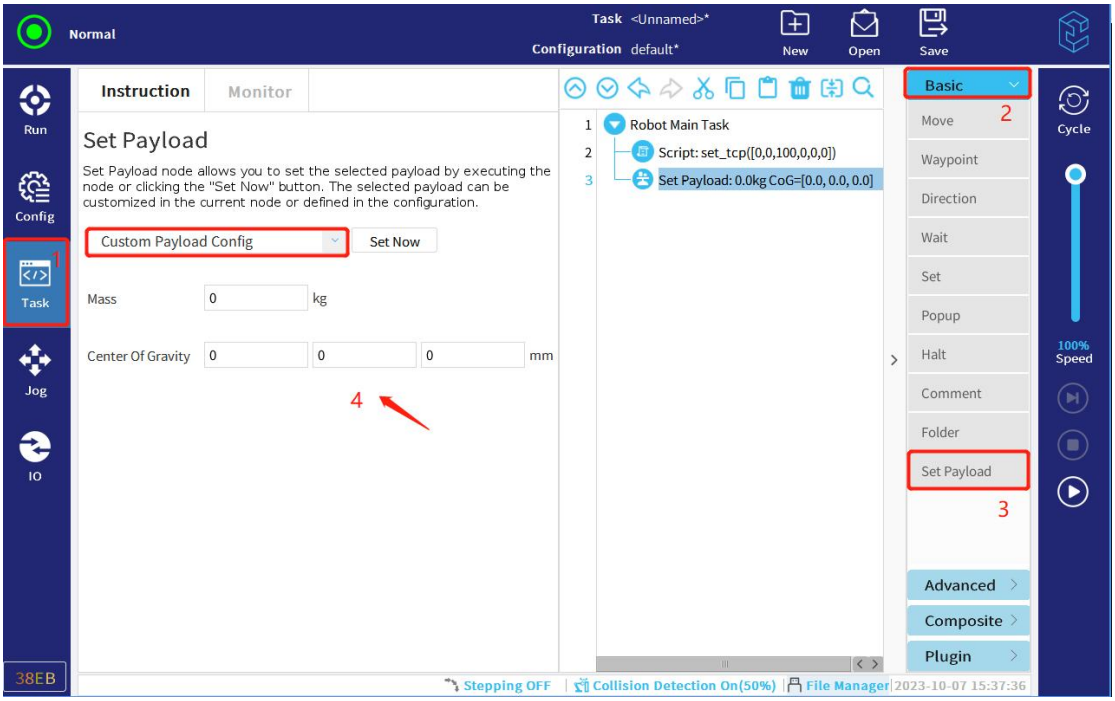


Figure 3-15

3.6 Payload Diagrams

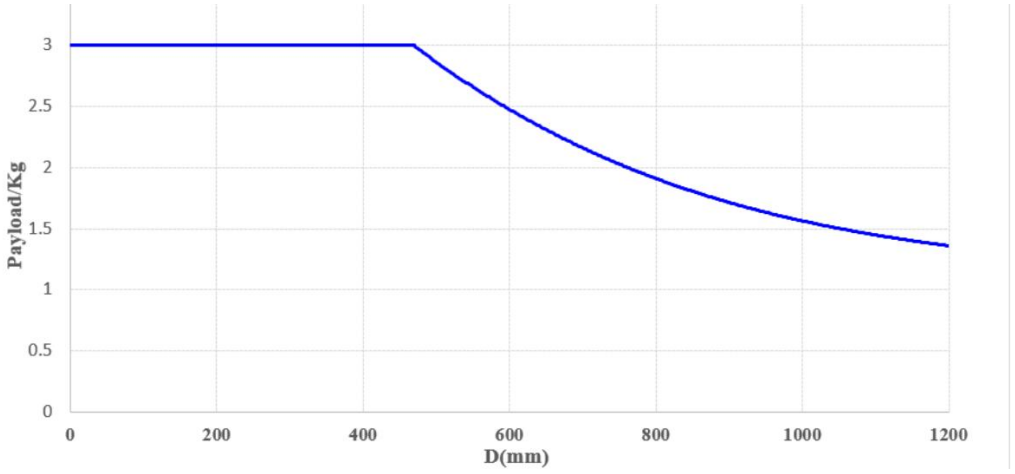


Figure 3-16

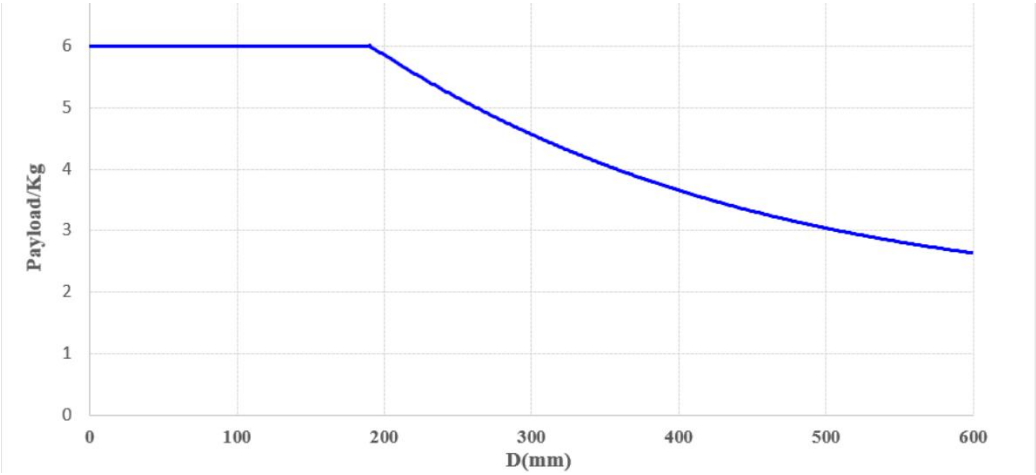


Figure 3-17

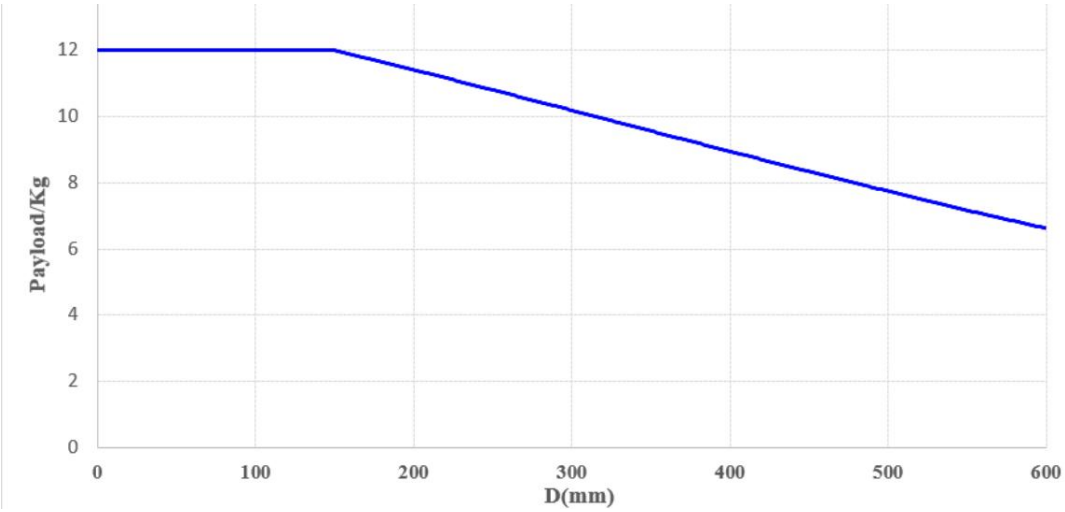


Figure 3-18

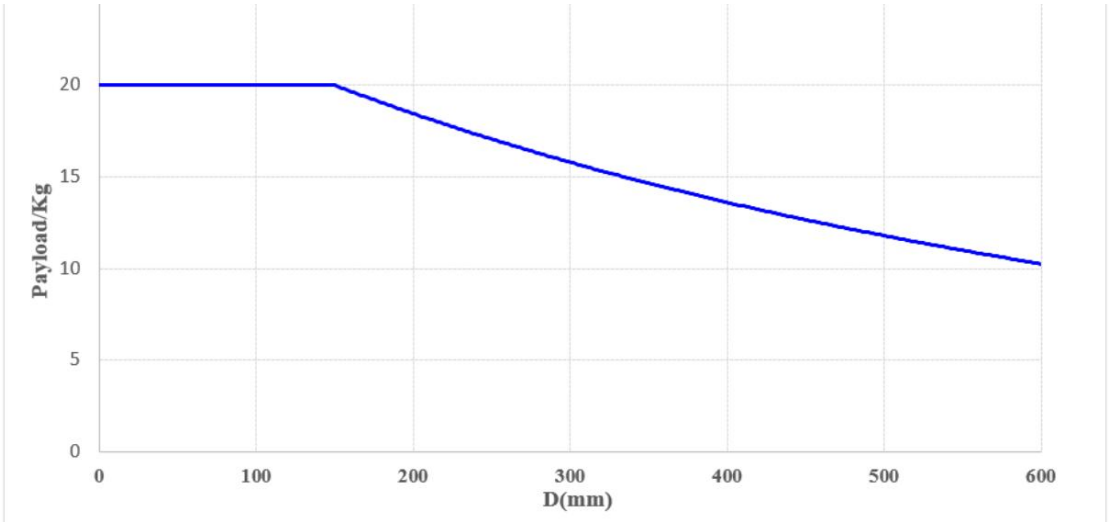


Figure 3-19

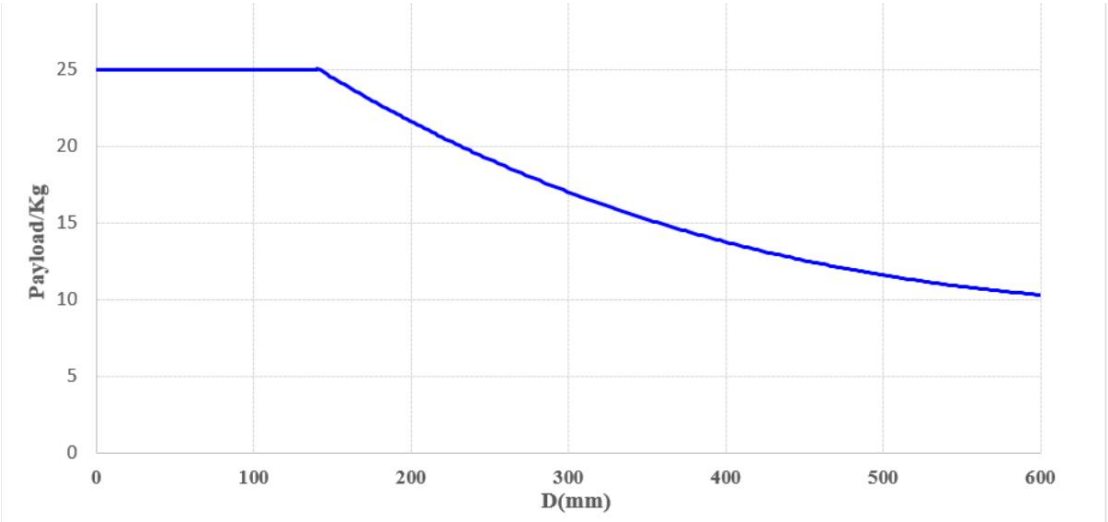


Figure 3-20

4. Define the Coordinate System

The user can customize the coordinate system in accordance with the actual movement of the robot.

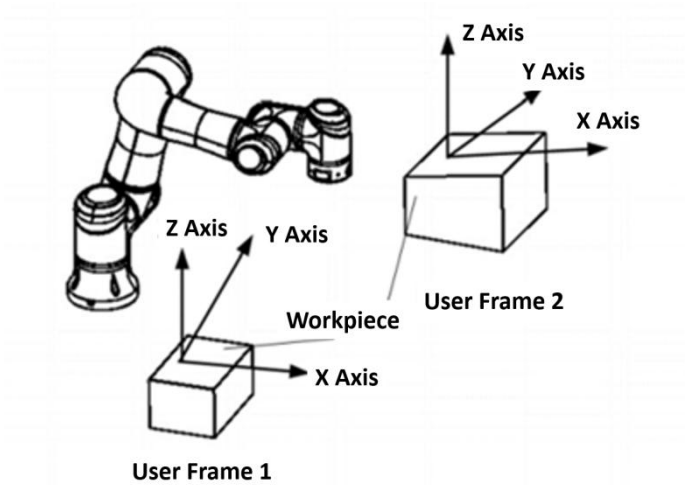


Figure 4-1

4.1 Enter the Values

Step: Click **Config -> Frame**. Enter the values of X, Y, Rx, Ry and Rz in the input box marked by a red square in the right.

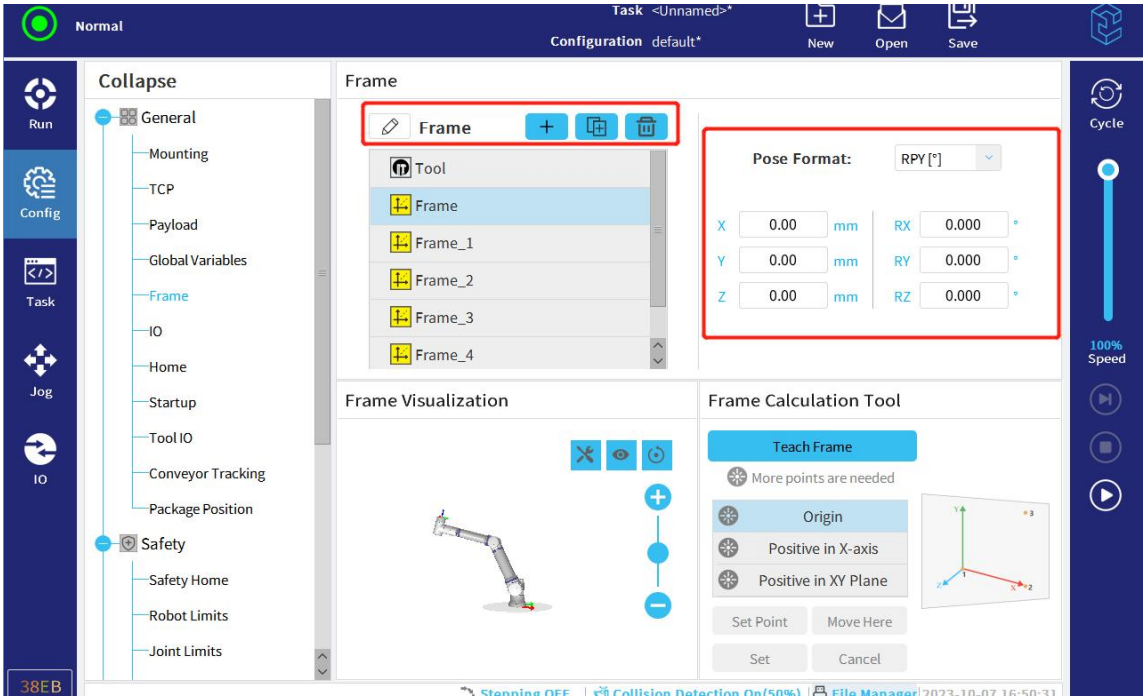


Figure 4-2

4.2 3-Point Teaching Method

Step: Click **Config -> Frame**. Click ”+” to add a frame. Click to rename the frame. Then Click the ”Teach Frame” button.

Click the ”Origin” button and the TCP will be moved to the home *O.* Then Click ”Set Point”.

Move the TCP to a point in the positive direction of the X axis and click ”Set Point”. Move the TCP to a point on the XY plane and click ”Set Point”. Then click ”Set”. The system will automatically work out the coordinate system data.

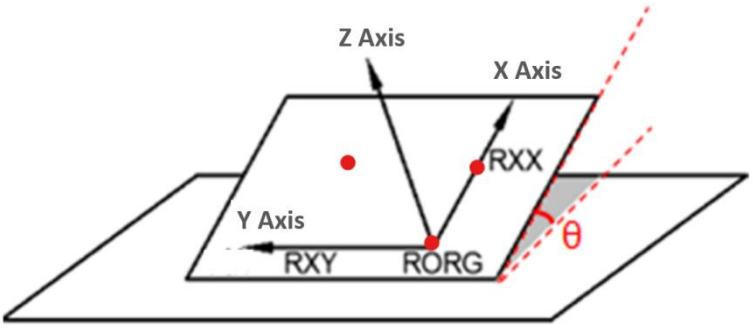


Figure 4-3

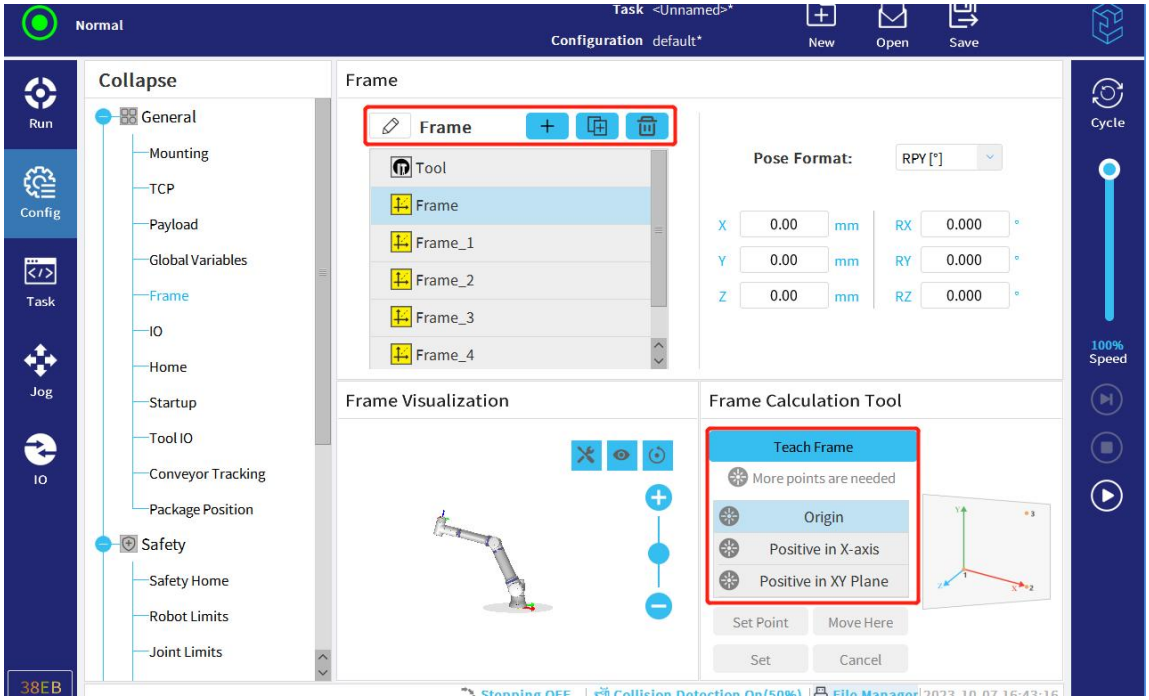


Figure 4-4

5. Move the Robot

5.1 Hand Drag

The user can hand drag the robot in the drag mode. When the drag function is enabled, the drag enable status light will turn blue.

Figure 5-1

Step:

Method 1: To hand drag the robot, press and hold the button 3 in the back of the teach pendant when the robot stops running and no alarm is reported.

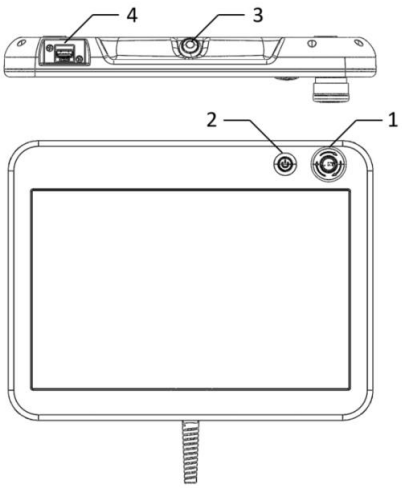


Figure 5-2

Method 2: The user can also hand drag the robot by clicking **Jog**, press and hold the ”Drive” button.

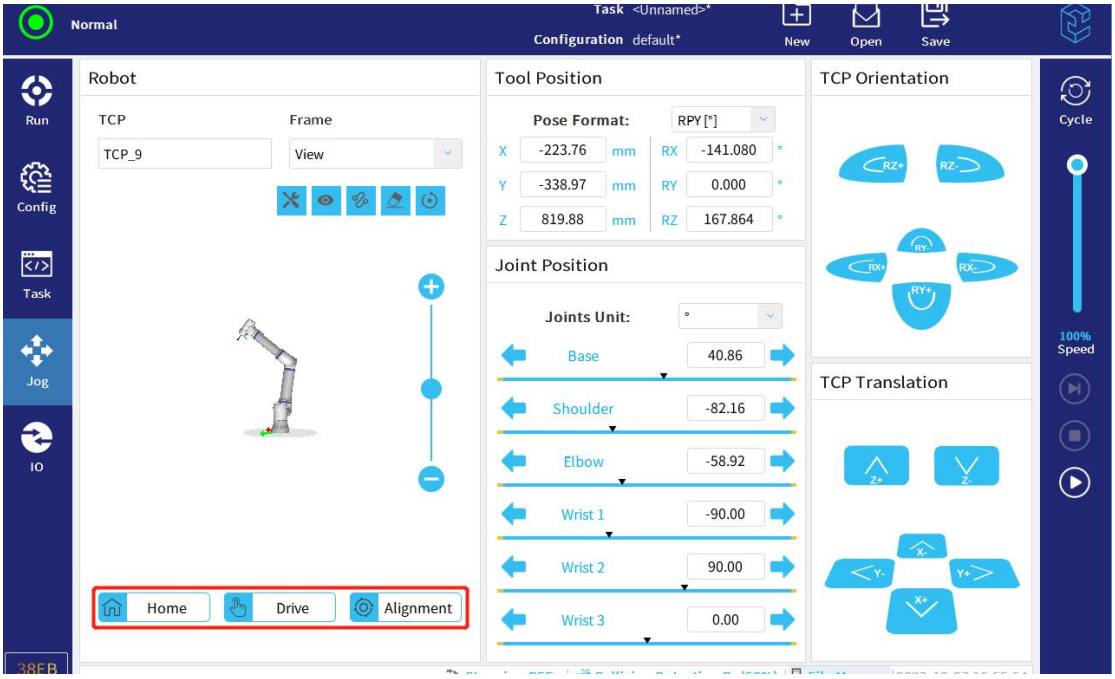


Figure 5-3

5.2 Alignment

Align the currently activited TCP with the selected frame in the XOY plane. A base frame or a user-defined frame is usually selected. The will simplify the motion trajectory of the robot. For instance, it allows the user to aim the gripper at the objects.

Step: Click **Jog** and select the desired frame. Press and hold the ”Align” button until the robot arrives at the target point and stops.

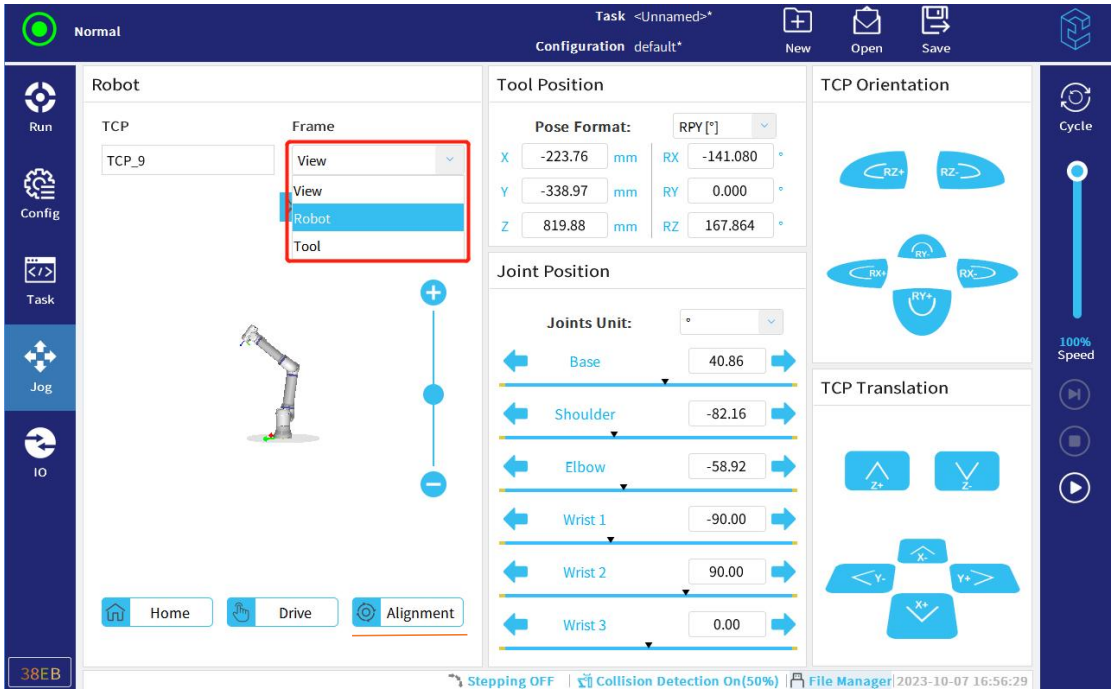


Figure 5-4

5.3 Stepping Mode

After the stepping mode is enabled, the CS series robot will move forward at a settled distance in a certain direction. The range of the fixed distance is 0.01-10mm and that of the fixed angle is 0.01-10°.

Step: Click the ”Stepping” icon, as shown in the following figure. Enter the value 0.5 in the input box of TCP translation, TCP Orientation and Joint Move. Click the arrows and the robot will be in the stepping mode.

Step: Click the ”Stepping” icon, as shown in the following figure. Enter the value 0.5 in the input box of TCP translation, TCP Orientation and Joint Move, which means one step value is set as 0.5mm or 0.5° Press the arrows until the robot reach thetarget pose.

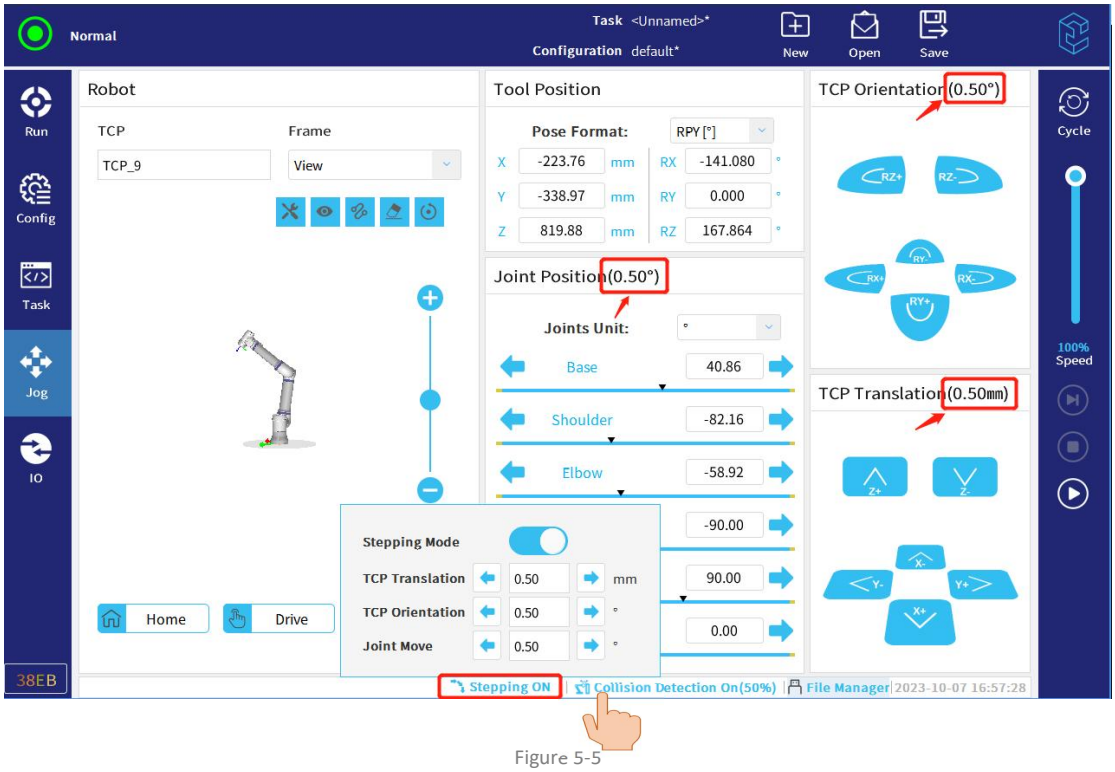


Figure 5-5

5.4 Collision Detection Shortcut Setting

In case that the collision detection is enabled, the robot will stop running when the collision is detected. Click the ”run” button and the program goes on. The collision sensitivity is expressed in percentage. The smaller the value is, the easier to stop when the robot is in a collision.

Step: The collision detection settings are as shown in the following figure.

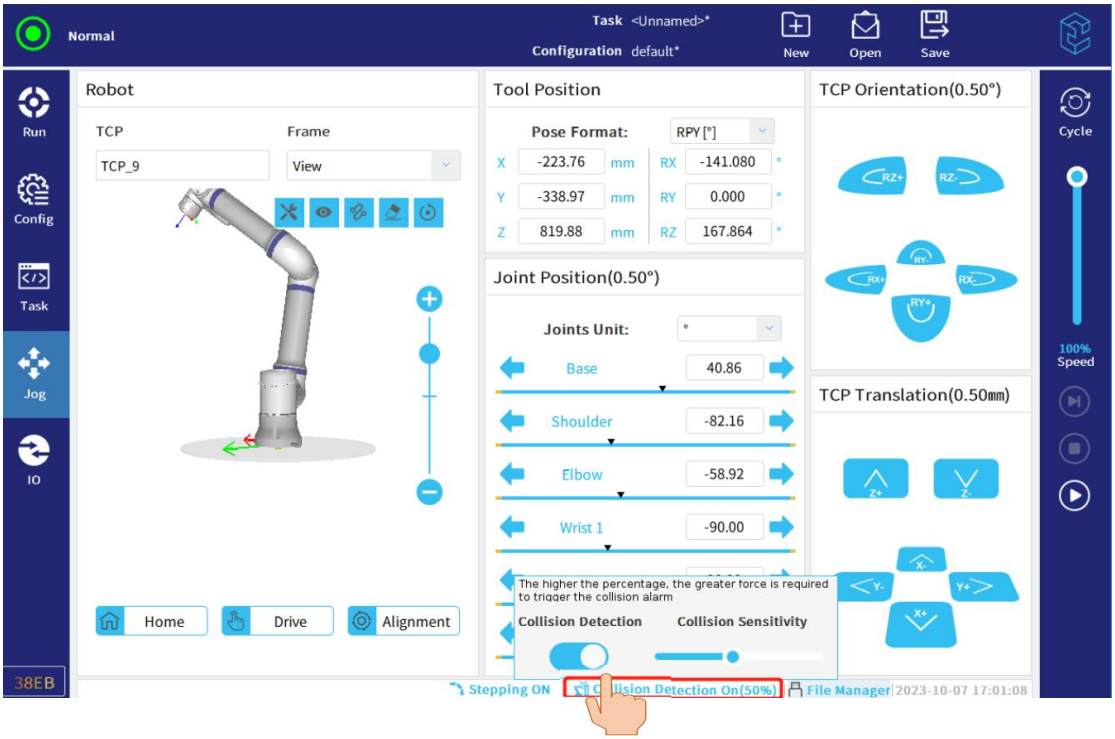


Figure 5-6

6 Task Programming

6.1 Create & Save a Task

The ”Task” tab includes the information about the programming (also called ”task”) of the CS series robot. The task files are with a suffix of task.

To create a new task, click **New -> Task**. In the current version, the task uses the currently opened configuration file when a new task is created.

Figure 6-1

To view an existing task, click **Open -> Task**. Select one of the task and click the “Open” button.

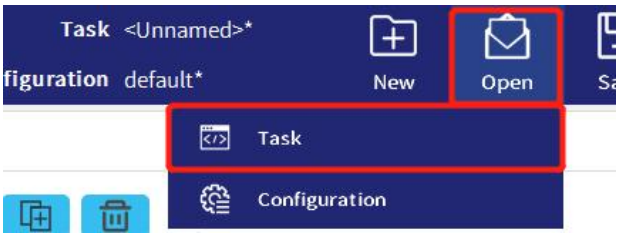


Figure 6-2

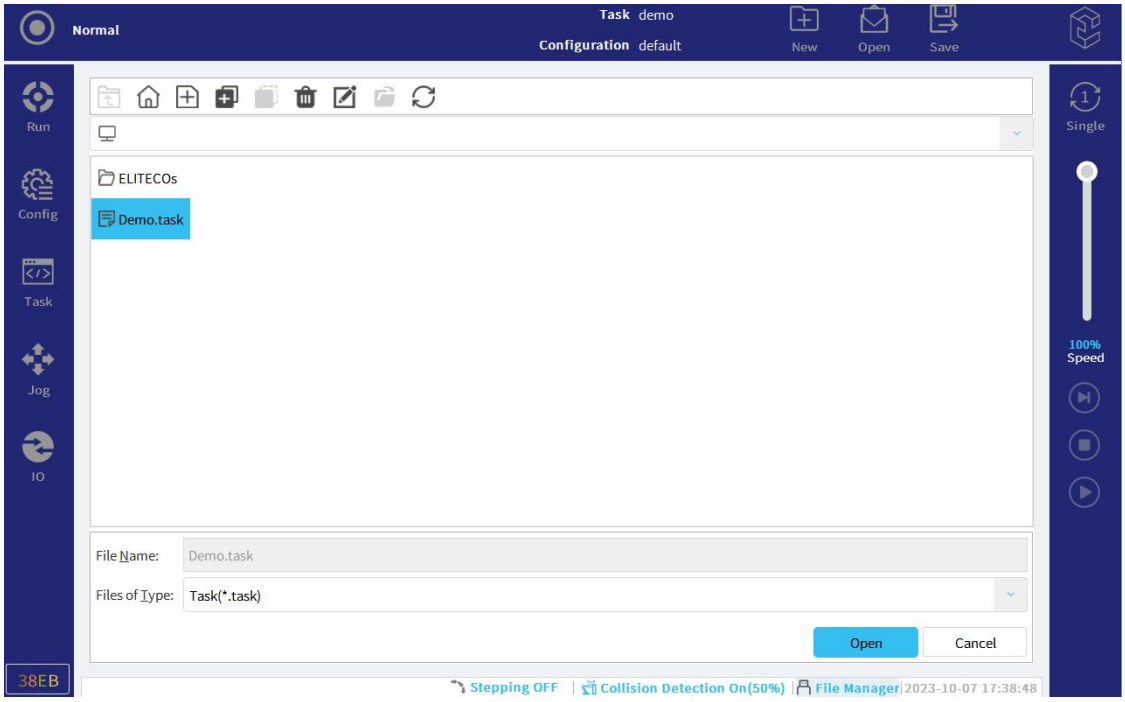


Figure 6-3

After opening a task, the configuration file used will be opened automatically. If it is not opened, the user can open it manually.

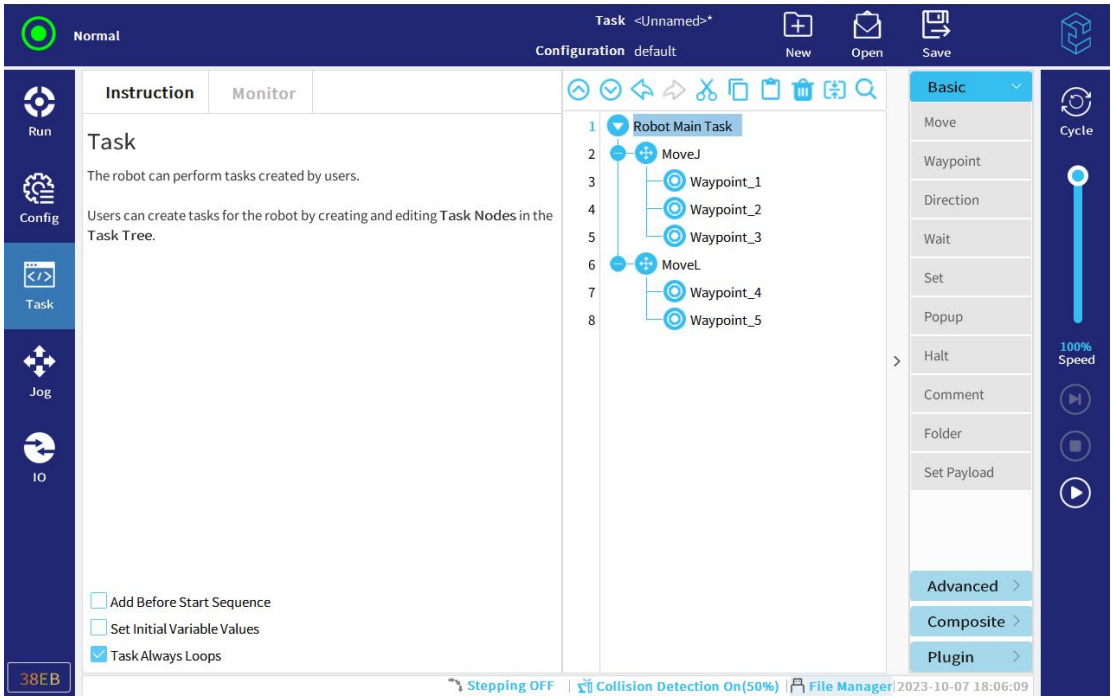


Figure 6-4

The configuration file includes the information in the “Config” tab. If the user expects to create a new configuration file, click **New -> Configuration**. At the time, the system will prompt the user whether to change the binding between the task and the configuration. If the previous configuration is still in use, please click “Not updated”. If it is ready to use a new configuration, please click “Update task”.

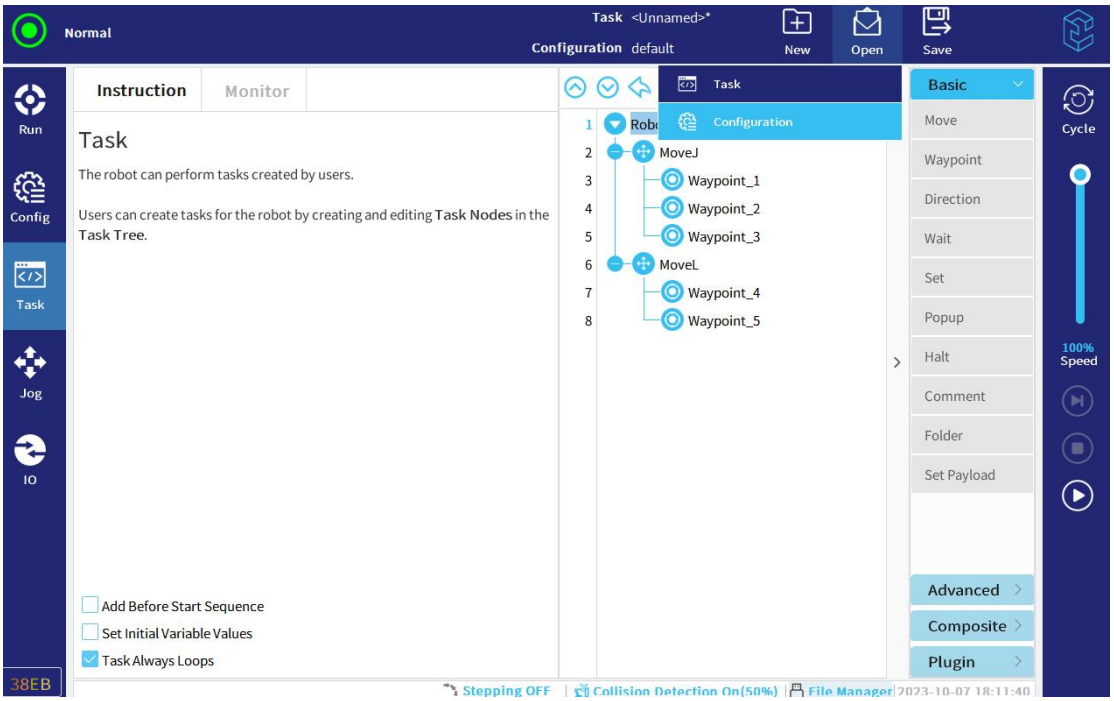


Figure 6-5

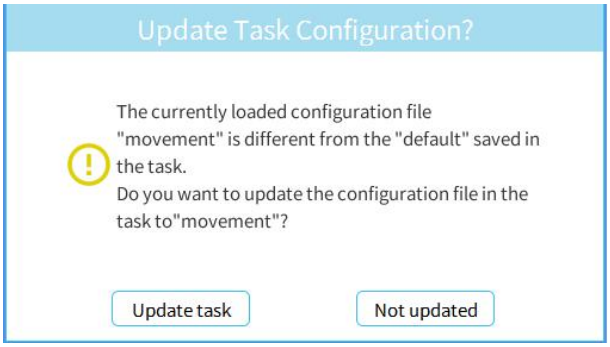


Figure 6-6

The dialog will also pop up when opening the existing configuration file. In addition, the robot will automatically power off when creating and opening a configuration.

Figure 6-7

Click “Save” to directly save the task and the configuration or click “Save -> Save as” to save them as the new ones. Note that the original configuration will be used if the configuration name is changed. Please re-open the configuration whose name is changed to make the adjustment. It is suggested to use the default name when naming the configuration file.

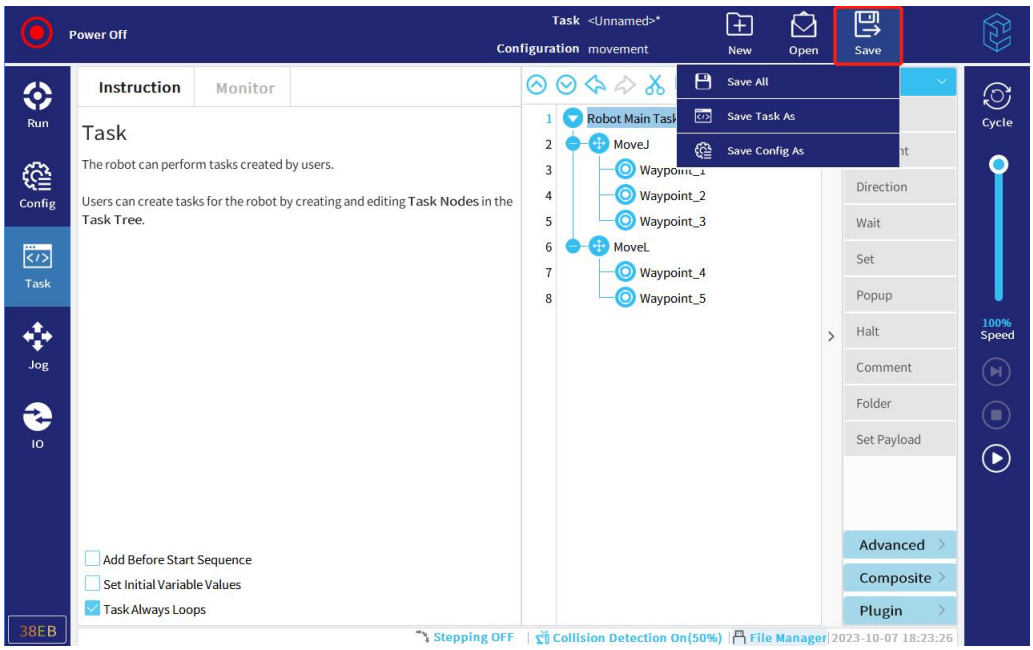


Figure 6-8

6.2 Task Tree

The user can edit the programs in the “Task” tab.

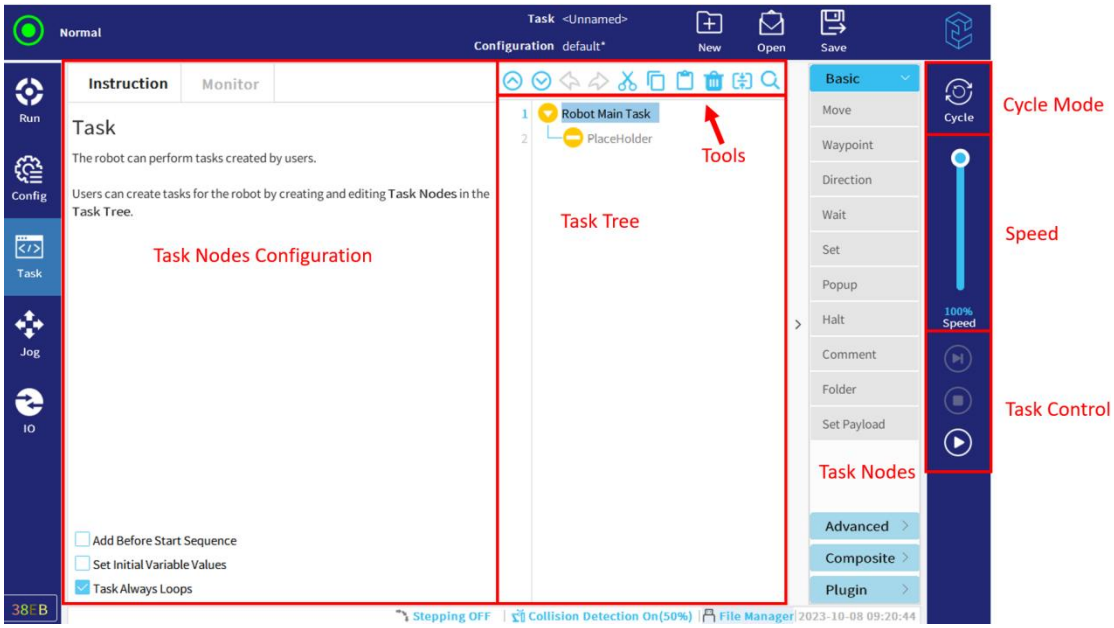


Figure 6-9

The tasks are carried out from top to bottom via the tree structure. Each branch of the task tree is called a “node”. Click a command in the right command list and it will be inserted below the selected node. After selecting a node, the user can edit the parameters in the left part of the interface.

Example:

Create a new task and enter the “Task” interface. The “Robot Main Task” line is selected. Click “Move” in the right basic command list and the MoveJ command will appear in the line 2, accompanied by “Waypoint_1” in the line 3.

When selecting “MoveJ” in the line 2, it allows the user to set the movement type, joint speed, joint acceleration and others in the left part of the interface. When selecting “Waypoint_1” in the line 3, the user can teach the point or set the required speed and time to move the robot to the waypoint.

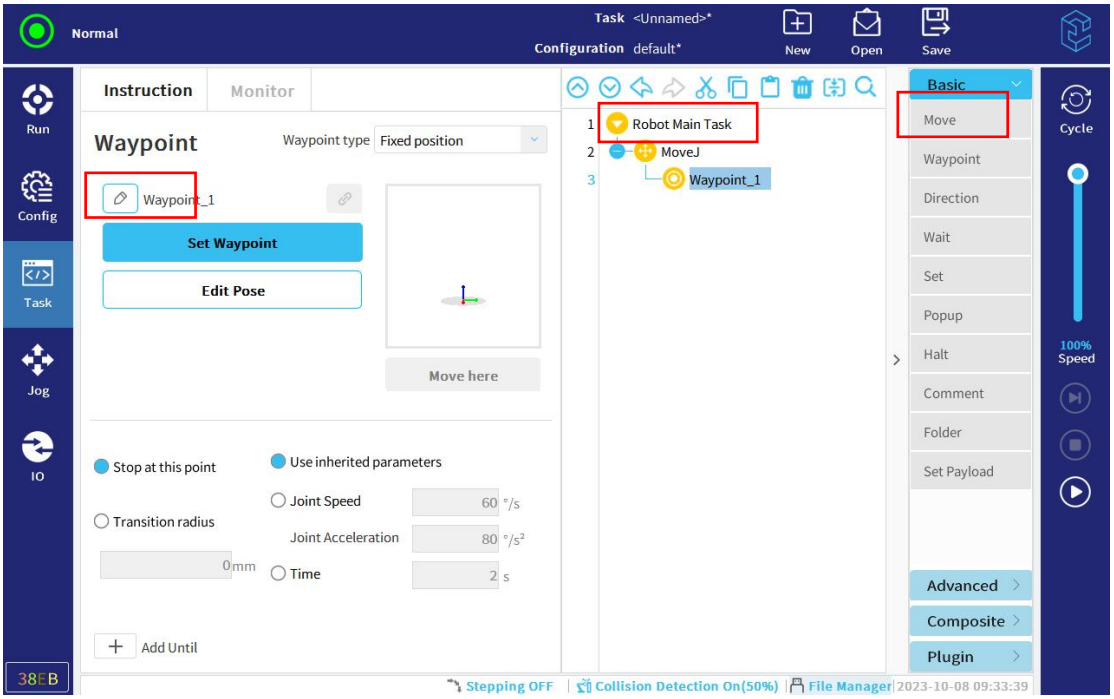


Figure 6-10

The task will not run until all nodes turn from yellow to blue.

6.3 Run a Task

There are 2 modes of operation, i.e. single and cyclic mode. To switch the mode, click 

or 

.

To adjust the speed percentage, please pull the slider. All speed will be subject to the changes of the speed percentage, e.g. the speed of the robot movement when pressing the button in the ”Jog” interface, the speed of ”auto move”.

Click  to start running a task

Click  to stop the currently running task. To run a task that has been stopped, please start again from the beginning.

Click  to pause the currently running task.

To continue a task that has been paused, click  .

To step through a task that has been paused, click  .

6.4 Task Editing Tools

The editing tools in the toolbar are as follows.

 Move the selected node up and down

 Undo and redo the modified commands

 Cut, copy and paste the node

 Delete the selected node

 Compress the selected node, the compressed node will not run. 选中的压缩节点在运行时会被忽略。

 Search the node in the search box, the user can enter the node name to search.

6.5 Task Frame Commands

Some nodes are designed to constract the task frame. A good task frame can reduce the programming and improve the process efficiency of maintenance. The nodes are pre-start sequence, file folder, subtask, thread, event and switch.

6.5.1 Main Task & Add Before Start Sequence

The user can select ”Add Before Start Sequence” in the “Robot Main Task” node. The node under "BeforeStart" will be executed earlier than the robot's main task and will only be executed once. The node is used to initialize the program. Note that only the robot main task loops.

