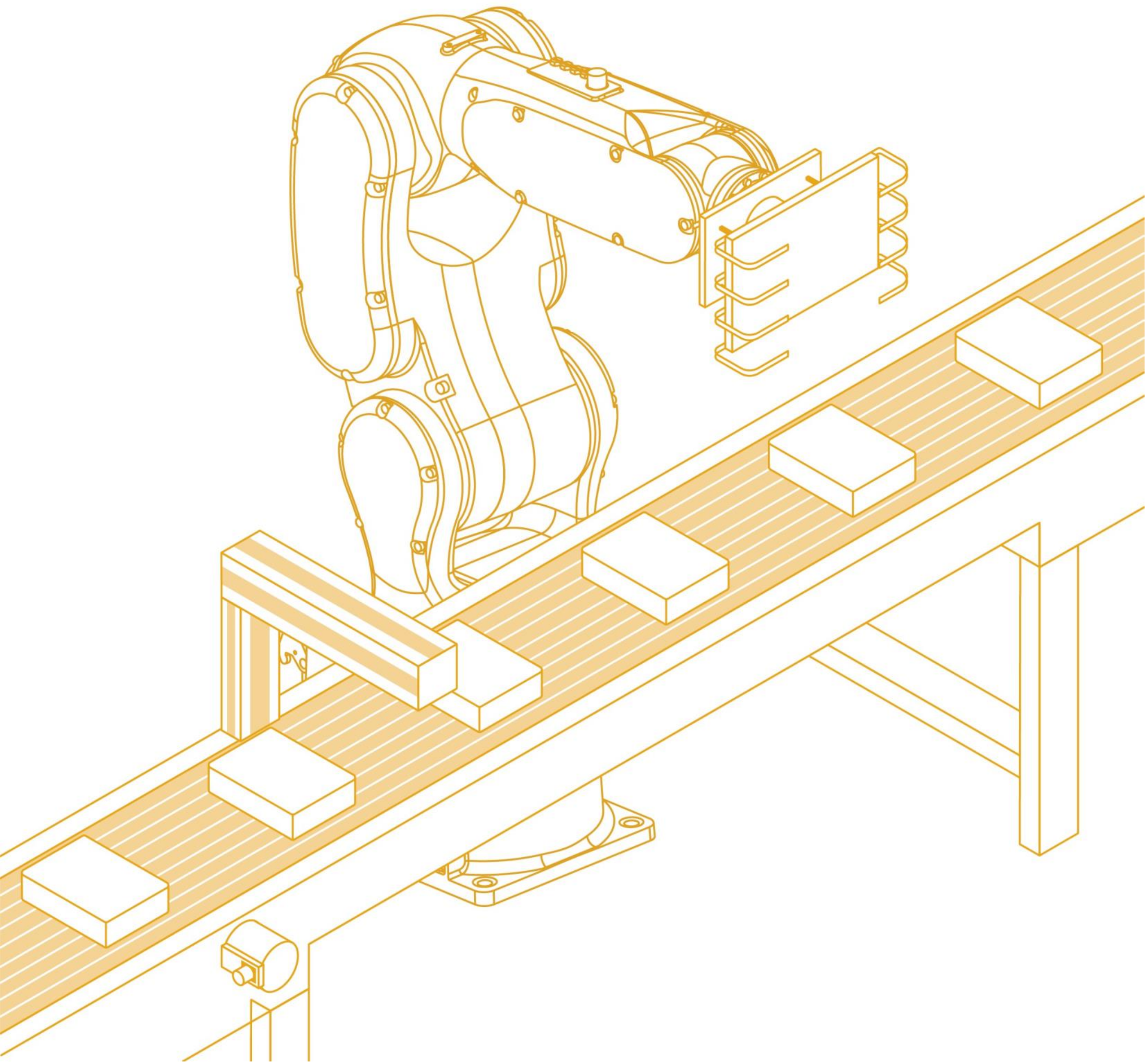
Conveyor Tracking Function Instruction

V1.5.0



Foreword

About this manual

This manual introduces the robot conveyor belt tracking function, and describes in detail the hardware coupled and software configuration process of the conveyor belt tracking function. Reading this document will help readers understand how the conveyor tracking function works and how to use it.

Operating prerequisites

Before operating the robot, please be sure to carefully read the relevant safety instructions and operating instructions of the product. Users must understand safety knowledge and basic operating knowledge before using the conveyor belt tracking function package.

Please refer to:

- "AIR-TP Teach Pendant Operation Manual"
- "ARL Programming Manual"
- "AIR Series Industrial Robot System Troubleshooting and Treatment Manual"

Target groups

- Operator
- Product technician
- Technical service personnel
- Teacher

Common logo meanings

The symbols appearing in the manual and their meanings are detailed in Table 1 below.

Table 1 Logo used in this article

Logo	Meaning
Danger	If instructions are not followed, an accident may occur, resulting in serious or fatal injury.
Warning	If instructions are not followed, an accident may occur, resulting in moderate or minor injuries, or only material damage.
Notice	Prompts you to pay attention to the environmental conditions and important matters, or quick operation methods

Logo	Meaning
Tip	You are prompted to refer to other literature and instructions for additional information or more detailed operating instructions for acquire

Manual description

The content of this manual will be supplemented and modified. Please pay attention to the "Download Center" of our company's website regularly for the latest version of the acquire manual.

Website URL: http://robot.peitian.com/

Revision history

Revision history accumulates descriptions of each document update. The latest version of document contain contains the updated content of all previous document versions.

Table 2 Logo used in this article

Version	Release time	Modify the description
V1.0	2019/07/18	First official release
V1.1.0	2020/06/30	Second official release Software version upgraded to V2.6.2
V1.2.0	2020/09/30	The third official release Software version upgraded to V2.6.3 Add program running multiplication function Added visual trigger configuration example
V1.3.0	2021/03/20	The fourth official release Software version upgraded to V2.6.4 Added arc conveyor belt tracking function
V1.4.0	2022/06/13	The fifth official release Software version upgraded to V2.6.5 Add visual calibration Add conveyor belt related functions
V1.5.0	2023/05/30	The sixth official release Software version upgraded to V2.6.6

Document number and version

See Table 3 for document related information.

Table 3 Document related information

File name	"Conveyor Tracking Function Instruction Manual"
Document number	UM-ZY0206-BJ-001
Document version	V1.5.0

File name	"Conveyor Tracking Function Instruction Manual"
HMI version	2.6.6

Notation convention

See Table 4 for document-related symbol conventions.

Table 4 Symbol Conventions

Format	Significance
<>	Angle brackets "<>" represent the button name, such as "Click the <Yes> button".
[]	"[]" with square brackets indicates the window name, menu name and data table, such as "pop up the [New User] window".
/	Multi-level menus are separated by "/". For example, [File/New/Folder], a multi-level menu represents the [Folder] menu item under the [New] submenu under the [File] menu.

Contents

1 Introduction to conveyor tracking function

1.1 Conveyor tracking system definition

The conveyor belt dynamic tracking system uses a robot to dynamically track work object moving on the conveyor belt. A synchronous switch or vision system is installed on the conveyor belt. When work object passes the synchronous switch or camera, a synchronous trigger signal is generated and transmitted to the robot control system, and then combined with the conveyor belt position data fed back by the encoder is used to calculate the position of work object on the conveyor belt. Subsequently, the robot adjusted its own movement speed and terminal posture in real time based on the position information of work object to track work object moving on the conveyor belt, and finally realized the robot's dynamic tracking and grabbing of the target work object.

The structure of the conveyor belt tracking system is shown in Figure 1-1, and the names of each part of the system are shown in Table 1-1.