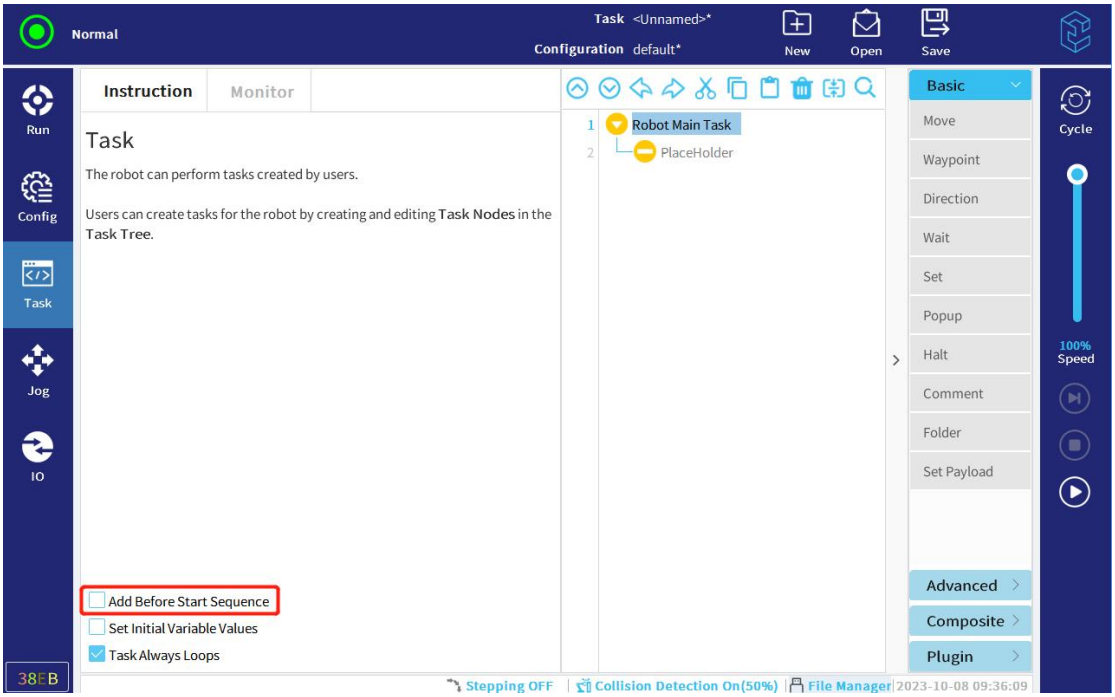


Figure 6-11

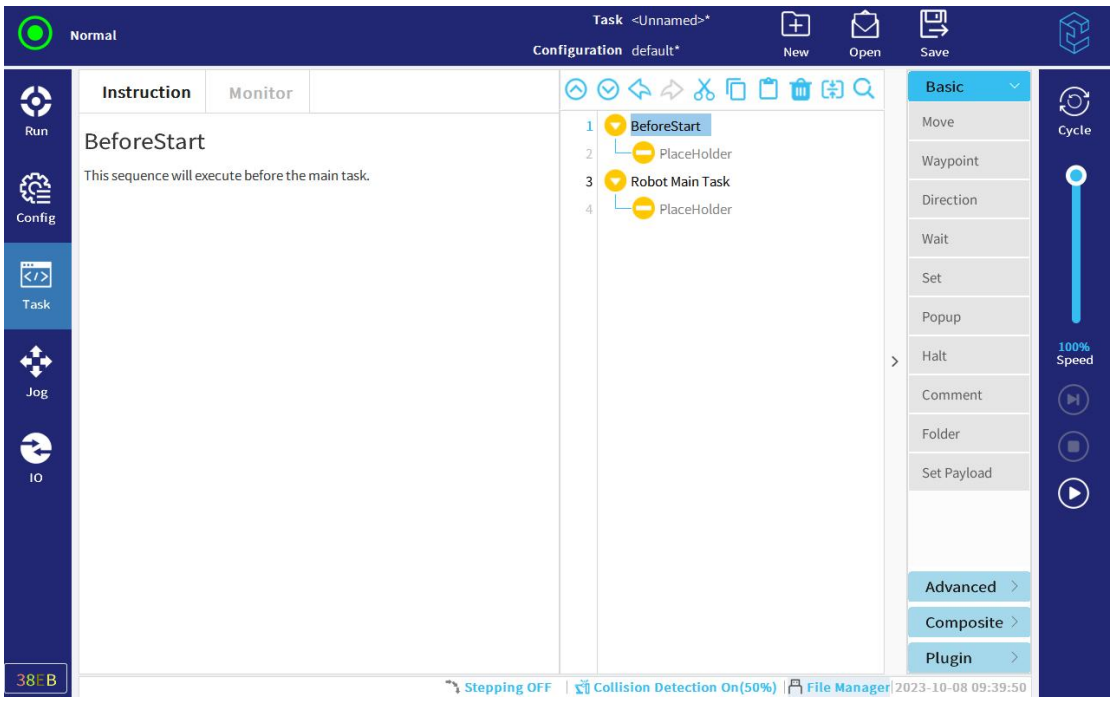


Figure 6-12

6.5.2 File Folder

To collapse or expand the subnodes, click the minus or plus sign. The ”File folder” node can also fold up the subnodes. Classification of the nodes like this improves the readability of the tasks.

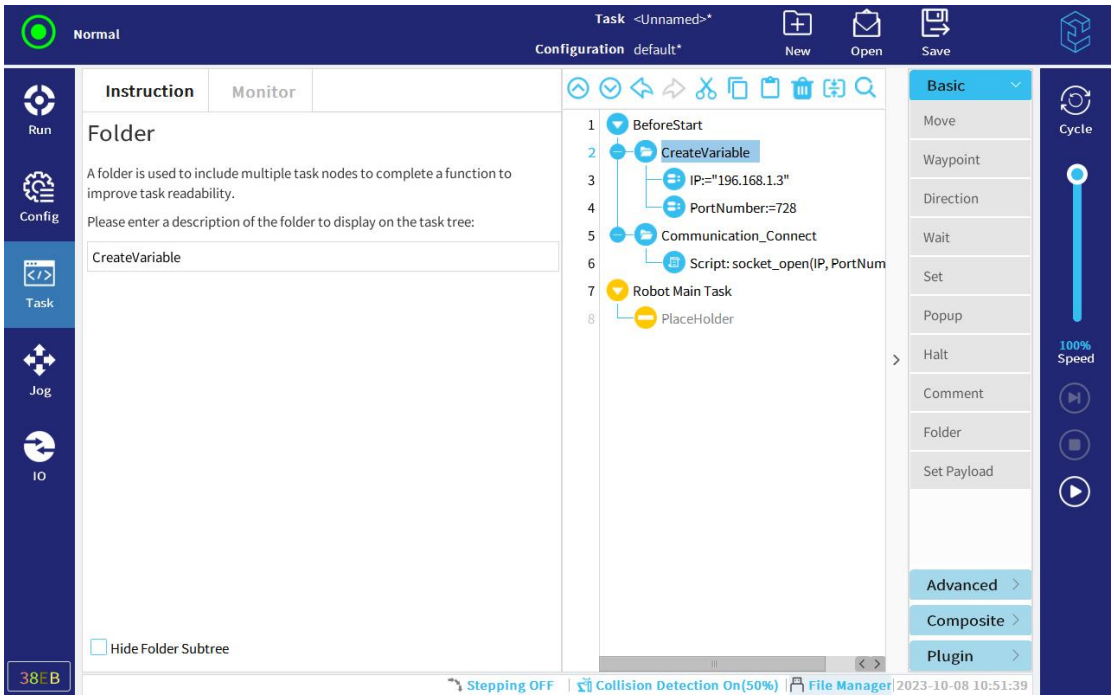


Figure 6-13

6.5.3 Subtask

Select the “Robot Main Task” subnode, click **Advanced -> Subtask** and a “Call” node will be inserted in the task tree. Then the user can call the subtasks by making a selection from the dropdown list in the left. If there isn’t a subtask, please create one by selecting “new” from the left dropdown list.

The user can also create a subtask by simply selecting the “Robot Main Task” node and clicking **Advanced -> Subtask**.

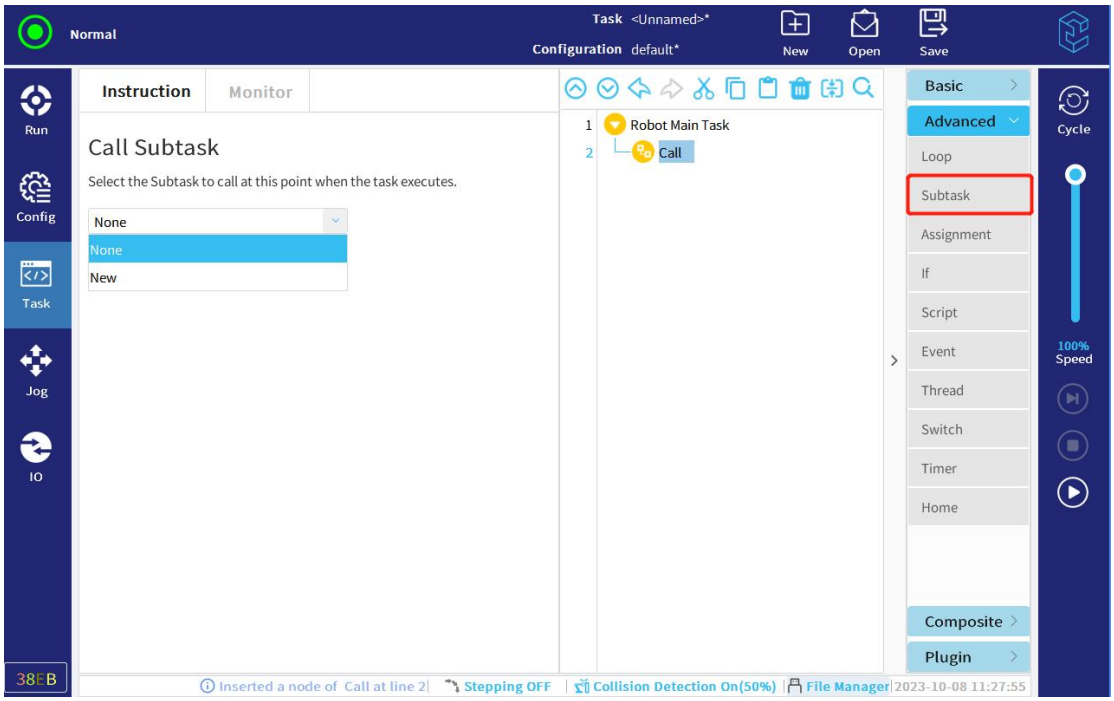


Figure 6-14

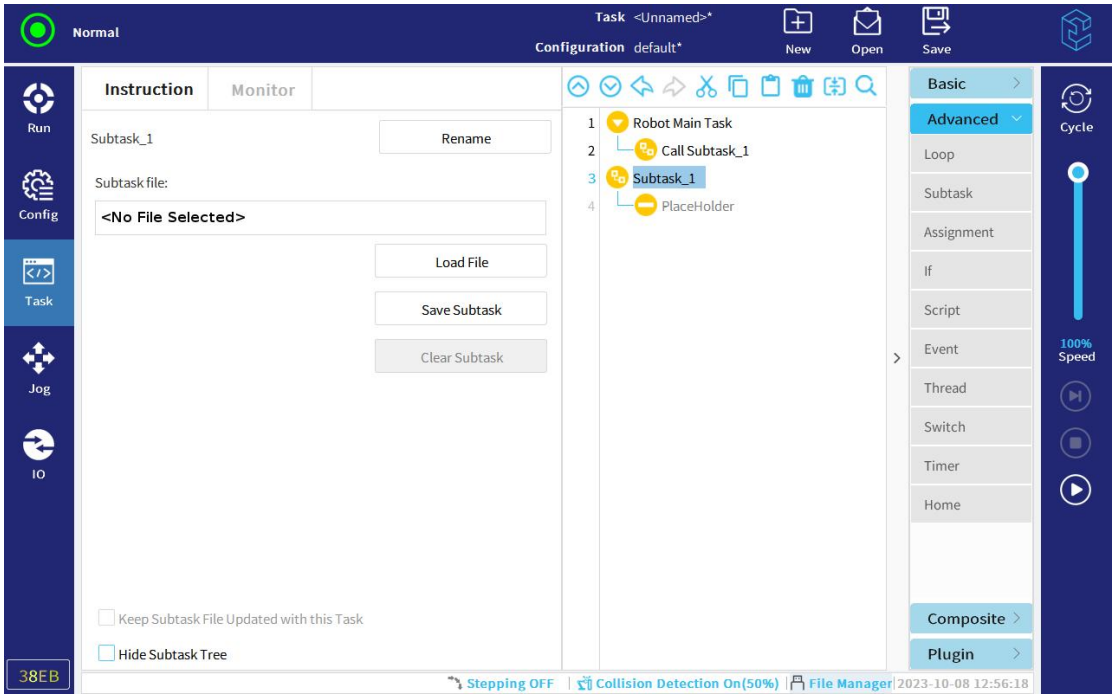


Figure 6-15

The user can directly edit the command(s) in the subnode. To save the edited subnode as a task file, click the “Save Subtask” button in the left. Click the “Load File” button in the left and the command(s) in the existing task file will be copied and pasted in the subtask node. Note that the configuration of the loaded subtask must be same as that of the current task. Otherwise, it will lead to a data conflict.

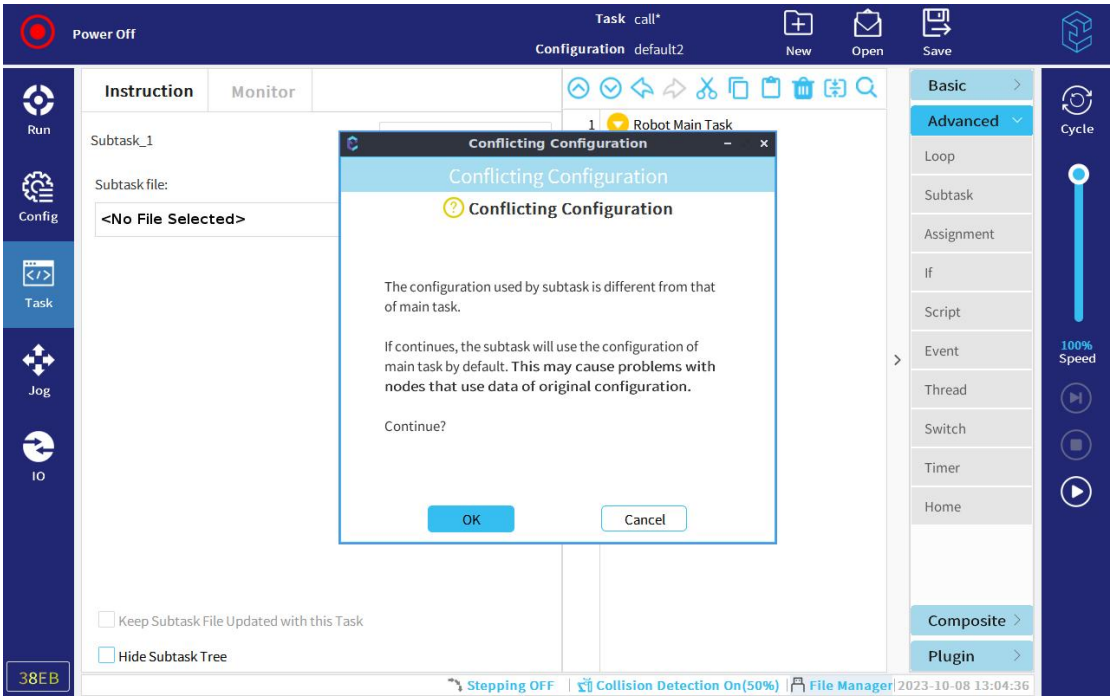


Figure 6-16

It is also possible to rename the subtask, thus improving the readability of the task.

When the contents of the task are repetitive, calling a subtask cuts down the programming work.

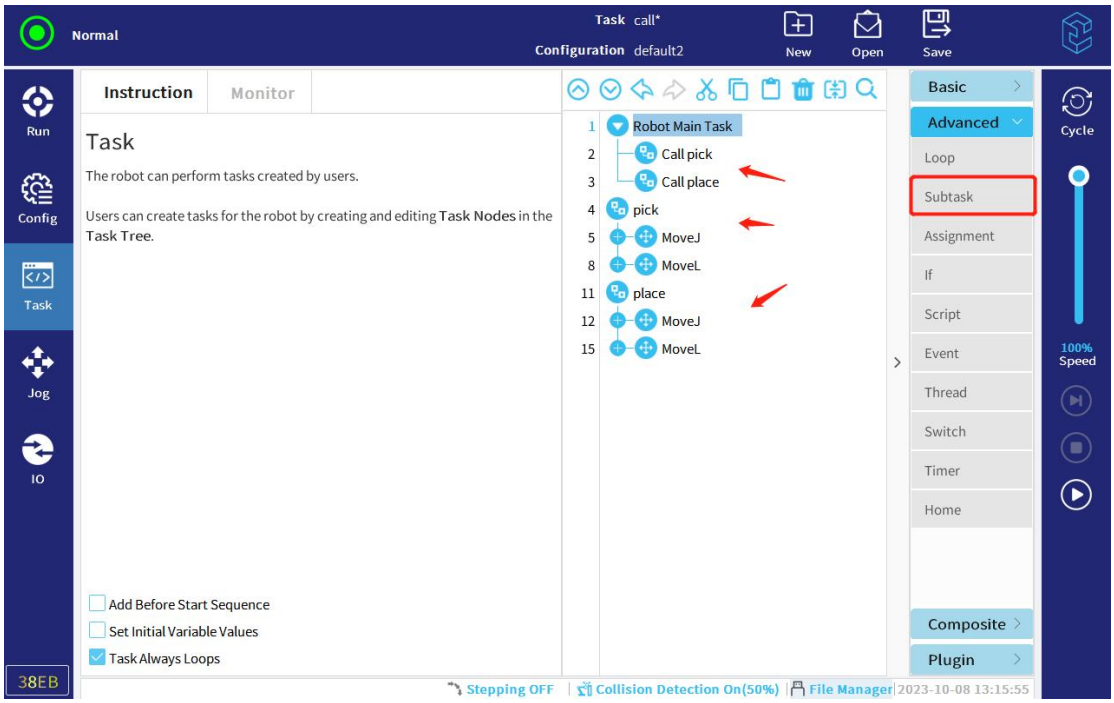


Figure 6-17

6.5.4 Thread

Select the “Robot Main Task” node, click **Advanced -> Thread** and the thread node will be inserted in the task tree. The thread, as a background task, ranks on the same level as the main task. It is usually used to monitor the IO and variables. After inserting the thread node, it is impossible to execute the motion instructions. Generally, it will simultaneously keep watch on the processes or perform some calculations while the main task is running. For instance, use the camera to calculate the position of the workpieces.

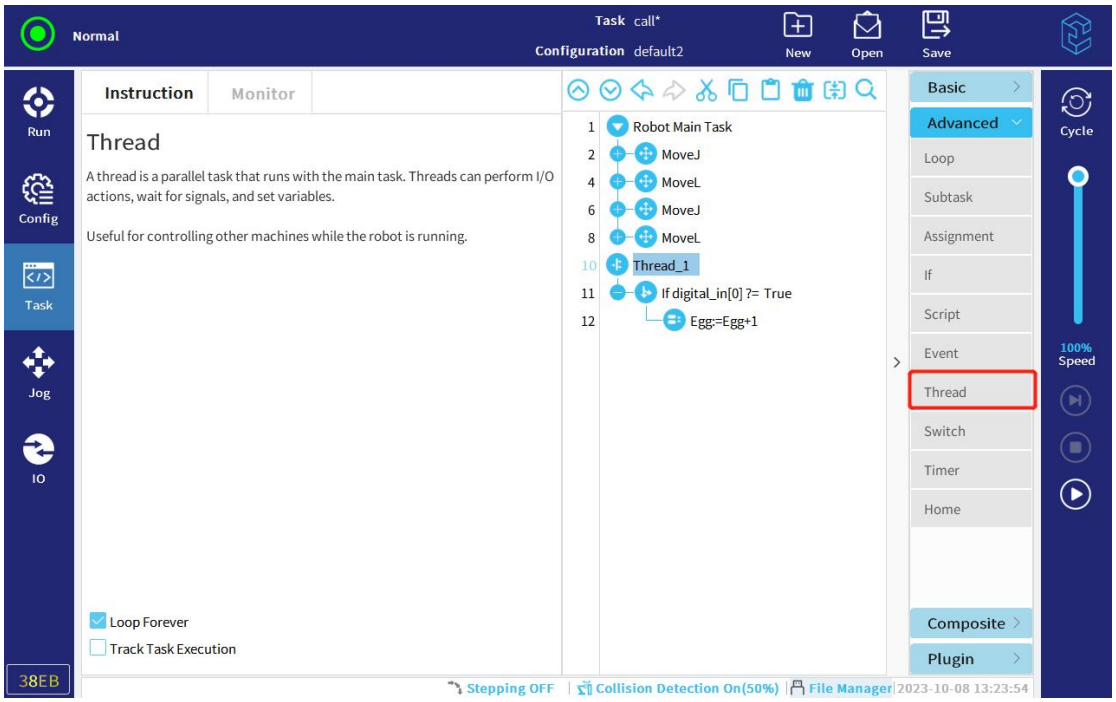


Figure 6-18

7 Motion Instructions

7.1 Movement Type

The motion instruction controls the movement of the robot through the basic waypoint. Waypoints must be placed under the motion command. The motion instruction defines the type of movement of the robot between waypoints. The motion instruction specifically refer to MoveJ, MoveL and MoveP. The user can add some brief descriptions about the node, set TCP, frame, speed and acceleration.

MoveJ: Joint movement, which controls the joints to complete the movement of the robot. This kind of movement can ensure that the robot arrives at the target waypoint in a fastest manner and avoid the singularity zone. The trajectory of the movement will not be confirmed until the robot is commissioning once. Please avoid the collision as much as possible.

When using MoveJ, the user can set the default joint speed and joint acceleration of the associated waypoint. The unit of the former is ˚/s and that of the latter is ˚/s^2.

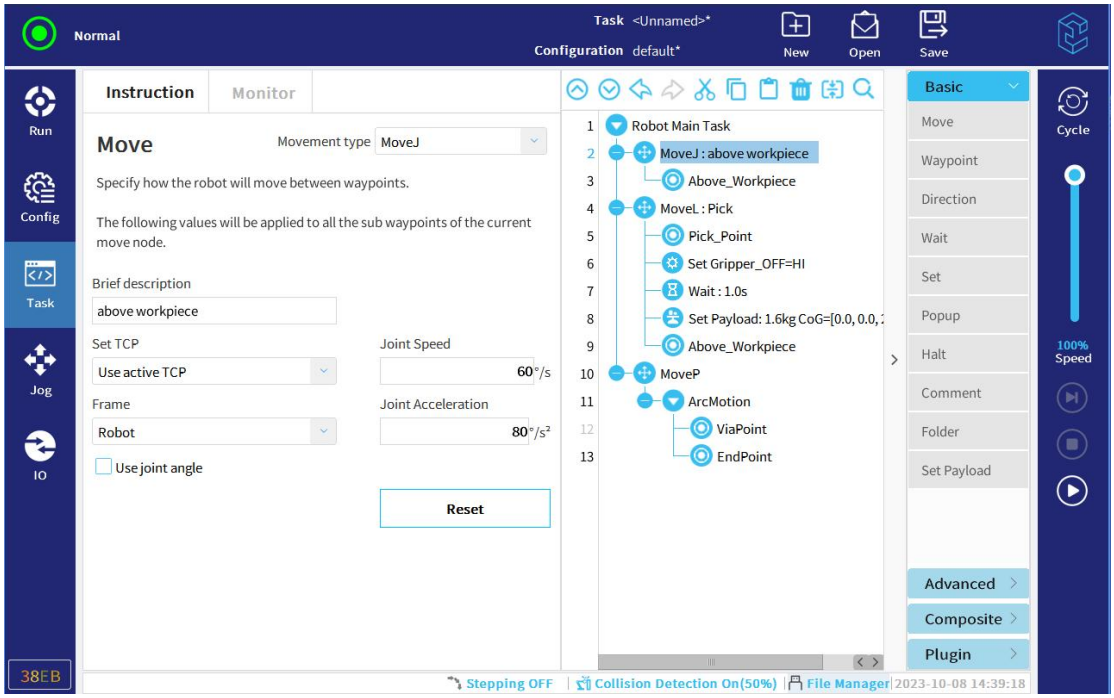


Figure 7-1

MoveL: Linear movement, which makes the tool center point (TCP) move in a straight line between the waypoints. It allows the user to configure the waypoint or the direction as the subnode. Note that it may not avoid the singularity zone.

When using MoveL, the user can set the tool speed and tool acceleration of the associated waypoint. The unit of the former is mm/s and that of the latter is mm/s^2.

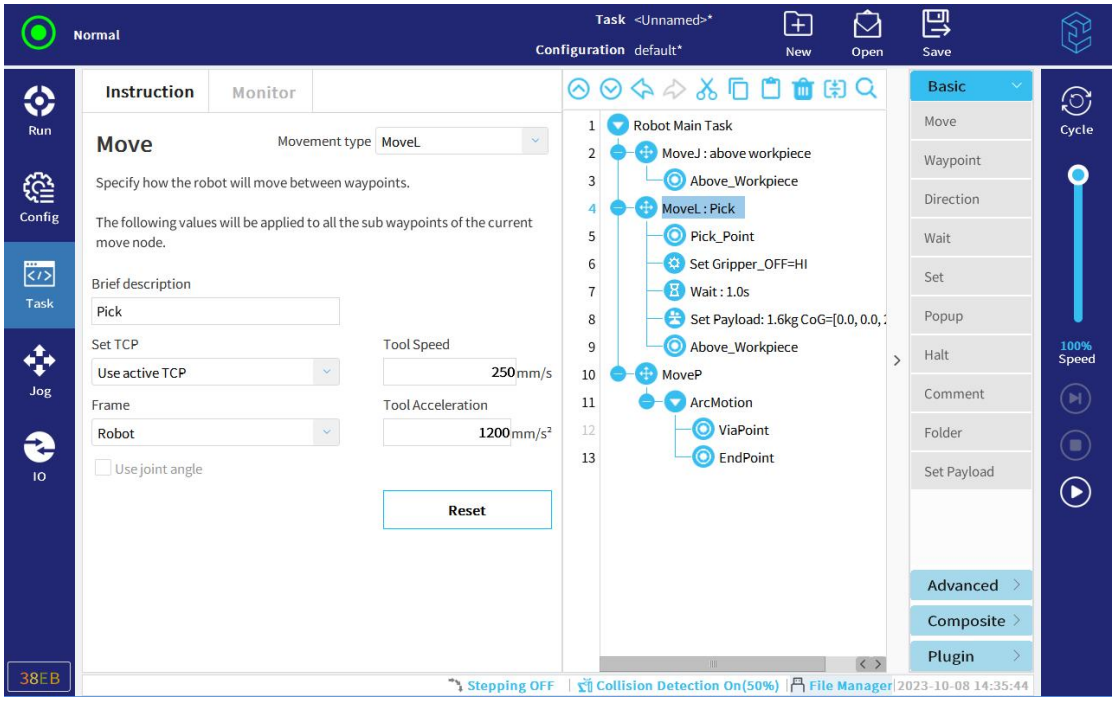


Figure 7-2

MoveP: Craft movement, which moves the robot along a straight line or allows the user to add the circular movement node. The arc path is complete by moving from a starting point (the current point), to a passing point (ViaPoint) and arriving at an ending point (EndPoint).

When using MoveP, the user can set the tool speed and tool acceleration of the associated waypoint. The unit of the former is mm/s and that of the latter is mm/s^2. The unit of the blend radius is mm. For more details about the blend radius, see the section 7.7 Blend Radius.

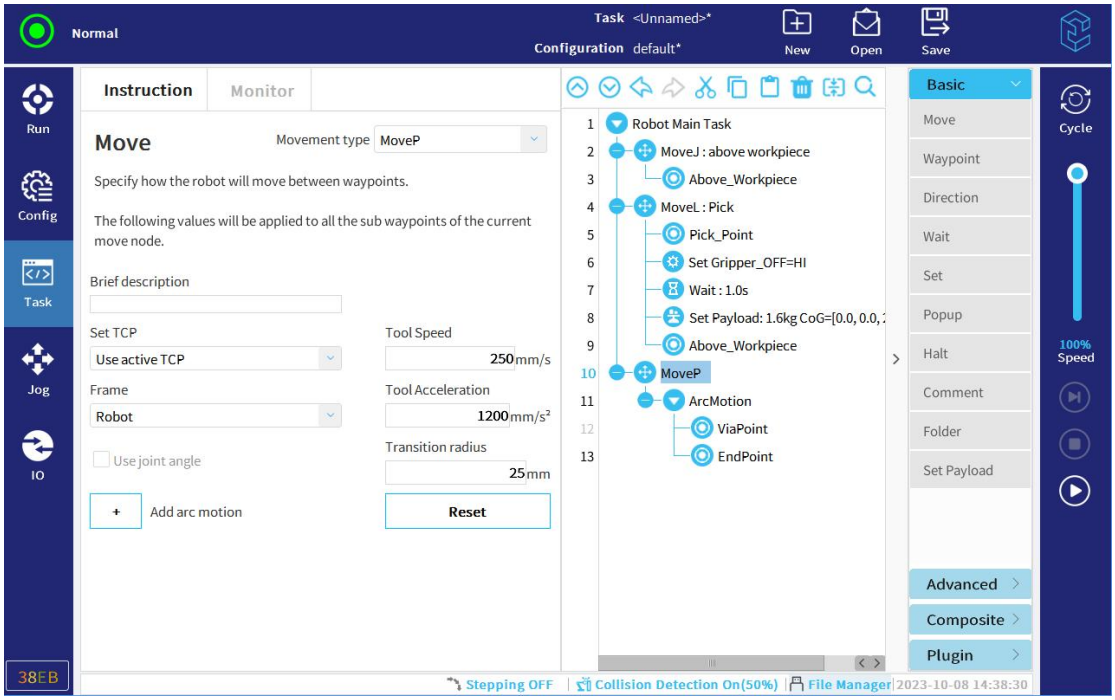


Figure 7-3

Note: The singularity zone refers to a designated zone where the robot cannot reach in a special pose in the working space but it arrives after changing the pose.

7.2 Circular Motion Node

The user can select the TCP direction relative to the arc in the circular movement node. When selecting “fixed”, the tangent angle between the robot pose and the corresponding point shall be consistent; when selecting “Unconstrained”, the robot will gradually adjust the pose as it approaches to the end point. Eventually, the robot pose and that at the end point shall be consistent.

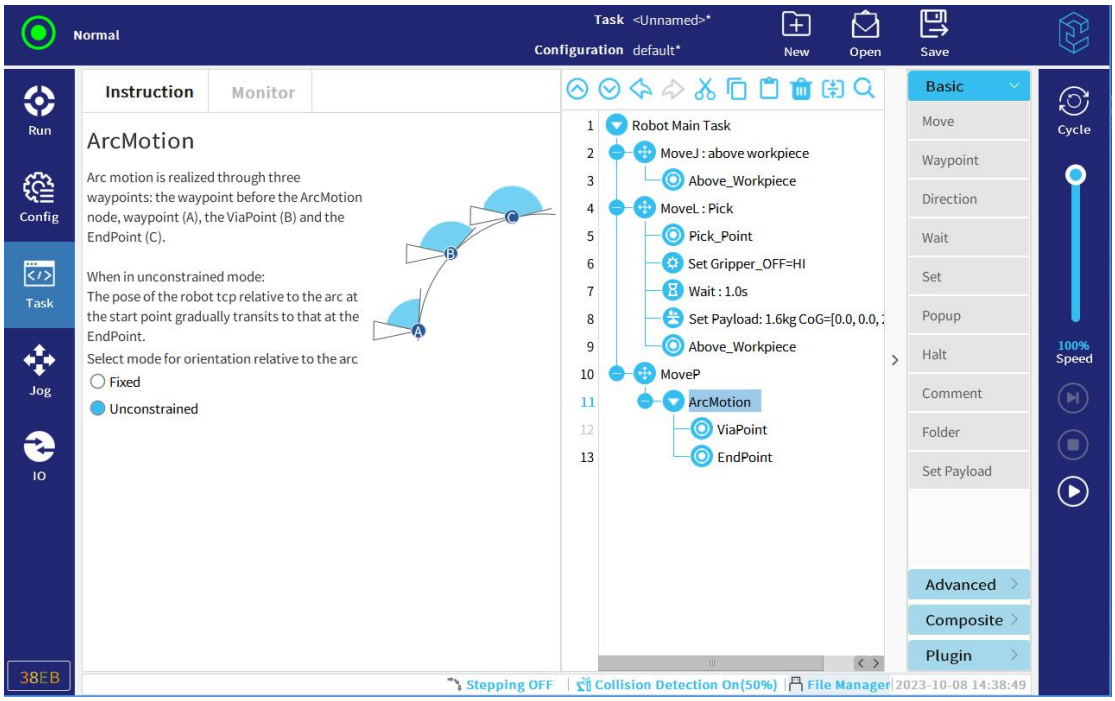


Figure 7-4

7.3 Set TCP in the Motion Mode

The TCP can be set by using MoveJ, MoveL and MoveP. The user can select one of the options, i.e. “Ignore active TCP”, “Use active TCP” and “Customized TCP”.

When selecting “Ignore active TCP”, the robot will move based on the center point of the flange.

When selecting “Use active TCP”, the robot will move based on the configuration of the currently activated TCP.

When selecting “Customized TCP”, the robot will move based on the configuration of the selected TCP.

When teaching the waypoints, the recorded coordinate data is mainly about the current TCP. The waypoint will be subject to the change of the TCP option.

Note: It is suggested to specify a required TCP in the motion instructions.

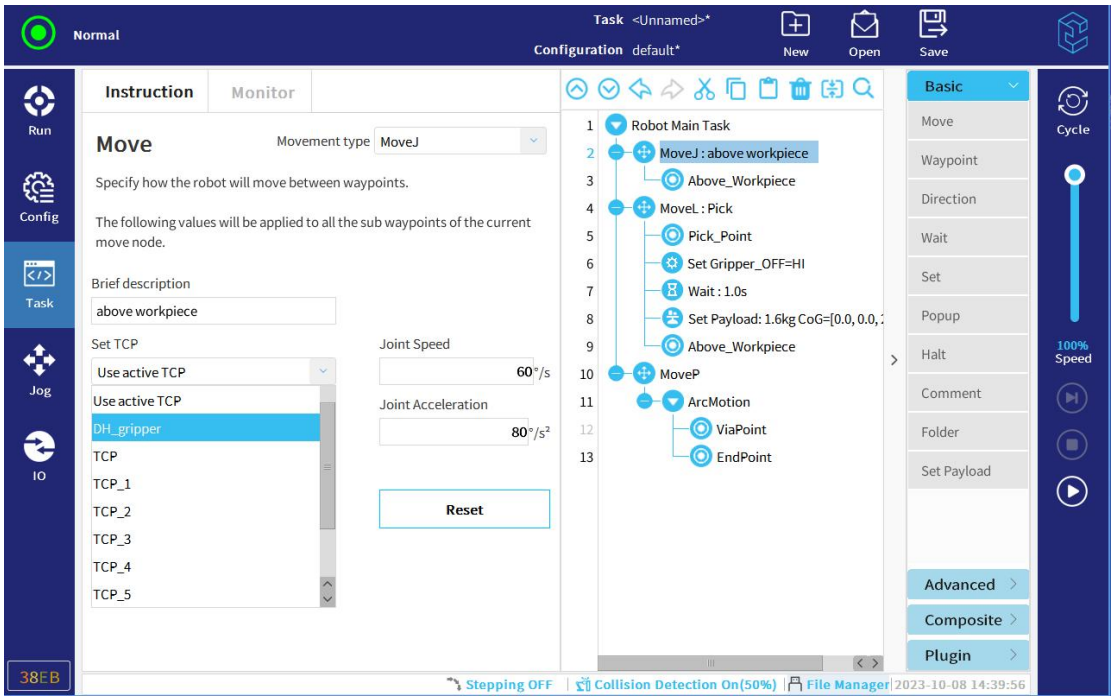


Figure 7-5

7.4 Select the Frame in the Motion Mode

The frame can be selected in case of using MoveJ, MoveL and MoveP. The user can select “Base” or “Custom frame”. When teaching the waypoint, the currently selected frame is the reference of the coordinate data recorded. The subnode incl. waypoint and direction will be subject to the change of the frame. This may lead to a collision.

Note: It is suggested to specify the frame in the motion instructions before teaching the point. If it is necessary to change the frame, we suggest rechecking the position of the waypoint or reteach the waypoint.

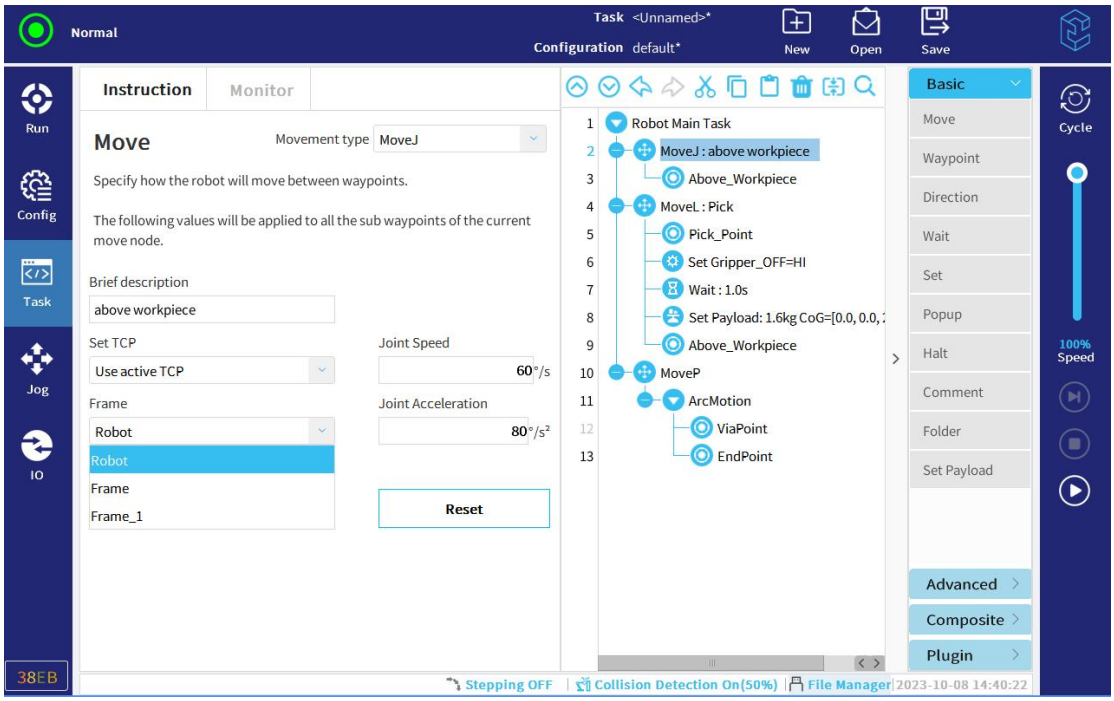


Figure 7-6

7.5 Use Joint Angle

When using MoveJ, the user can select the checkbox ”Use joint angle”. After that, the option ”Tool” and ”Frame” will be disabled and the waypoints under the movement type will inherit the parameter ”Use joint angle” when running. If checked, when the waypoint under the movement type moves, the pose will no longer be considered. The joint angle at the time of recording will be taken as the motion target.

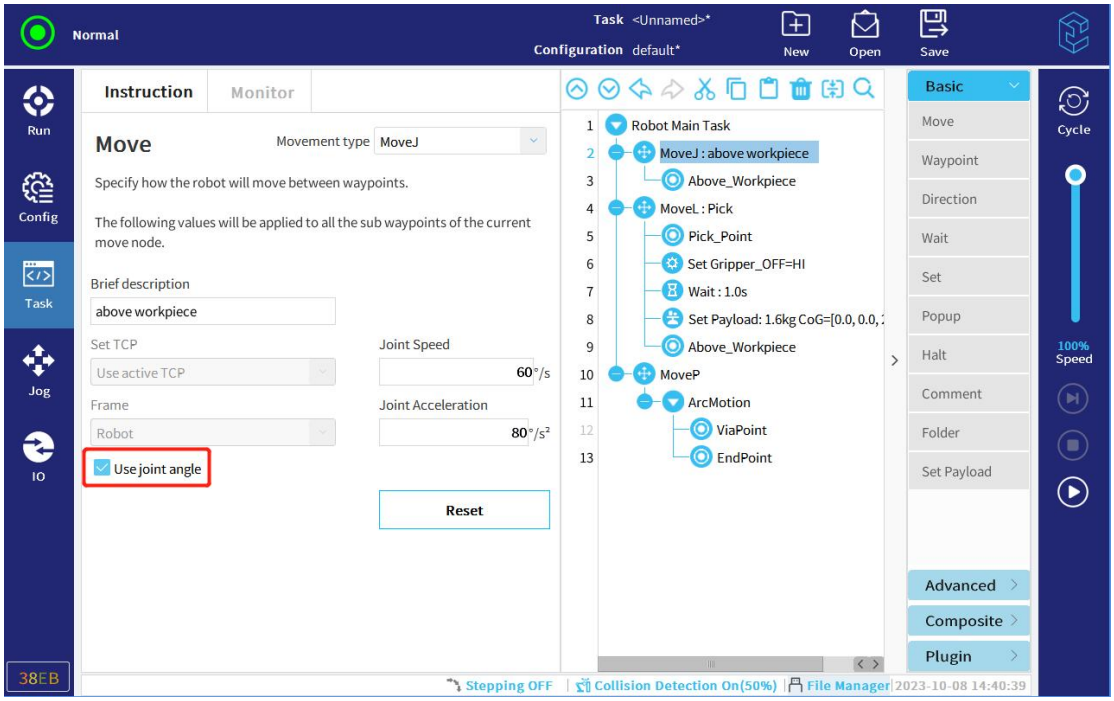


Figure 7-7

7.6 Waypoint

The waypoints are the core element of the robot task and are used to record the target points where the robot reaches.

There are three types of waypoints, i.e. fixed position, relative position and variable position.

Fixed position waypoint: after teaching, the robot arm moves in line with the taught point position data. The user can teach the points by clicking ”Set Waypoint” or ”Edit Pose”.

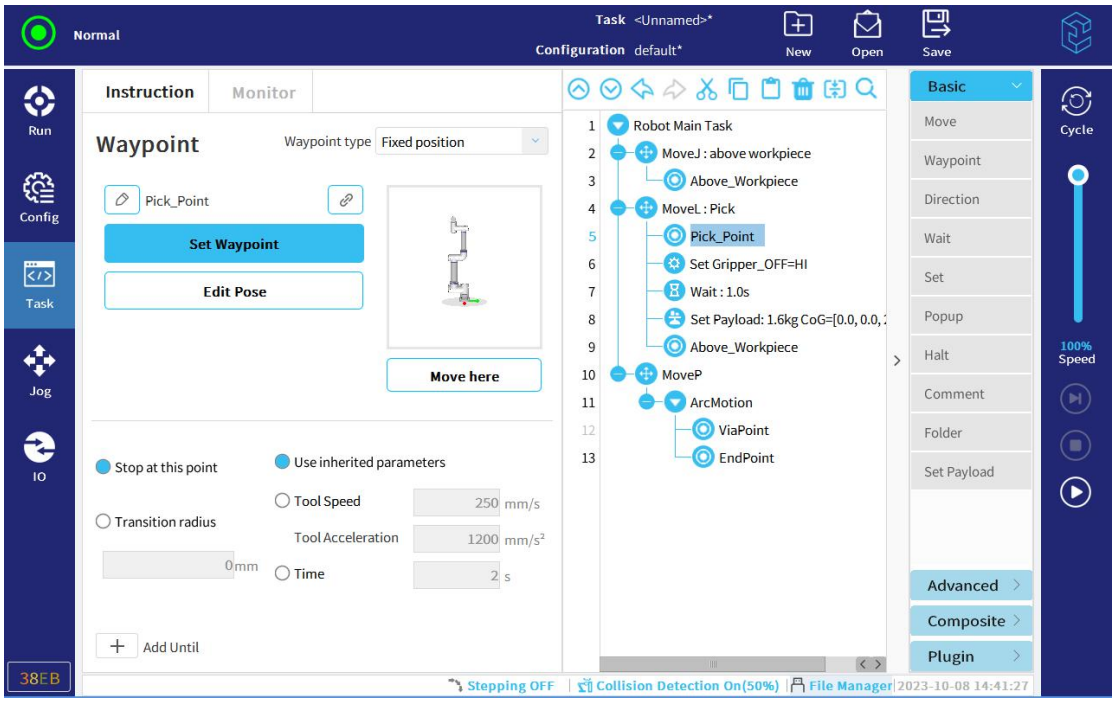


Figure 7-8

When clicking “Set Waypoint”, the robot must be moved to the position after the setting if the user adjusts the waypoint by modifying the position parameters directly. When clicking “Edit pose”, there is no need to move the robot to the new position after the position parameters are modified.

Click  and the user can link the existing waypoint. The linked waypoint will synchronize the data. Any modifications on the data will lead to the updates of all nodes that calls the waypoint. Click  and the user can cancel the link.

Relative position waypoint: after teaching two waypoints and calculating the pose difference between the two waypoints, the robot will move in accordance with the current pose and the calculated pose difference.

Variable position waypoint: variable waypoint refers to the location of the waypoint given by the variable. For the variable waypoint in the MoveJ instruction, it will run as the joint position. For the variable waypoint in the MoveL and MoveP instructions, it will run as the pose. Therefore, please make sure that the variable data matches the motion type. Otherwise, the unexpected problems may occur in the movement.

7.7 Blend Radius

The blend radius can improve the cycle time and facilitate the smooth movement of the robot. After setting the blend radius, the robot will turn to the next target point before the arrival at the pre-set target point. At the end, the robot will bypass the pre-set target point. That meant, of course, the robot will not accurately reach the point. When setting the blend radius of a waypoint, it cannot overlap with the blend radius of the previous waypoint or the next waypoint.

There are three modes of the blend radius.

Stop (no blend radius): When the movement type is "MoveJ" or "MoveL", the waypoint parameter can be set to "Stop at this point". After the setting, the robot will stop at the waypoint first and then move to the next waypoint. When executing a task of transporting the workpieces, it is necessary to set the blend radius to ”Stop at this point” in the pick-up or placement waypoint.

Custom: It is possible to customize the blend radius for the waypoints in all movement types. After setting the blend radius, the robot will turn to the next waypoint before reaching the current waypoint and the movement will be smoother.

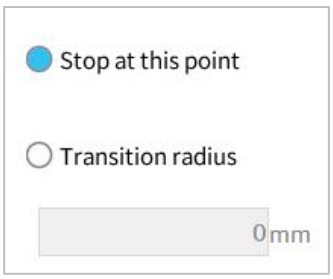


Figure 7-9

Inheritance: when the movement type is "MoveP", the waypoint parameter can be set to "Use inherited transition radius". In this case, the blend radius of the waypoint will be same as the configuration under the "MoveP" node.

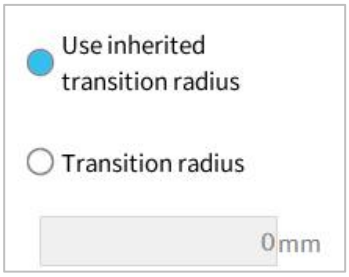


Figure 7-10

Note: It is suggested to use the blend radius to improve the 节拍 under the condition that the robot does not need to reach the waypoint very precisely. If it requires the robot to reach the waypoint accurately or the waypoint is the last one that the robot reaches, please set the blend radius to 0.

7.8 Speed & Acceleration

There are three modes of the waypoint speed.

Inheritance：After selecting “Use inherited parameters”, the speed and acceleration are the same as those set in the “MoveJ” or “MoveL” node.

Speed/Acceleration: The user can separately set the speed and acceleration that the robot moves to the waypoint. When the movement type is “MoveJ”, the unit of the speed is °/s and the unit of the acceleration is °/s^2. When the movement type is “MoveL” and “MoveP”, the units are mm/s and mm/s^2, respectively.

Time: When the movement type is “MoveJ” or “MoveL”, the user can set the waypoint parameter to “Time”, which refers to the time when the robot moves from the current point to the target point. The unit of the time is s.

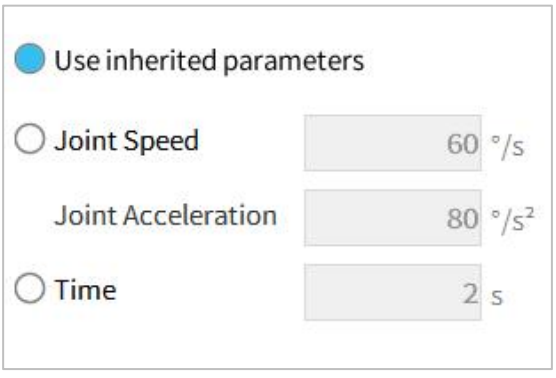


Figure 7-11

7.9 Add Until

In the task tree, the user can add the ”Until” node under the ”Waypoint” or ”Direction” node. When one of the ”Until” stop conditions is satisfied, the robot will stop moving towards the ”Waypoint” or the ”Direction” and the task will enter the next node. The user can add multiple stop conditions in each waypoint or direction.

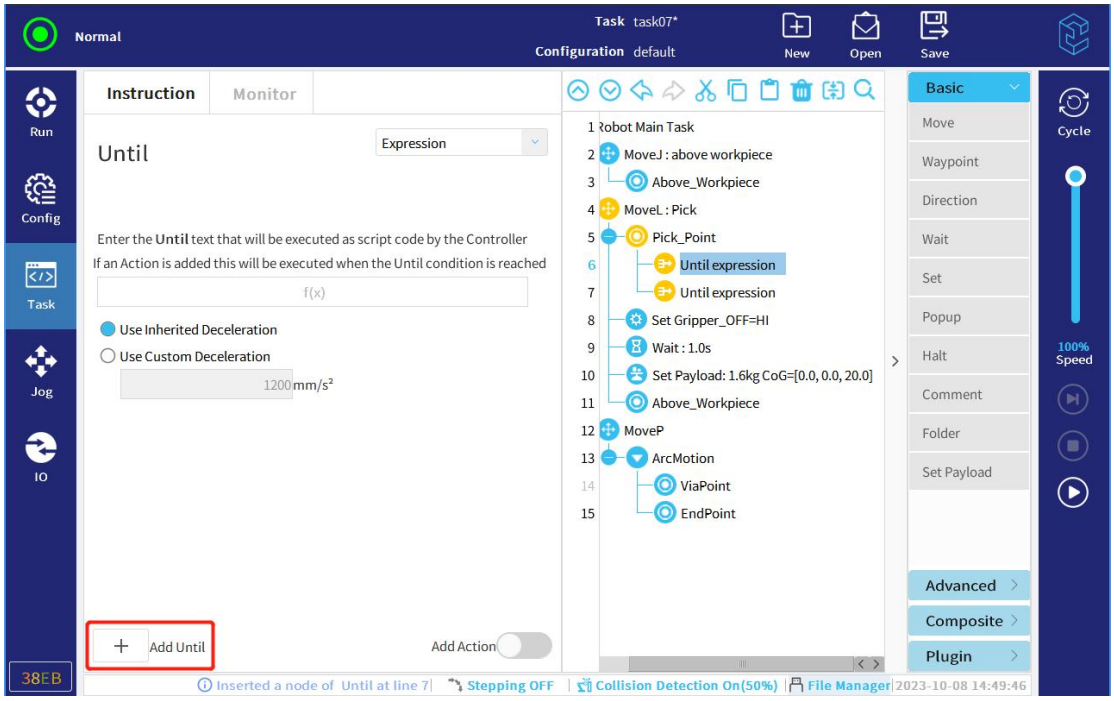


Figure 7-12

The stop conditions of the “Until” node are as follows: “Expression”, “Waypoint reached” and “IO input”.

When selecting “Expression”, the user can customize the stop conditions and adjust the deceleration.

If “User Inherited Deceleration” is checked, the deceleration of the waypoint will be same as the acceleration under the waypoint node.

If “Use Custom Deceleration” is checked, the user can enter the customized deceleration.

When selecting “Waypoint reached”, the task will enter the next node when the robot reaches the waypoint.

When selecting “IO input”, the user can designate the digital input, configurable input and analog input as the stop conditions.

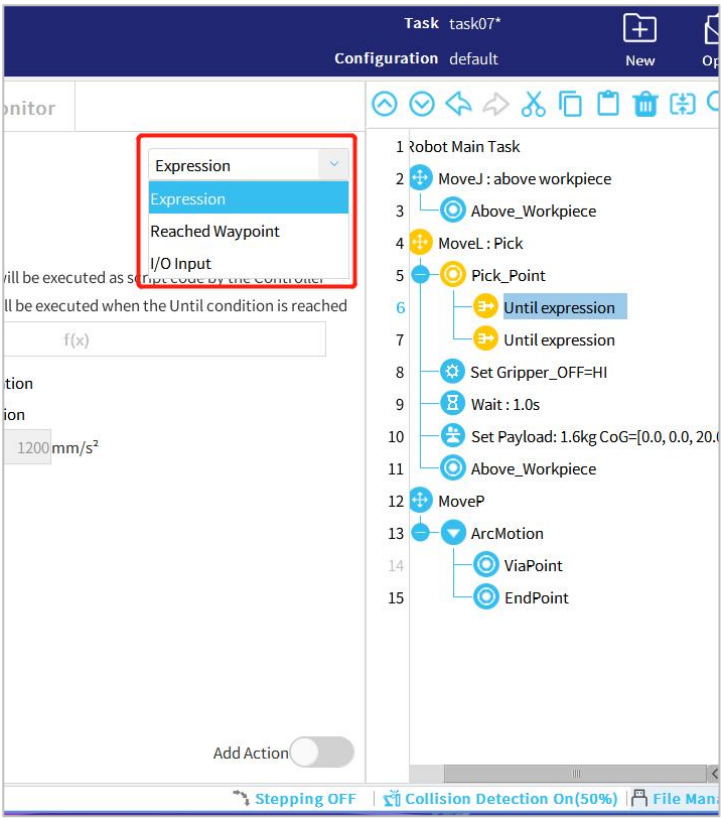


Figure 7-13

Since that the multiple “Until” nodes can be added under the ”Waypoint” node, a main node can be divided into several sub-nodes with the ”add action”. The following figure shows that the workpieces will not be taken in and out when the gripper works improperly. It avoids that a defective gripper collides with the workpieces.

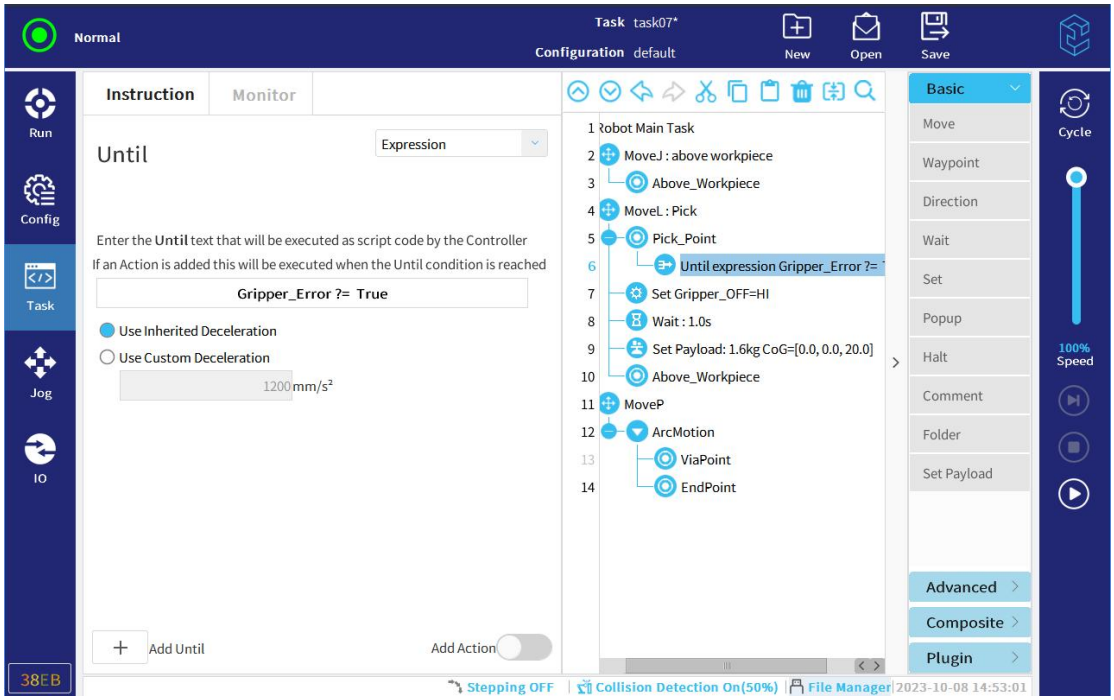


Figure 7-14

7.10 Direction

The ”Direction” command can only be placed under the MoveL node and a ”Until” node will be auto-generated. The command specifies that the robot moves linearly in a given direction until it is stoppd by the ”Until” condition.

The user can select an inherited frame or a customized one.

For the direction, the user can select one of them (X+, X-, Y+, Y-, Z+, Z-) from the dropdown list.

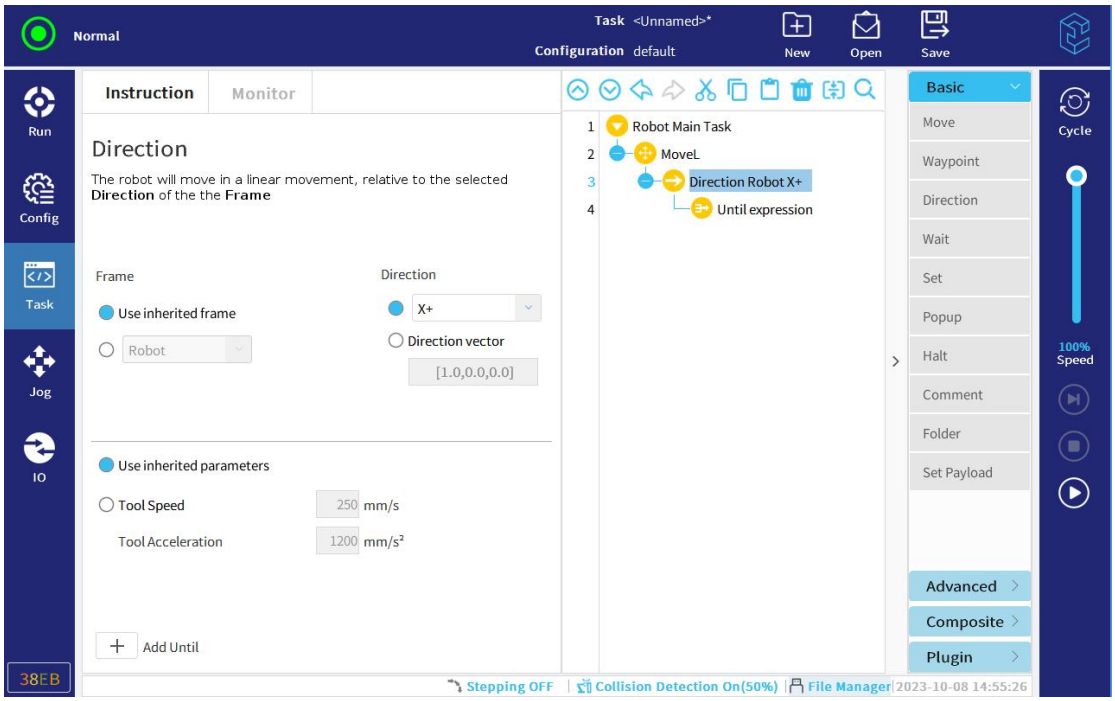


Figure 7-15

The user can also configure the direction by setting the direction vector. The vector is distributed in a proportion of x, y and z. The range of the value is [-1, 1]. The following figure shows that the robot moves towards the vector merge direction of x+ and z+ axis. The speed that the robot movs towards the x axis is twice as fast as that to the z axis.

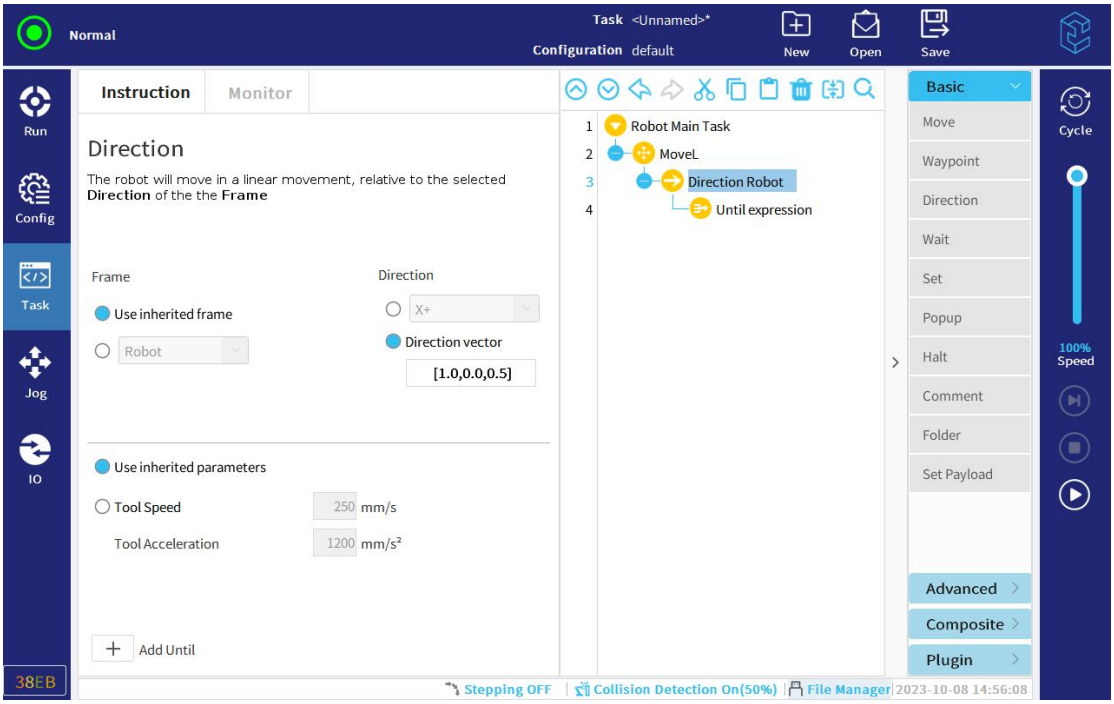


Figure 7-16

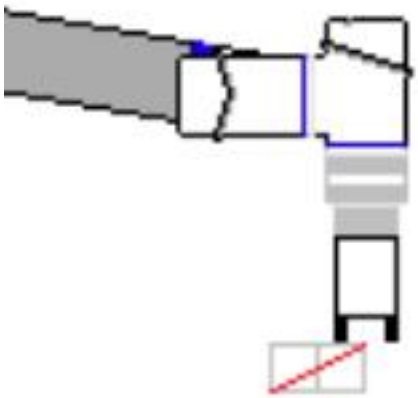


Figure 7-17

For the speed and acceleration, the user can select “Use inherited parameters” or set them separately.

The option “Waypoint reached” is not available in the “Until” node under the “Direction” node, but the user will find a new option of “Distance” in the dropdown list. The robot stops when it moves a specific distance.

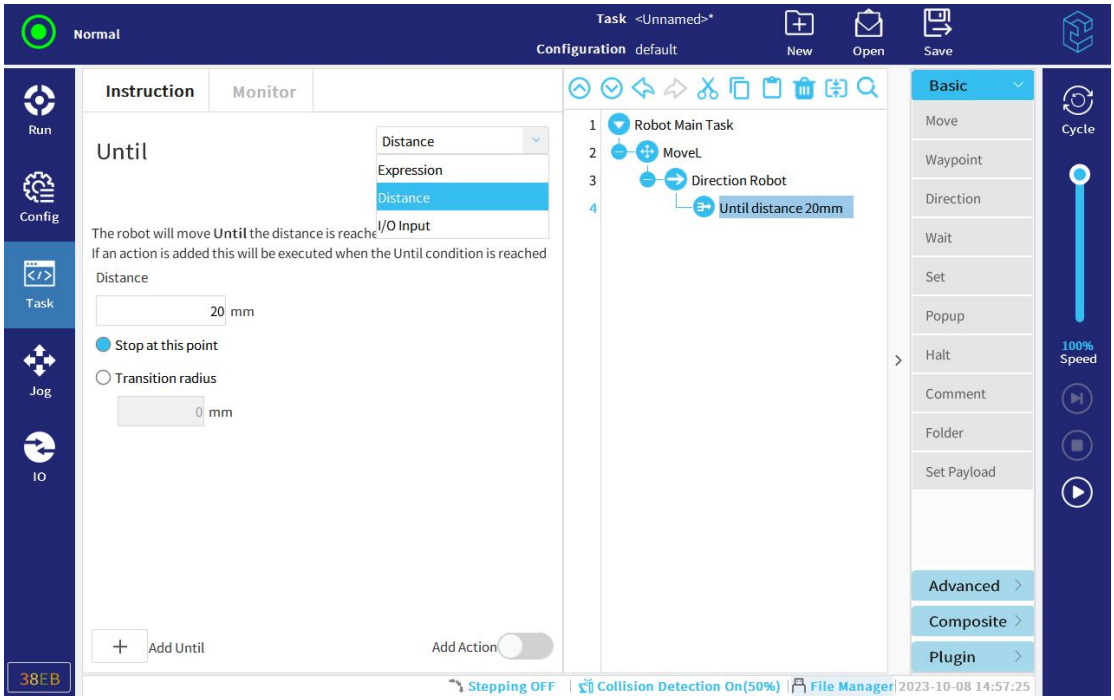


Figure 7-18

8 Variable

The variable can be used to save the values. The rule on the variables of the CS series robot are identical to that of Python. There are 5 sorts of the basic variables, namely, boolen, integer, float, string and list of 6 elements. The system can tell the variable type when assigning at the time of definition.

Boolen: the value is True or False

Integer: int, incl. negative integer, e.g. 5，0，-2

Float: number with a decimal point, incl. negative numbers, e.g. -3.14, 2.33, 0.00

String: characters composed of letters, numbers and underlines and enclosed in the double quotation marks, e.g. “5.28”, “Car”, “love4Cats”, “box_length”

List of 6 elements: each element can only be numbers and is enclosed in the square bracket. The list generally refers to the joint angle or the pose. When it indicates the joint angle, the 6 elements are the angles of the joint 1-6, respectively. The range is [-360°,360°]. When the angle is out of range, the exception may occur. When it indicates the pose, the first three elements are the values of the X, Y and Z. The unit is m and the maximum distance cannot be exceeded. The last three elements are the values of the Rx, Ry and Rz and indicate the rotation angles around the 3 coordinates. The unit is radian and the range is [-2π, 2π]. When the value is out of range, it may lead to an exception. For instance, the

values can be [280.33, -144.23, 54.23, 47.98, -12.03, 32.23], [2.334, -1.576, 0.245, 1.234, 0.453, -2.321].

8.1 Global Variable

To create global variables, click **Config -> General -> Global Variable**. The data will be saved in the configuration file and can be called by the task being ties up with the same configuration file. It supports for being retentive.

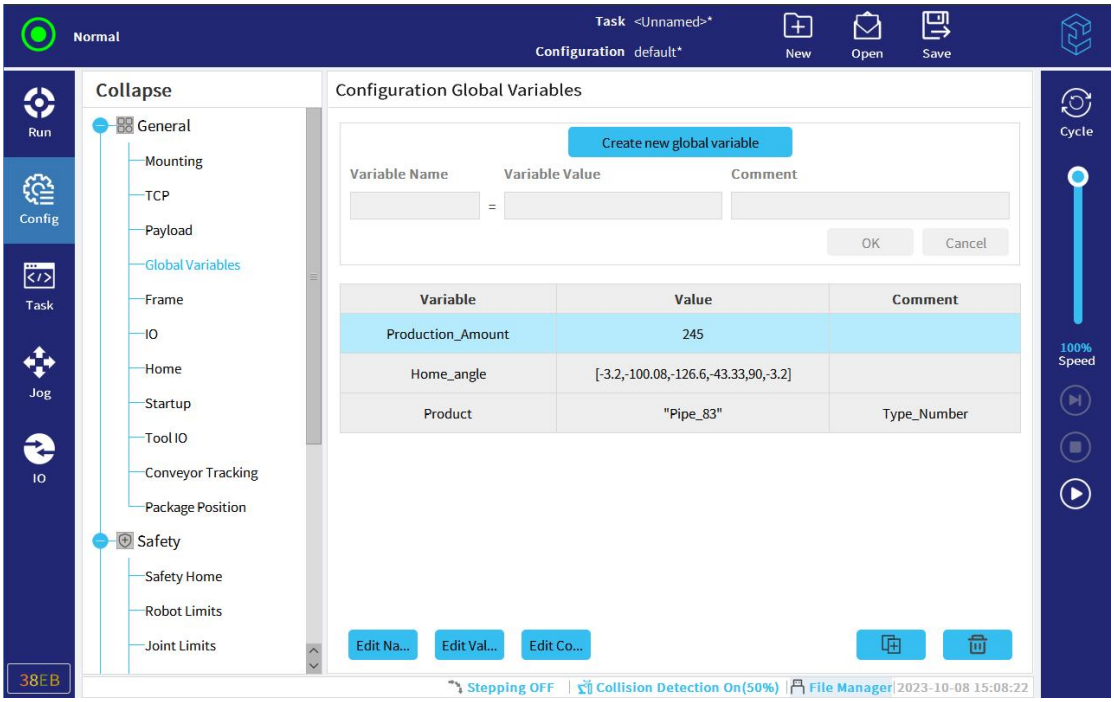


Figure 8-1

Step: click “Create new global variable”, enter the variable name and value and click “OK”.

Select the existing variable and the user can edit the variable name and values, as well as add the remarks. Click  to copy the variable. Click  to delete the variable.

8.2 Local Variable

The local variables can be created in the assignment command. The data is saved in the task file and can only be called by the current running task.

If a local variable of not being defined is used in a calculation, the error will be reported when running the task.

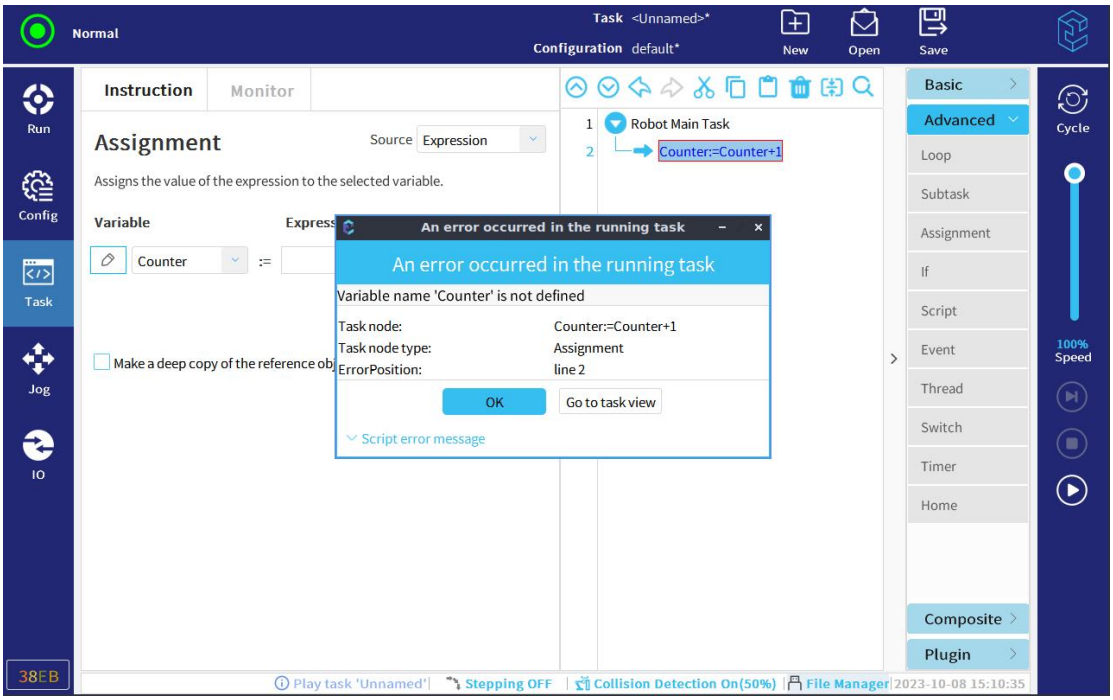


Figure 8-2

The user can pre-define the local variables by executing the assignment command in the before start sequence ”Beforestart” node for the easy use of the main task.

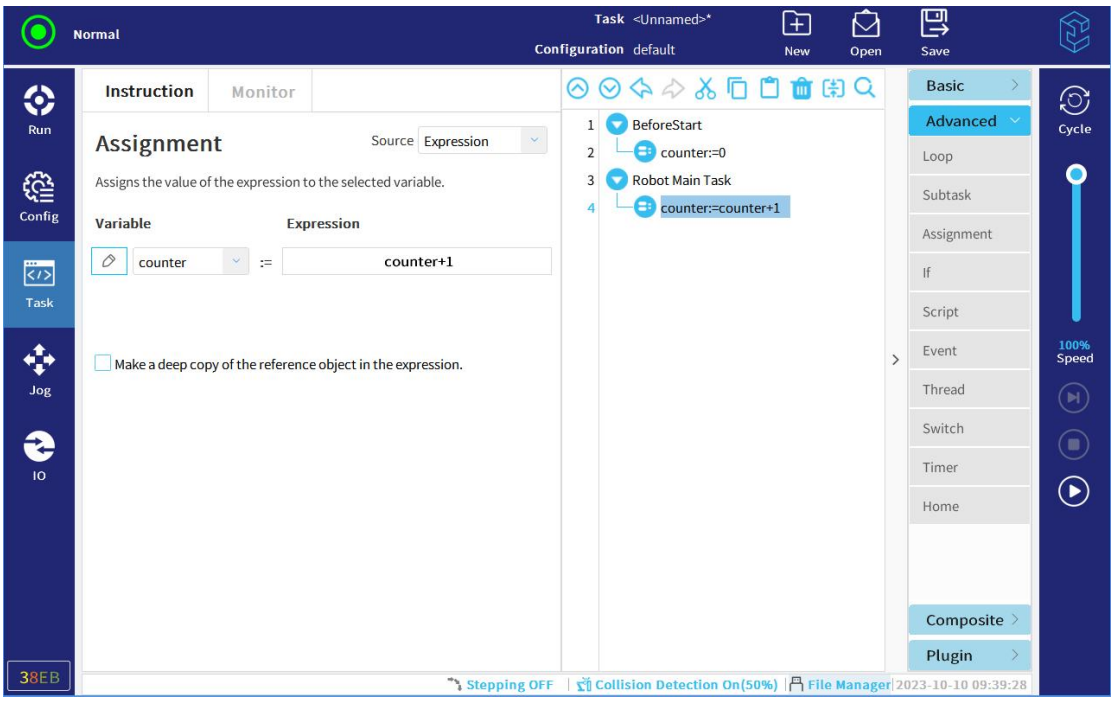


Figure 8-3

It is suggested to select “Set Initial Variable Values”.

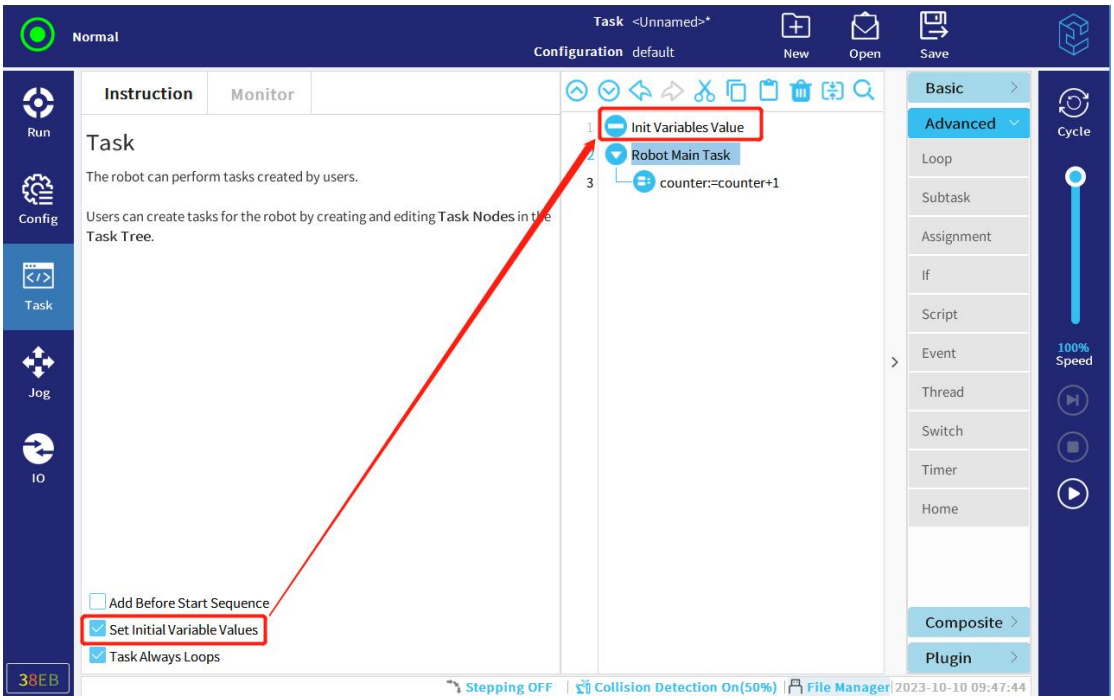


Figure 8-4

All local variables used in the task will be automatically added into the “Init Variables Value” node. The user can set the initial value. The variable will be reset to the initial value each time when the task restarts. If the option “Keep Value From Previous Run” is checked, the variable will be reset to the initial value only when the task starts for the first time.

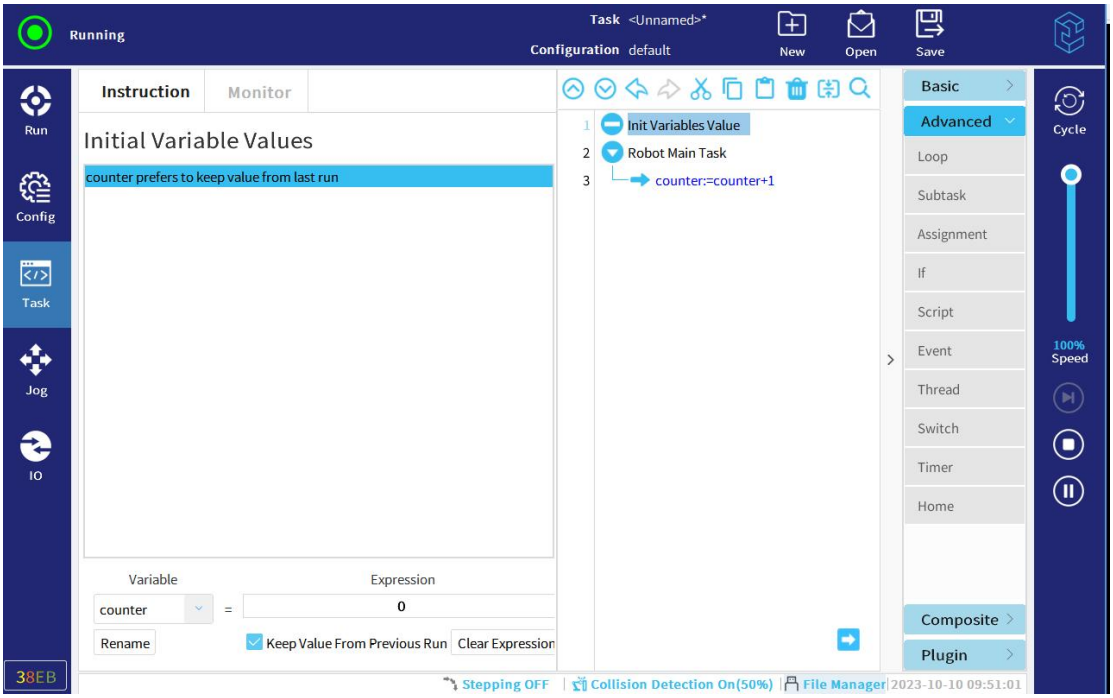


Figure 8-5

8.3 Assignment

The global variable and local variable can be created with the assignment command.

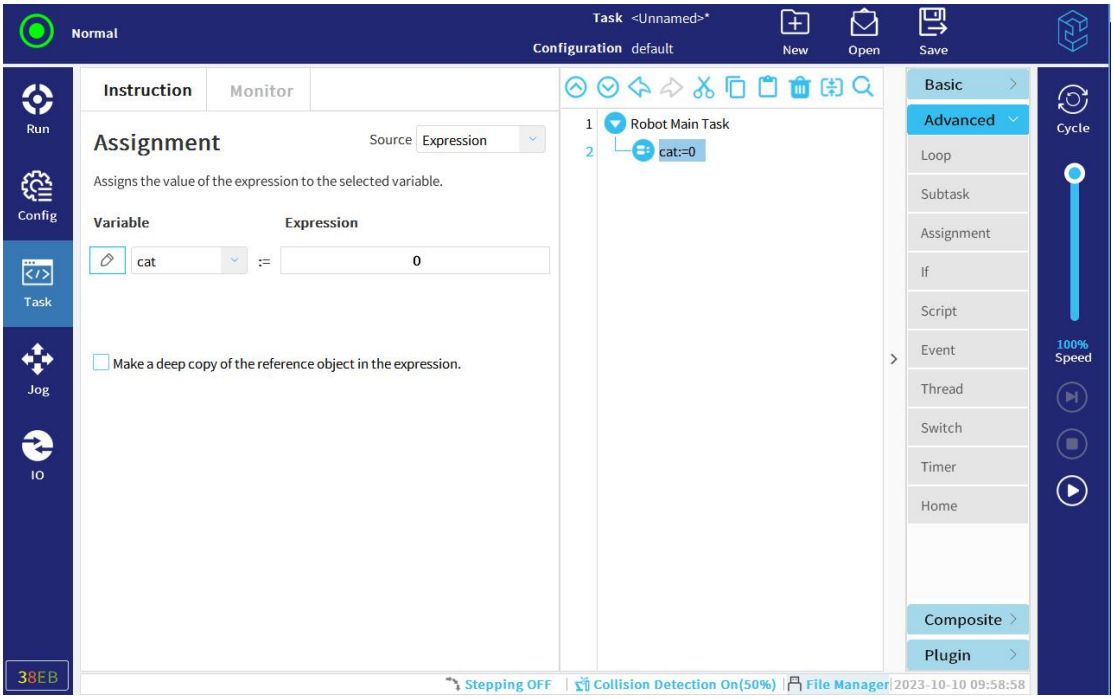


Figure 8-6

To modify one of the element in the variable list, please use the script command, enclose the element in a square bracket and make an assignment via the equal sign.

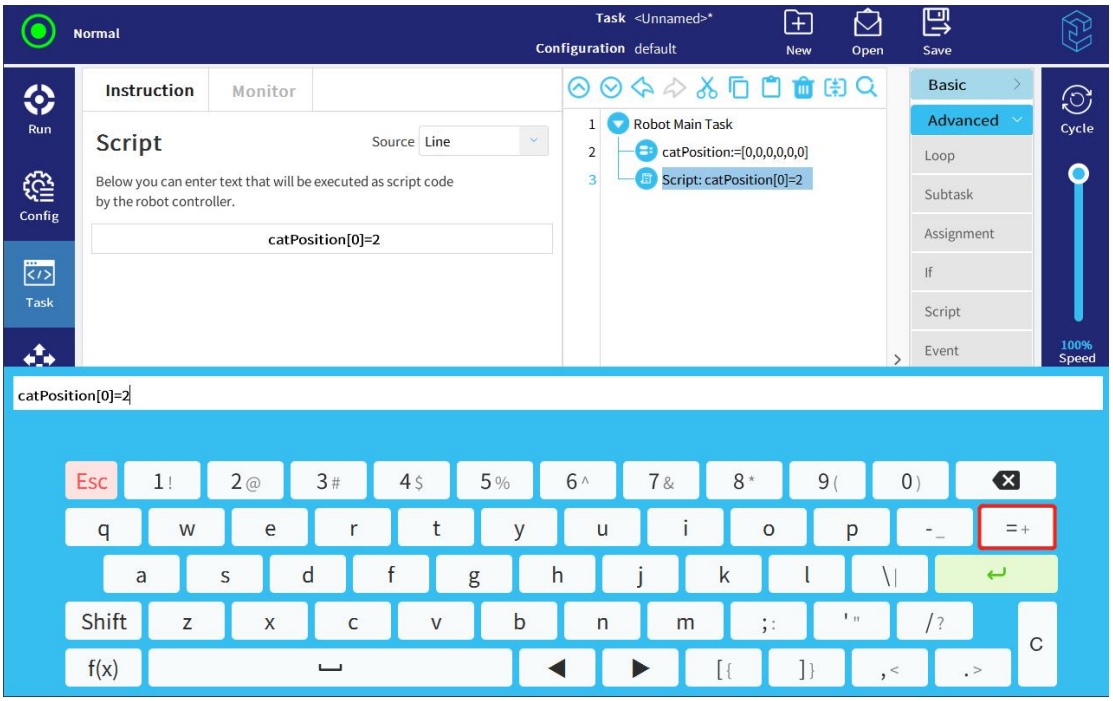


Figure 8-7

* There is a feature for the variable of list type. If variable A of the list type is assigned to the variable B of the list type, the two variables will be directed to the same list.

As shown in the 5th node in the following figure, the variable catPosition is assigned to the variable tigerPosition. The value of the variable catPostion is modified (see 7th node). When the 1st element of the variable catPosition is increased (+1), the 1st element of the variable tigerPosition is also added up (+1).

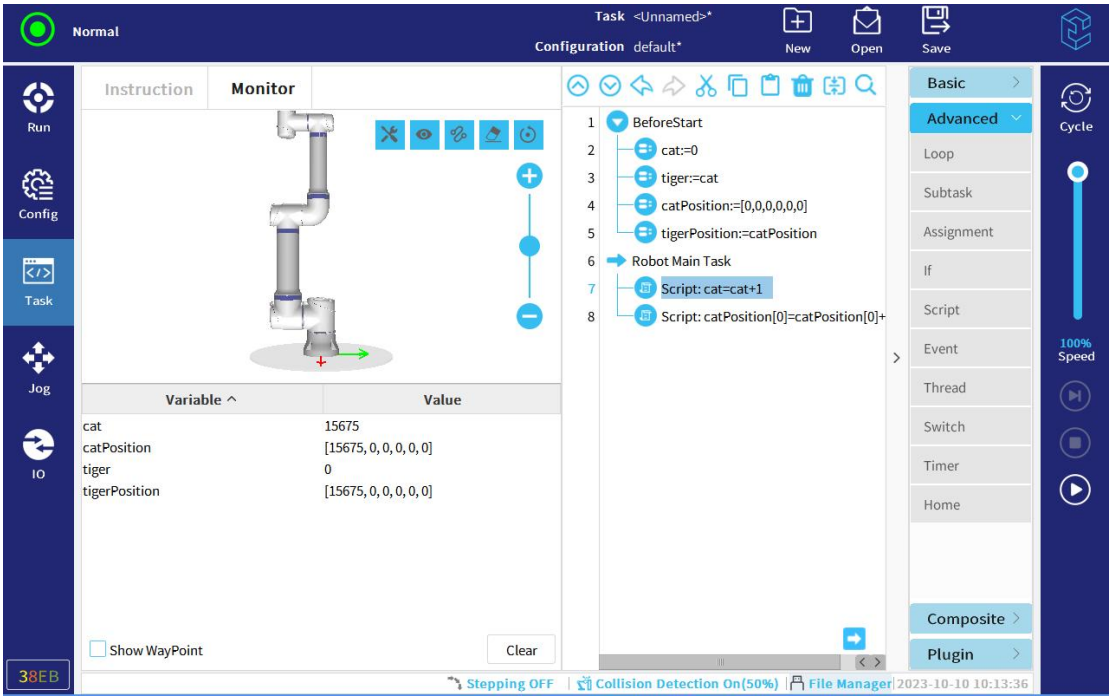


Figure 8-8

There are two ways to assign one variable of list type to another one.

One is to select “Make a deep copy of the reference object in the expression”.

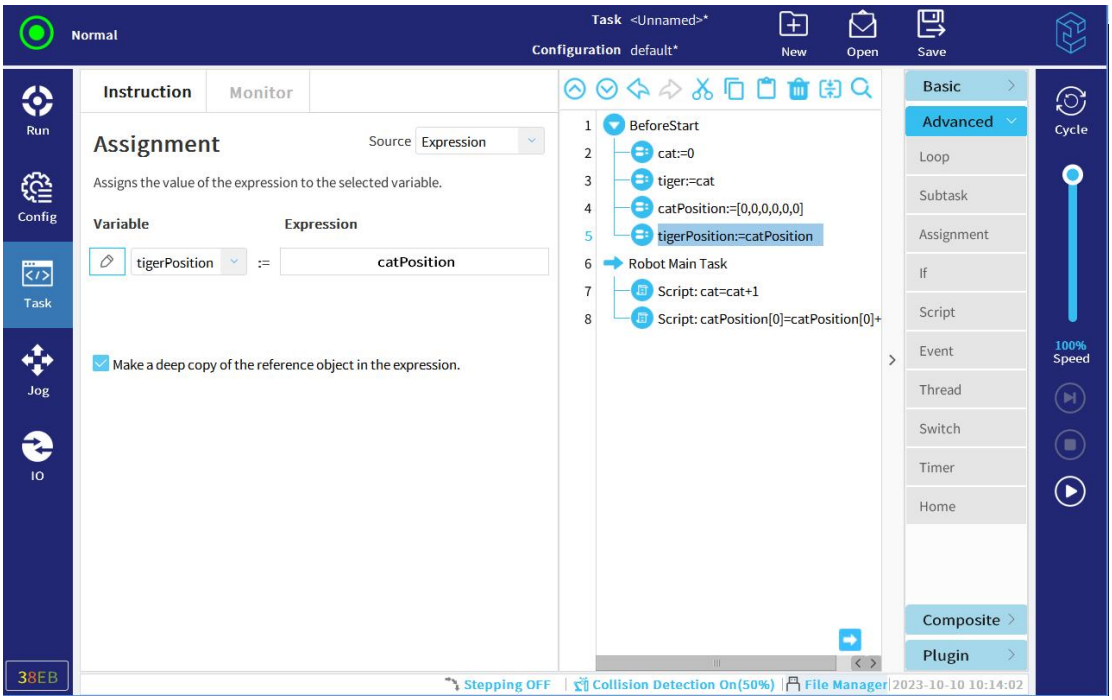


Figure 8-9

The other is not to select “Make a deep copy of the reference object in the expression”. The user needs to add the suffix of .copy() in the source list of the assignment.

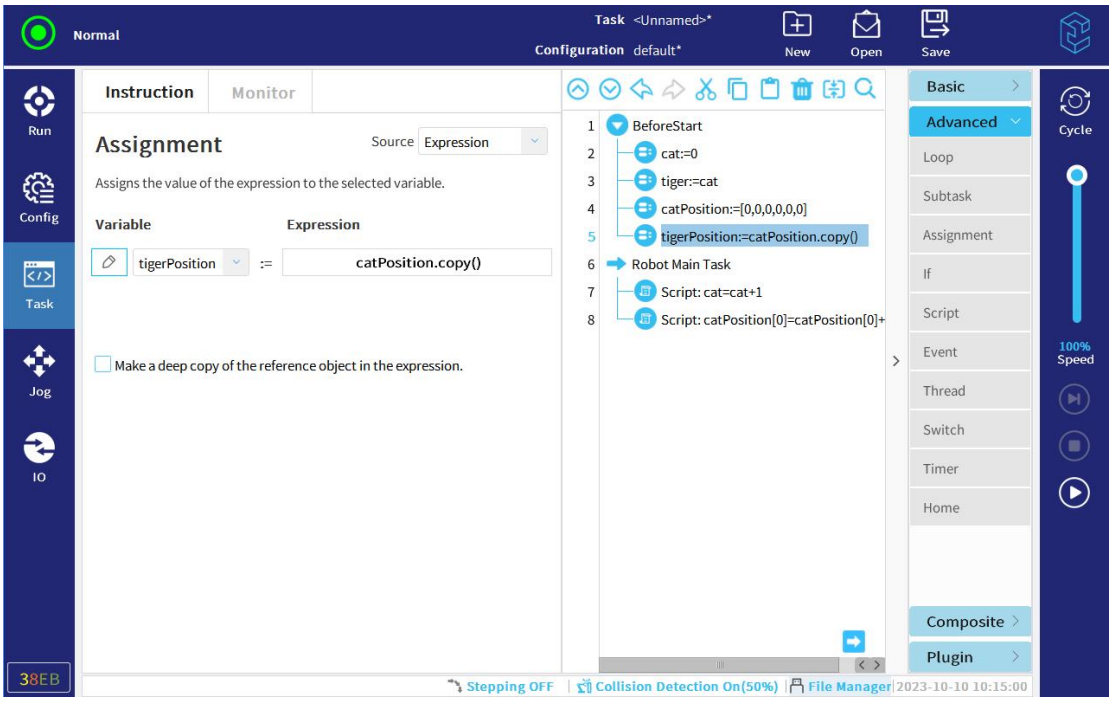


Figure 8-10

With the two methods mentioned above, the assigned object will only copy the values from the source list instead of simply directing the assigned variable to the same list.

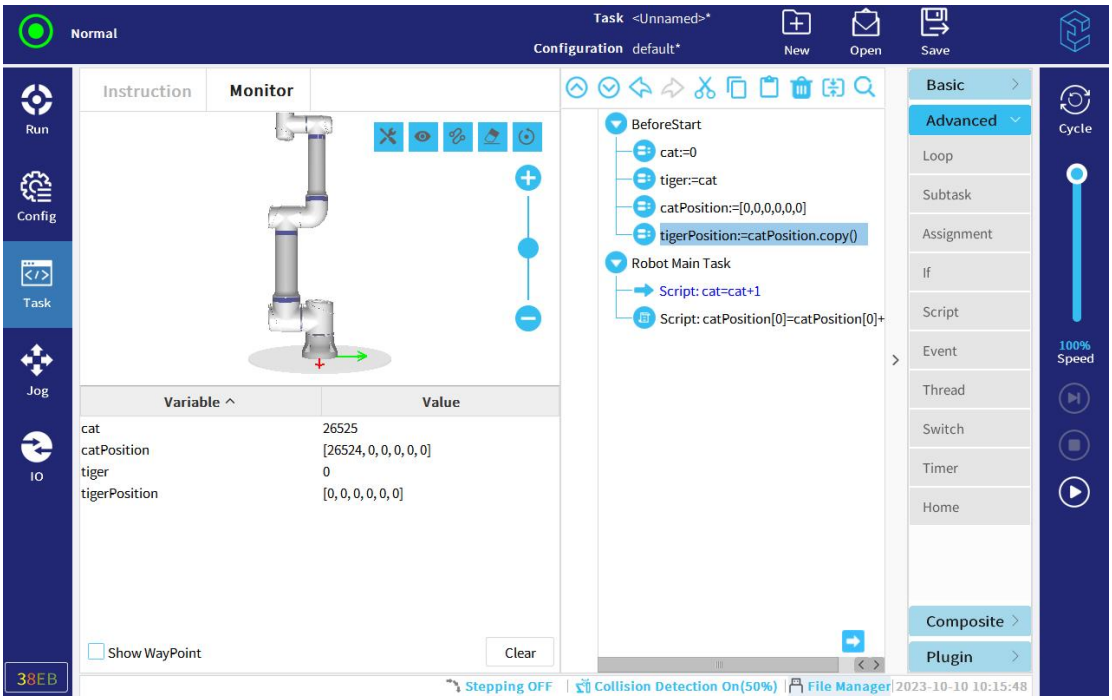


Figure 8-11

9 I/O Signal

9.1. Digital IO

The user can see 16 digital inputs (DI) and 16 digital outputs (DO) after opening the controller.

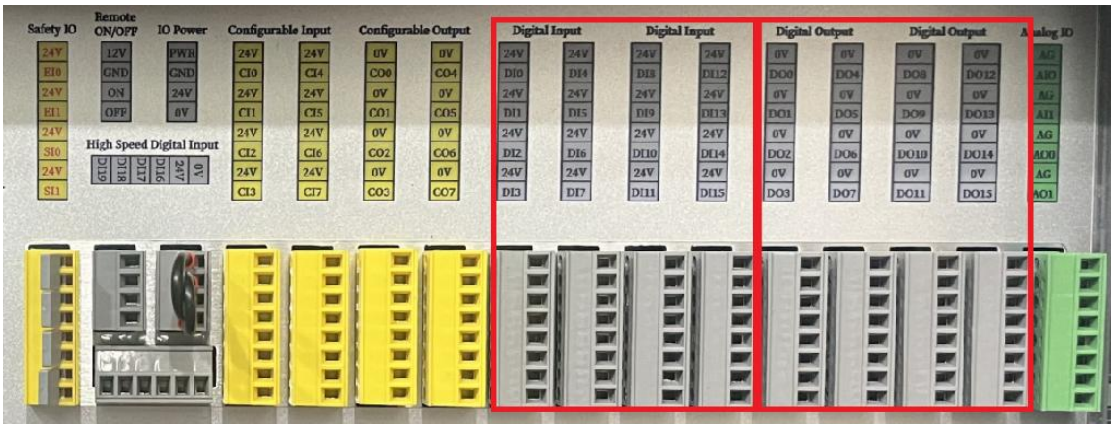


Figure 9-1

In the IO tab, the user can directly monitor the state of the digital IO signal. The output signal can be manually changed at any time by default.

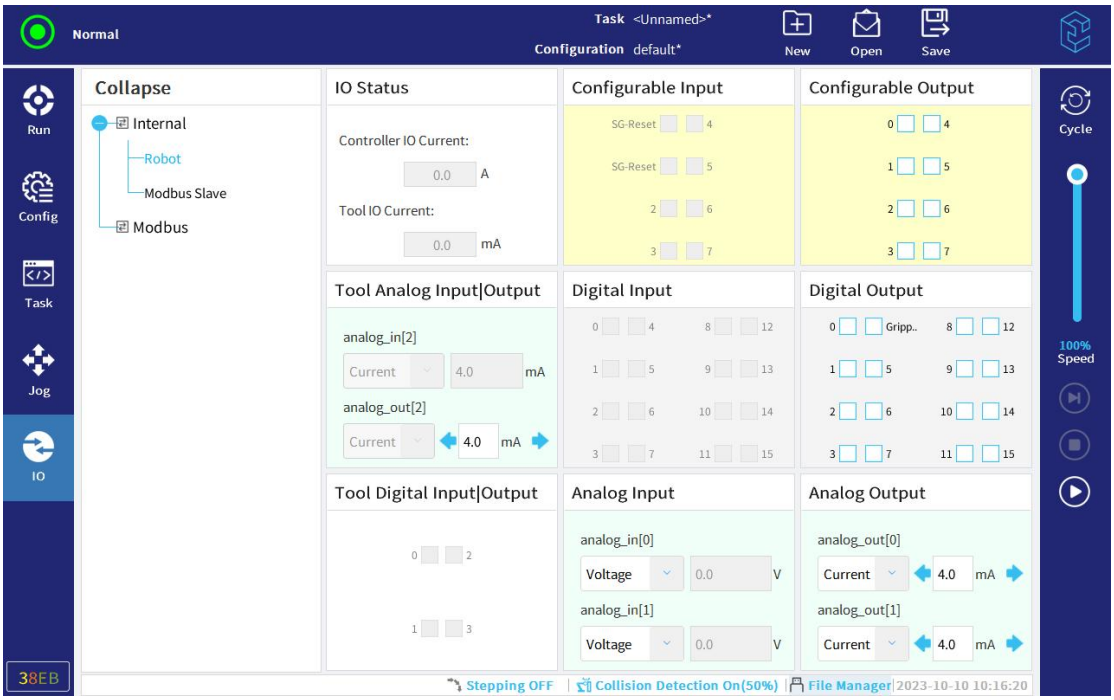


Figure 9-2

Click **Config -> IO**, and the user can select the diginal signal and configure it.

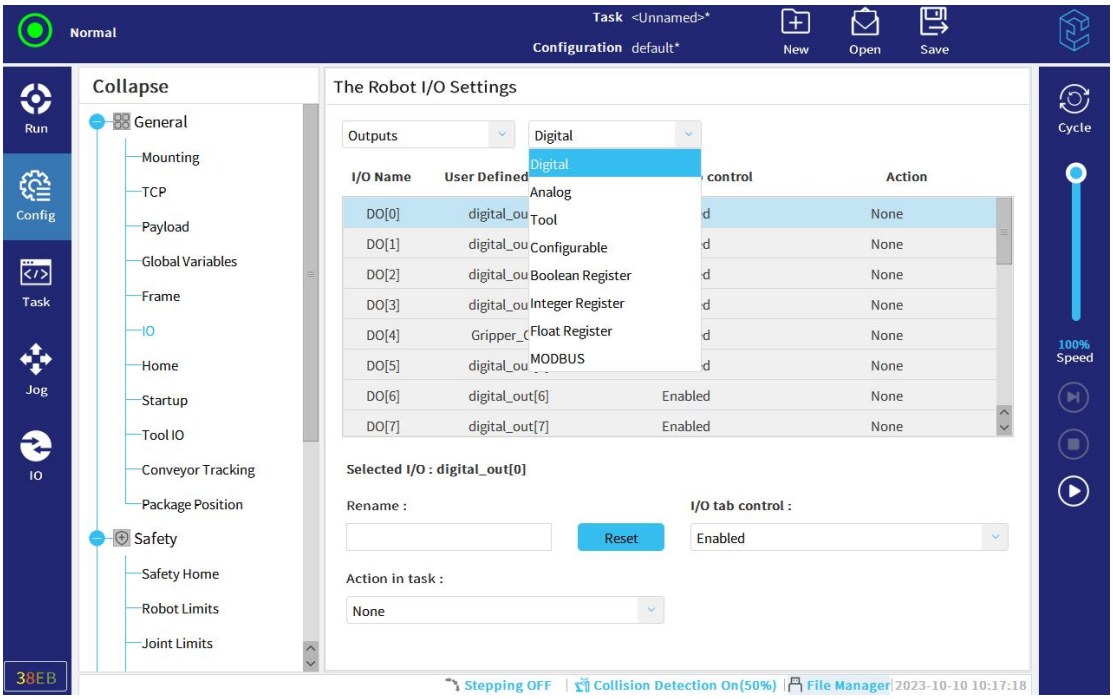


Figure 9-3

The user can rename the signal.

The user can set the control of the digital output.

When selecting “Enbled”, the user can modify the output status in the IO tab.

When selecting “Manual mode only”, the user can only modify the output status in the manual mode.

When selecting “Disabled”, the output status cannot be modified.

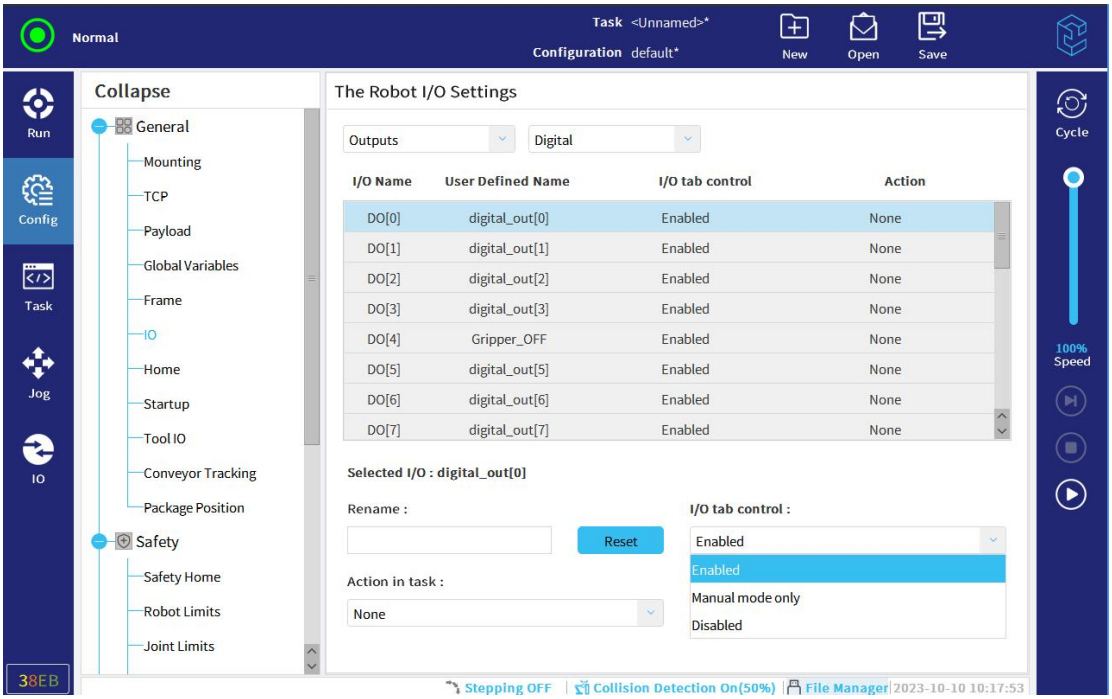


Figure 9-4

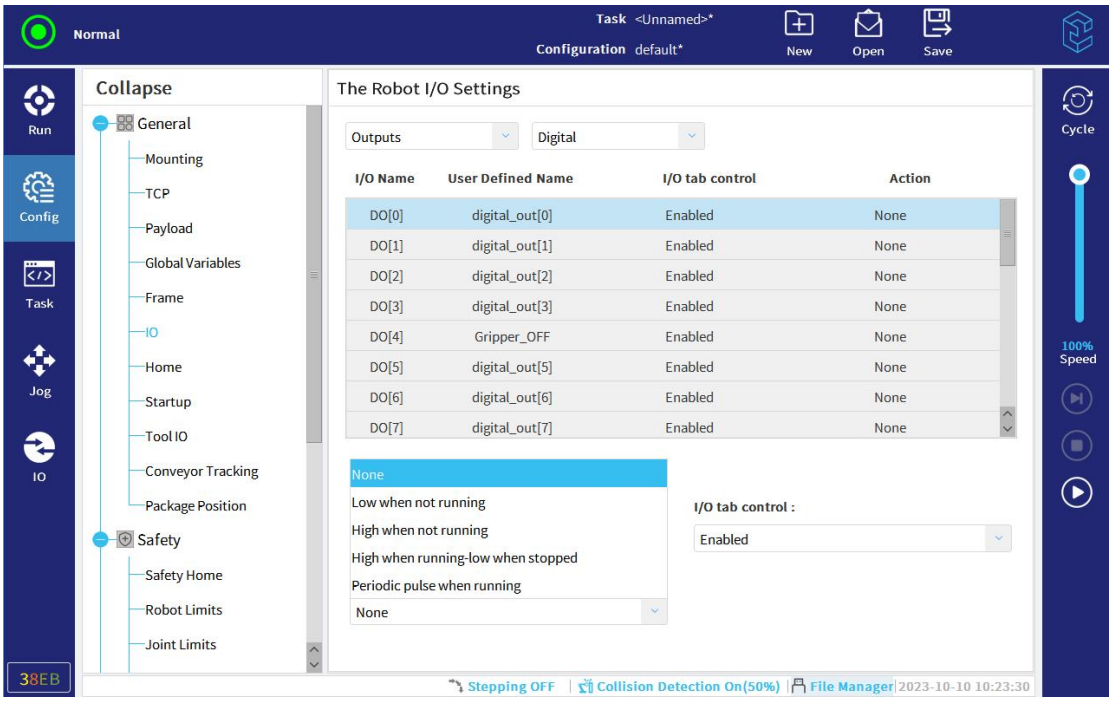


Figure 9-5

The digital input can be used to trigger some functions, incl. starting, stoping or pausing a task. It can also be used to enable the drag function.

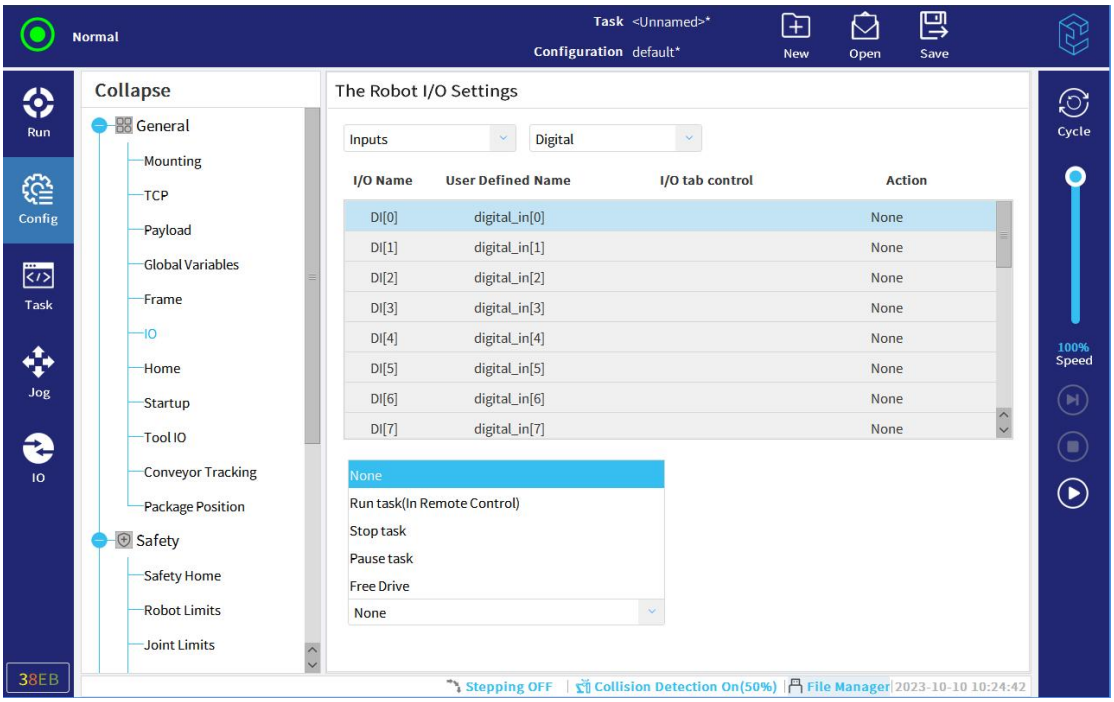


Figure 9-6

9.2 Configurable IO

There are 8 configurable inputs and 8 configurable outputs in the controller.

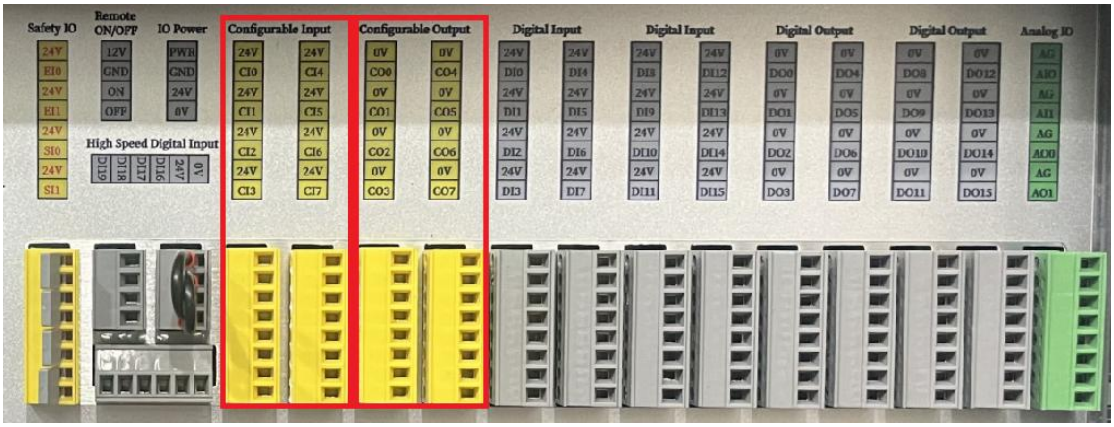


Figure 9-7

In the IO tab, the user can directly monitor the state of the configurable IO signal. The output signal can be manually changed at any time by default.

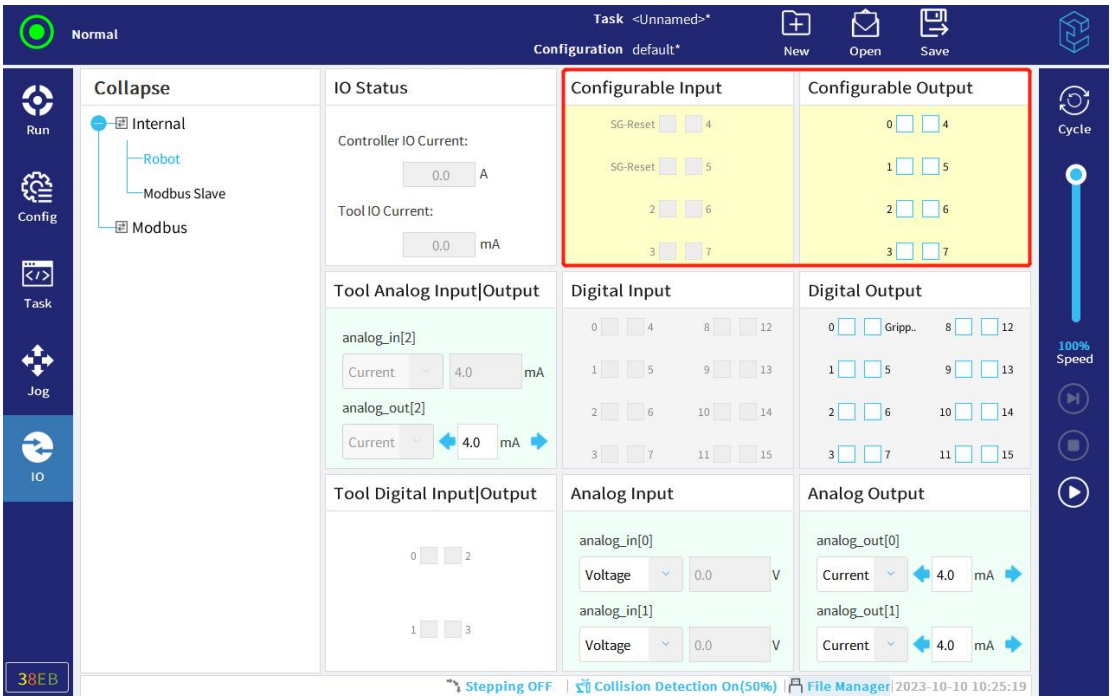


Figure 9-8

The configurable IO can be used as a safety IO signal and the dual-loop signal is connected. When it is not used as a safety IO signal, treat it as an extended digital IO signal, which can be used in a single channel.

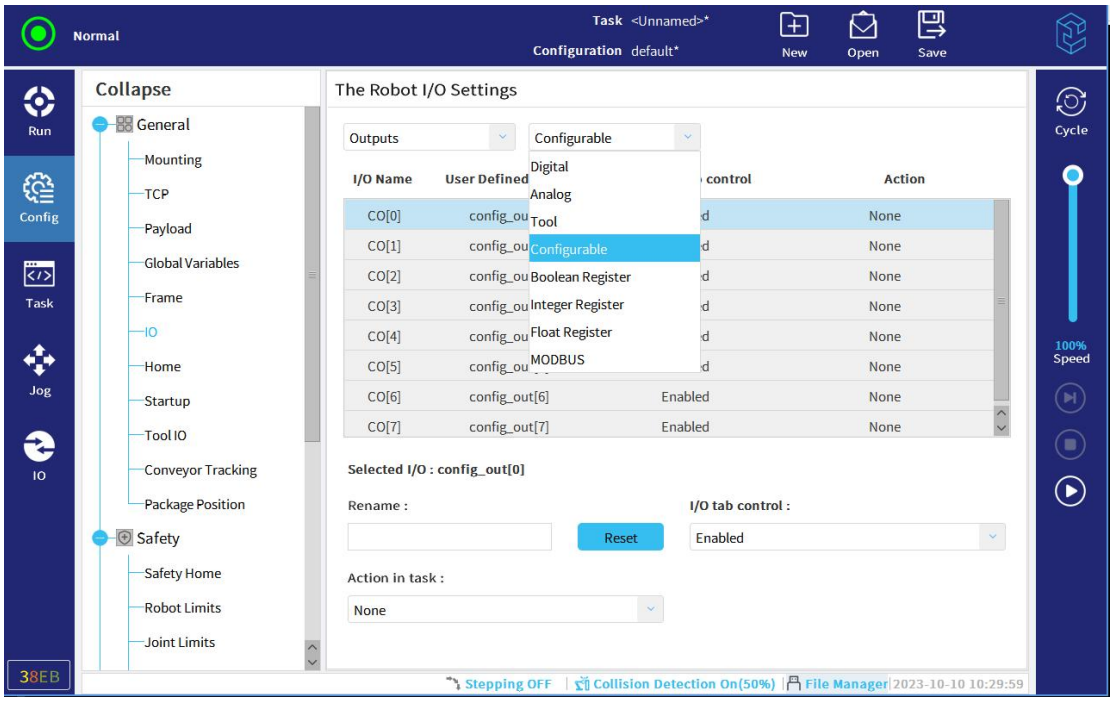


Figure 9-9

Note: The configurable input implements the safety functions, e.g. trigger the safety plane, trigger the emergency or safeguard stop, help switch automatically and manually. For the details, see the chapter 20 Safety Setting.

9.3 Analog IO

The controller is equipped with 2 analog inputs and 2 analog outputs.

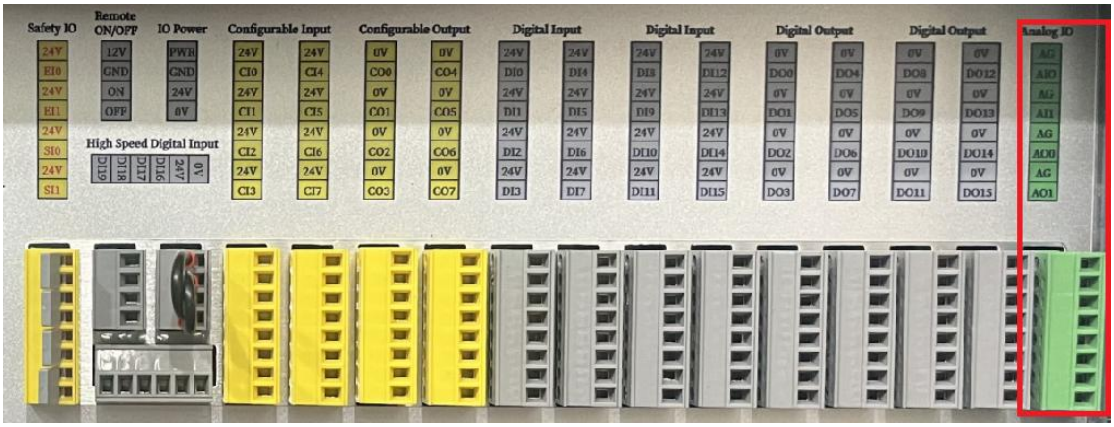


Figure 9-10

In the IO tab, it is possible to configure the analog IO to use the current mode or the voltage mode.

The user can monitor the state of the signal and by default, manually change the output signal at any time.

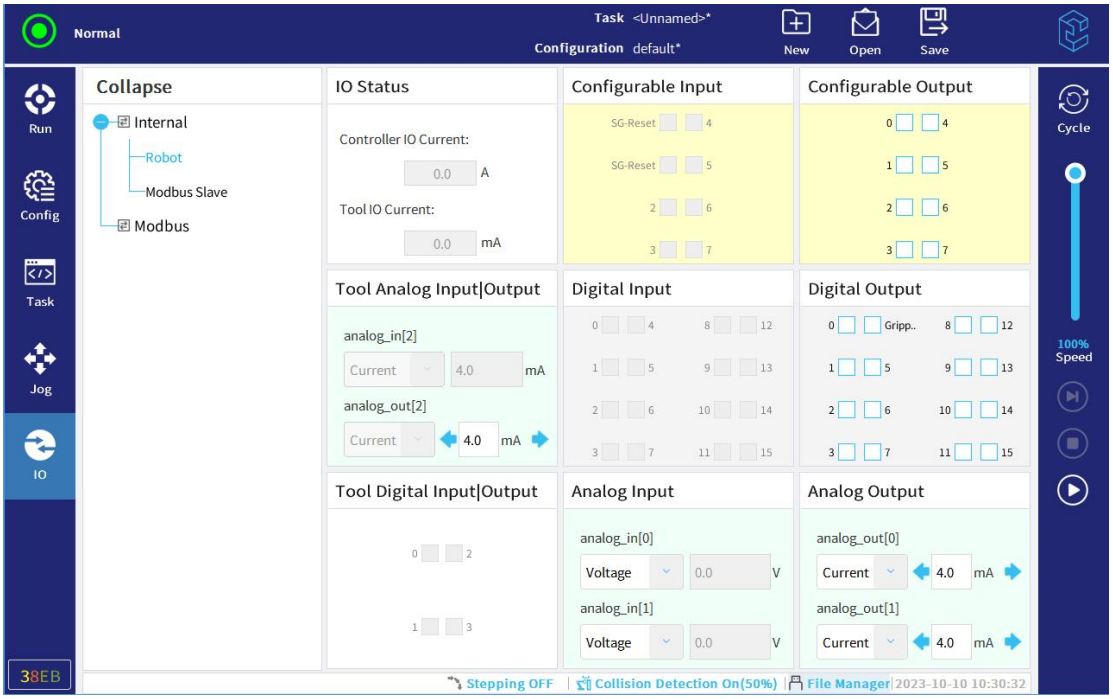


Figure 9-11

Click **Config -> IO**, and the user can select the analog signal and configure it.

Note: AI[2] and AO[2] are from the flange IO of the robot.

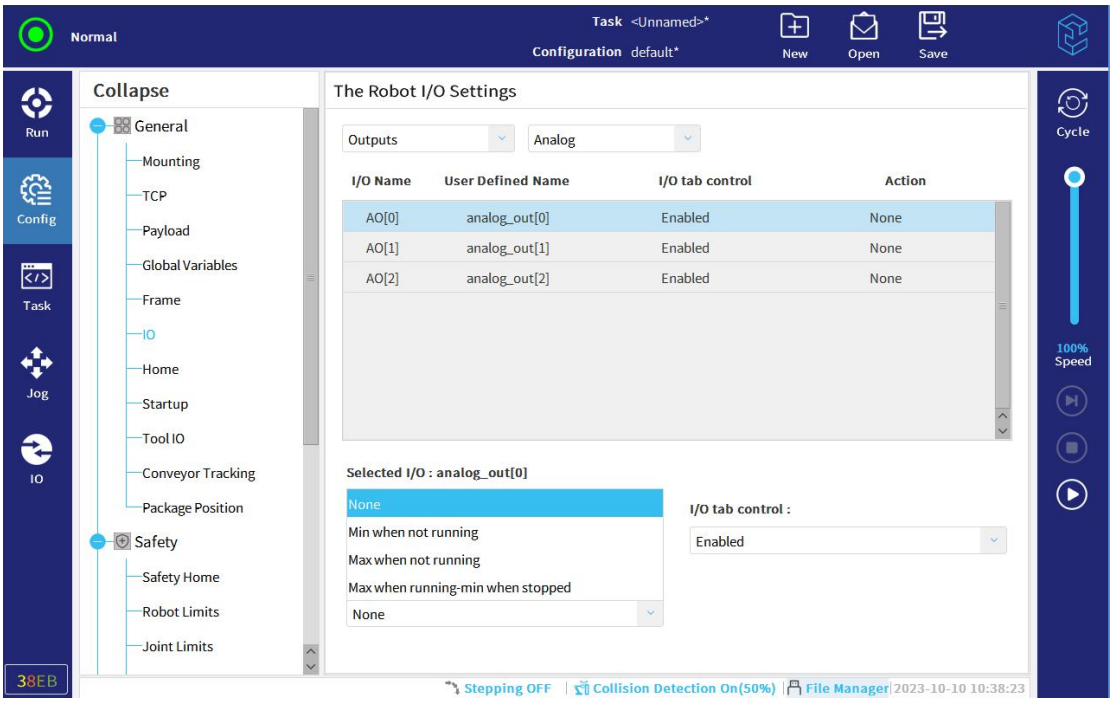


Figure 9-12

9.4 Tool IO

There are 8 pins on the tool flange, among which the 3rd, 4th, 6th and 7th pin can be used as the digital IO in accordance with the configuration, the 1st pin can be configured as an analog input and the 2nd pin can be configured as an analog output.

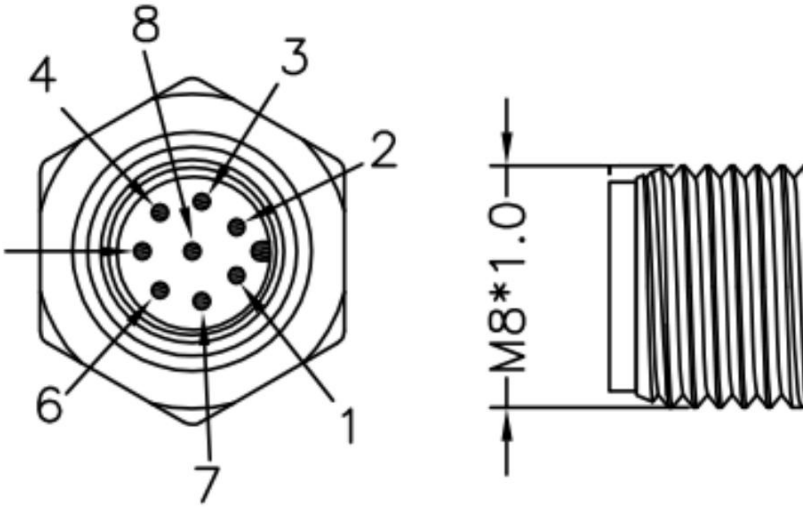


Figure 9-13

The tool IO is used as a RS485 interface in the USART mode. In this case, however, the user cannot use the analog input and output.

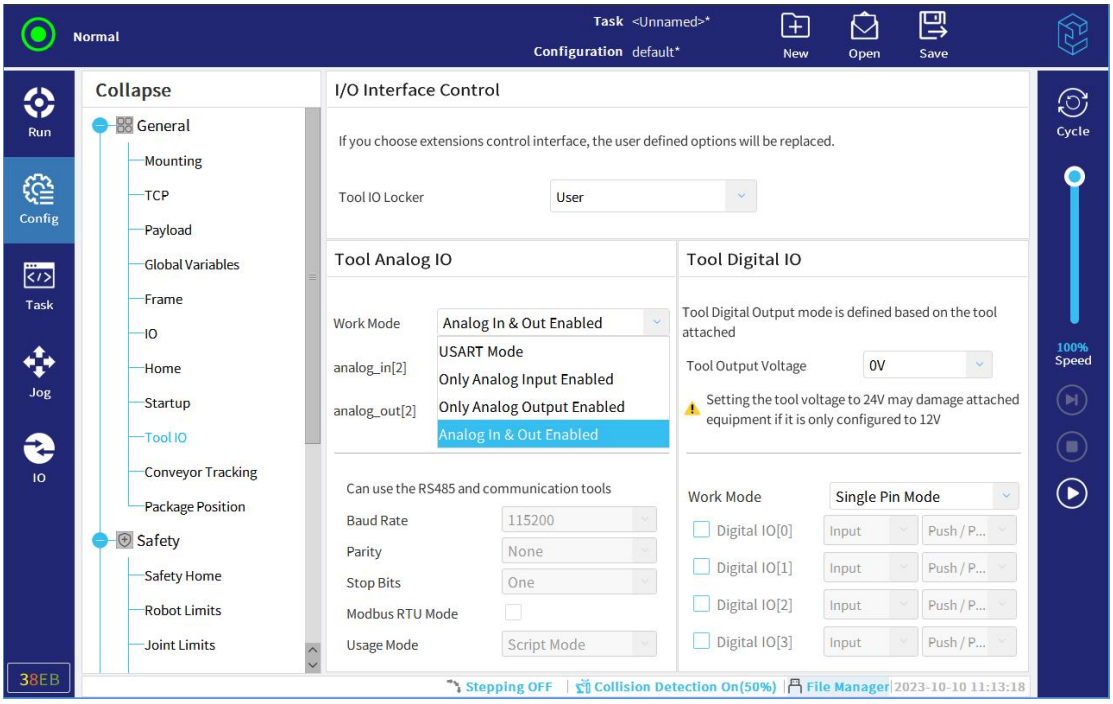


Figure 9-14

When selecting “Analog In & Out Enabled” from the dropdown list of “Work Mode”, the user will see two options, i.e. current or voltage.

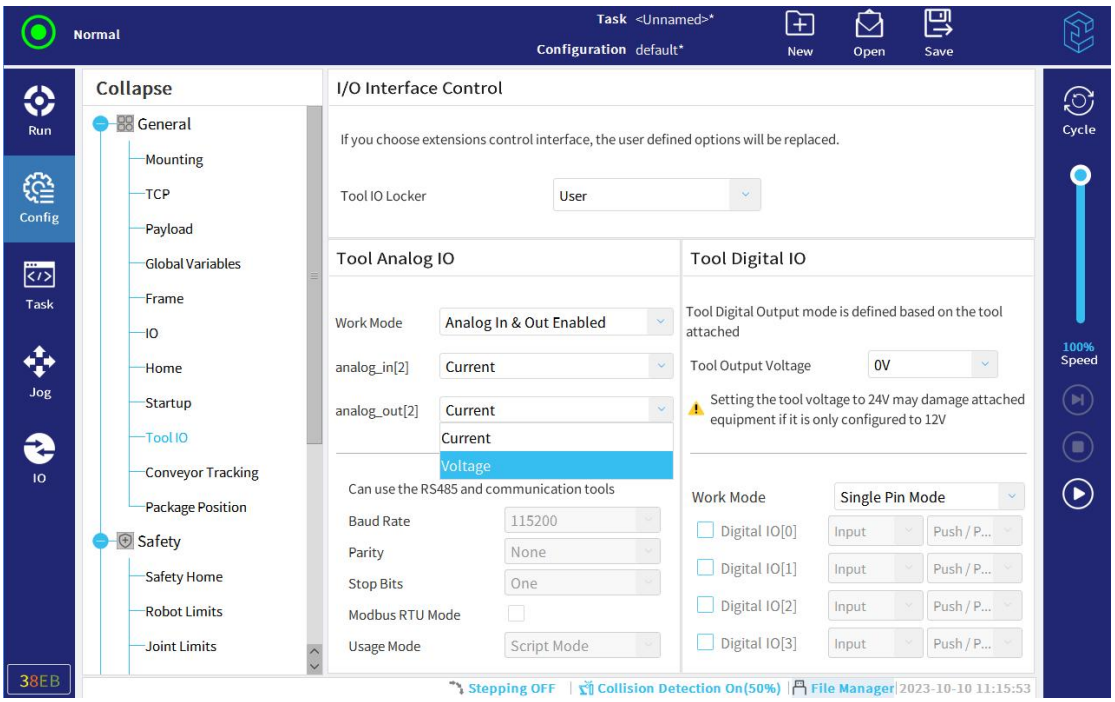


Figure 9-15

The digital IO has many configurations, such as I/O signal switching, actual amount of channels and signal modes, i.e. PNP, NPN and Push/pull mode (connect the circuit only). The user can also set the voltage output of the tool terminal, i.e. 0V/12V/24V.

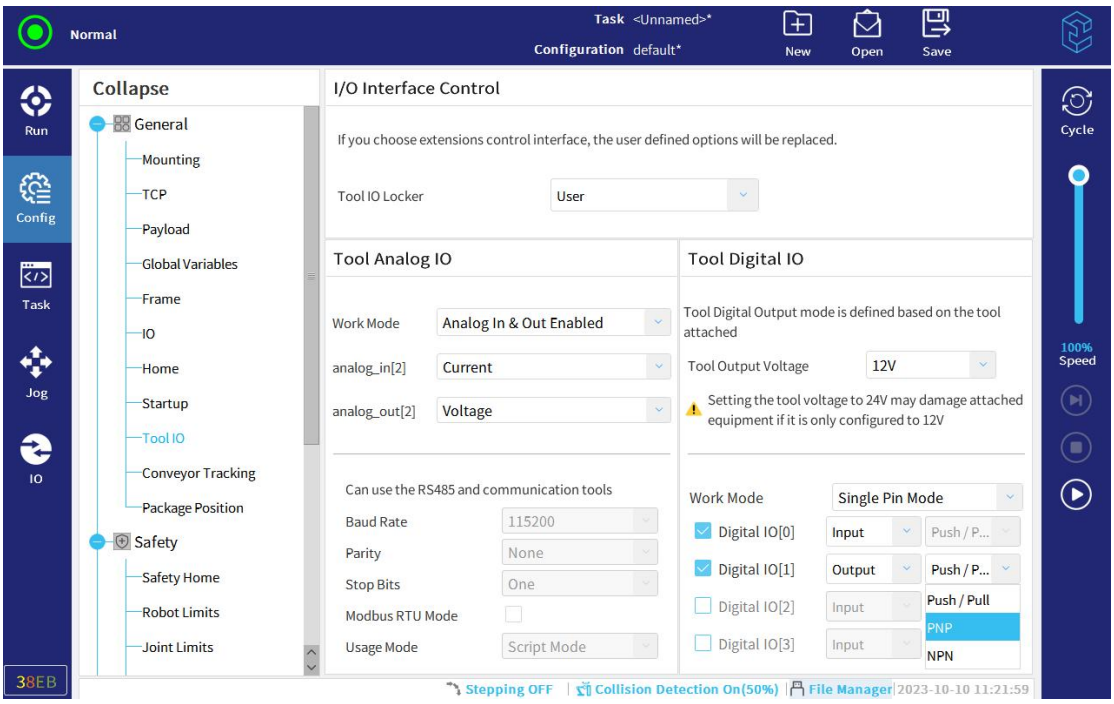


Figure 9-16

Note: It is suggested to set the output voltage to 0V when replacing the end-effectors and change the voltage to the supply voltage after the replacement.

In the IO tab, the user can directly monitor the state of the configurable IO signal after the configuration of the tool IO is complete. The output signal can be manually changed at any time by default.

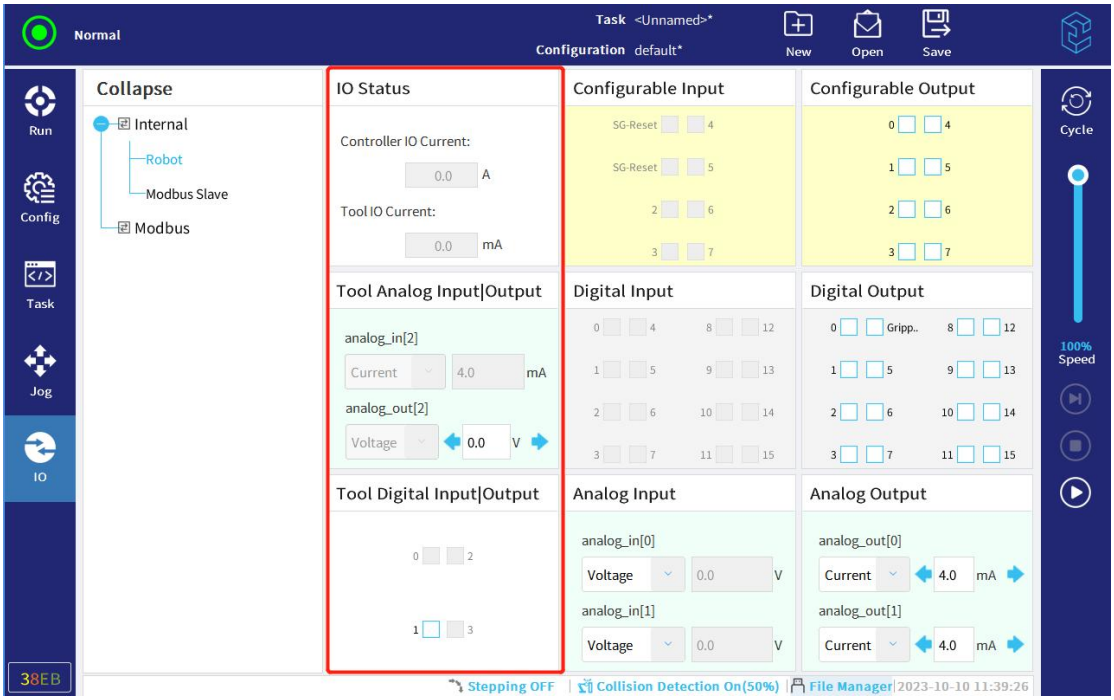


Figure 9-17

In the IO interface, the digital IO of the tool has the same configuration as that of the controller. Only the signal that has been configured in the tool IO interface will display.

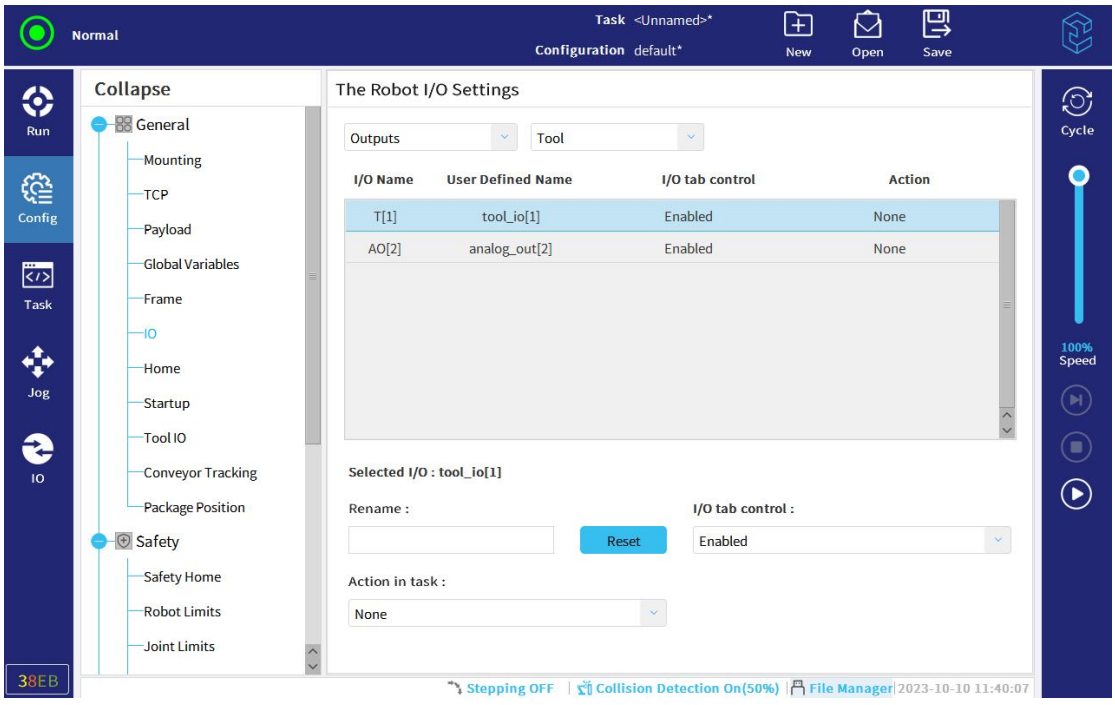


Figure 9-18

10 I/O Signal

10.1 Use the Input Signal

In the task tree, the instructions like assignment and script involving the expression can adopt the IO signal as the calculation conditions.

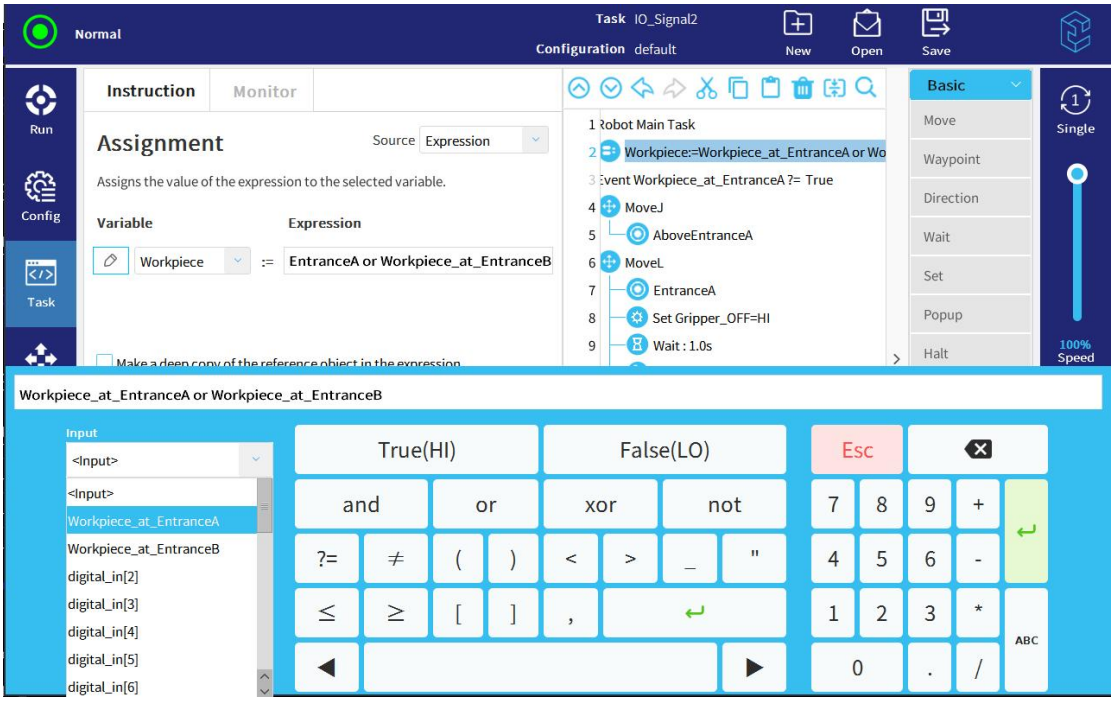


Figure 10-1

10.2 Set the Output Signal

The user can set the state of the output signal with the Set instruction.。

The function of the Set node includes:

- Set Digital Output: the user can select a digital output, a tool digital output or a configurable output which is not occupied by the safety functions and set the value to Low or High.

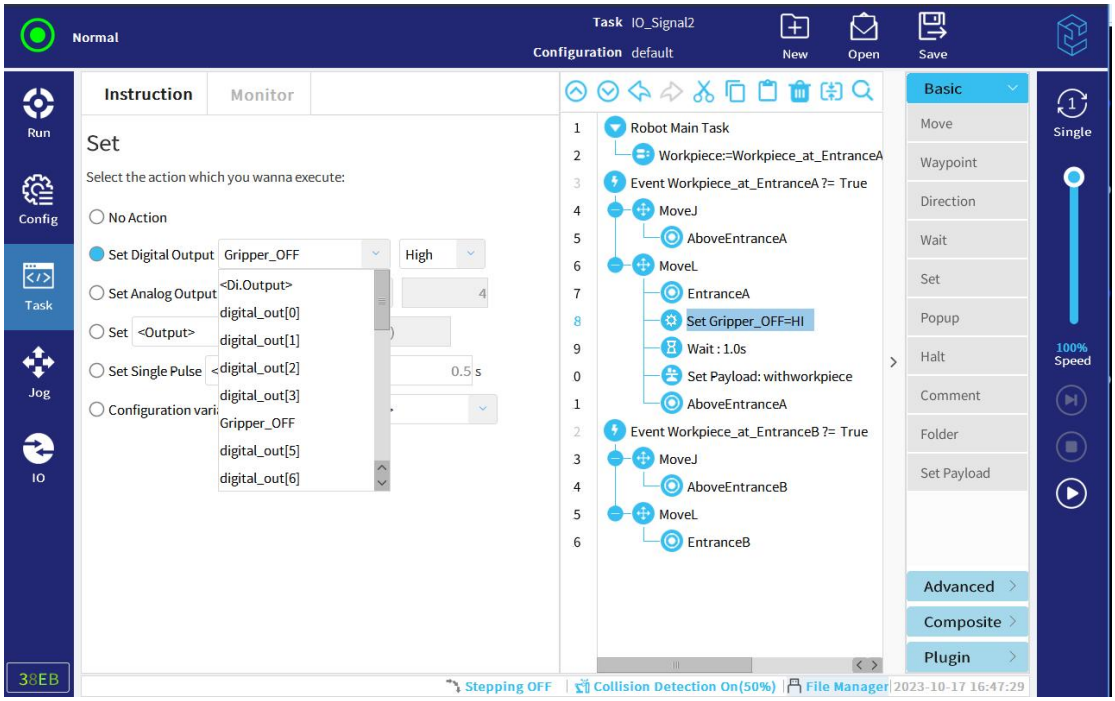


Figure 10-2

- Set Analog Output: the user can select an analog output and set the value of the current or the voltage, which depends on the type of the analog output in the ”IO” tab. The tool analog output type is determined by the IO settings of the tool in the ”Config” tab.

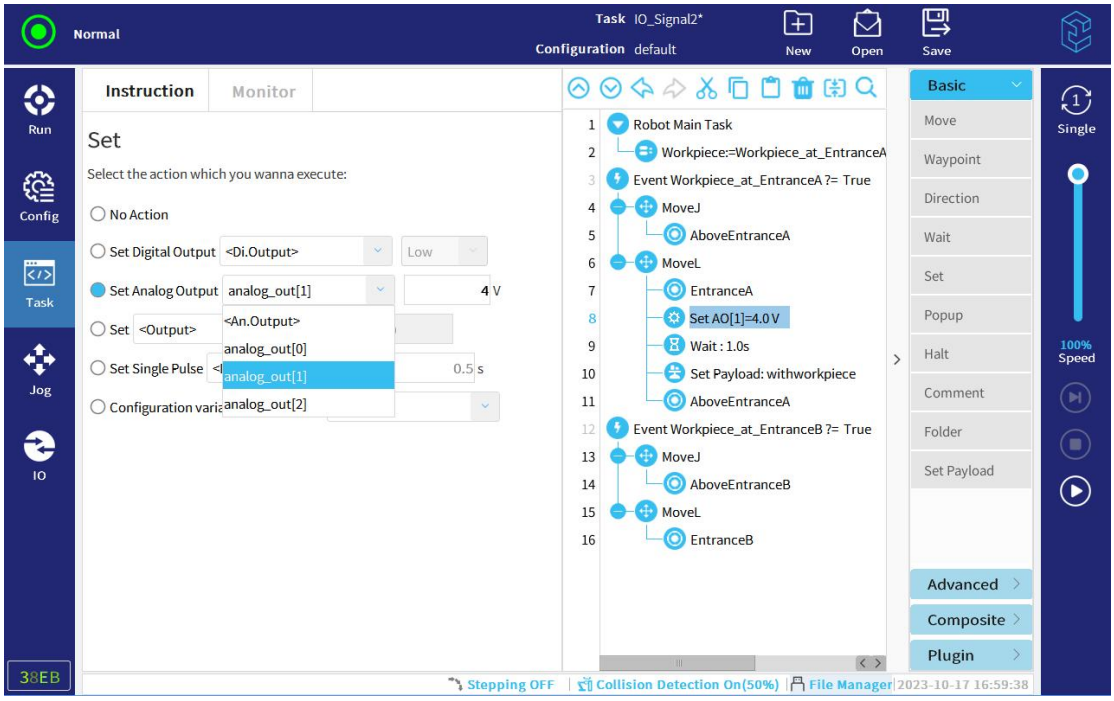


Figure 10-3

- Set: the user can select an output, incl. digital output, analog output, configurable output, tool output and Modbus output. The signal can be set in accordance with the expression.

When selecting the digital output, configurable output, tool digital output and Modbus digital output, the expression must evaluate to a Boolen value. Otherwise, the system will report an error.

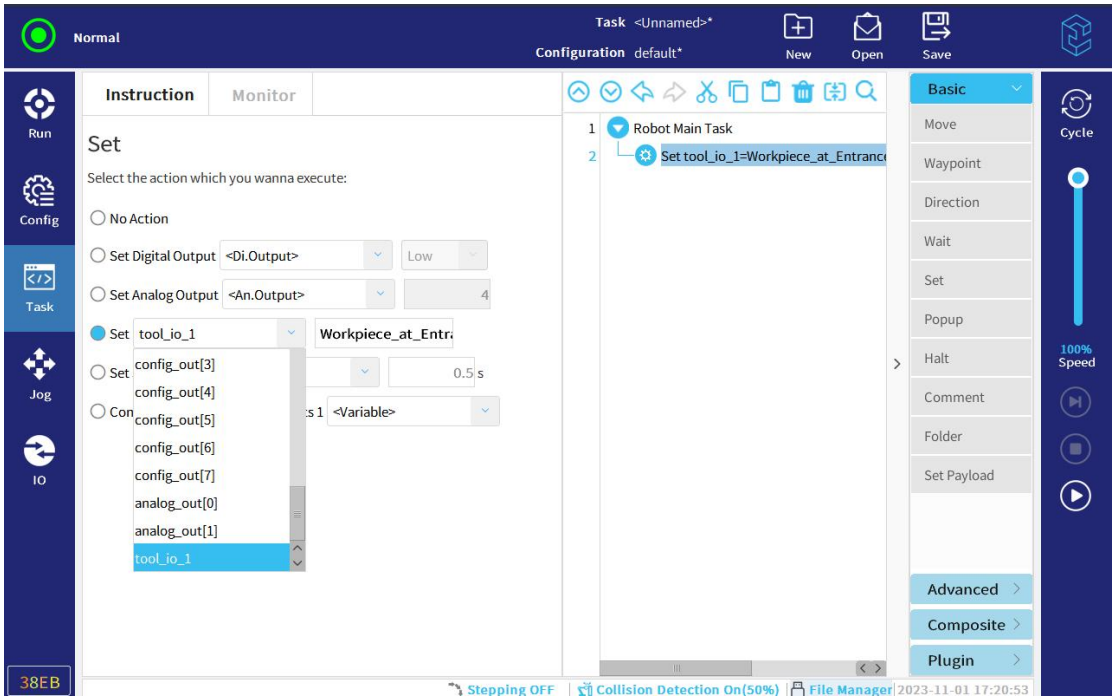


Figure 10-4

When selecting an output of the Modbus register, the expression must evaluate to an integer. Otherwise, the system will report an

error.

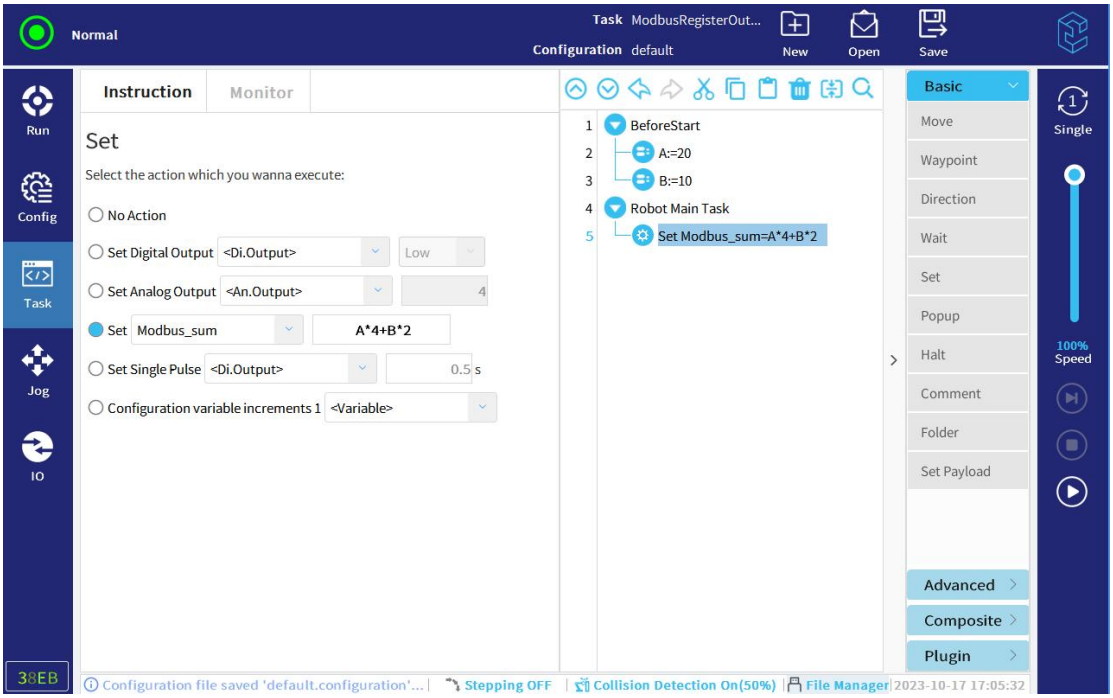


Figure 10-5

Example 1: the expression that evaluates the current output: range of the current is [4-20mA], when the value is 0.2, then the formula will be 4+0.2*(20-4) = 7.2 mA.

Example 2: the expression that evaluates the voltage output: range of the voltage is [0-10V], when the value is 0.2, the formula will be 0.2*10 = 2 V

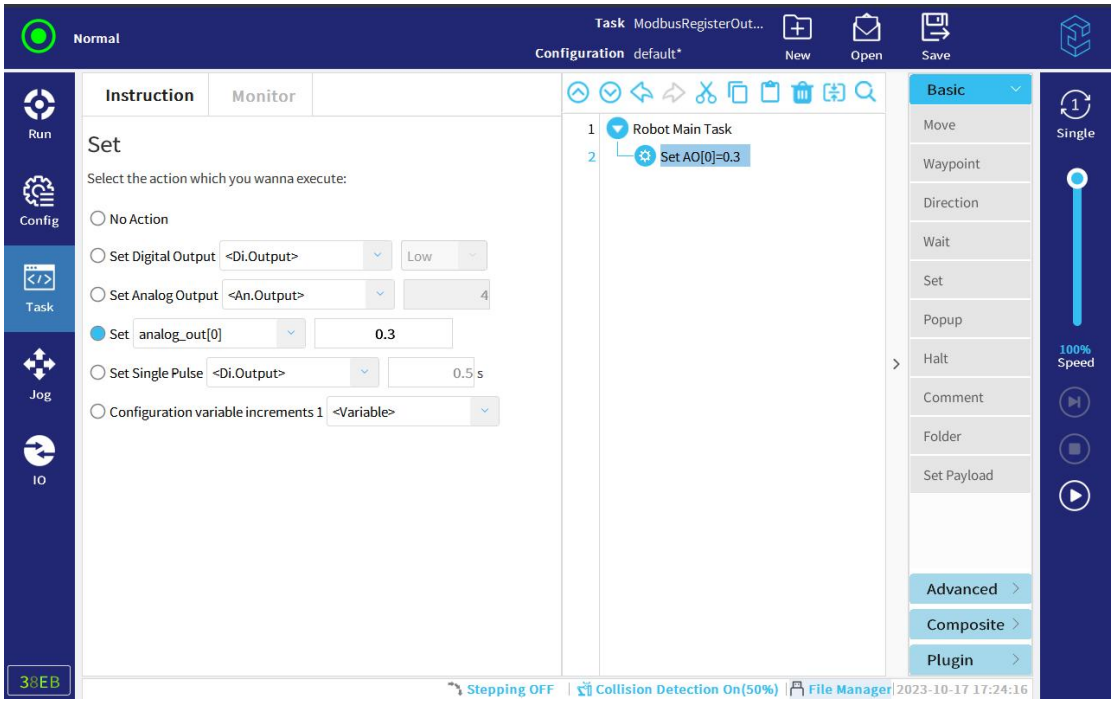


Figure 10-6

- Set Single Pulse: the user can select the digital output, configurable output, tool digital output and Modbus digital output and set the time of the single pulse (pulse width). That is, set the corresponding signal to High and then set it to Low after a certain of time elapses.

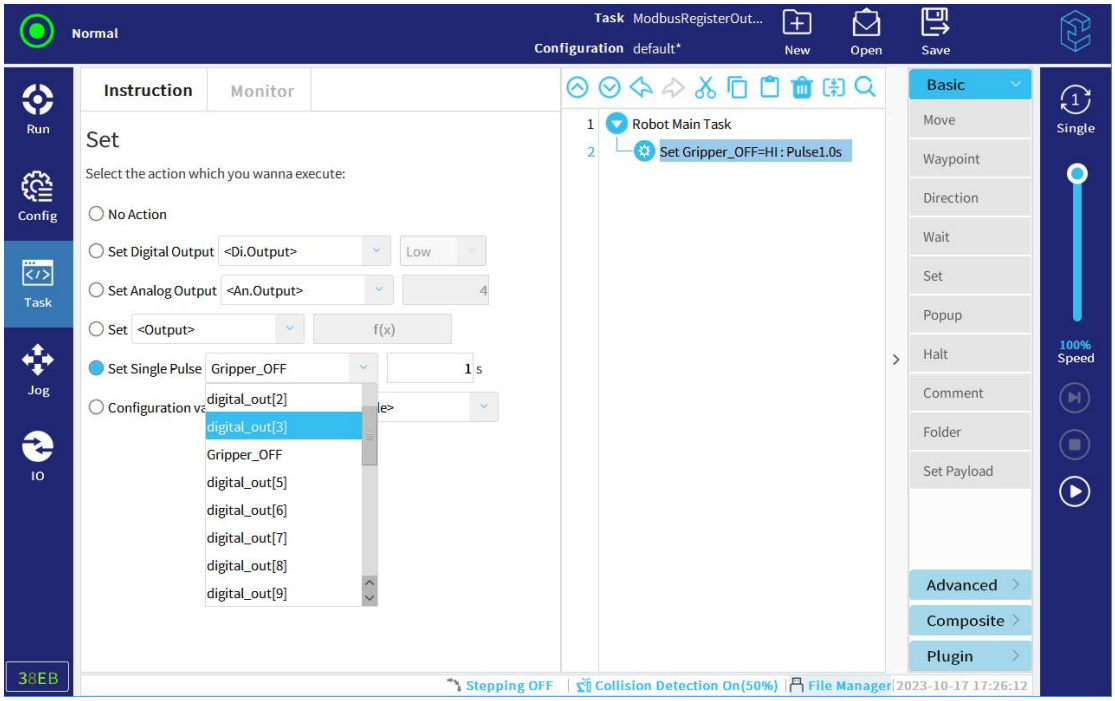


Figure 10-7

- Configuration variable increments 1: the user can select an integral or floating glocal variable and adds 1 to the value.

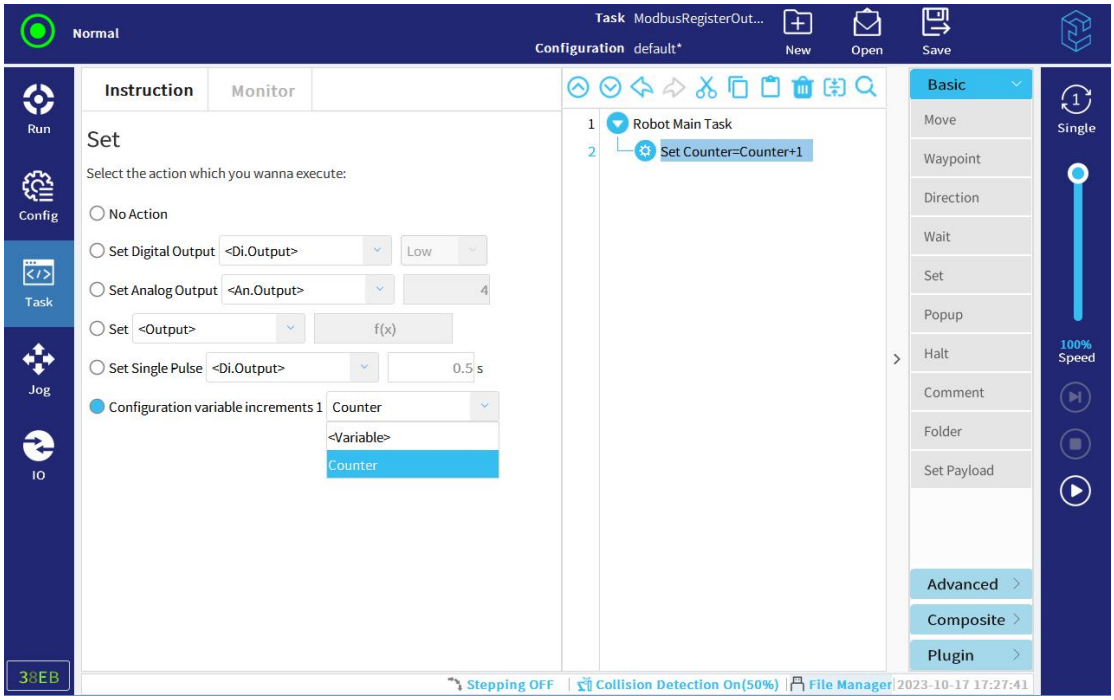


Figure 10-8

11 Payload Setting Command

The “Set Payload” command is used to dynamically switch over the actual payload of the tool.

The user can select ”payload” that has already been set in the ”Config” tab or ”Custom Payload Config” from the dropdown list.

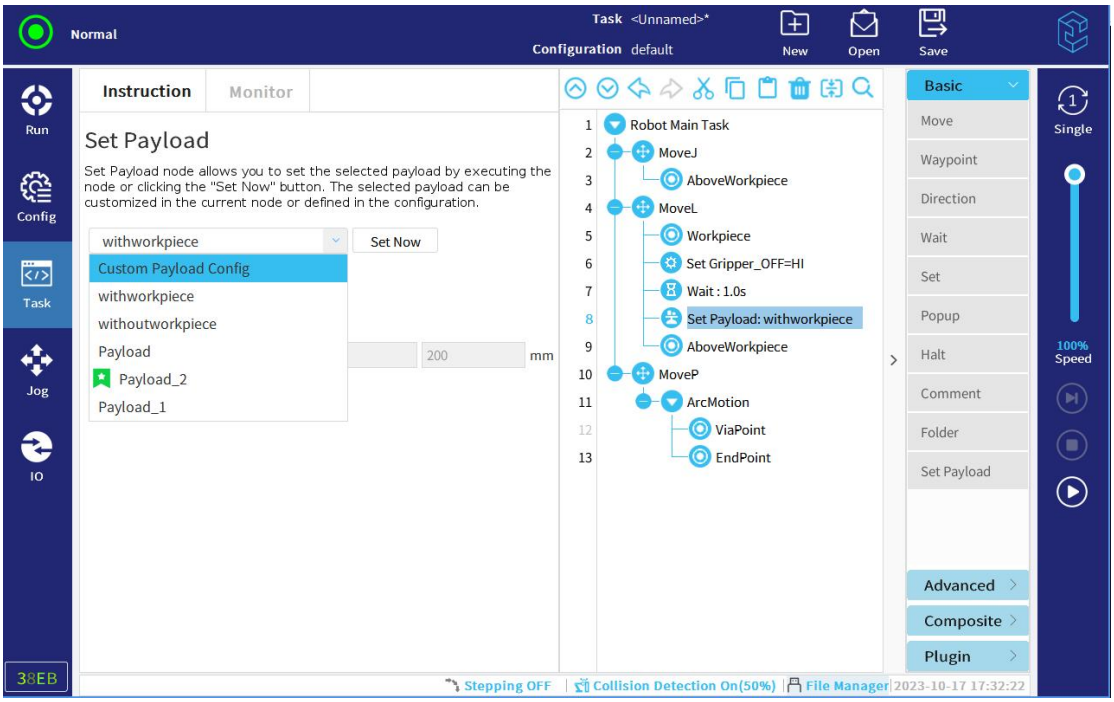


Figure 11-1

When selecting the custom payload config, the user can enter the mass and center of gravity.

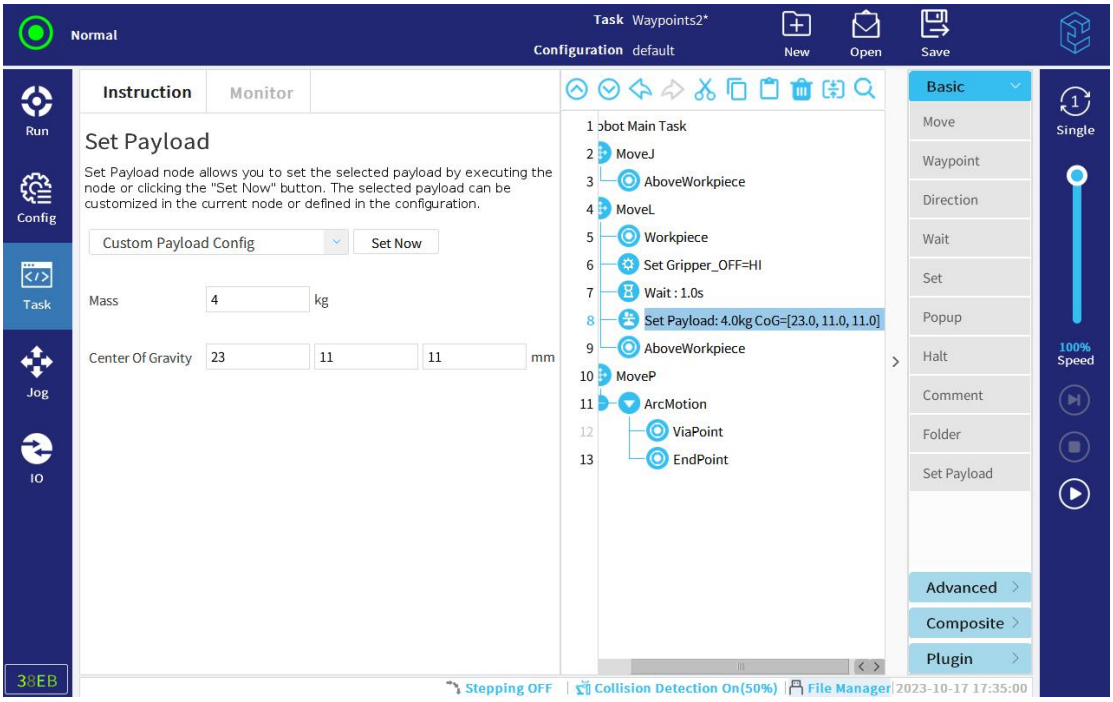


Figure 11-2

The center of gravity refers to the distance of the payload centroid from the center of the flange in the x, y and z direction. The default direction of the flange is as shown in the following figure.

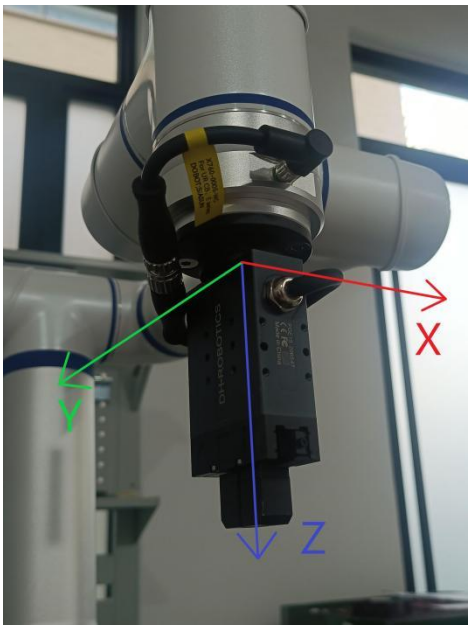


Figure 11-3

Click the “Set Now” button and the custom settings in the current node will take effect immediately.

12 Logic Control Commands

12.1 If Command

The If and If...Else sentence can be used to execute the corresponding subnodes in accordance with the expression.

If the expression is true, it will execute the subnodes in the if command.

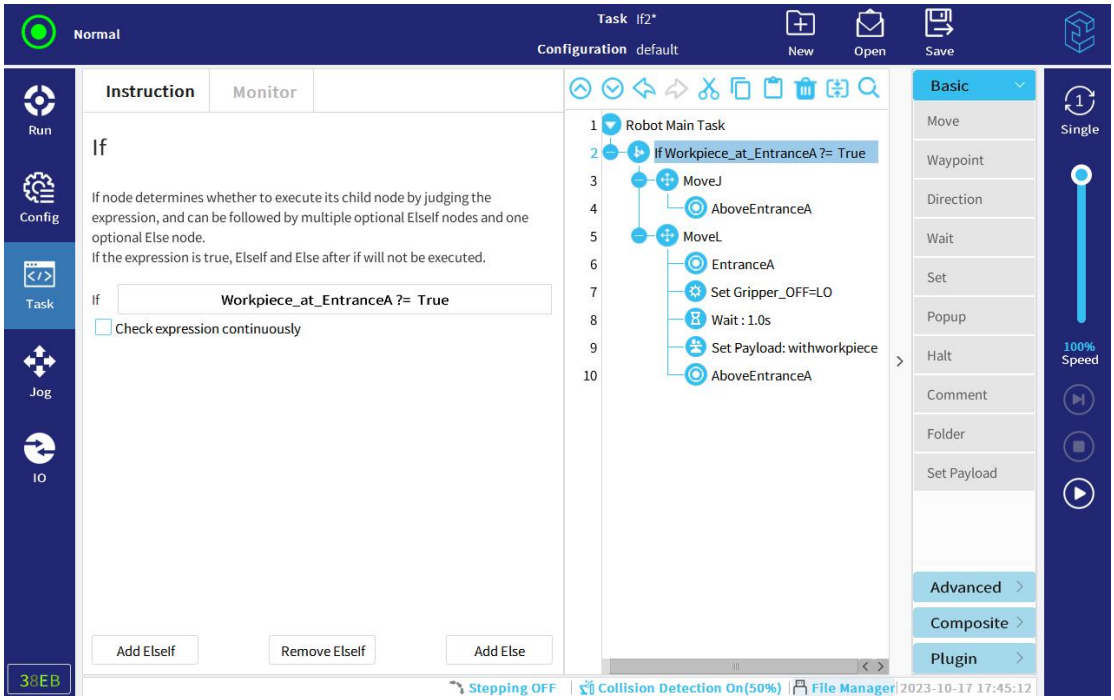


Figure 12-1

The user can add multiple ElseIf nodes and an Else node in an If node. If the conditions in the If node and the ElseIf node are not satisfied, it will execute the subnode in the Else node.

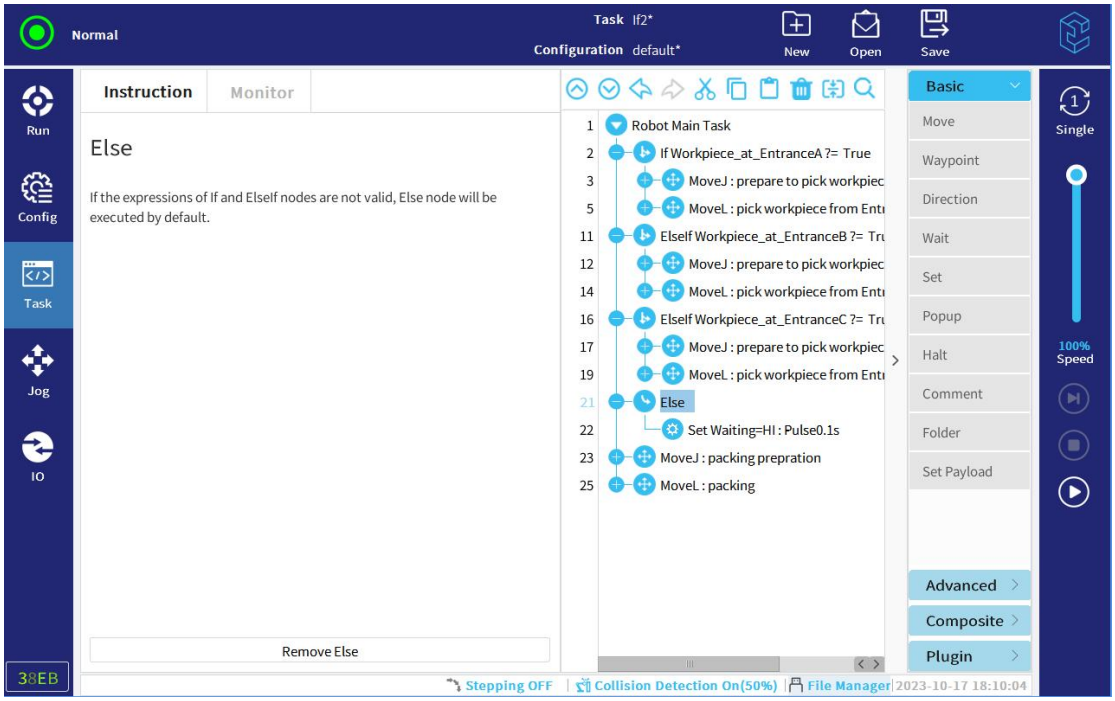


Figure 12-2

If “Check expression continuously” is checked, then the if expression will be checked in the background and see if the conditions are fullfilled, wherever the program runs. If the conditions are not satisfied, it will navigate to the else node and execute the task of the node.

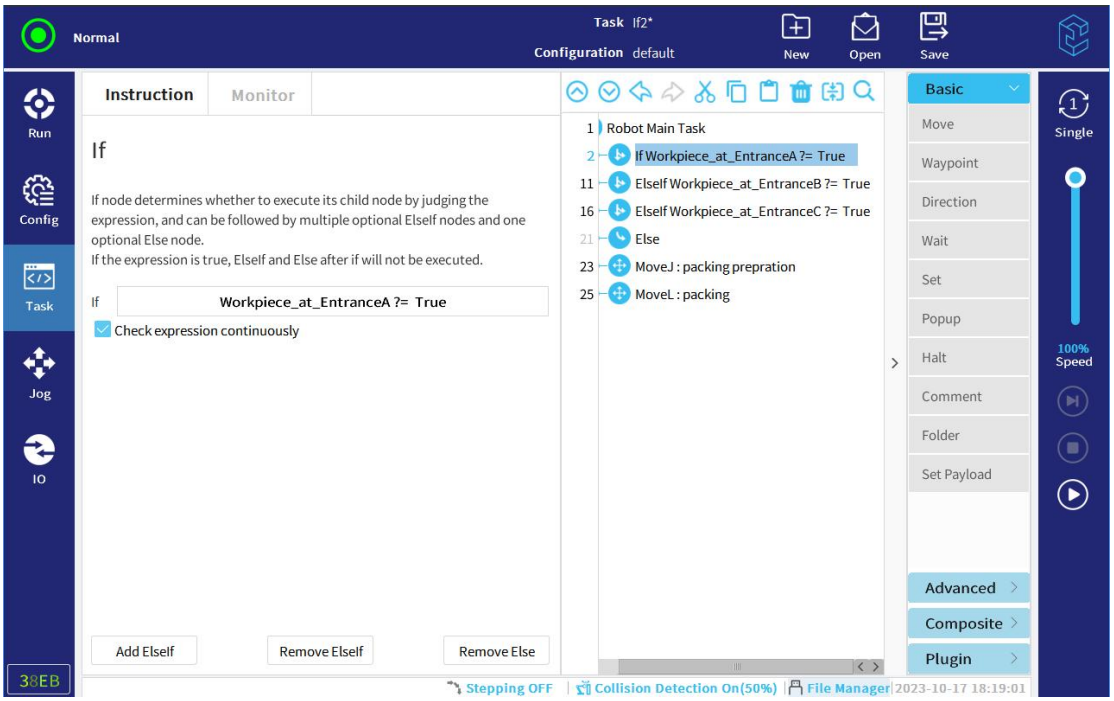


Figure 12-3

12.2 Loop Command

The loop command is used to repeatedly run the subnodes. There are 3 loop modes, i.e. Loop Always, Loop N Times and Loop when the expresssion is True.

When selecting “Loop Always”, the task loops all the time.

When selecting “Loop N Times”, it will specify the number of loops. A loop variable for counting the number of loops will be created. The variable can be used in some special operations. The following figure shows that the number of loops is 3. When the value of the variable ”Loop_1” is set to 0, 1 and 2, it will execute the subnodes of the loop command. After the execution of the loop command, the robot moves back and forth twice in the X axis and then revolves around the current TCP by 90°. Afterwards, it moves back and forth twice again.

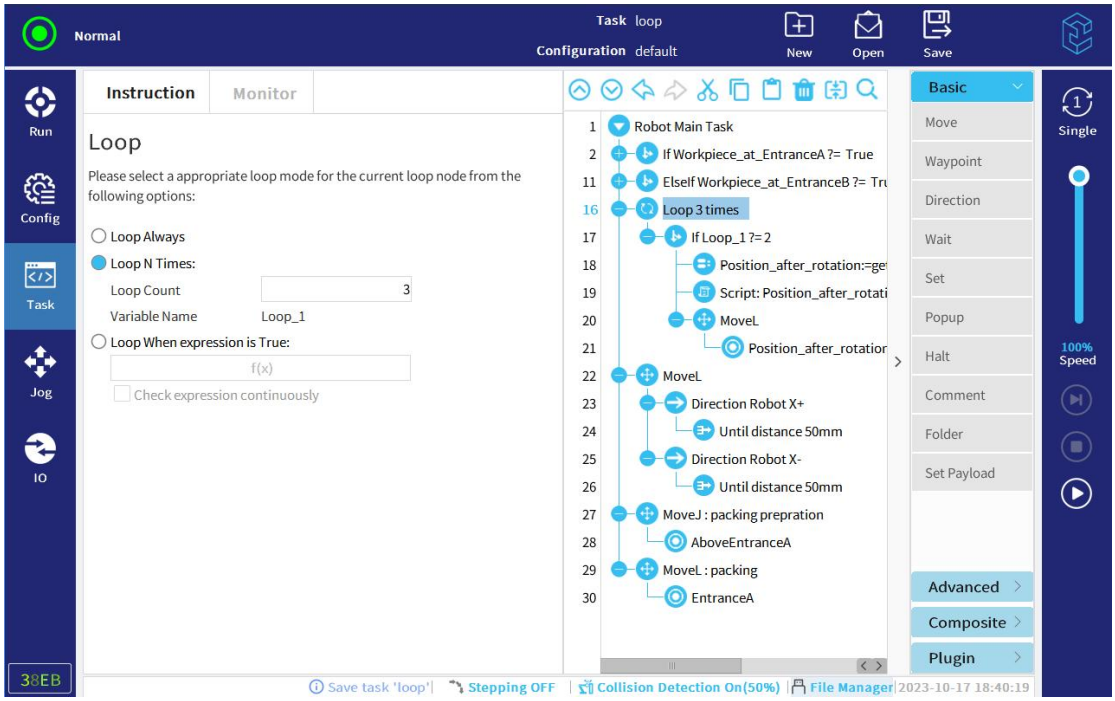


Figure 12-4

When selecting “Loop When expression is True”, the child node will be always looping so long as the expression is true.

Note: When “Check expression continuously” is checked, it will jump out from the loop node as long as the expression is not true in the process of running a node and go to the next node.

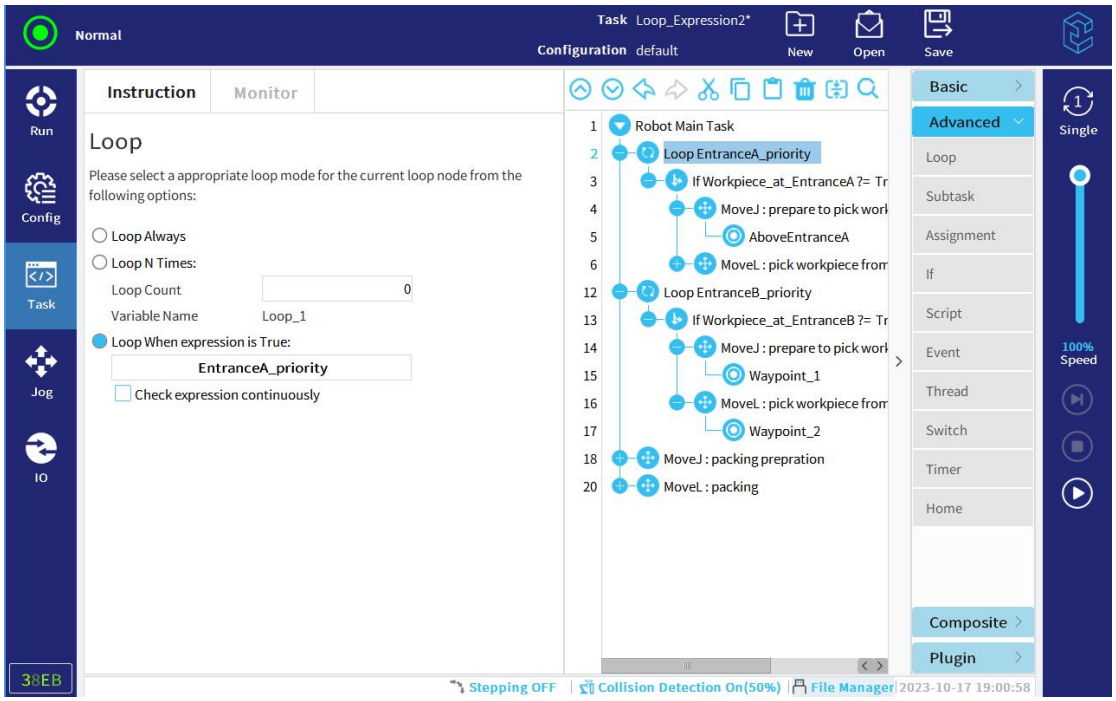


Figure 12-5

12.3 Wait Command

The “Wait” node is used to pause a running task. The paused task will not continue to until the condition of no longer waiting is satisfied. In the “Wait” node, the user can enter the specific waiting time, select from the dropdown list of “Wait for Digital Input” and “Wait for (<An.Input>)”, as well as select True or False from the list of “Wait for” (the expression is true).

When selecting “Wait (time)”, the user can enter the specific waiting time. The unit is s.

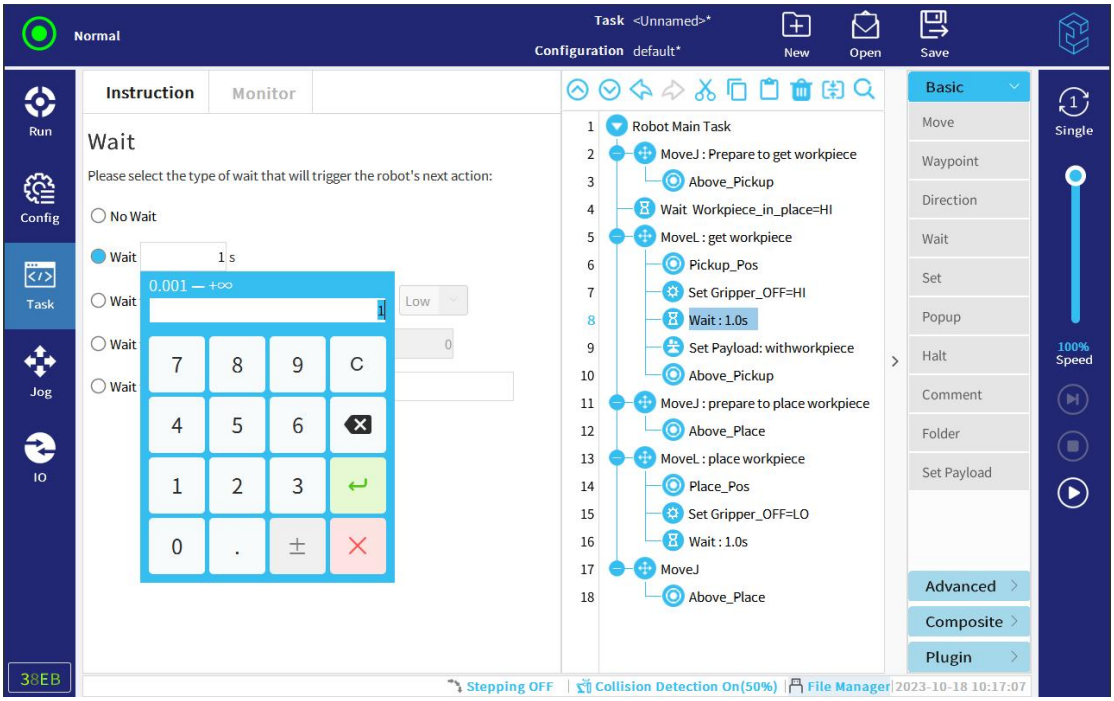


Figure 12-6

When selecting “Wait for Digital Input”, the user can select “High” or “Low” signal as the condition for ending the waiting.

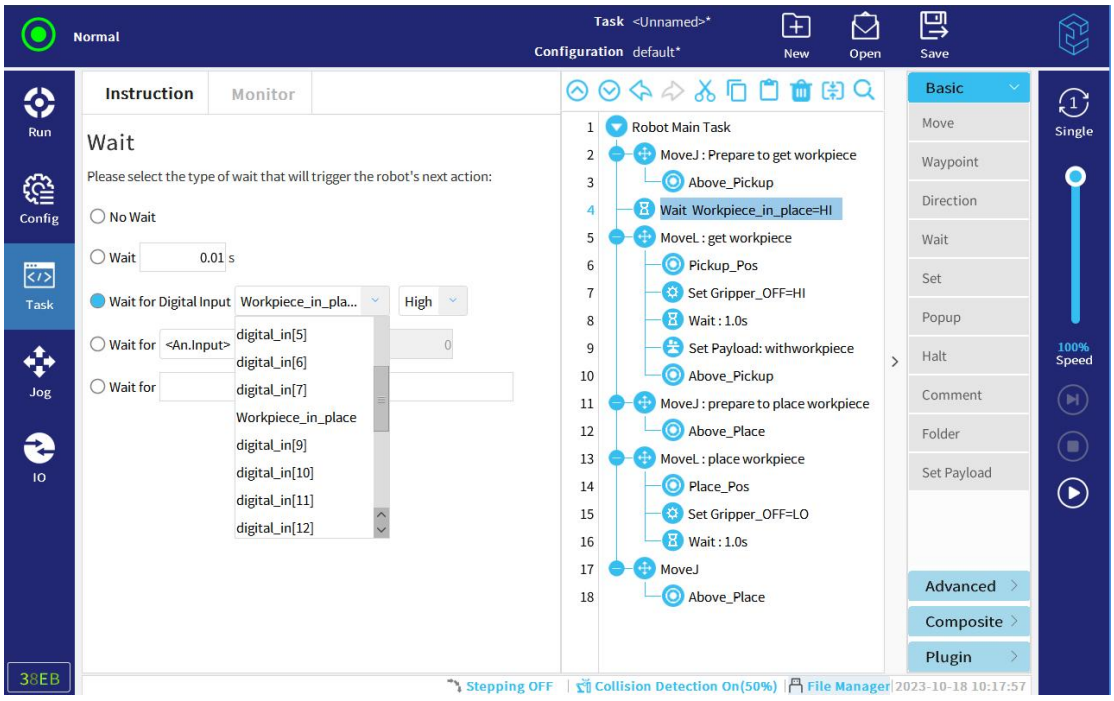


Figure 12-7

When selecting “Wait for (<An.Input>)”, the user can determine whether to keep waiting based on the fact that the signal is greater or smaller than a value. What the option is, current or voltage, depends on the configuration of the signal. To set the signal type, navigate to the IO tab or click **Config -> Tool IO**.

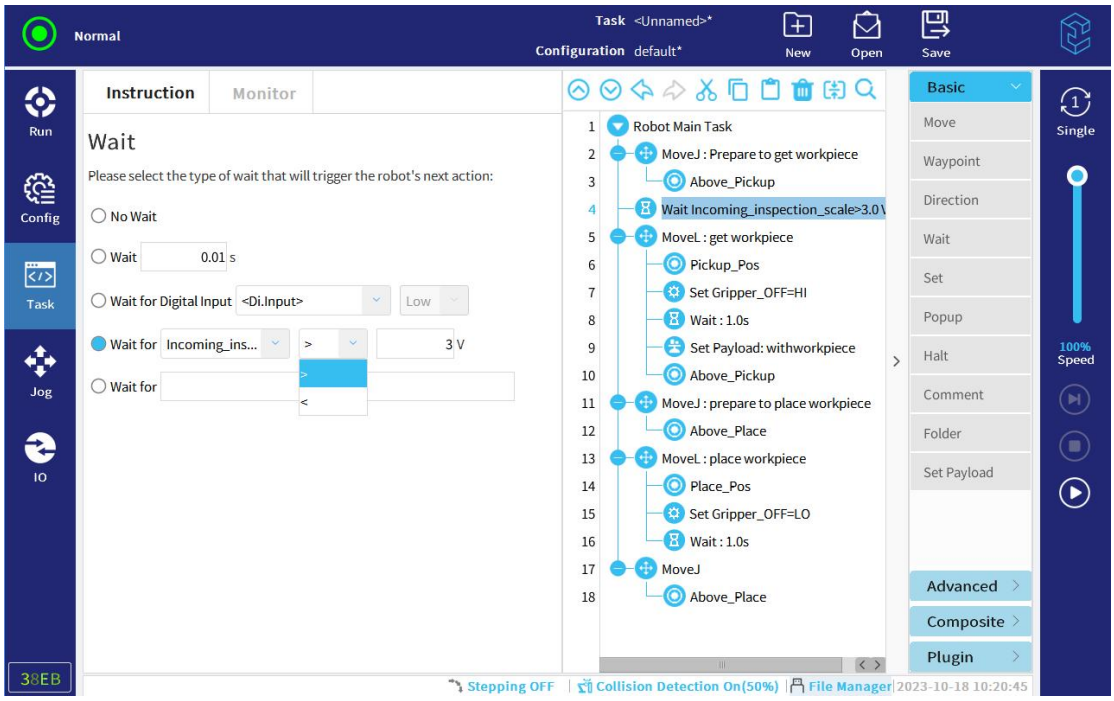


Figure 12-8

The user can also set the waiting condition via the expression, as shown in the following figure.

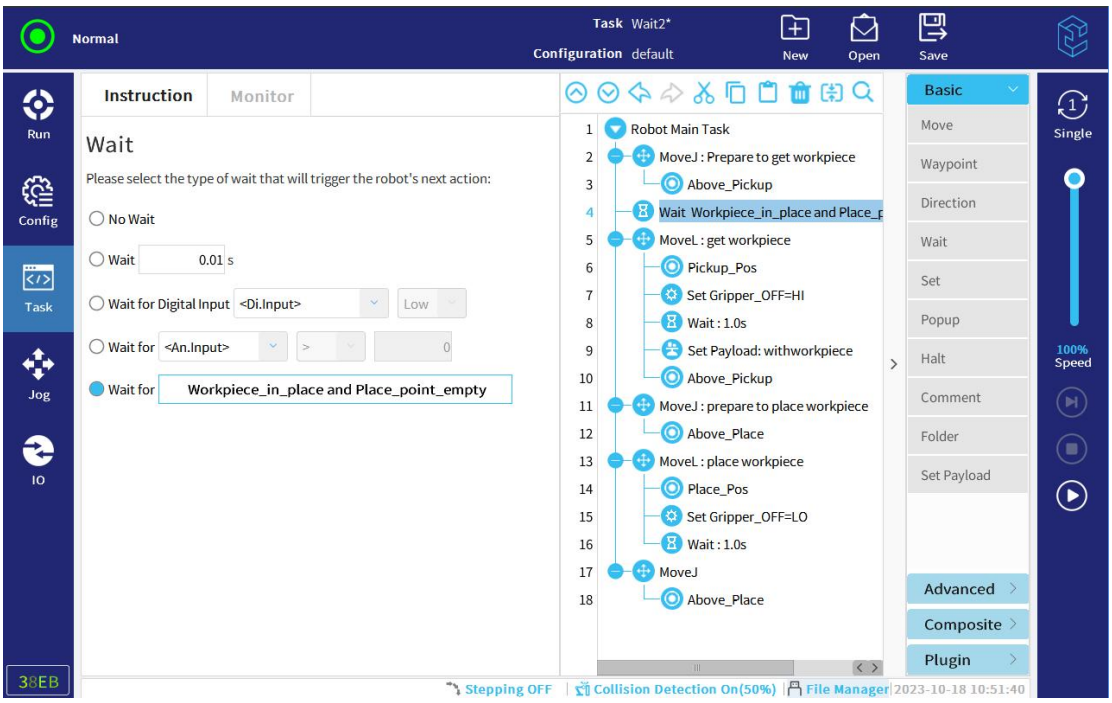


Figure 12-9

13 Halt Command

The halt command is used to stop a running task. A stopped task will run from the first line.

The halt command is usually used together with the conditional instructions. For instance, the task will be automatically stopped when the production times are reached. The task will also be stopped in case that some other equipments work abnormally.

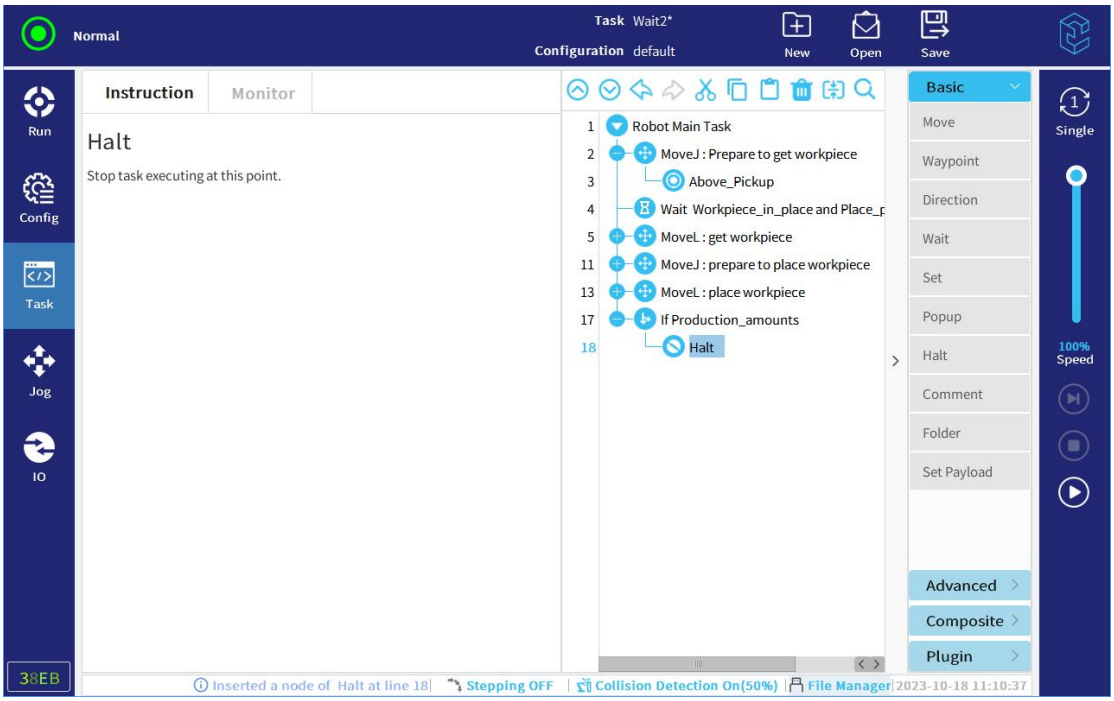


Figure 13-1

14 Popup Command

The popup command is used to pop up a dialog when the program is running and halt a task. The user can press the ”OK” button or the ”Stop Task” button with the teach pendant. The user can also make the selections through the IPC.

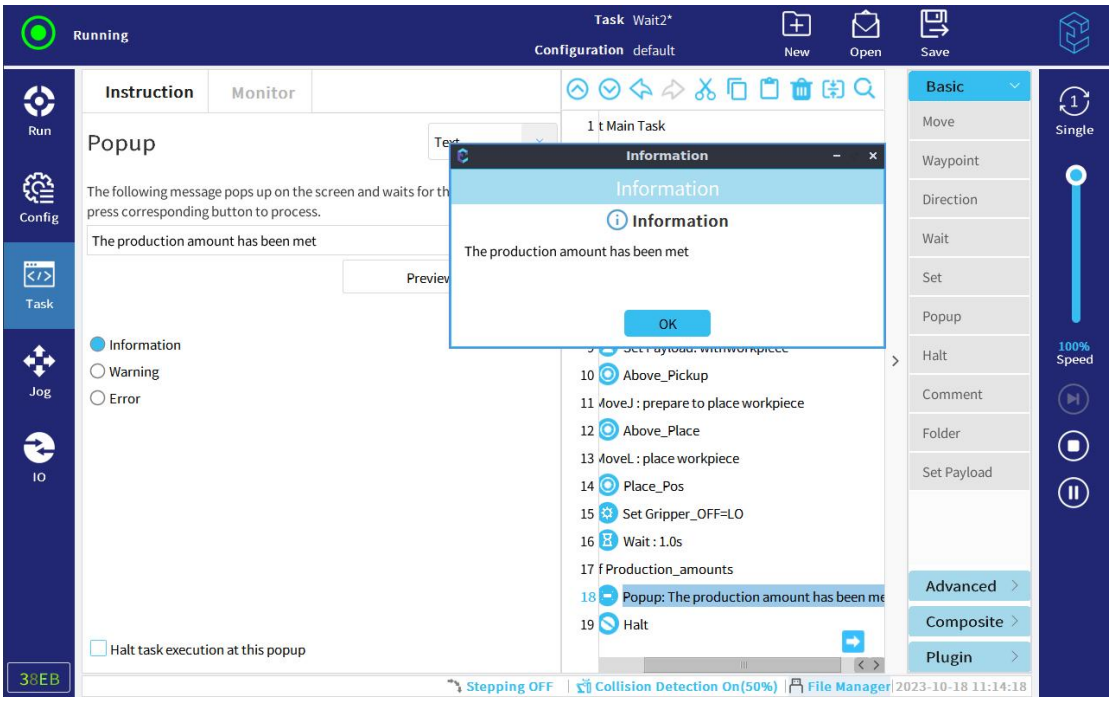


Figure 14-1

The user can select “text” or “variable” when selecting the popup type.

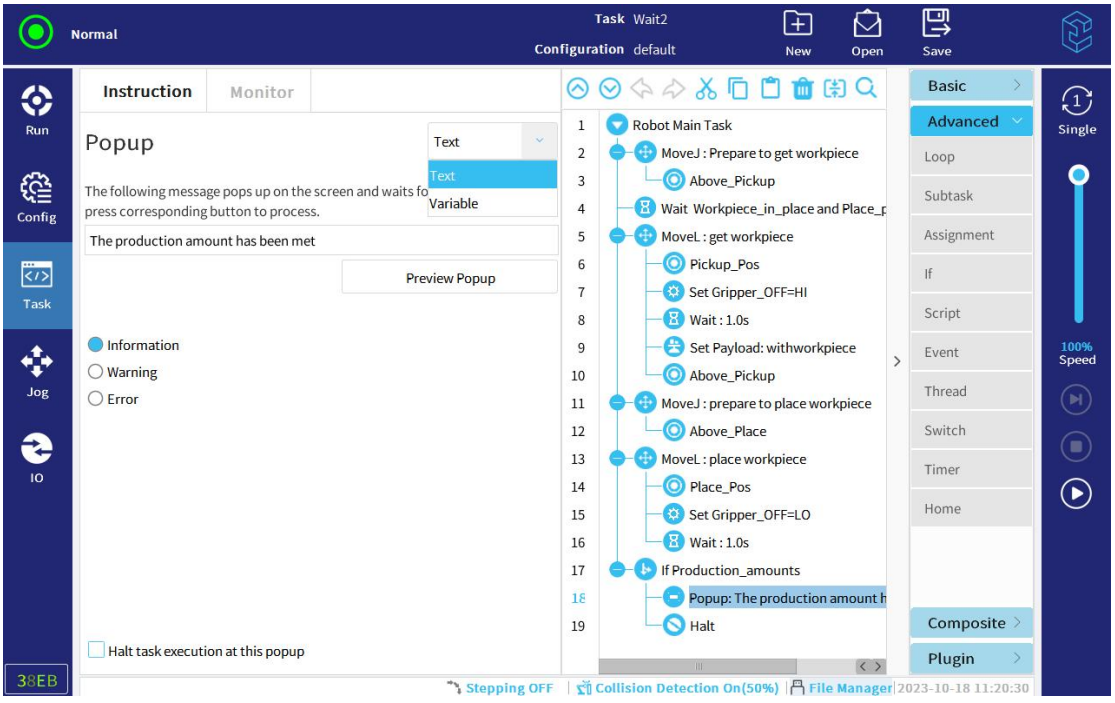


Figure 14-2

When selecting “text”, the texts entered before will appear in the popup dialog. When selecting “variable”, the current value of the selected variable will appear in the popup dialog.

The user can select “Information”, “Warning” or “Error”, each of which stands for an information dialog, a warning dialog and an error dialog.

15 Program Debugging

15.1 Single-step Debugging

The function allows the user to run a node in the single-step mode, that is, executing only a line of the command once.

Step: The currently running task can be paused by clicking the “Pause” button in the right side of the menu. After the pause, the single-step button will be highlighted. Click the button and the node will run. The user can switch over between the running states in the single-step mode and non single-step mode. This will not stop the proceedings of the current running task.

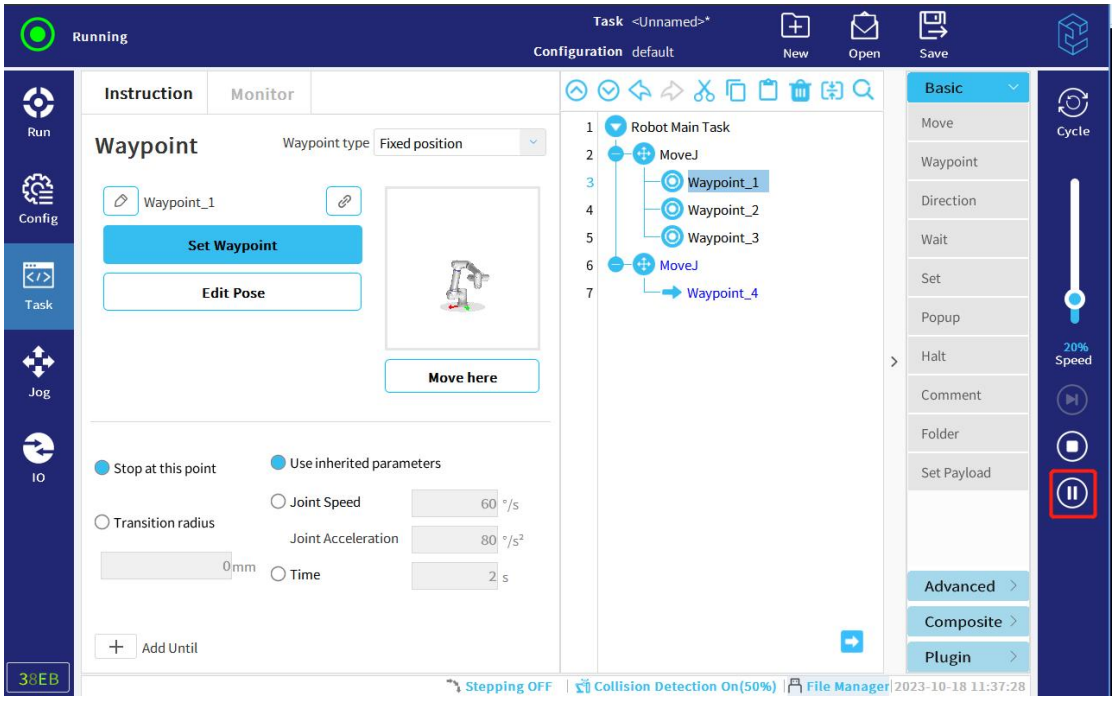


Figure 15-1

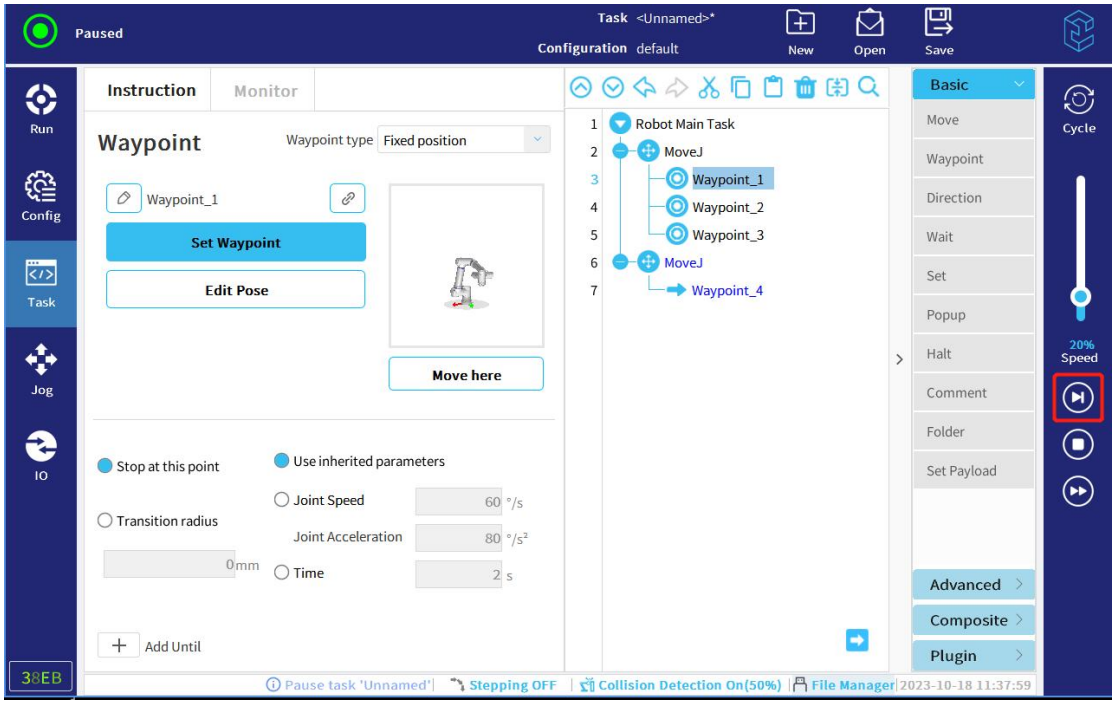


Figure 15-2

Note: The function can be used together with the breakpoint. For the steps of setting the breakpoint, see the follow-up chapter.

15.2 Breakpoint Debugging

The breakpoint is composed of a red spot and a red line. The red spot means that the program will stop at the point. The red line indicates when the function will be triggered. If the red line is above the red spot, it indicates that the function will be triggered before running the node. If the red line is below the red spot (e.g. waypoint), it indicates that the function will be triggered before running the next node. When the red spot turns dark, it indicates that the function has already been triggered and the task is paused.

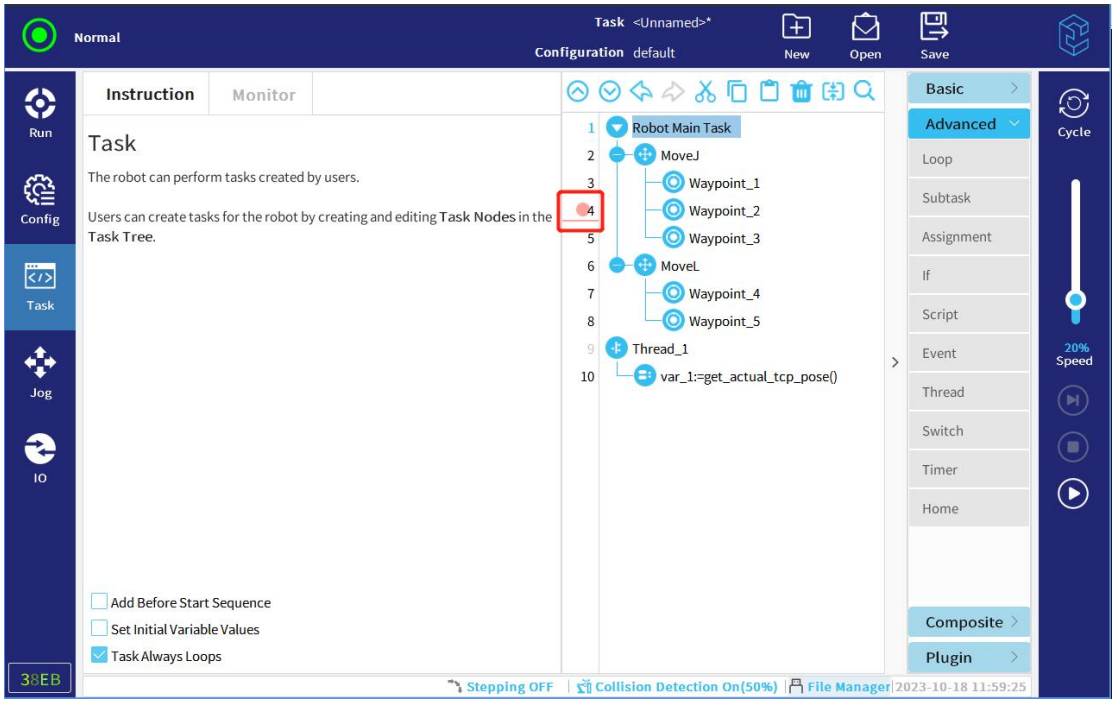


Figure 15-3

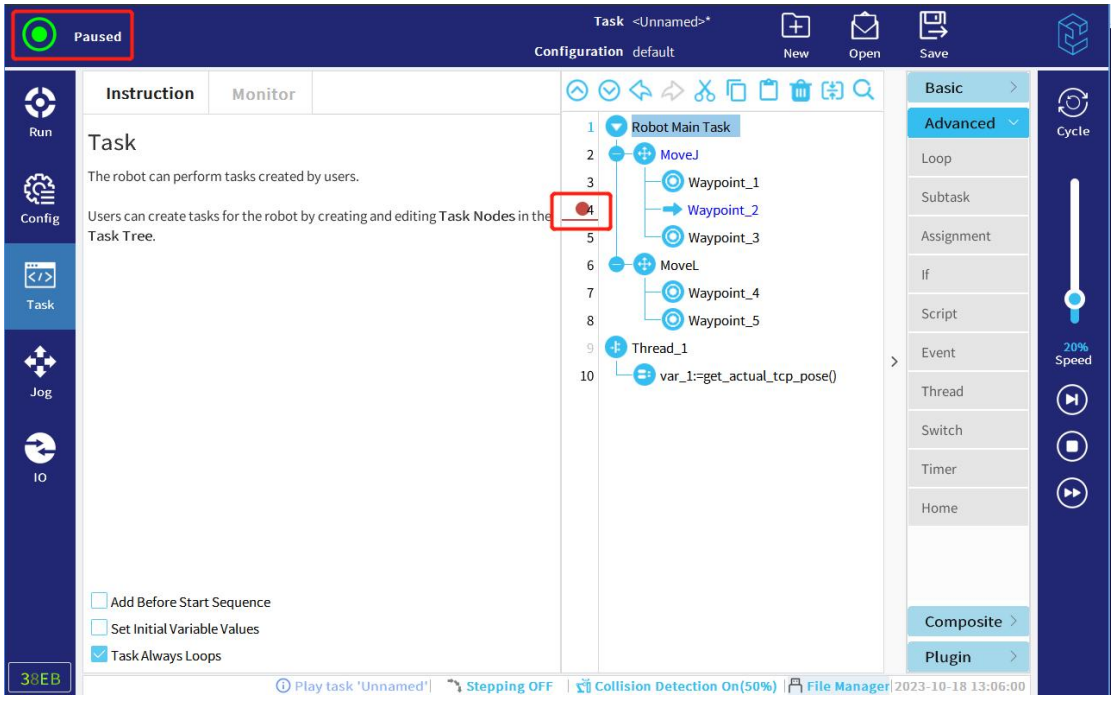


Figure 15-4

The breakpoint can be set by clicking the line number of the node. To cancel the breakpoint, click the line number again. The breakpoint can be set and deleted dynamically when the task is running. This will take effect when running the same task next time. It is impossible to set the breakpoints in some nodes, i.e. the compressed node, the place-holder node, the thread node and other nodes in the plugin that do not allow the users to insert from the command line (e.g.

the layer node and the layout node in the palletizing). If such nodes are selected, the line number goes to light blue. If they are not selected, the line number is marked as light grey.

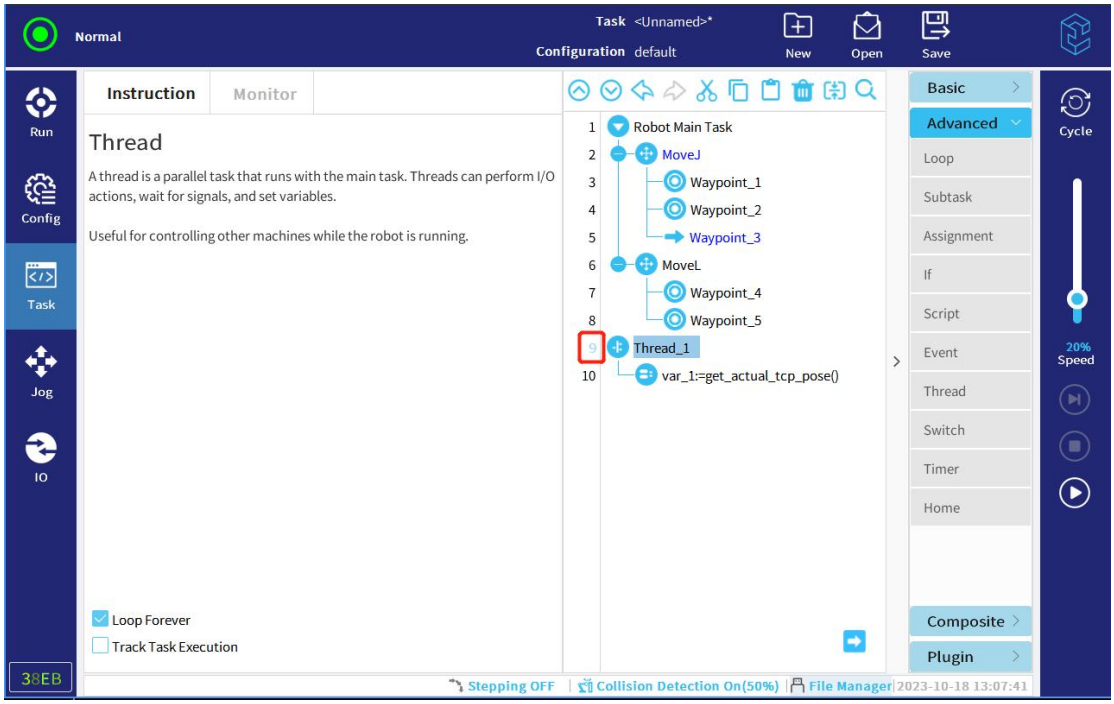


Figure 15-5

15.3 Cyclic Mode

To run the program, click 

or 

.

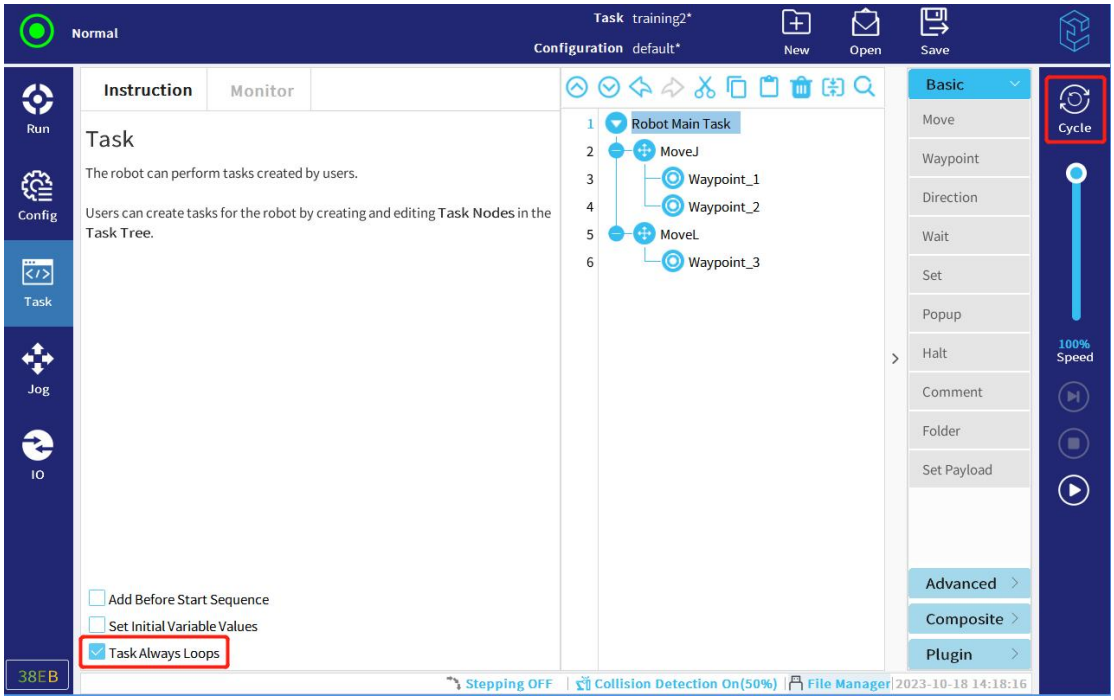


Figure 15-6

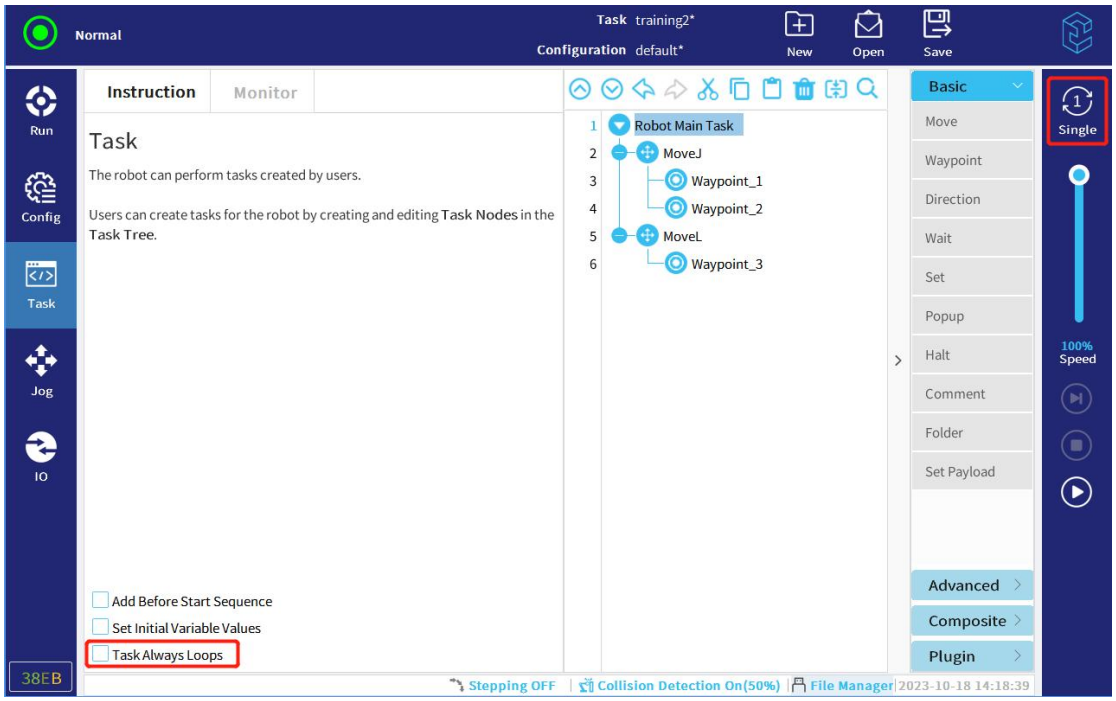


Figure 15-7

16 PC & Robot Connection

16.1 File Transfer

The CS series robot supports for the file transfer and download with the SFTP software. Please make sure that the robot and the PC are in the same LAN network and the connection is good.

Step:

FileZilla:

1. Fill in the IP address (of a robot to be connected) in the Host, as shown in the following figure
2. Username: root
3. Password: elibot
4. Port: 22
5. Click “Quickconnect”
6. Note that the remote site shall be set to /home/elite/user/program after it is successfully connected.
7. The local site is in the left and the robot is in the right. Select the required file and click “Upload” or “Download”.

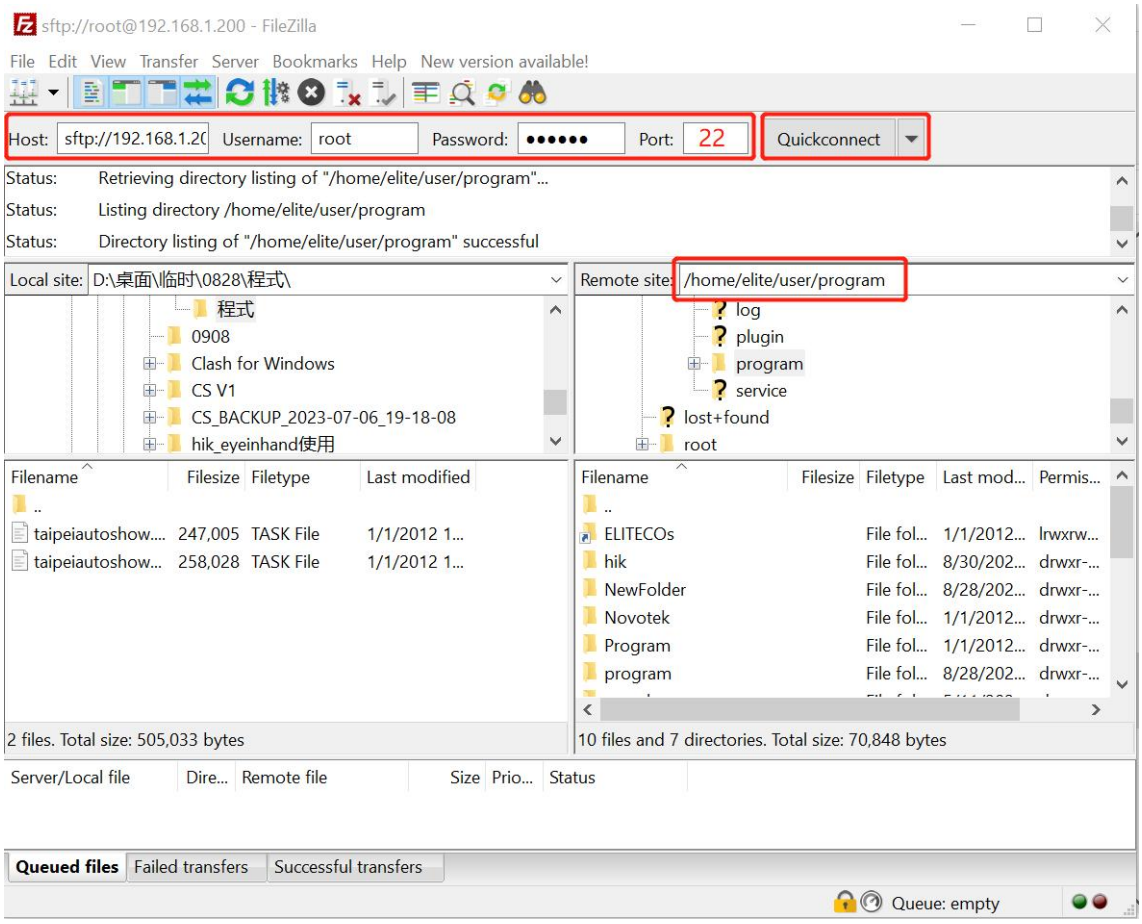


Figure 16-1

Visual Studio Code:

1. Create a file folder named “WorkSpace” locally and right-click “Open via vscode”. If there is no such an option, please reinstall the vscode and select the option of “Right click to open”.

2. Click “Extensions”, search for the SFTP plugin and install it.

3. Press Ctrl+Shift+P to display the configuration window, i.e. SFTP: config, and enter the following texts:

```
{
    "name": "My Server",
    "host": "robot IP address ",
    "protocol": "sftp",
    "port": 22,
    "username": "root",
```

"password": "elibot",

"remotePath": "/home/elite/user/program",

"uploadOnSave": true

}

4. Make sure that a SFTP configuration file folder exists in the WorkSpace file folder.

Reference: https://bbs.elibot.cn/forum/detail/topic/291.html

16.2 VNC Viewer

It is used to monitor the teach pendant interface of the robot and the user can perform some basic operations with the VNC viewer

Step:

1. Open the VNC viewer
2. Click “File”> “New connection” in the top left corner
3. Enter a VNC Server address and click “OK”. Then the figure with the IP address will appear. Double click the figure and a popop dialog will appear. Click “Continue” in the popup dialog and it can monitor the teach pendant interface of the robot without entering a password.

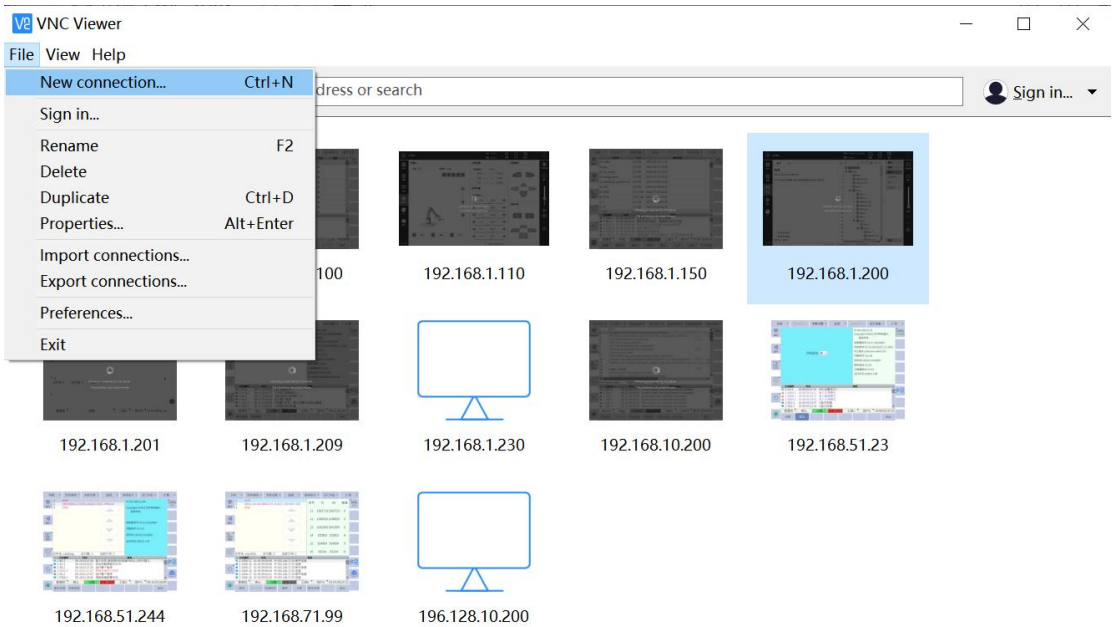


Figure 16-2

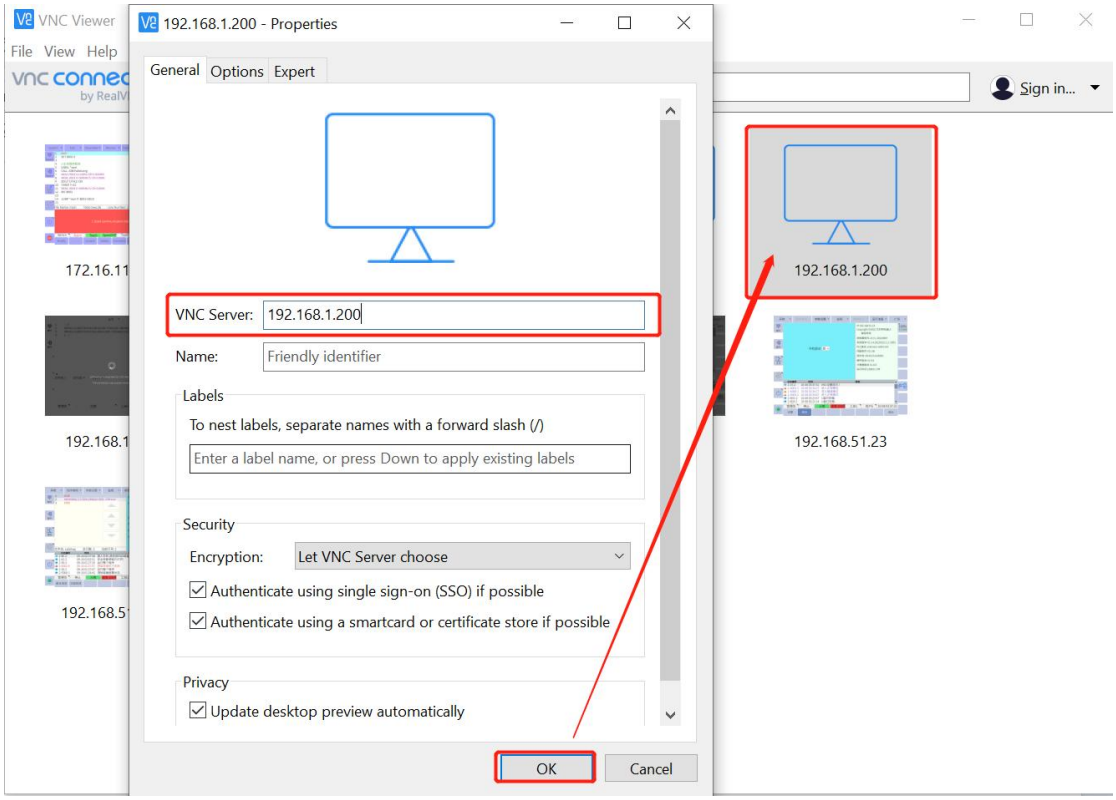


Figure 16-3

17 File Backup & Restoration via a USB flash disk

17.1 Backup

The user can back up the tasks and configuration files to the USB flash disk.

Step:

1. Click the Elite logo in the top right corner.
2. Click **Settings -> System -> Backup**
3. Click ”Select Path”, switch over to the USB flash disk and select the storage path where the files are saved. A file folder can be created and renamed.
4. Click ”Start Backup” and wait for a while patiently. Once the backup is finished, it will prompt ”EliRobot backup done” and ”EliServer backup done”.

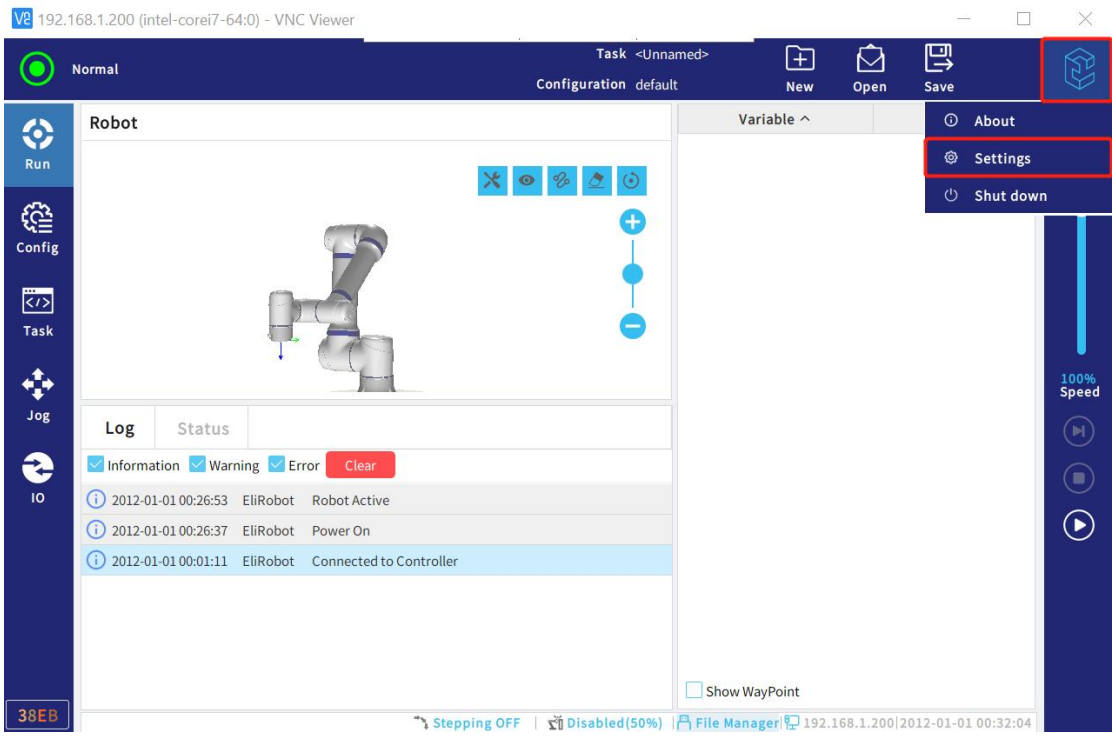


Figure 17-1

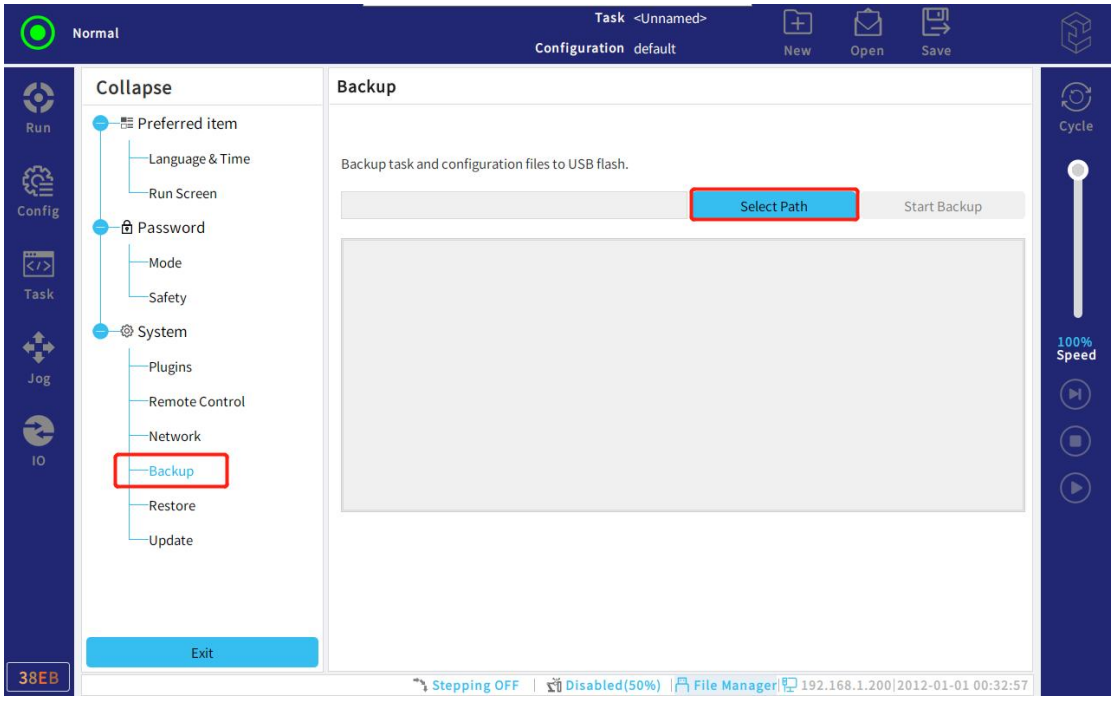


Figure 17-2

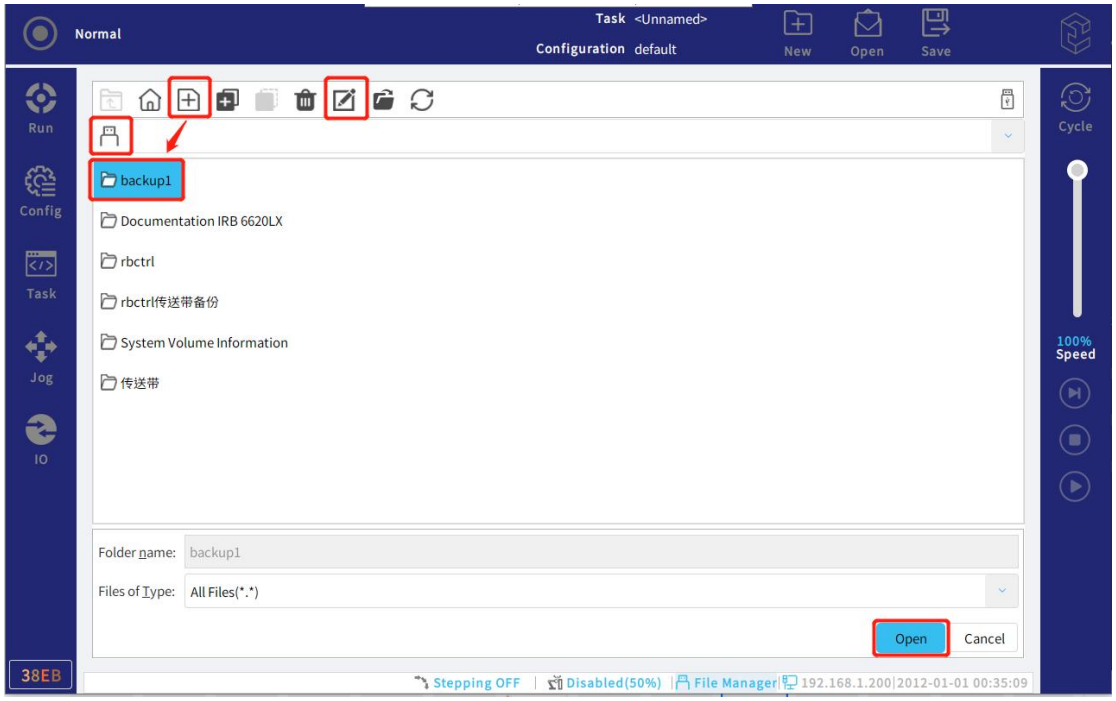


Figure 17-3

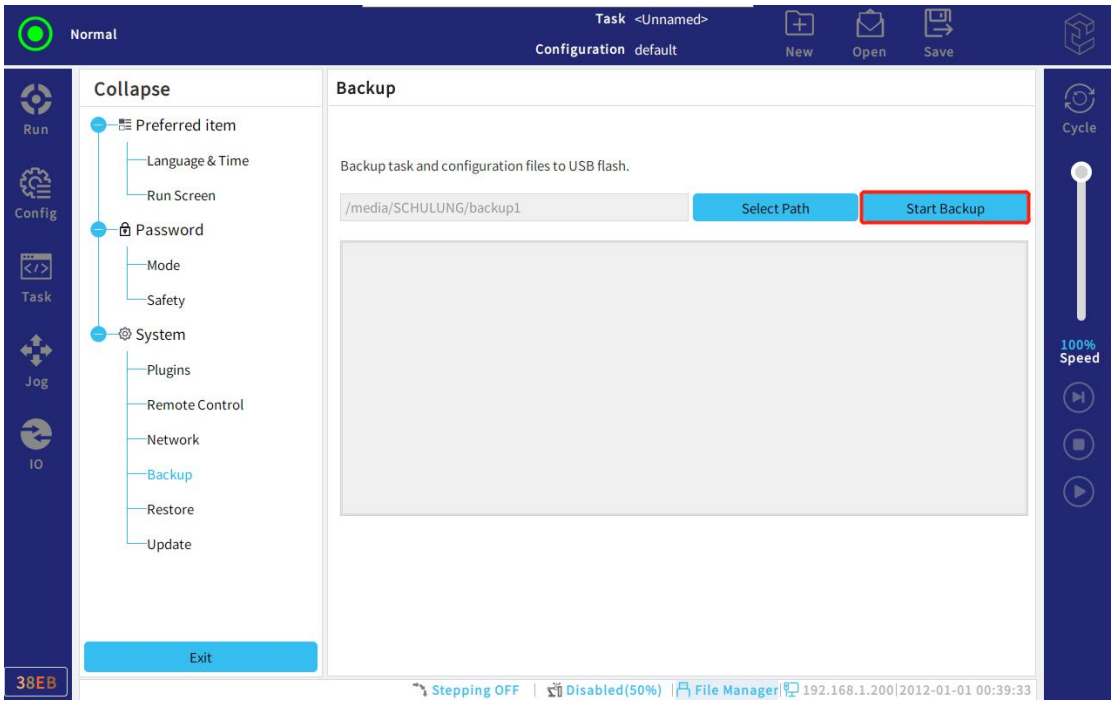


Figure 17-4

17.2 Restore

The user can restore the tasks and configuration files in the USB flash disk to the system.

Step:

1. Copy the file Backup.zip into the USB flash disk
2. Click the Elite logo in the top right corner
3. Click **Settings -> System -> Restore**
4. Click “Select file” and open the file Backup.zip in the USB flash disk
5. Select “Start Restore”, click ”OK” and wait patiently. After a while, the restoration will be done.

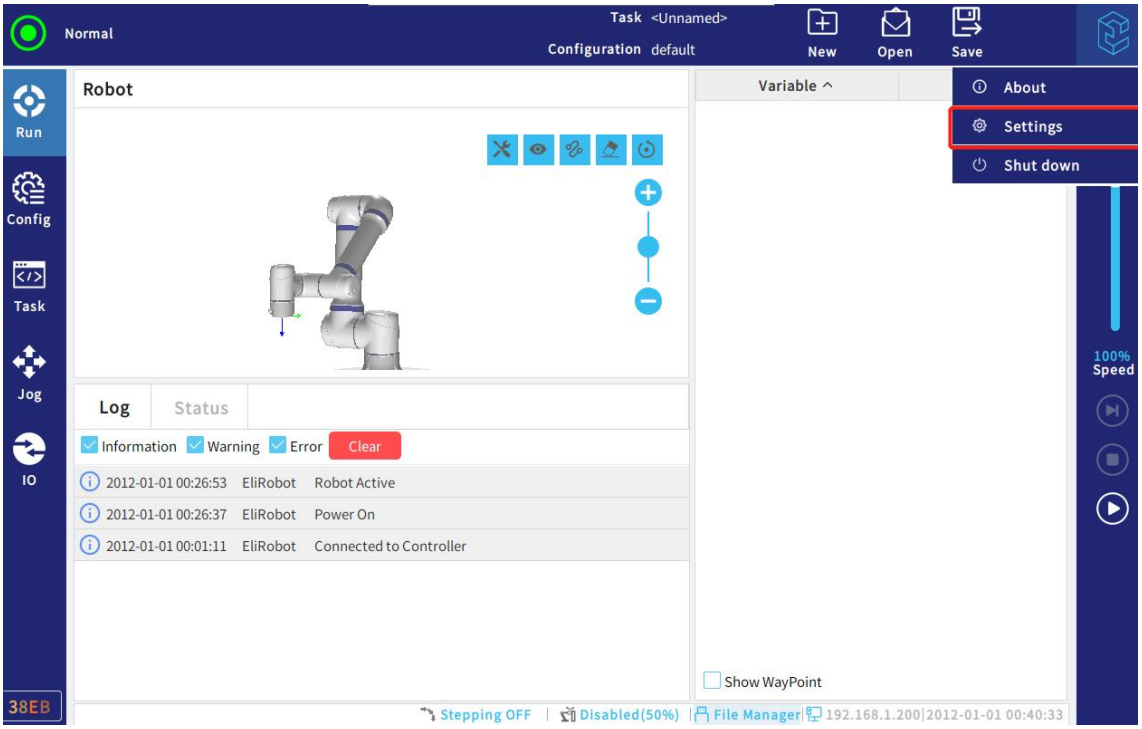


Figure 17-5

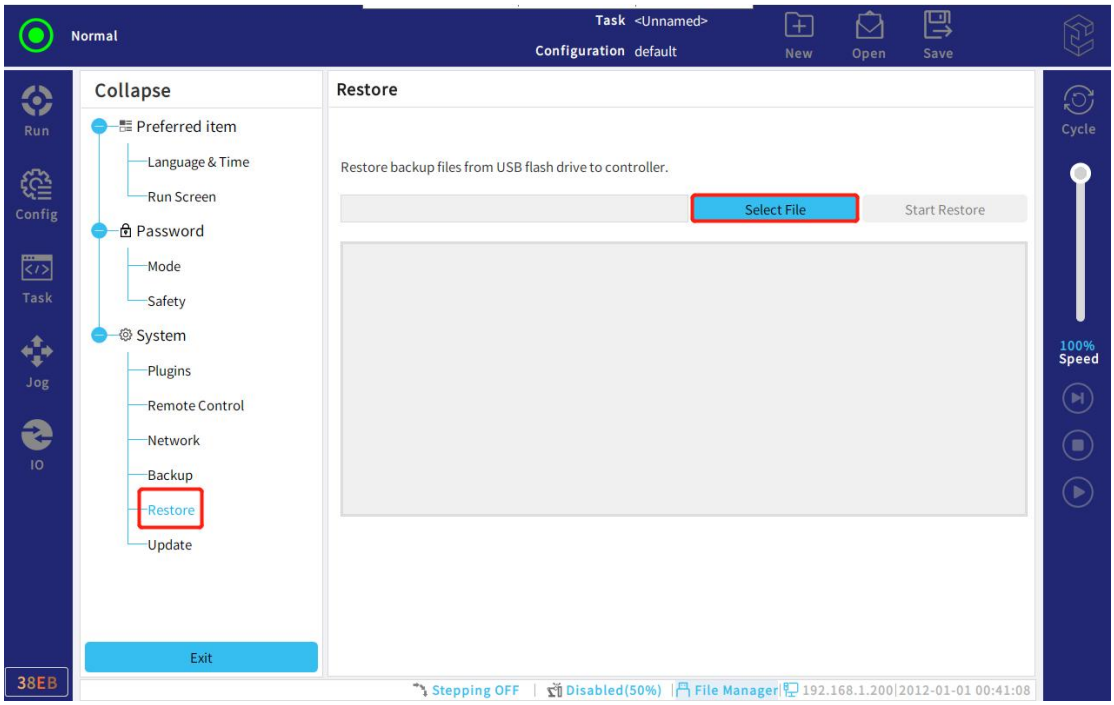


Figure 17-6

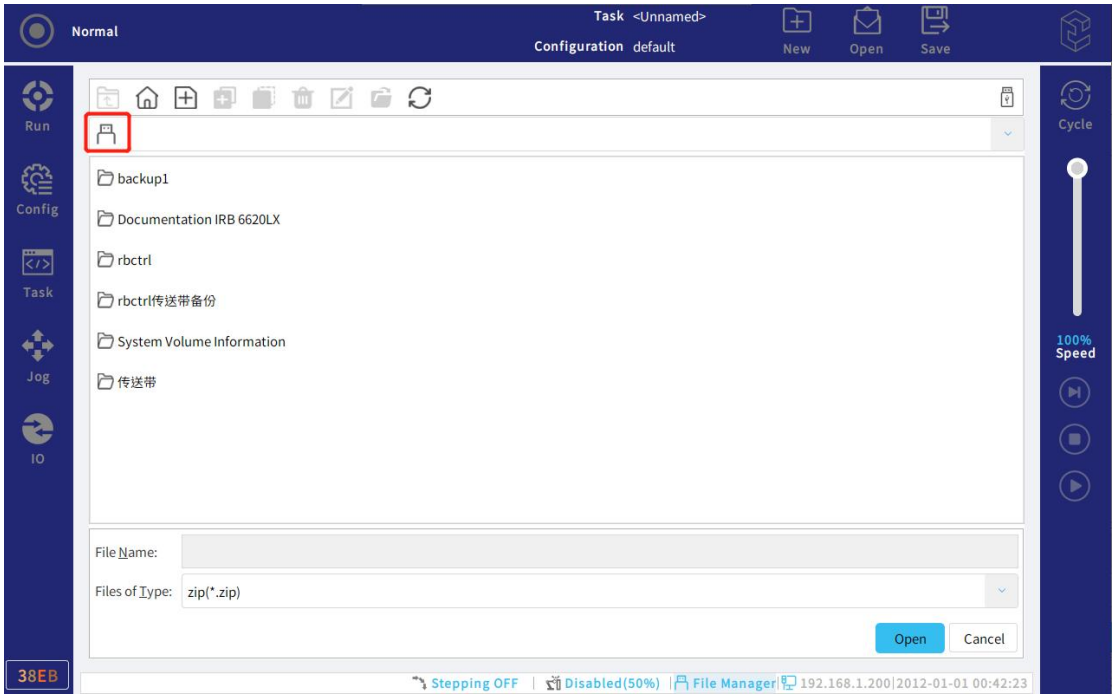


Figure 17-7

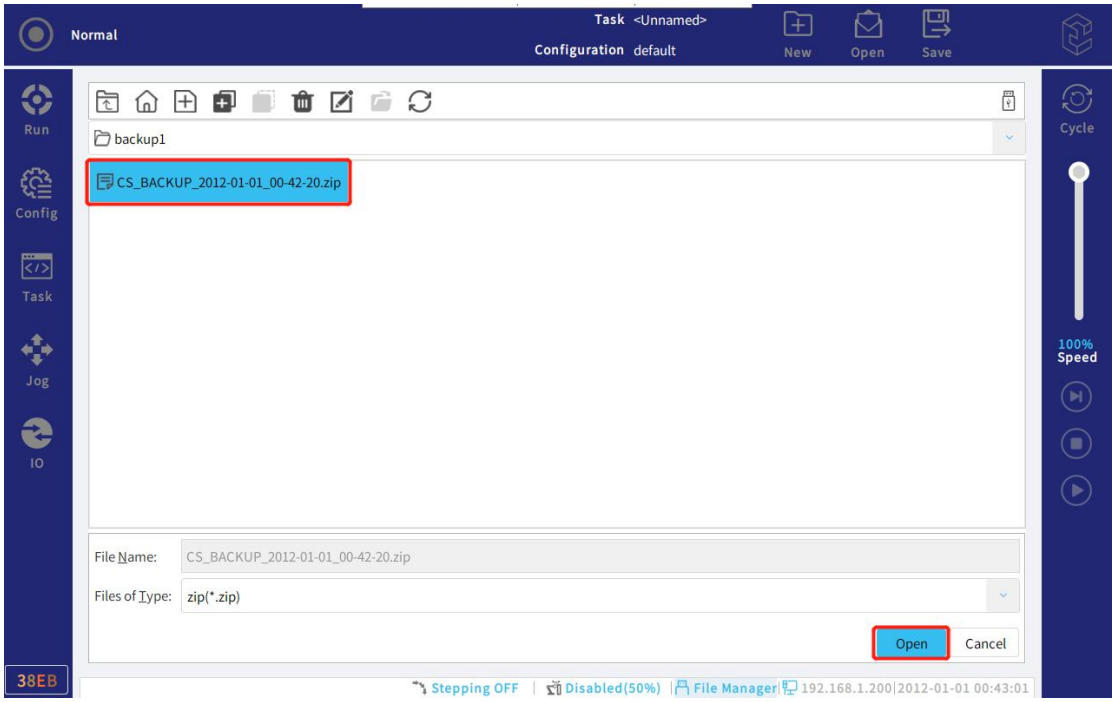


Figure 17-8

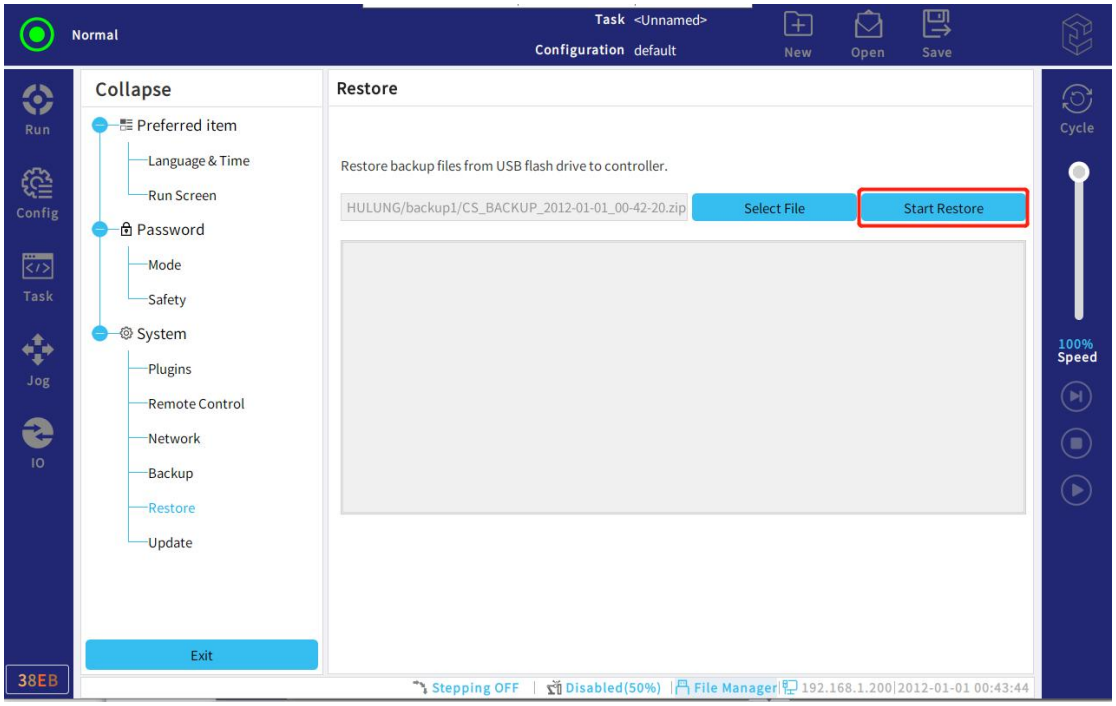


Figure 17-9

18 Robot Update

18.1 System Update

Update the system.

Step:

1 Click the Elite logo in the top right corner of the teach pendant interface, click ”About” and view the robot version.

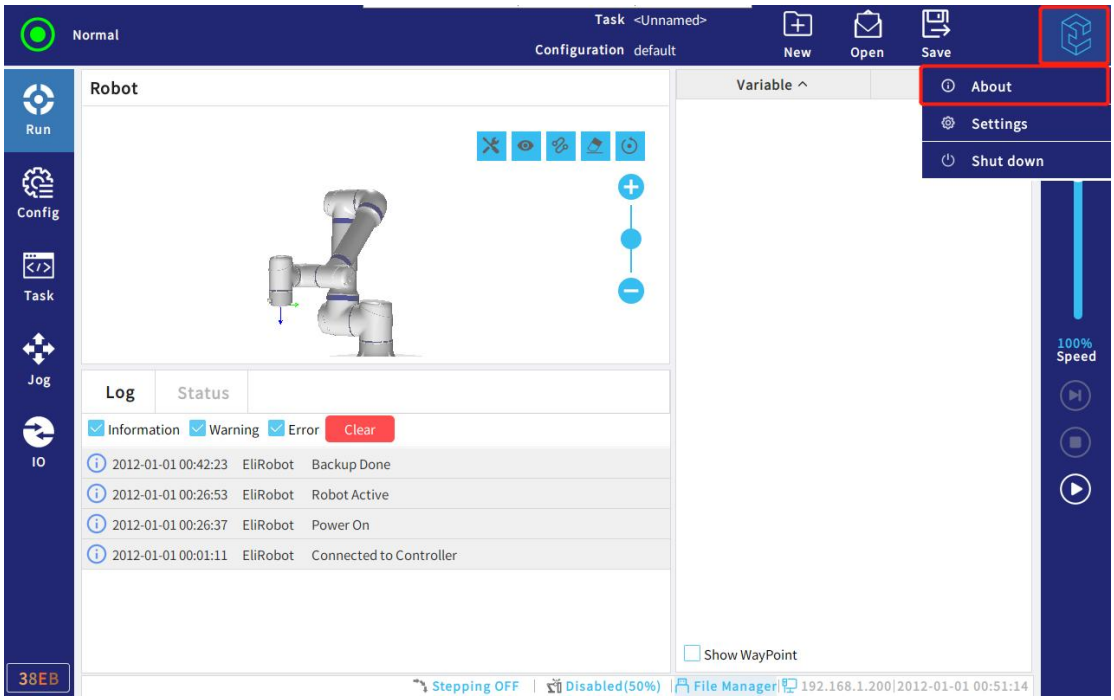


Figure 18-1

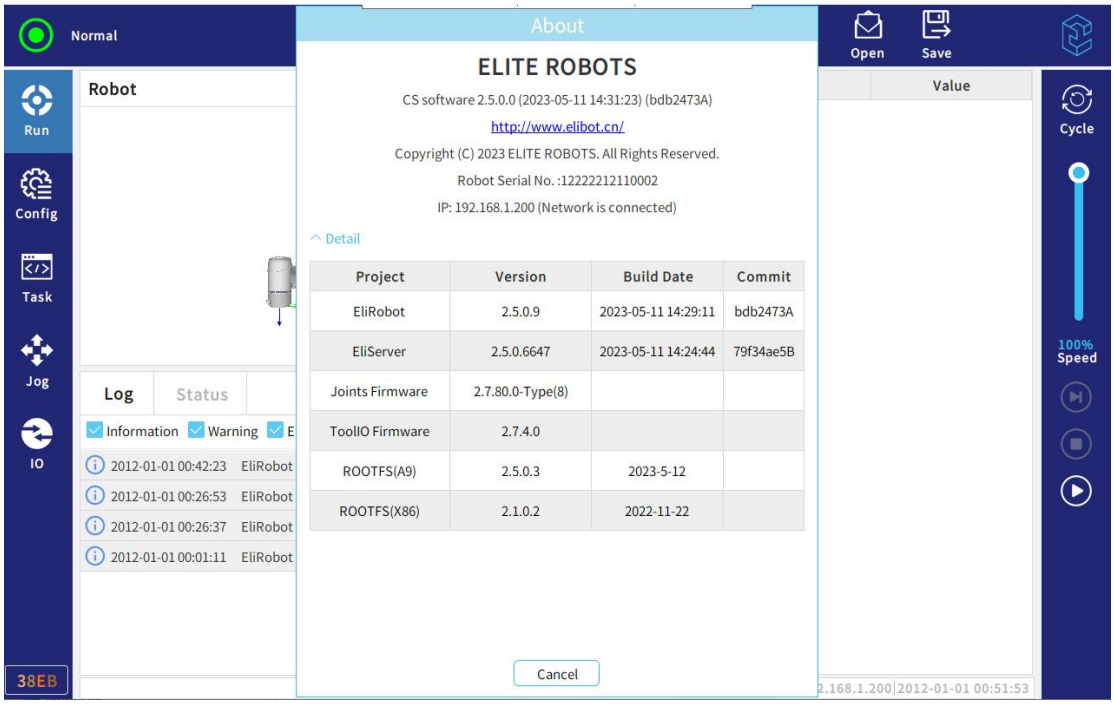


Figure 18-2

2. Copy the update file into the USB flash disk and insert the USB flash disk into the interface

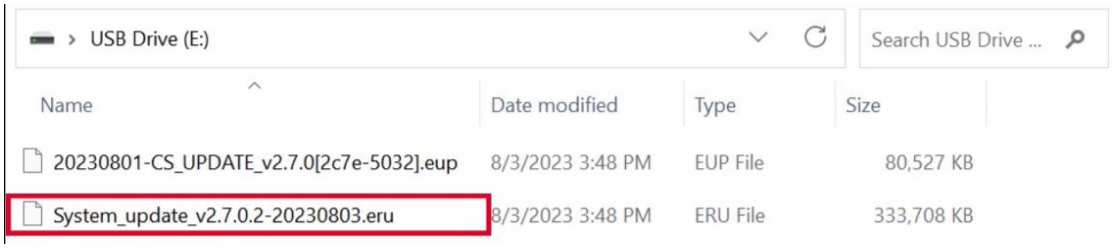


Figure 18-3

3. Select “System”, search for the files with the suffix of ”.eru” and wait patiently.

Note that please update the system first and then the application!!!

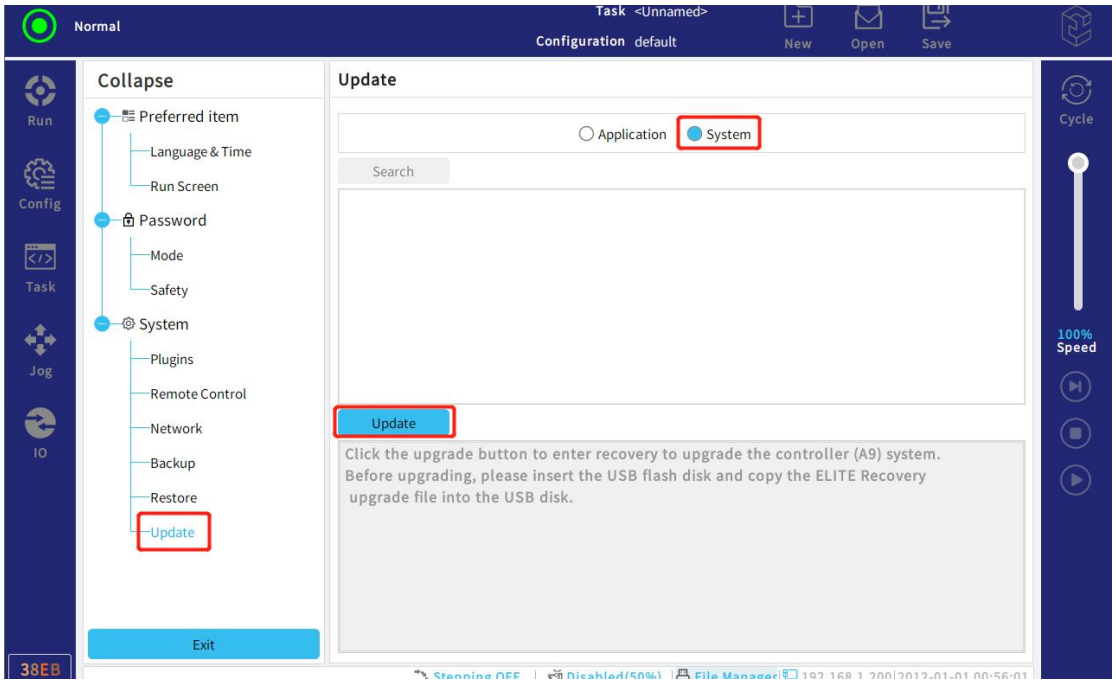


Figure 18-4

Click “Start” when the following figure appears.

Figure 18-5

Click the file package and click “OK”.

Figure 18-6

Please wait patiently and click “OK” to restart the robot once the progress bar is loaded to 100%.

Figure 18-7

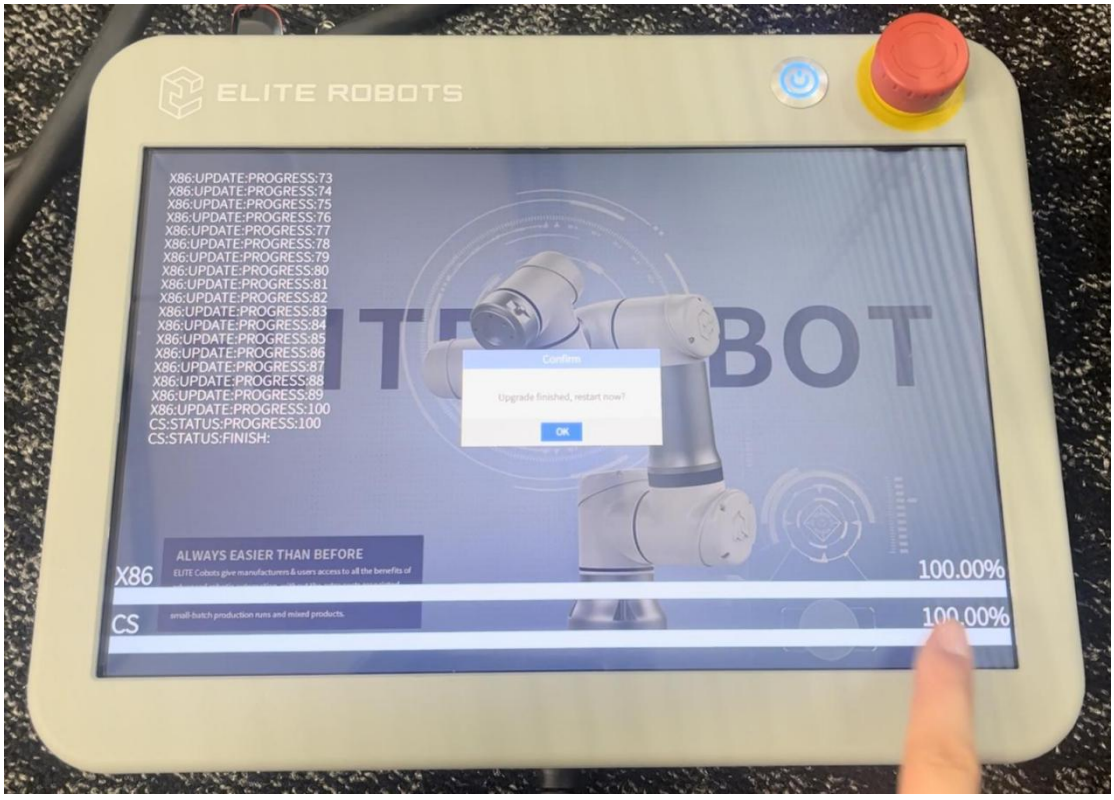


Figure 18-8

18.2 Application Update

The user can update the application, master MCU, slave MCU and drive (IO).

Step:

1. Copy the application upgrade package with the suffix of ".eup" into the USB flash disk and insert the USB flash disk into the interface.

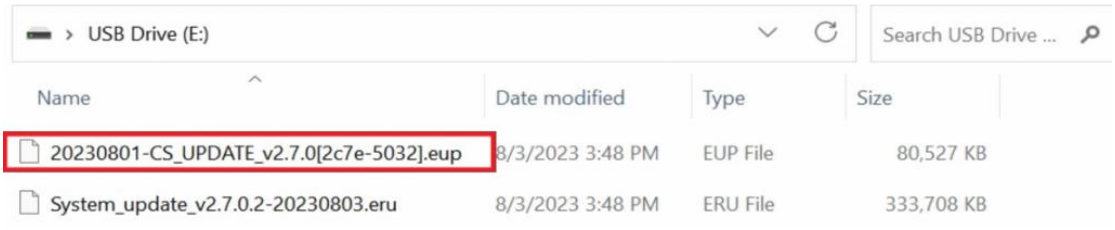


Figure 18-9

2. Click the Elite logo in the top right corner, click **Settings -> System -> Update** and select "Application". Click "Search" and search for an one-stop upgrade package with the suffix of ".eup". Then click "Update". The user can select all modules or select the modules to be updated as required. Click "Ok" and the module(s) will be updated. Please wait patiently. Once the upgrade is done, the robot will automatically restart.

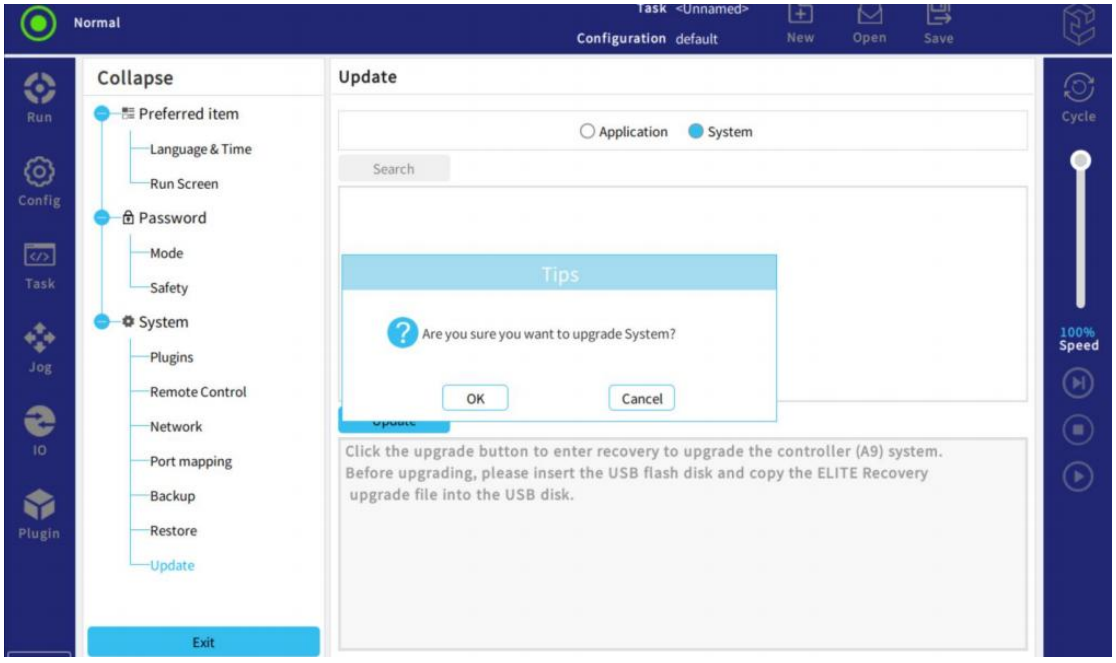


Figure 18-10

19 Safety IO

19.1 Safety IO

Safety devices and equipment must be installed in accordance with safety instructions and risk assessment, see the user manual. The safety stop includes the emergency stop and protective stop. The former is used in the emergency stop devices only for contingency while the latter can be used in all types of the safety protection equipment. The user can also set the other safety IO functions with a configurable IO.

Note:

Do not connect the safety signal to a device with the improper safety level. All safe IO have the redundancy (two independent channels). Before putting the robot into use, be sure to check the safety funtion and test it regularly.

The following figure shows the difference between the emergency stop and the protective stop. After clearing the emergency stop alarm, please re-power on the robot. In case of the protective stop, the robot will not disconnected from power.

	Emergency Stop	Safeguard Stop
The robot stops moving	Yes	Yes
Task execution	Stop	Pause
Robot power supply	Close	Open
Resetting	Manual	Automatic or manual
Operating frequency	Not often used	No more than once per run cy
wntime category (IEC 60204)	1	2
formance level (ISO 13849-1)	PLd	PLd

Figure 19-1

The safety IO associated with the emergency stop is marked out by the red squares, as shown in the following figure.

Figure 19-2

The following figure shows the IO related to the protective stop, marked by the red squares.

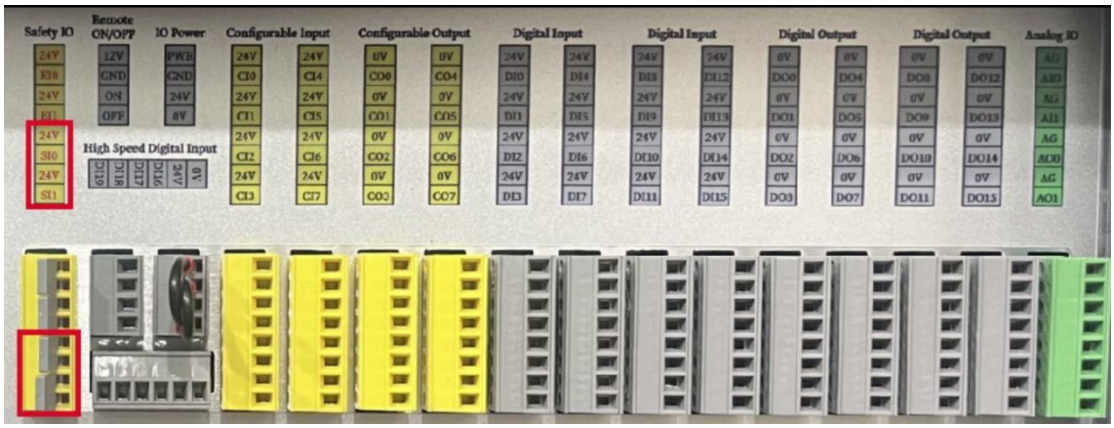


Figure 19-3

The protective stop buttons are connected as shown in the following figure (e.g. security lock).

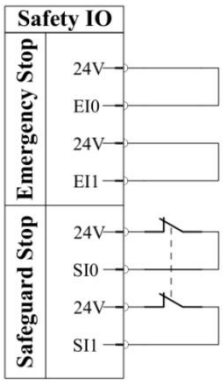


Figure 19-4

Note: Please connect at least one external emergency stop button in case that the teach pendant is removed.

19.2 Use Configurable Safety IO as Safety IO

The user can set the emergency stop, the reduced mode, the three-position switch and others with the 8 configurable inputs and 8 configurable outputs.

Figure 19-5

Step: Click **Config -> Safety -> Safety IO** and enter the password. Select the configurable input and output to be configured and select the corresponding function from the dropdown list. Note that the user needs to configure the parameters before selecting the safety plane and safety tool.

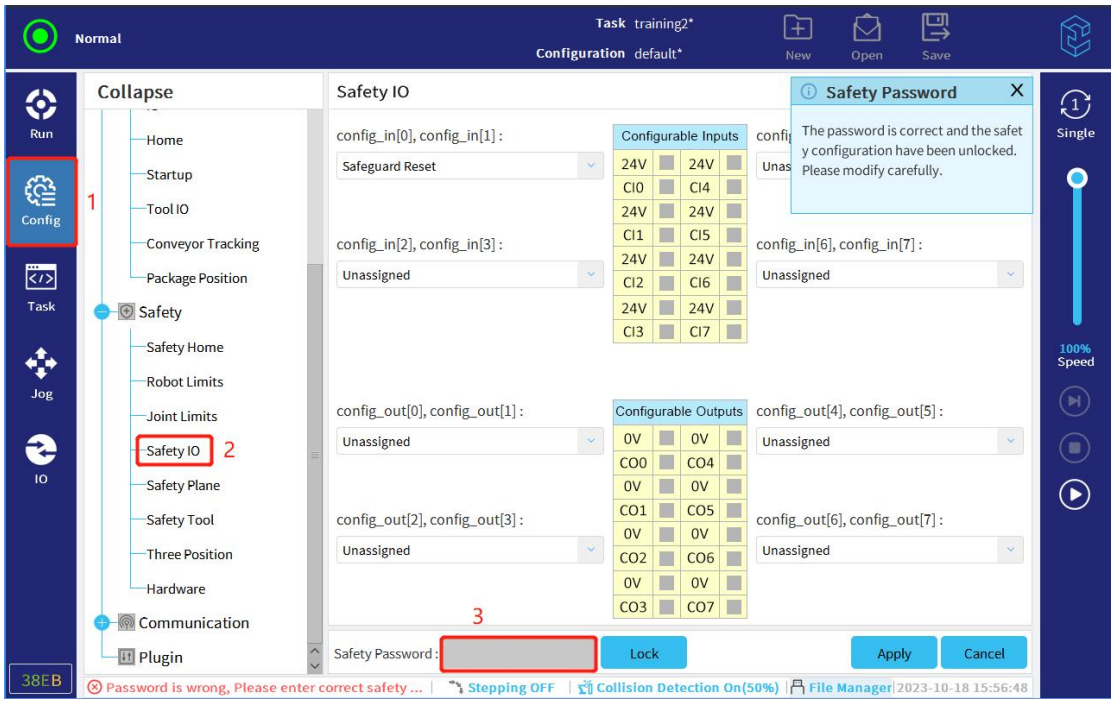


Figure 19-6

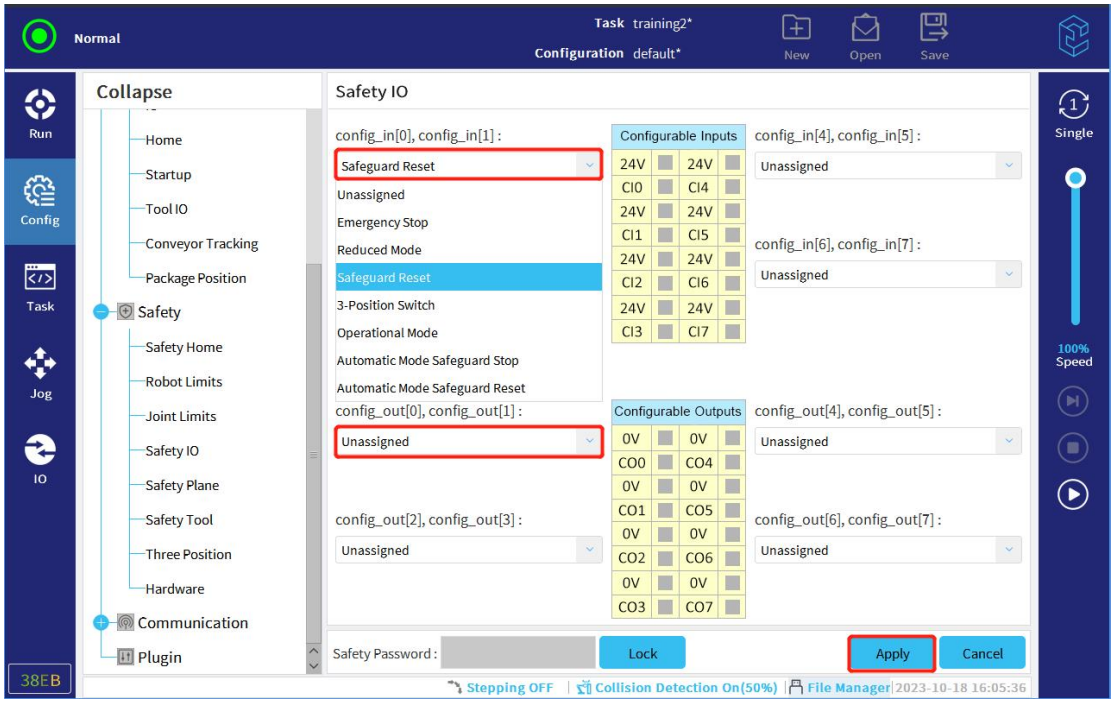


Figure 19-7

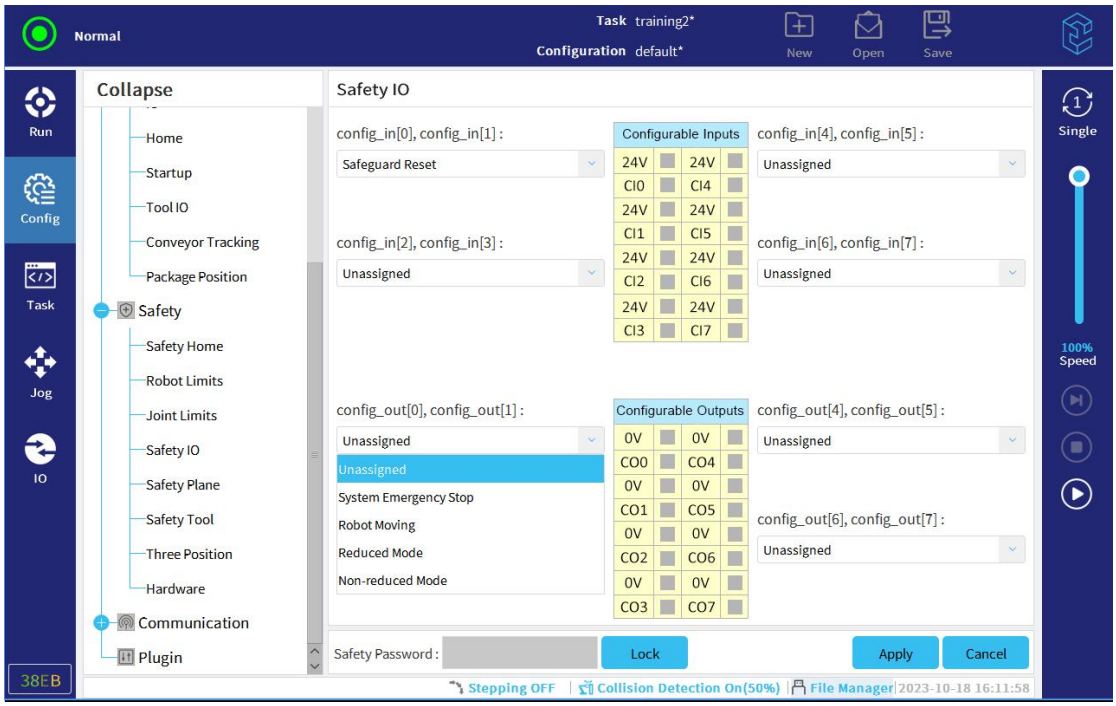


Figure 19-8

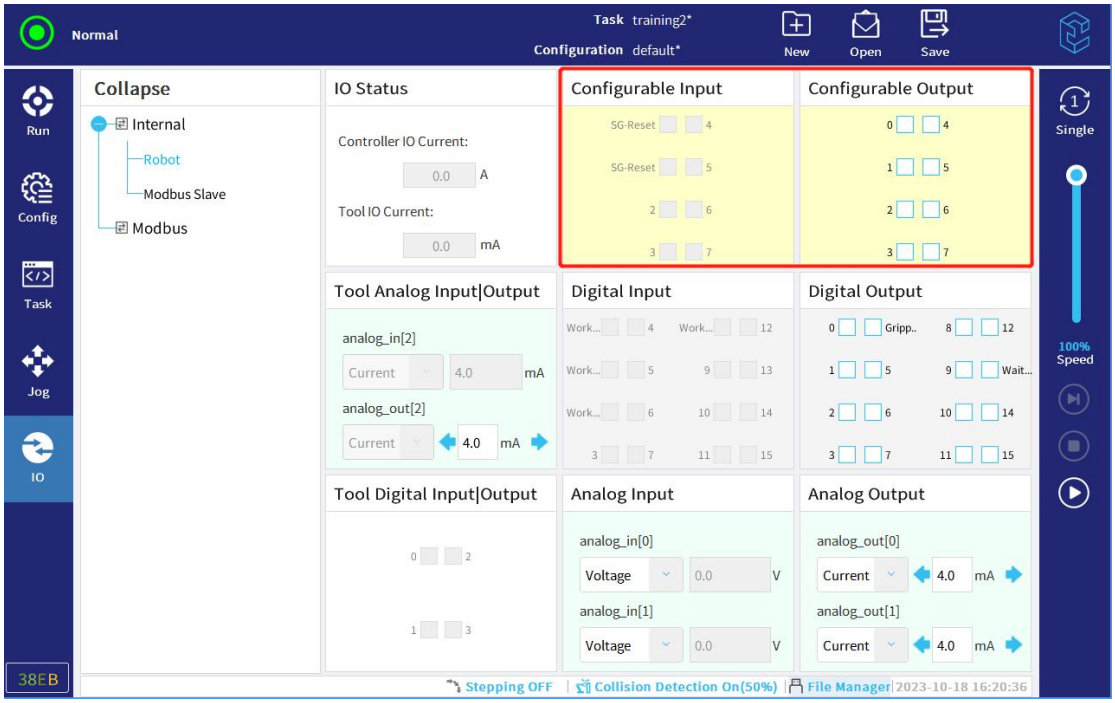


Figure 19-9

The protective reset button is connected as shown in the following figure (e.g. safety light curtain).

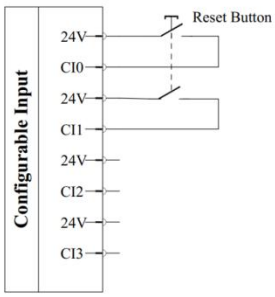


Figure 19-10

20 Safety Setting

20.1 Safety Password

Please enter the safety password before modifying the safety settings.

Step:

1. Click the Elite log in the top right corner
2. Click **Settings -> Password -> Safety**. There is no initial password. The user can directly enter a new password and confirm it. Then click ”Apply”.

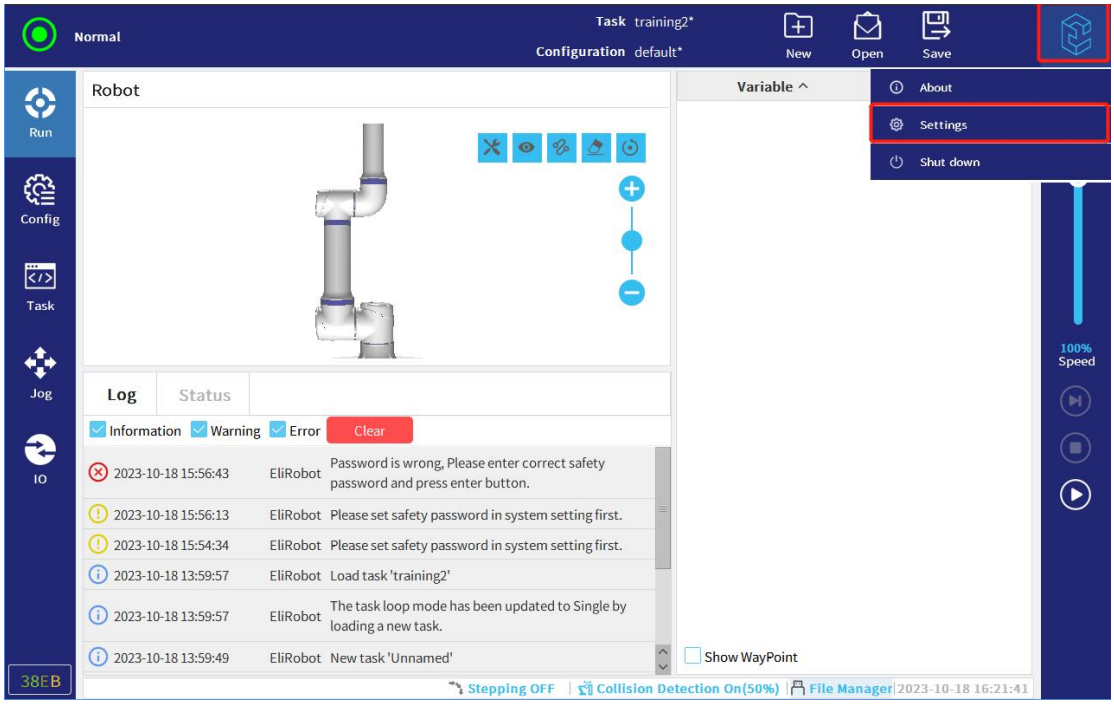


Figure 20-1

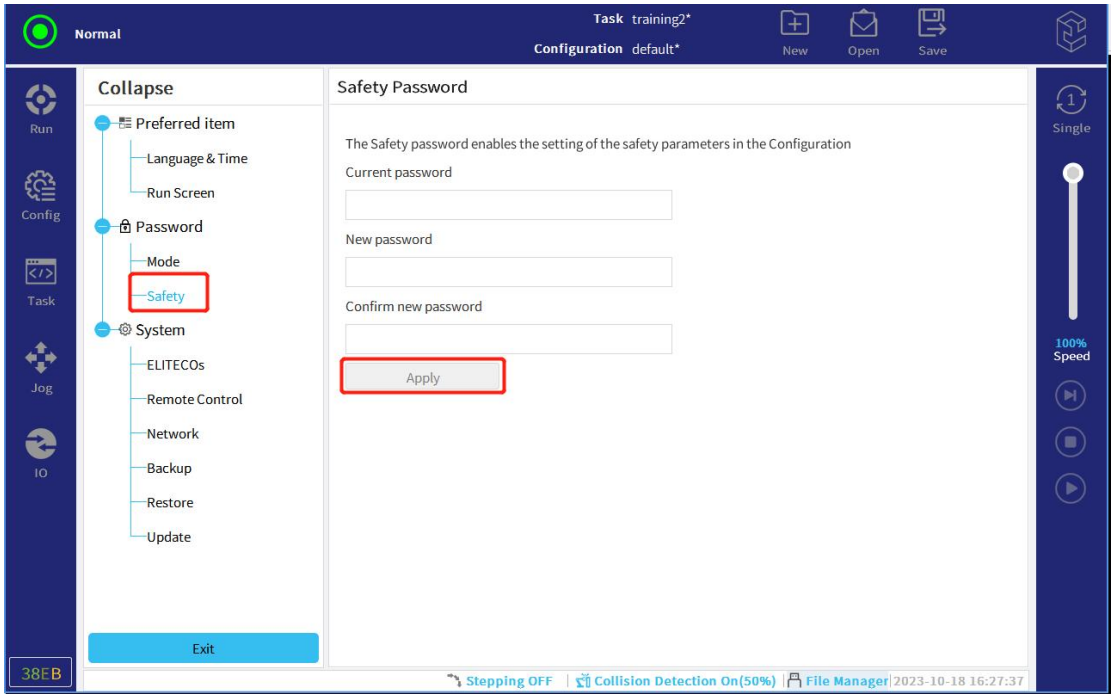


Figure 20-2

Note: Please contact the Elite support department if the password is forgotten (service@elibot.com).

20.2 Safety Home

The safey home is a user-defined home position. The user can only select the safety home from the home available.

Step:

1. Click **Config -> Safety -> Safety Home** and enter the safety password.
2. Select the home from the dropdown list of “Reference Home” in the right.

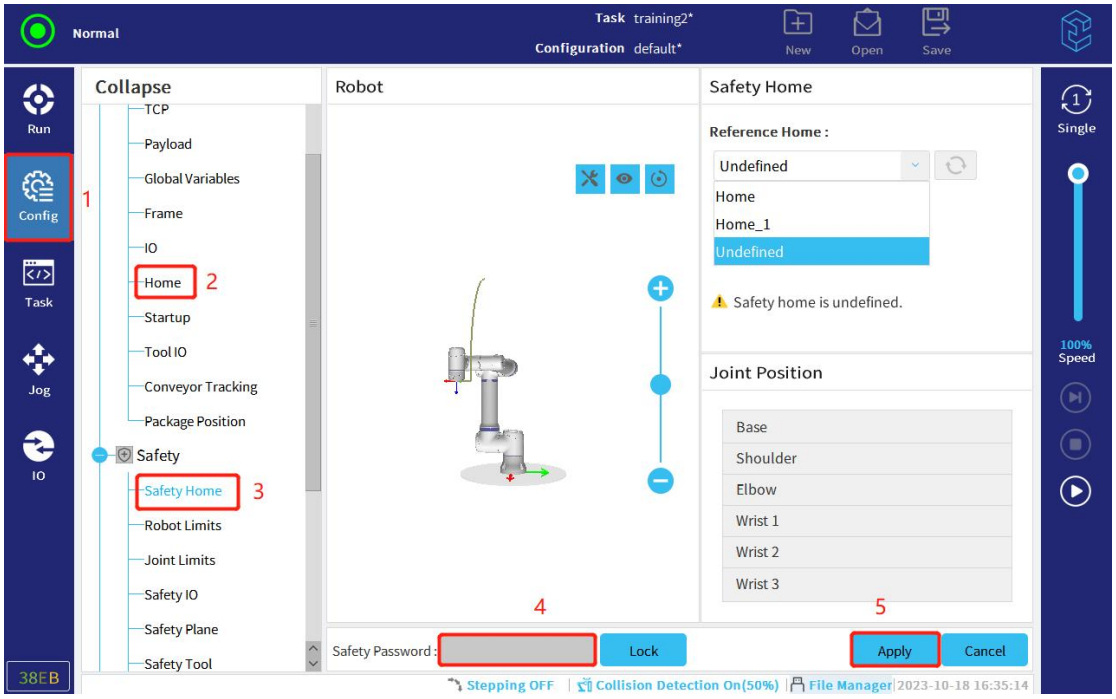


Figure 20-3

Note: After the settings, the user can configure CO (configurable output) to send the signal whether the robot is at the safety home to the external device.

20.3 Robot Limits

There are 5 configuration options available on the robot limit screen, i.e. Least Restricted, 2nd Least Restricted, 2nd Most Restricted, Most Restricted and Custom Restricted. Only the “Custom Restricted” option allows the user to modify the limit parameter of the robot.

Stopping Time: limits the maximum time that takes robot to stop from moving

Stopping Distance: limits the maximum distance that the robot tool can move before stopping

Tool Speed: limits the maximum speed that the robot tool moves

Elbow Speed: limits the maximum speed that the robot elbow moves

Note: When setting the robot limit parameters, the values in the normal mode is larger than those in the reduced mode.

Step:

Click **Config -> Safety -> Robot Limits** and enter the safety password. The user can select the limit level by clicking  according to the applications. The icon  indicates that it is activated.

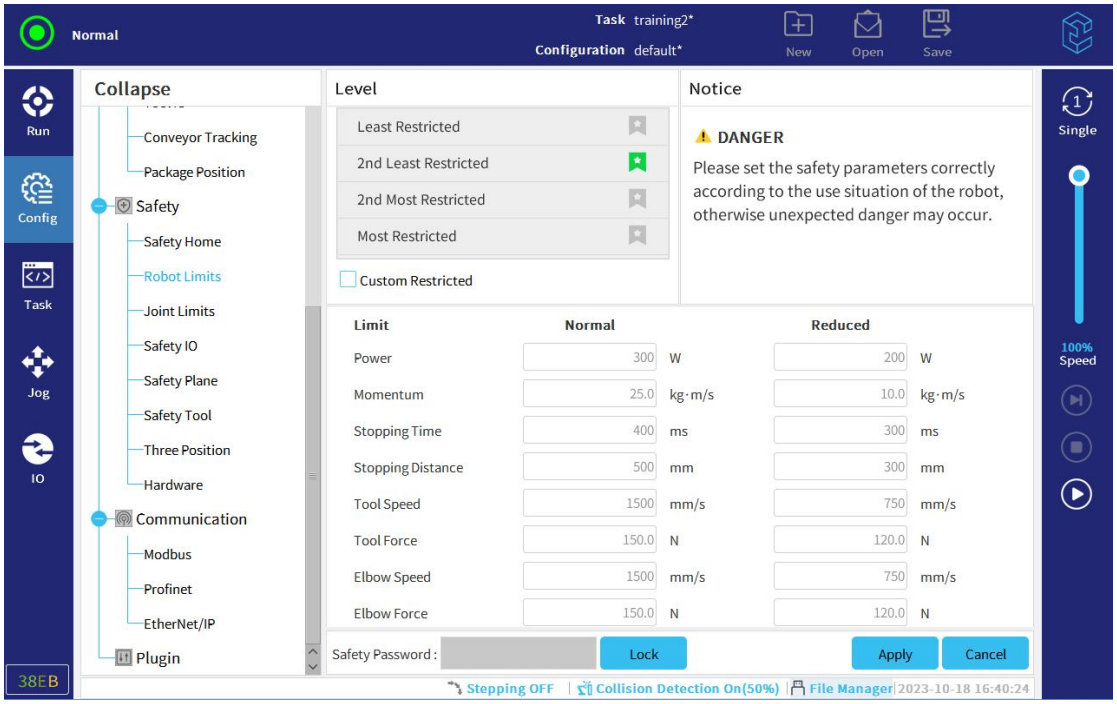


Figure 20-4

20.4 Joint Limits

It is used to limit the range of the position and the maximum speed of each joint.

Step:

1. Click **Config -> Safety -> Joint Limits** and enter the safety password.
2. Set the range of the joints and the maximum speed in accordance with the actual demands.

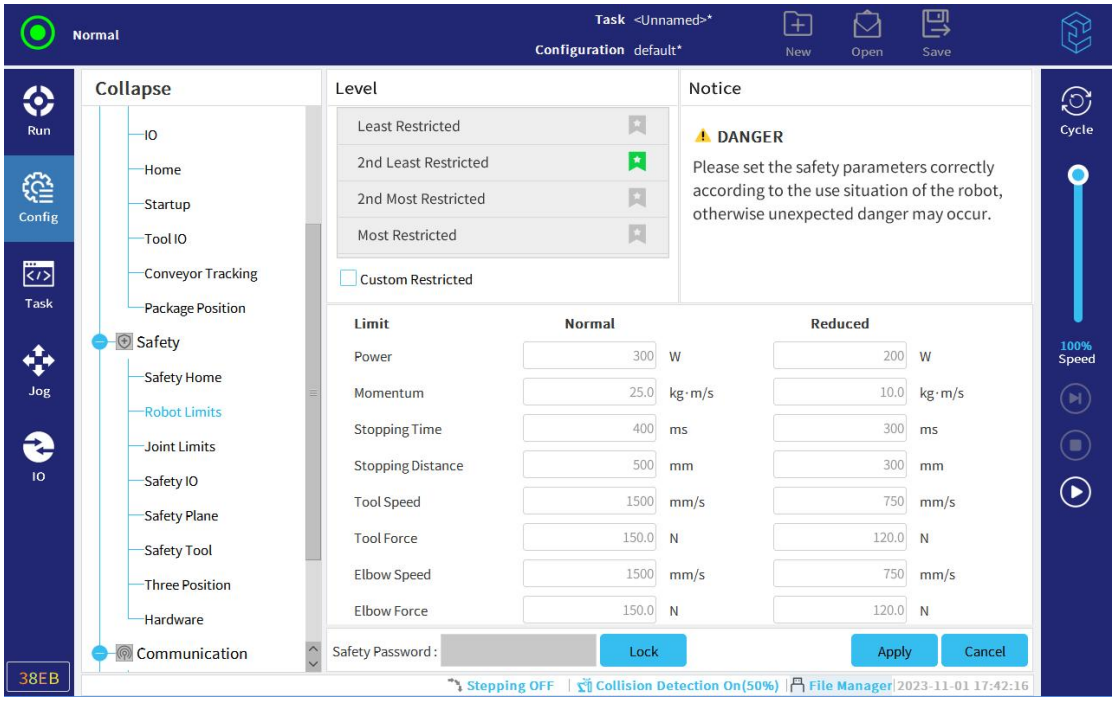


Figure 20-5

Note:

1. When setting the position range, the difference between the maximum and minimum values should be at least 7°, and the range value in the reduced mode cannot exceed that in the normal mode.

2. The reduced mode can be triggered by setting the safety plane or the safety IO. If there are no trigger conditions, it indicates that the parameters in the reduced mode cannot be set.

20.5 Backdrive (in an emergency)

The function can be used to release the stuck state. When entering the backdrive mode, the user can directly drag by hand. Note that the force exerted shall be slightly greater than that of the hand drag.

Step:

When the robot is in standby state, press the drag button to enter the backdrive node.

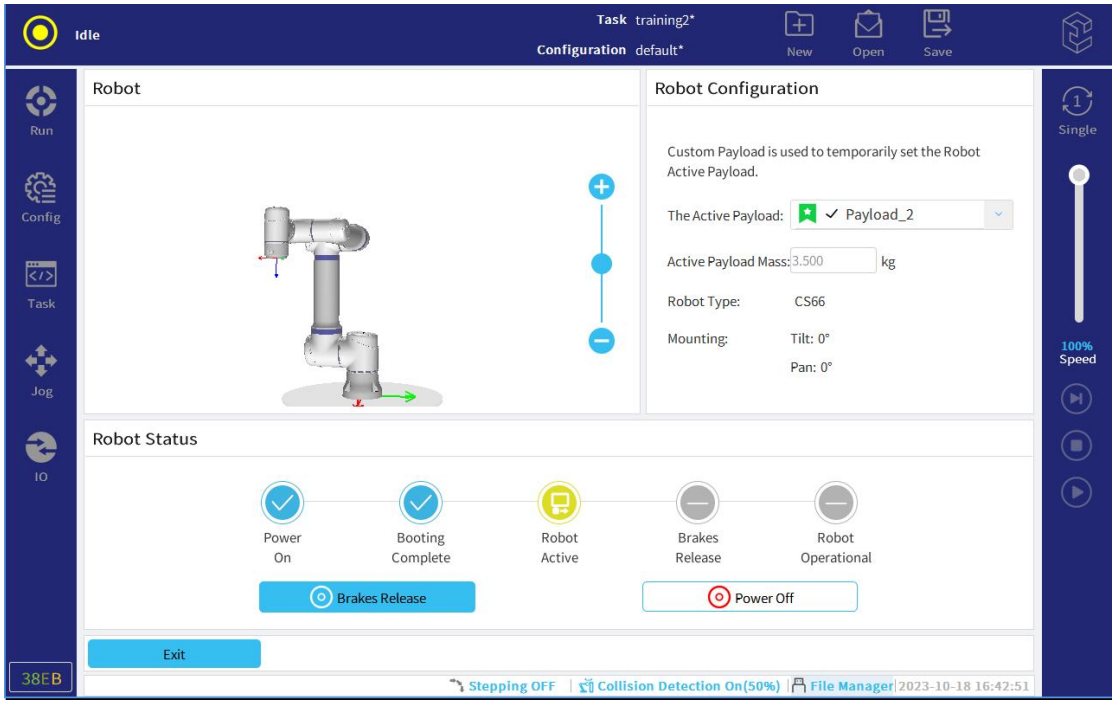


Figure 20-6

Figure 20-7

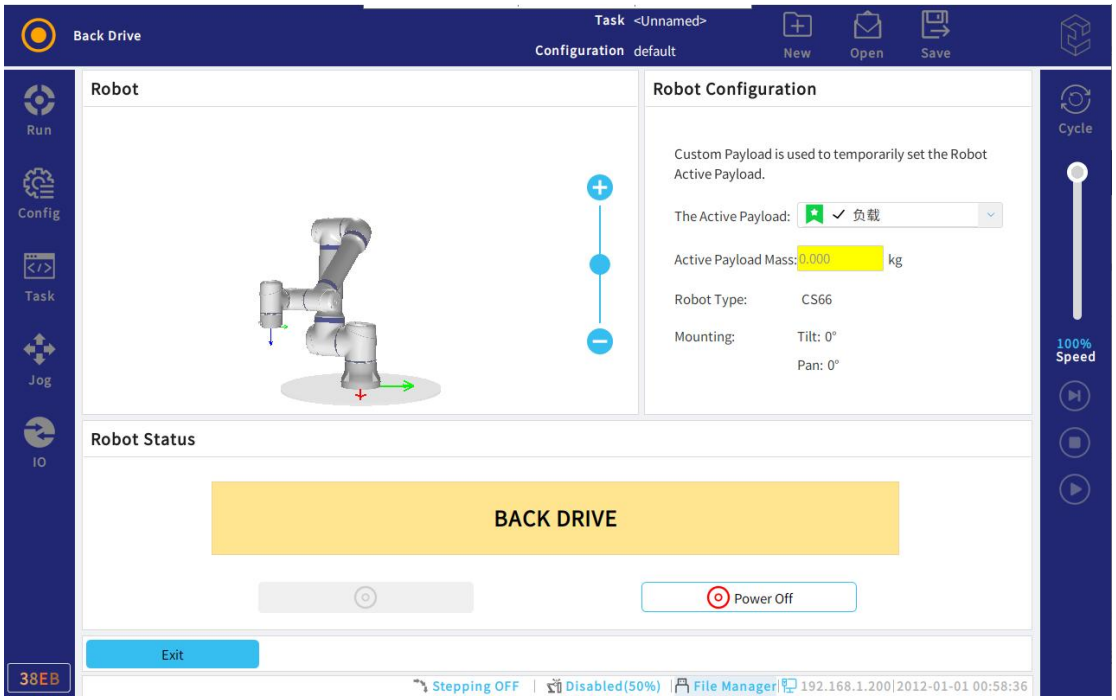


Figure 20-8

20.6 Safety Plane

The safety plane limits the workspace of the robot. It can limit the safety tool or elbow and determine if they cross the plane. The user can define up to 8 safety planes. Please define the frames first before defining the safety plane.

Step:

1. The safety plane is configured based on the frame.
2. Click **Config -> Safety -> Safety plane** and click + to create a safety plane. Click to rename it.

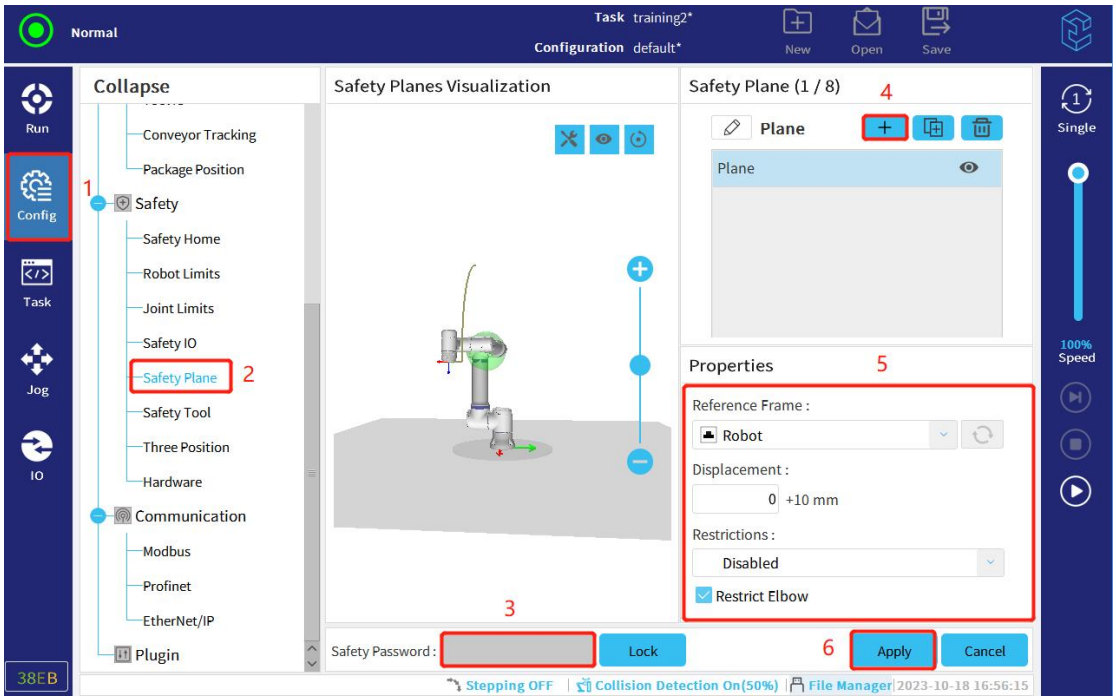


Figure 20-9

3. In the attribute field at the bottom right corner of the screen, set reference frame, displacement and restrictions.

➢ Reference Frame: For the definition method, please refer to the aforementioned chapter.

➢ Displacement: If the user enters a number in “Displacement”, it indicates the moving distance relative to the Z axis of the selected frame. For instance, if the frame is selected as “Base”, and the displacement is 200, then the coordinate of safety plane is the frame that moved 200mm on the Z axis from the base frame.

➢ Restrictions: The user can configure restrictions for each safety plane. In the disabled state, the safety plane is never activated. When the robot is in the “normal” mode, the safety plane takes effect. When the robot tool or elbow exceeds the safety plane, the robot stops protectively. When the robot is in the “reduced” mode, the safety plane takes effect. When the robot tool or elbow exceeds the safety plane, the robot stops protectively. When selecting both (i.e. normal and reduced mode), the safety plane takes effect. Whenthe robot tool or elbow exceeds the safety plane, the robot stops protectively. The “Trigger Reduced mode” indicates that the system will switch to the reduced mode when the robot tool or elbow exceeds the safety plane.

The user can configure restrictions for each safety plane.

Disabled: in this state, the safety plane is never activated;

Normal mode valid: when the robot is in "Normal" mode, the safety plane takes effect. When the robot tool or elbow contacts or exceeds the safety plane, the robot stops protectively;

Reduced mode valid: when the robot is in "Reduced" mode, the safety plane takes effect. When the robot tool or elbow contacts or exceeds the safety plane, the robot stops protectively;

Normal & Reduced mode valid: when select "Both", the safety plane takes effect when the robot is in "Normal" mode or "Reduced" mode. When the robot tool or elbow contacts or exceeds the safety plane, the robot stops protectively;

Trigger Reduced mode: when the robot tool or elbow contacts or exceeds the safety plane, the system switches to reduced mode.

Note: If the reference frame is modified, the warning icon will appear in the left part of the “Frame” text. The frame of the safety plane, however, is still the frame before the modification. That is, the frame will not be modified synchronously.

20.7 Safety Tool

The safety tool TCP is tested and limited by the safety plane. In the safety tool, the user can add/copy/delete the safety tool items. The tool flange is the default safety tool and the parameters cannot be modified (the default radius of the tool flange and TCP position are both 0). The user can set/customize up to 3 safety tools, as shown in the following figure.

Figure 20-10

Step: The user can select the default tool flange or a user-defined new one as the safety tool. Click “Edit Position” to edit the properties in the “Copy TCP” and modify the values of X, Y and Z. The visualization of the safety tool will also display the status attributes of the corresponding tool.

After the settings, click “Apply” and the robot will automatically restart.

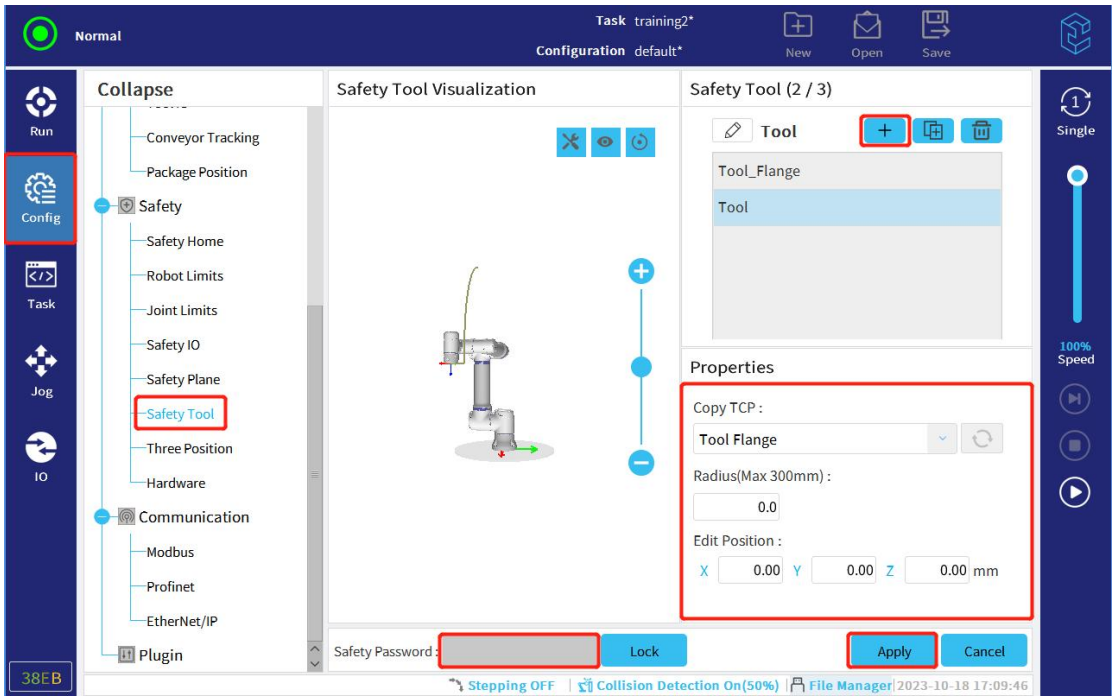


Figure 20-11

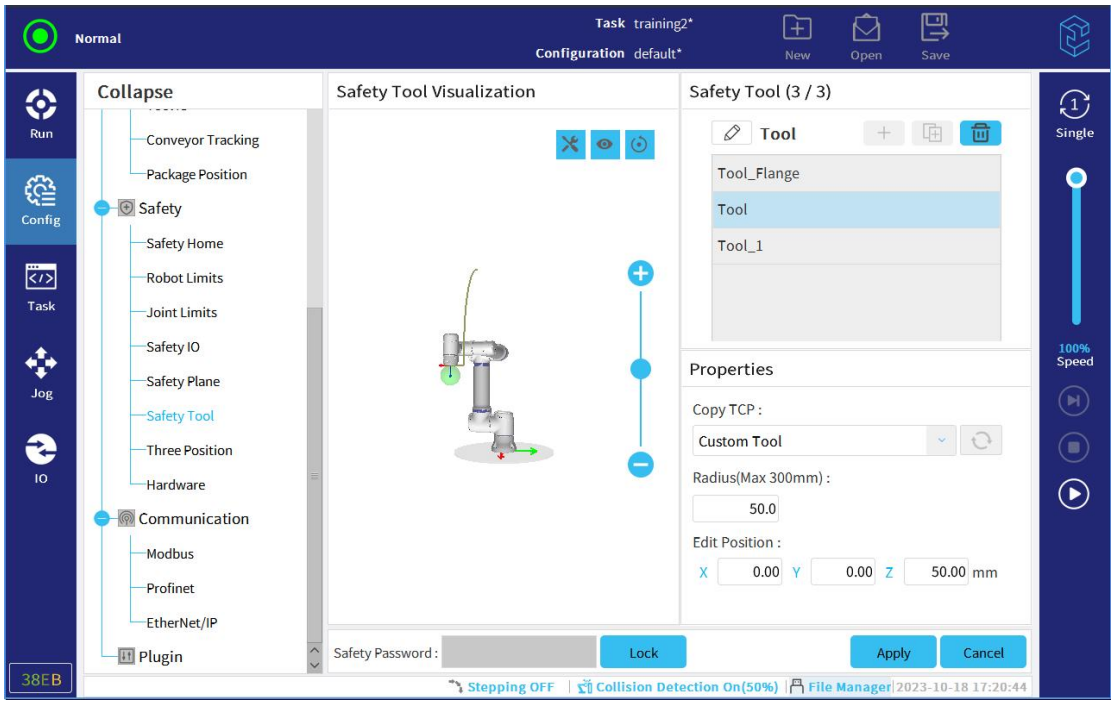


Figure 20-12

20.8 Three-position Switch

The controller connects the three-position switch externally by setting the configurable safety IO. After the configuration, the user can enable the manual high speed, which is used to dynamically adjust the speed limit in the manual mode when the program is running. The user can switch over the manual mode to automatic mode.

The user needs to enter the password when switching the mode if there be one. In the manual mode, turn the three-position switch to the middle part (that is, after turning on the servo) and then teach the robot. When the three-position switch is not on, the robot will be in a protective stop status.

Step: Click **Config -> Safety -> Three Position**

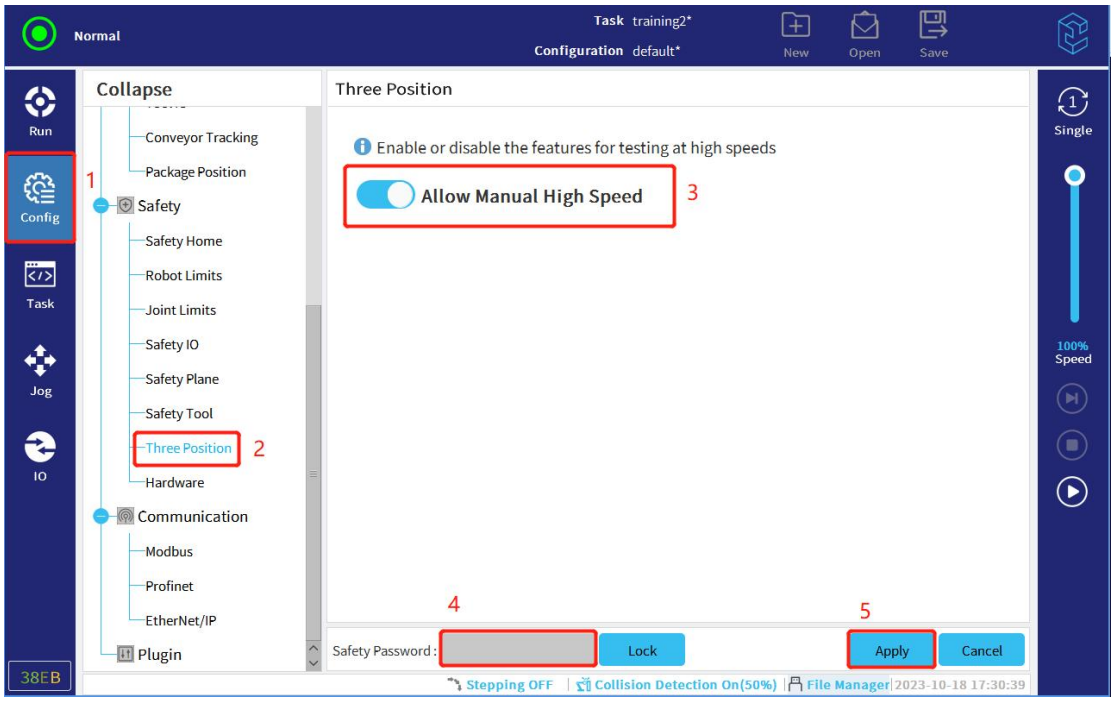


Figure 20-13

20.9 Remove the Teach Pendant

The teach pendant can be removed as follows.

Step: Click **Config -> Safety -> Hardware**, enter the safety password and select the type of the teach pendant.

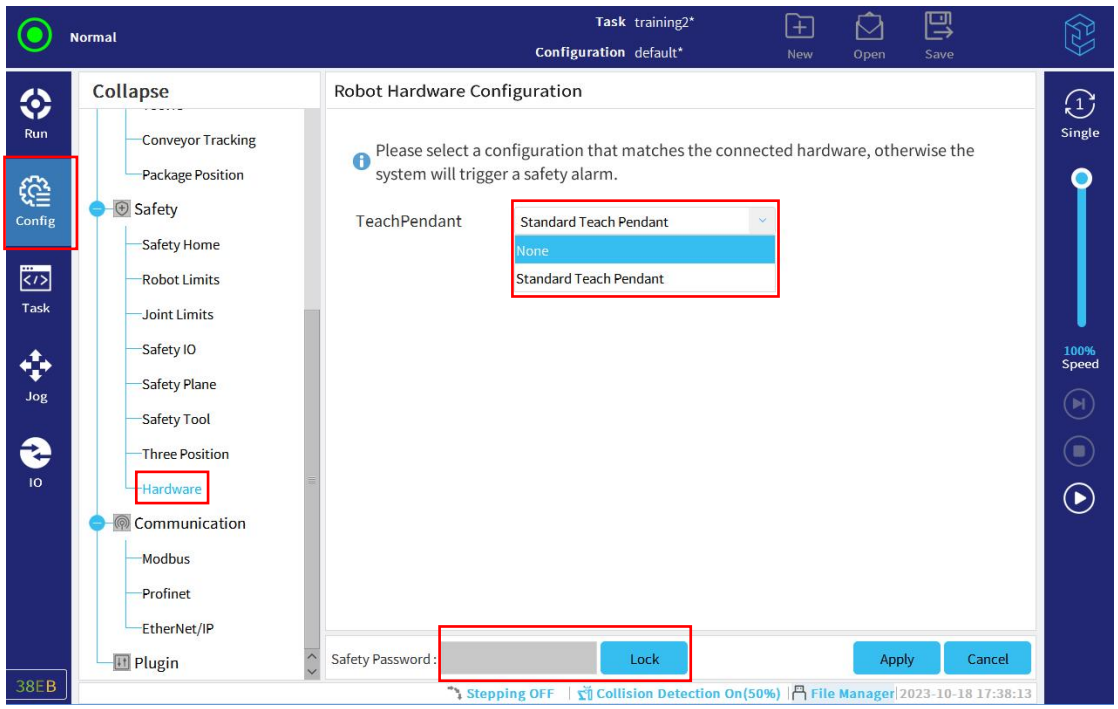


Figure 20-14

20.10 Collision Detection (Advanced)

Click the section marked by a red square five times, as shown in the following figure and enter the password (i.e. elite) to enter the expert mode.

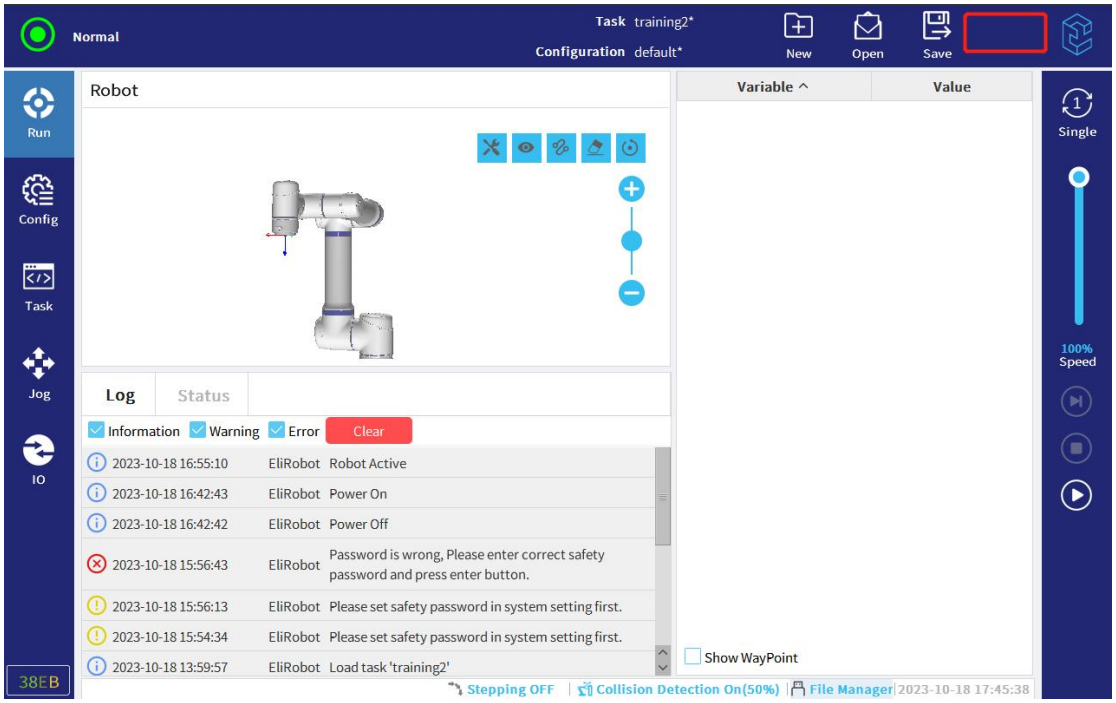


Figure 20-15

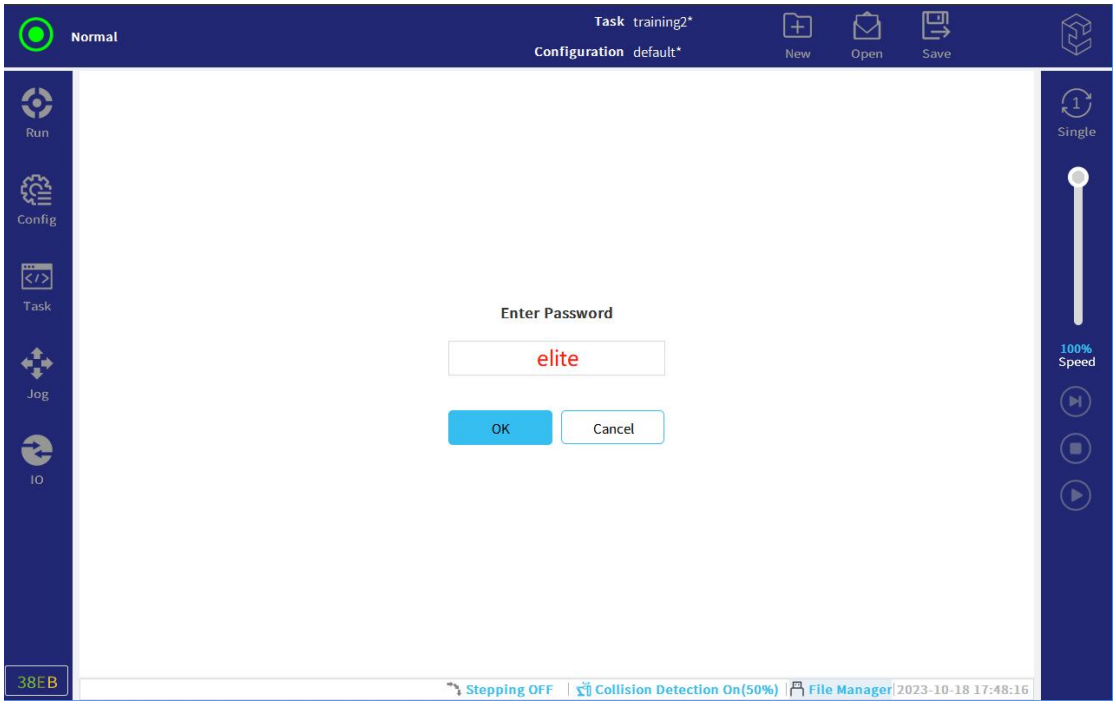


Figure 20-16

Click “Force Control” to set the collision detection on/off, sensitivity and drag speed.

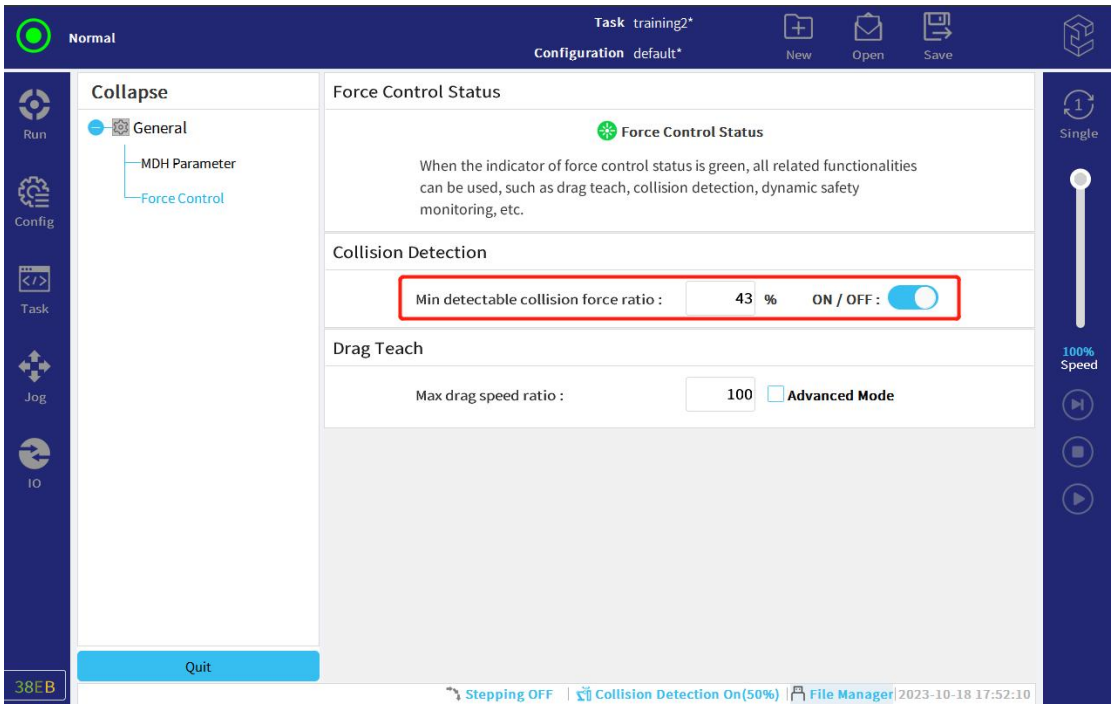


Figure 20-17

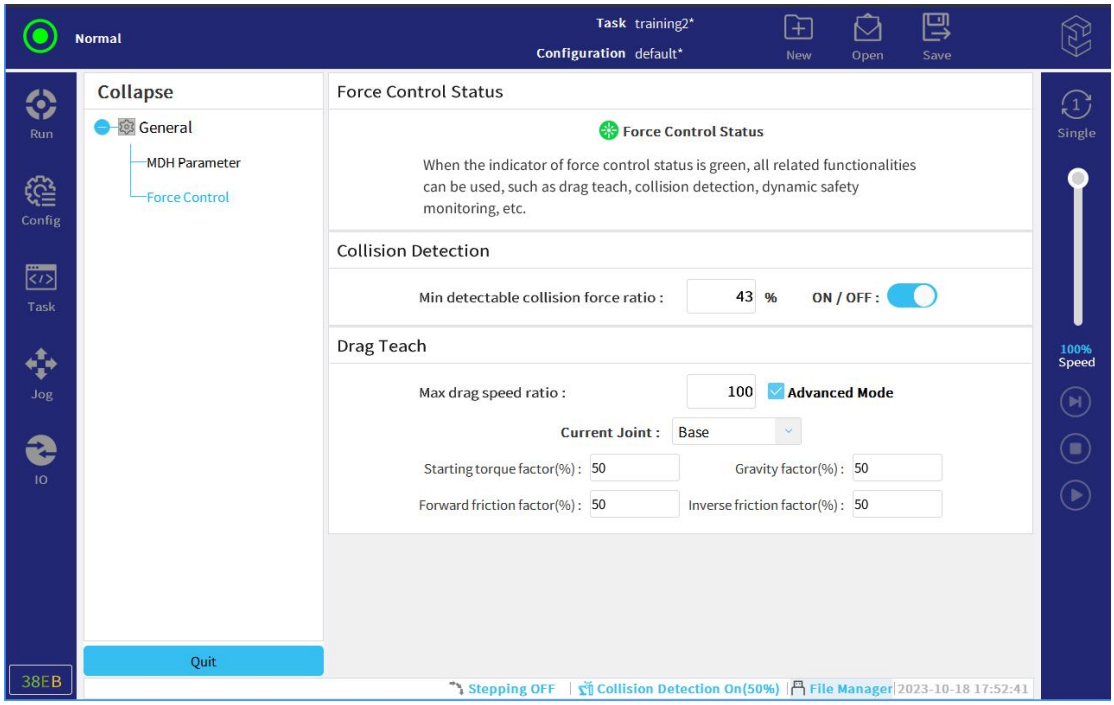


Figure 20-18

20.11 Startup via External Device

The remote ON/OFF functional terminal provides a 12V auxiliary power supply. When the remote ON is enabled, the control system will be on. When the remote OFF is enabled, the control system will be off.

Note: After removing the teach pendant, the robot can automatically start and stop when enabling the auto load and start task with the external start signal.

The following figure shows the connection of the remote ON/OFF (The OFF input must be used for remote shutdown control because this signal allows the controller to save open files and close them normally.).

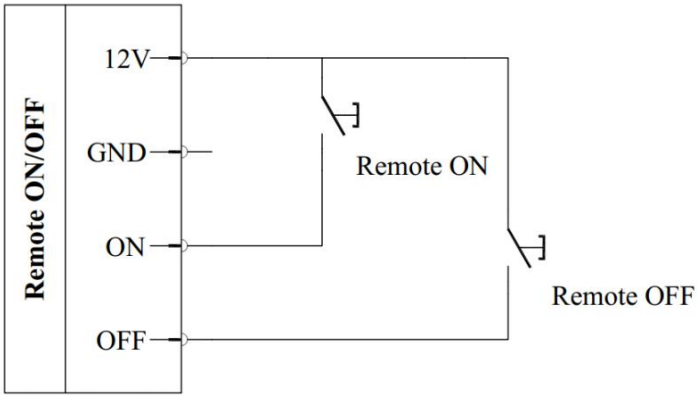


Figure 20-19

* Appendix (follow-ups)

Appendix 1-1 Modbus

Robot is the server

The Elite robot supports Modbus TCP and Modbus RTU. It can also be used as a server (slave) and a client (master). When it is regarded as a server, it will provide other devices with the data. The settings are very simple and here are the steps. Click **Config -> Communication -> Modbus**, and click “Add MODBUS Unit” (a new Modbus address and the CS Modbus can support up to 200 signals).

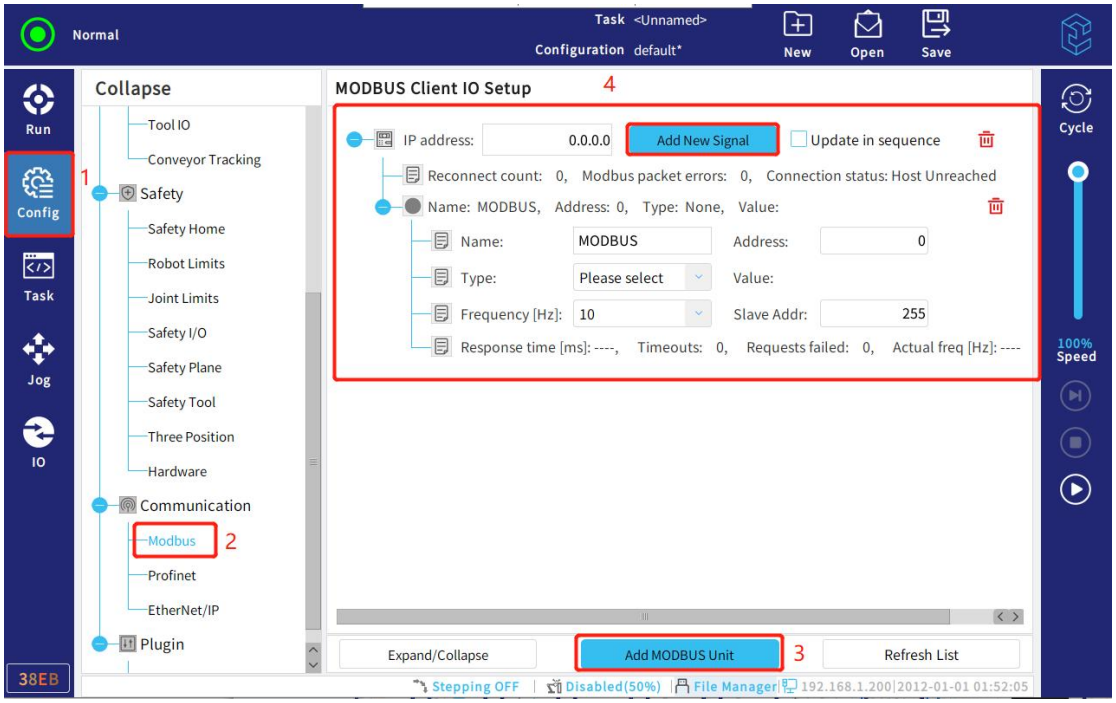


Figure Appendix-1

Enter an IP address, read or write the register address and type, as well as add a new signal.

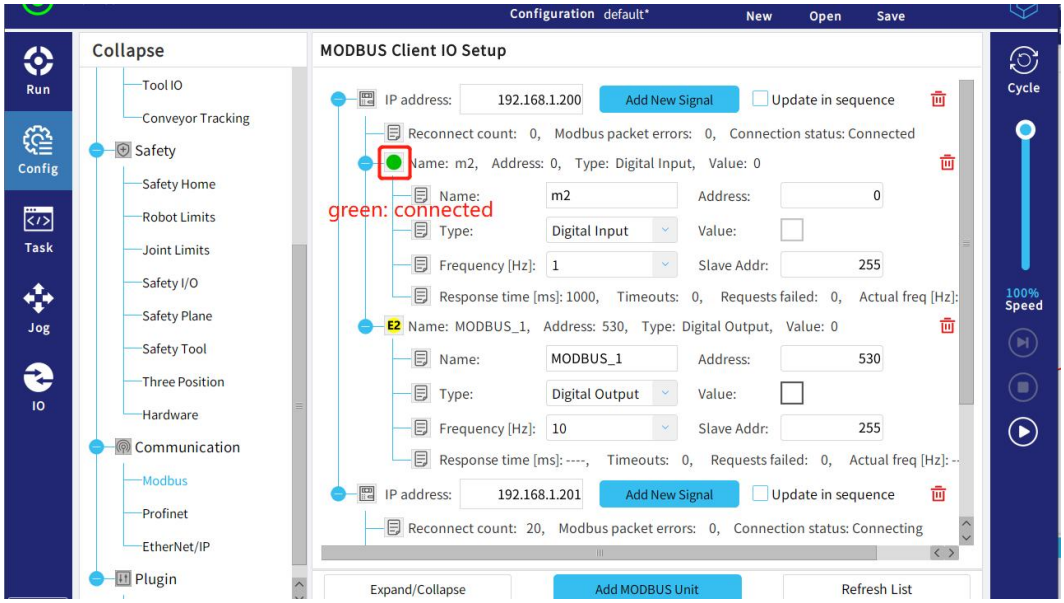


Figure Appendix-2

Robot is the client

When the robot is used as a client, the user can estalibsh a connection to the server with the script commands and read the data. For the details of the script commands, please refer to the script manual. The communication with DH gripper is taken as an example for the illustration.

_init.script

VorkSpace > Program > dh_init.script > ...

```
IO_MODE = 0x0402
INIT = 0x0100
TARGET_WIDTH = 0x0103

tool_serial_config(True, 115200, 0, 1,8,True)
tool_modbus_write_registers(1, IO_MODE, 0)
# Turn off the gripper IO mode
#tool_modbus_write_registers(1, INIT, 1)
# Initialize the gripper

IO_MODE = 0x0402
INIT = 0x0100
TARGET_WIDTH = 0x0103

def GripperOpen():
    tool_modbus_write_registers(1, TARGET_WIDTH, 1000)

def GripperClose():
    tool_modbus_write_registers(1, TARGET_WIDTH, 0)

def GripperSet(num):
    tool_modbus_write_registers(1, TARGET_WIDTH, num)

sleep(1)
```

Figure Appendix-3

Connect the gripper to the tool terminal of the robot and set the communication parameters with the command tool_serial_config. The command tool_modbus_write_registers is used to write the data to the gripper. The user can get the register address of the gripper in the grippper manual.

The user can view the register address via the monitor.

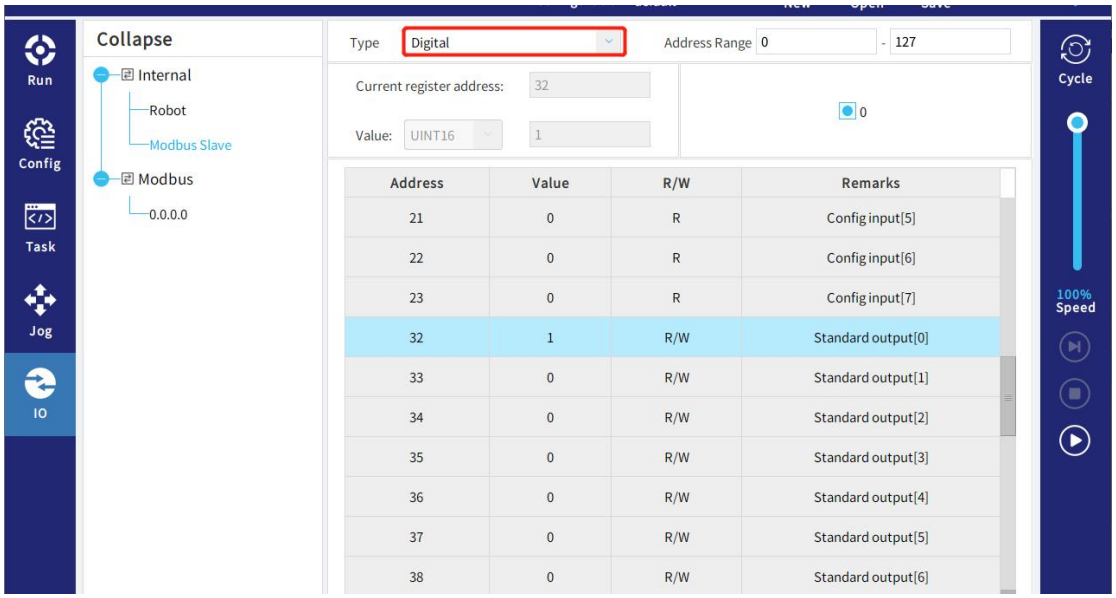


Figure Appendix-4

When the IP address is 127.0.0.1, it indicates that the data is read from the controller server. For instance, read the value of the inner register address 256 and assign it to the variable of the mstart, i.e. mstart=99. The user can directly visit the signal mstart via name in the program.

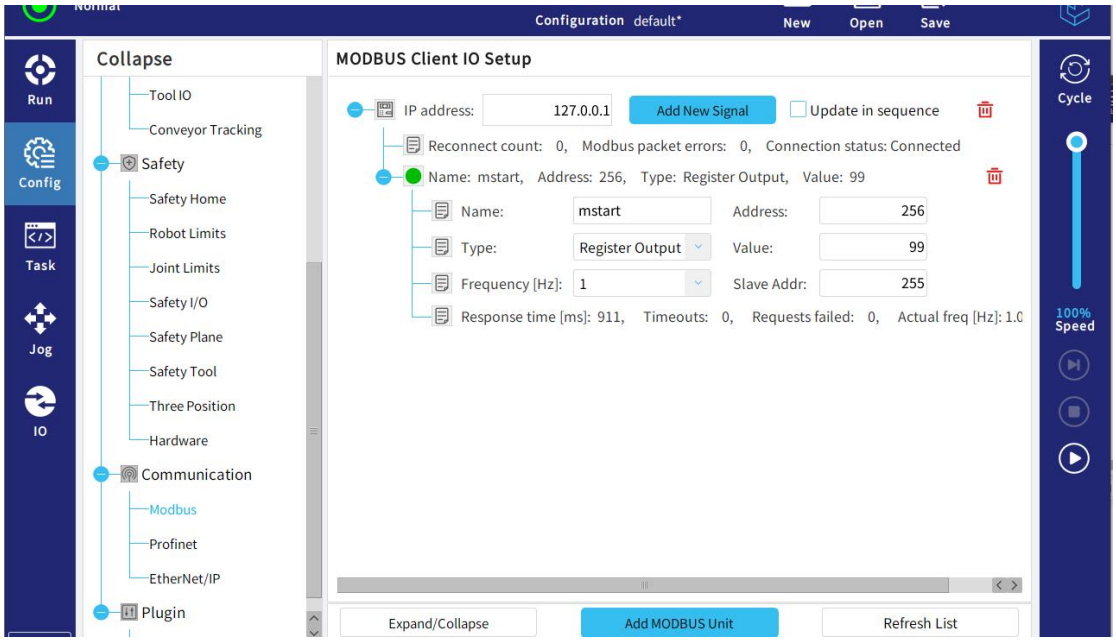


Figure Appendix-5

Appendix 1-2 Profinet Setting

The Profinet interface mainly includes the Profinet status bar, notification bar, the robot IO module and the insert status of the register module.

Step: Click **Config -> Communication -> Profinet**. Enter the profinet interface and click ”Enable”.

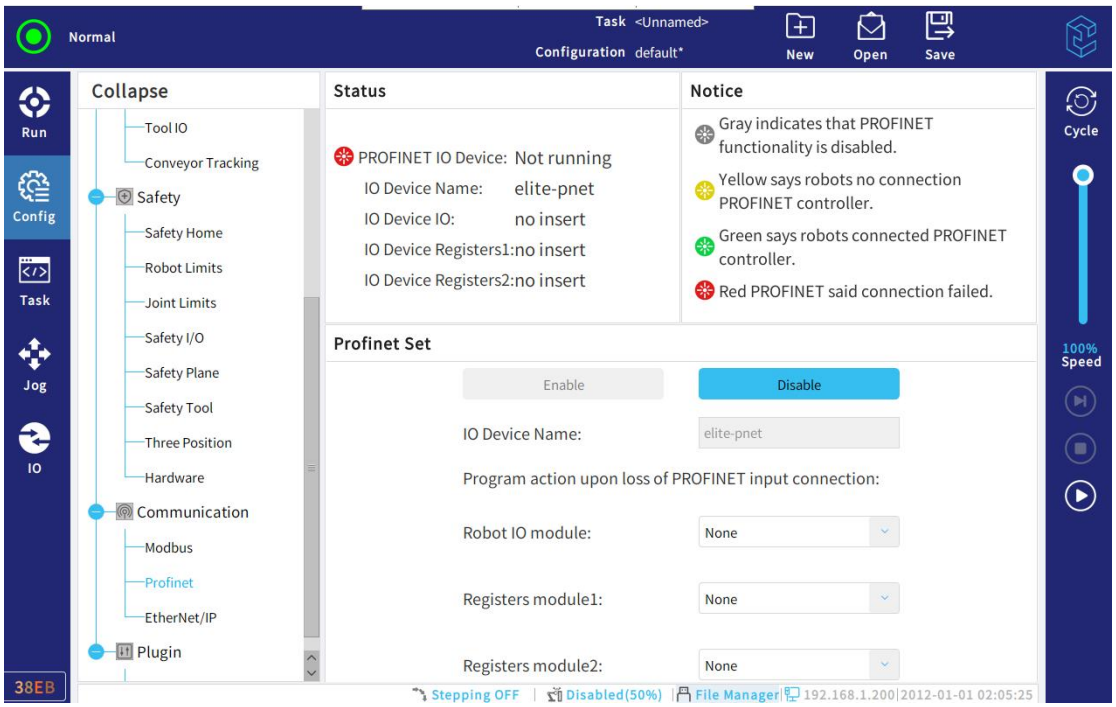


Figure Appendix-6

To get the GSD file of the CS series robot and the data interpretation, please visit: https://bbs.elibot.cn/forum/detail/topic/309.html.

Appendix 2-1

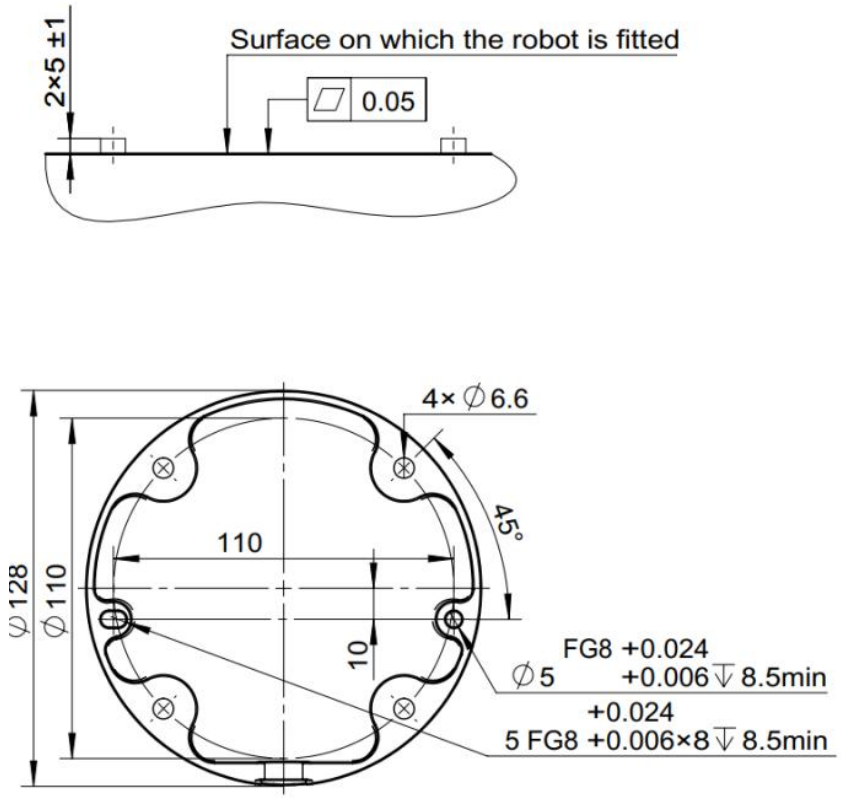


Figure Appendix-7

Mounting dimension drawing of the CS63 robot base, with unit of mm

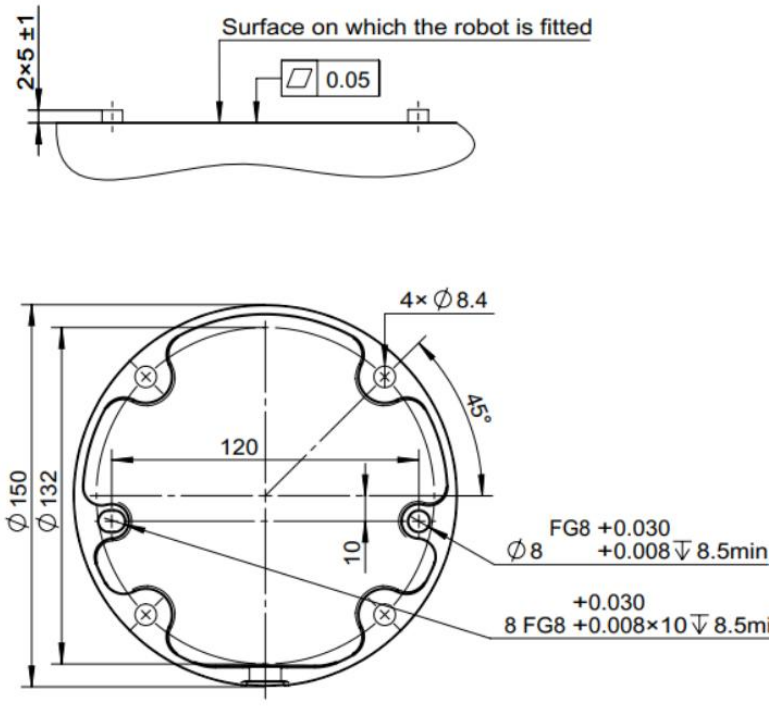
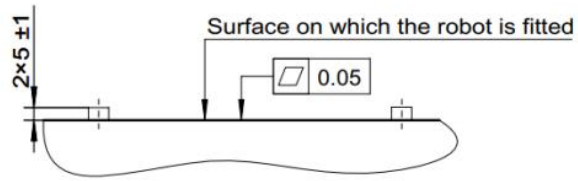
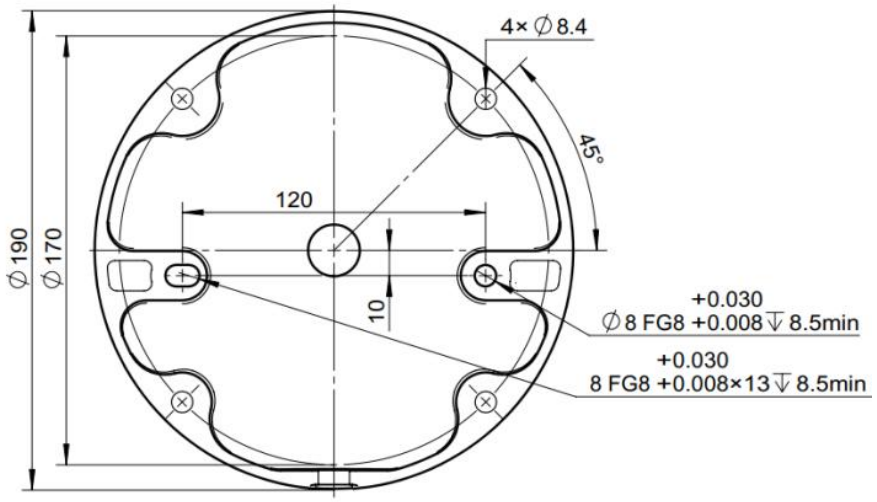


Figure Appendix-8

Mounting dimension drawing of the CS66 robot base, with unit of mm





m

Figure Appendix-9

Mounting dimension drawing of the CS612 robot base, with unit of mm

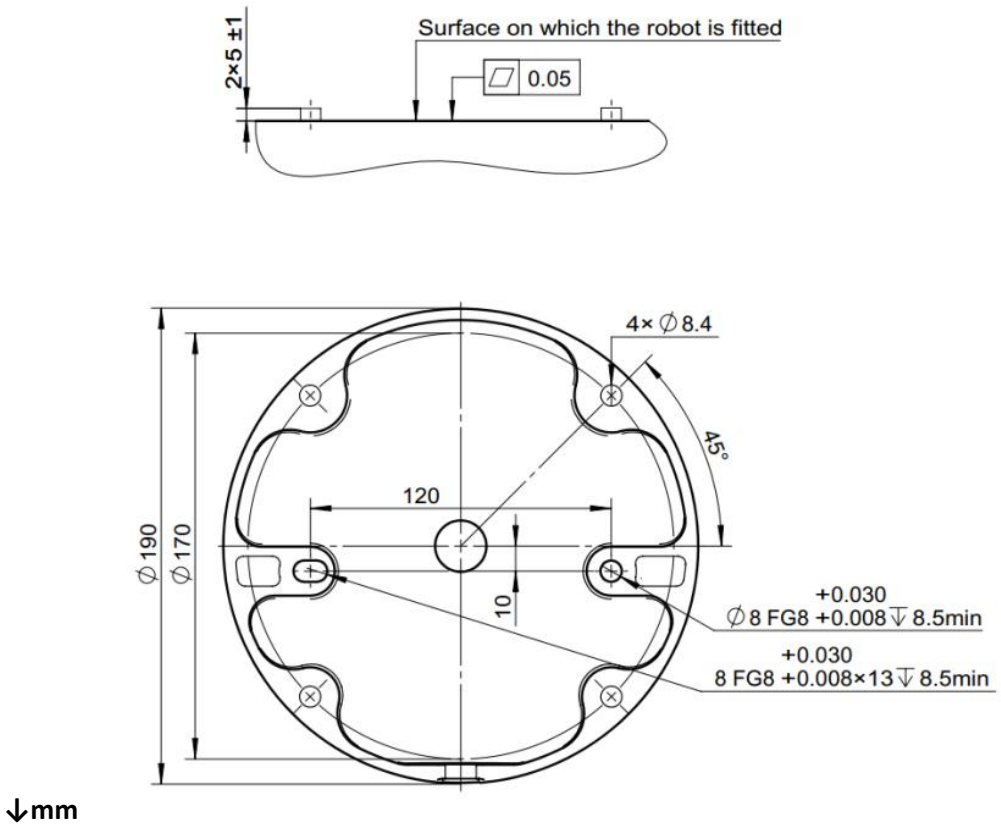


Figure Appendix-10

Mounting dimension drawing of the CS620 robot base, with unit of mm

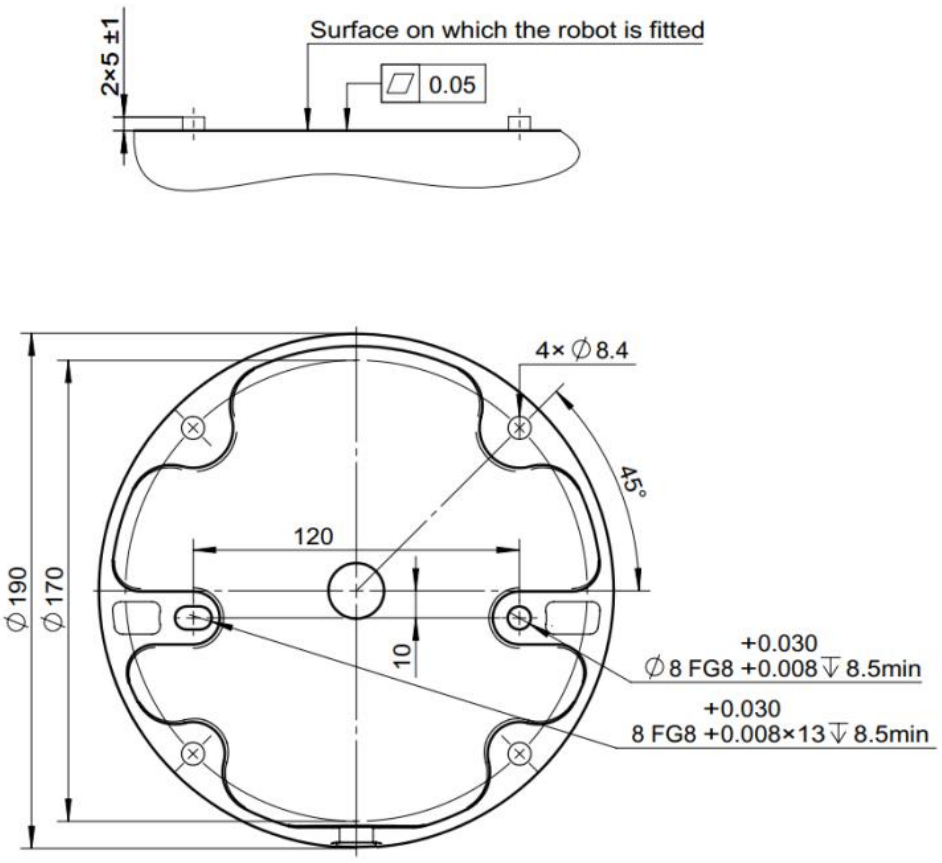


Figure Appendix-11

Mounting dimension drawing of the CS625 robot base, with unit of mm

Appendix 2-2

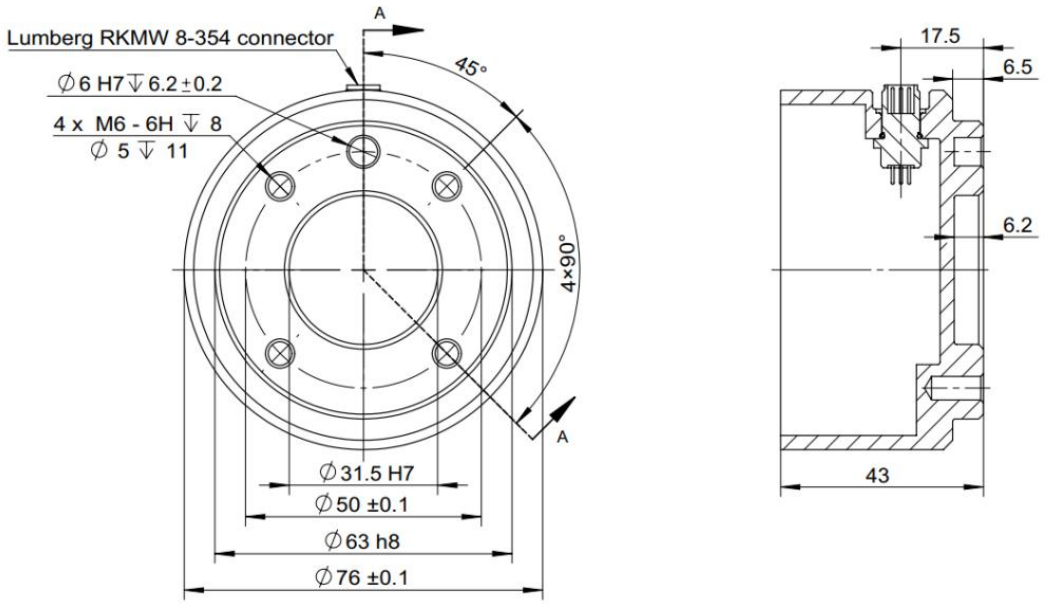


Figure Appendix-12

Mounting dimension drawing of the CS63 robot tool flange, with unit of mm

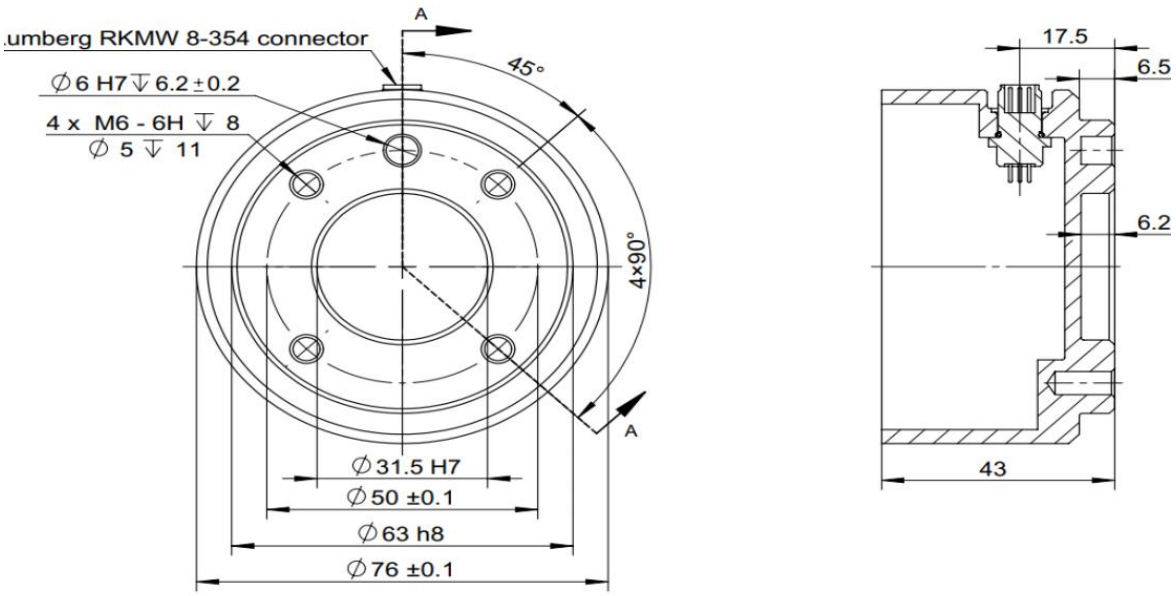


Figure Appendix-13

Mounting dimension drawing of the CS66 robot tool flange, with unit of mm

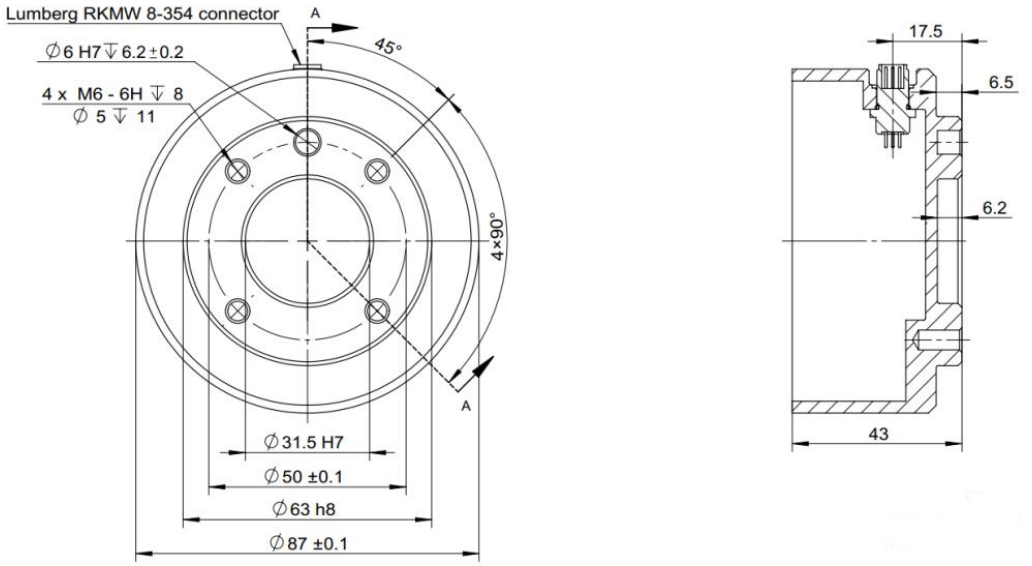


Figure Appendix-14

Mounting dimension drawing of the CS612 robot tool flange, with unit of mm

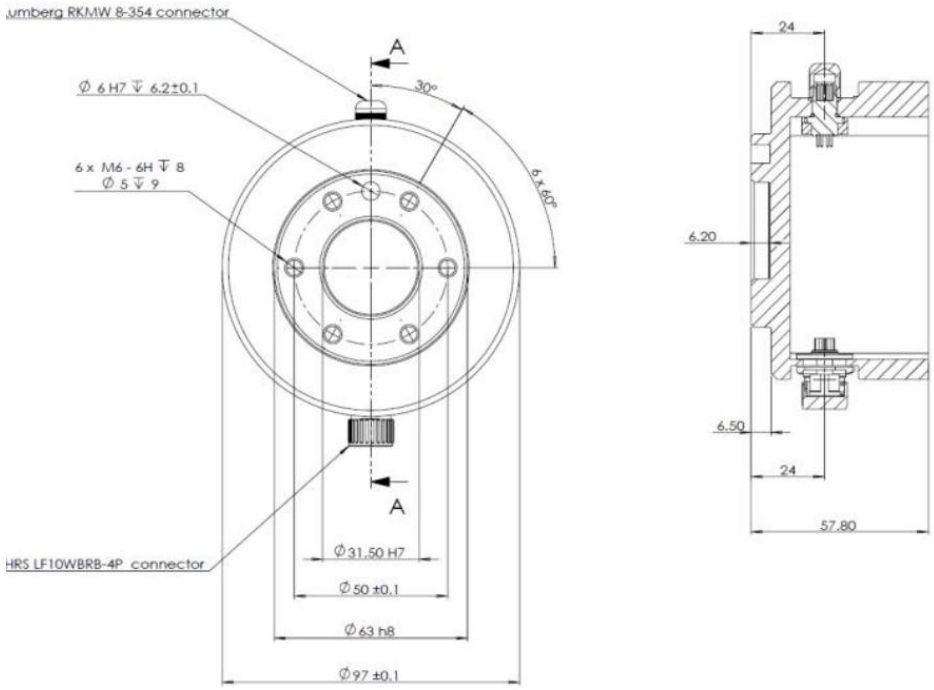


Figure Appendix-15

Mounting dimension drawing of the CS620/CS625 robot tool flange, with unit of mm

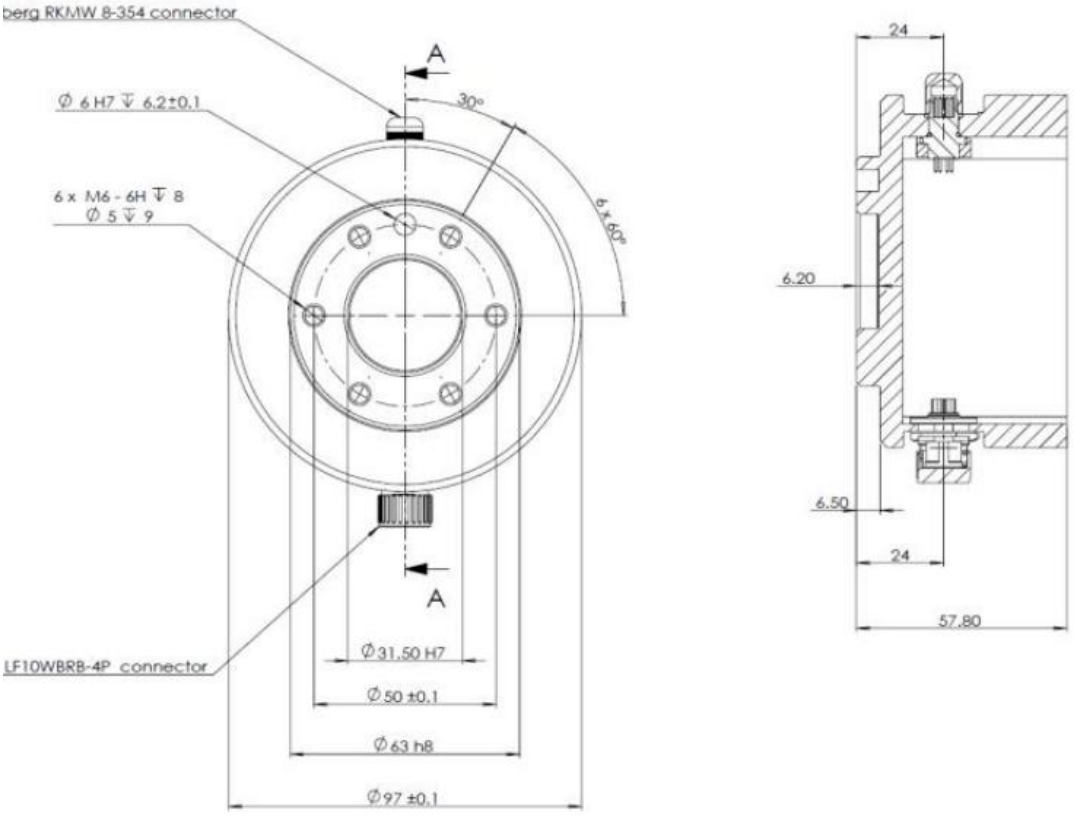


Figure Appendix-16