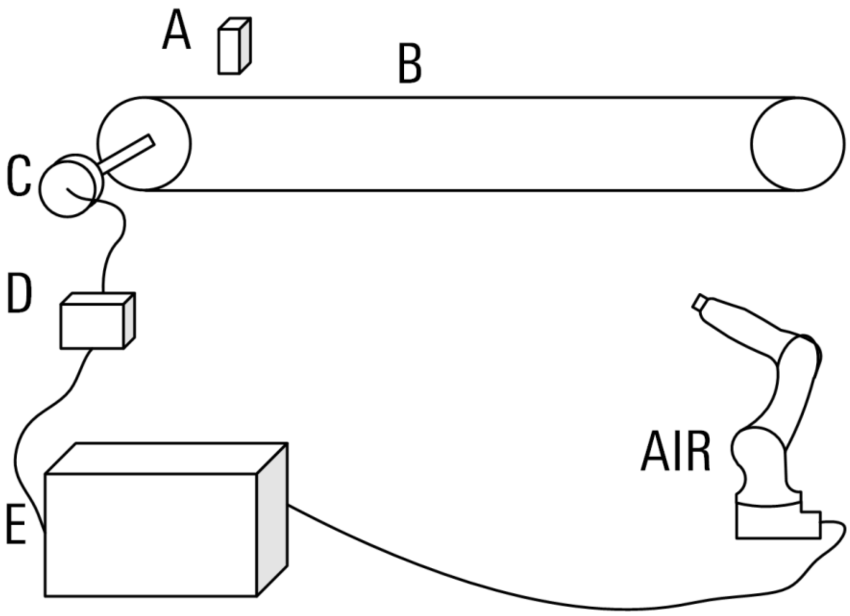


Figure 1-1 Diagram of conveyor tracking system

Table 1-1 Number description

Serial number	Illustrate
A	Photoelectric switch/camera
B	Conveyor belt
C	Encoder/motor (motor + servo drive mode)
D	Servo drive (motor + servo drive method)
E	Control cabinet
AIR	Manipulator

1.2 Explanation of proper nouns

The following terms will be used in the requirement definition of the conveyor tracking function, and their specific definitions are:

■ Record	After work object crosses the photoelectric switch on the conveyor belt, its position data is written to the work object queue and updated in real time.
■ Association	work object is connected to the mobile work object coordinate system, and the mobile work object coordinate system moves with work object.
■ Freed	The connection between the associated mobile work object coordinate system and work object is released.
■ Waiting for association	After the currently associated work object is released, the next work object to be associated first is in a waiting state for association.
■ Get on	The process of the robot TCP moving from the target point based on the fixed work object coordinate system or other work object coordinate system independent of the conveyor belt to the target point on the mobile work object coordinate system associated with the conveyor belt is the "get on the car" process.
■ Get Off	The process of the robot TCP moving from the target point on the moving work object coordinate system associated with the conveyor belt to the target point based on the fixed work object coordinate system or other work object coordinate system independent of the conveyor belt is the "getting off" process.

1.3 Feature

Accuracy

In automatic mode, when the conveyor belt runs at a constant speed of 150mm/s, the robot's tool center point (TCP) will remain within +/-2mm of the calibration point. This value depends on the calibration of the robot and the conveyor belt, and only applies to linear conveyor belt tracking.

Work object queue

In the conveyor belt tracking function, a work object that has passed the photoelectric switch will enter the work object queue in order. The work object queue can have up to contain 100 work objects.

Getting on area

After work object in the work object queue enters the getting on area with the conveyor belt, they will start waiting to be tracked in turn.

Access queue

ARL programs can access the current position of the conveyor belt through the cjoint function. You can also delete the first work object in the tracking queue or all work object in the queue through the ARL instructions.

Processing interval

By specifying the processing interval, you can prevent the robot from tracking in non-working areas or non-safe areas.

Calibration of conveyor belts

The teach pendant provides a calibration method to calibrate the position and direction of conveyor belt motion in the robot's workspace. Linear conveyor belts can be used in any position and direction.

1.4 Conveyor belt tracking principle

Describe

In conveyor tracking, the robot tool center point (TCP) will automatically follow the associated work object on the conveyor. When tracking the conveyor belt, when the speed of the conveyor belt changes, the relative speed specified by the program will still be maintained between the robot TCP and work object. The conveyor belt tracking control is established on the work object coordinate system.

Conveyor belt as mechanical unit

In the conveyor tracking function, the conveyor belt is treated as a mechanical unit but is not controlled by the teach pendant. As a mechanical unit, the teach pendant can activate and deactivate conveyor belts.

Waiting for work object on the conveyor belt

For the conveyor belt tracking function, the work object on the conveyor belt is detected by a photoelectric switch. All work objects that pass through the photoelectric switch and are within a specified distance in the working area can be tracked. Before starting to track the conveyor belt, the program must first check if there is a work object available on the conveyor belt. If work object is available, program execution continues; if work object is not available, the program will wait until work object is available.

Linked to work object

ARL instruction waitwobj is used to wait for work object on the conveyor before starting conveyor tracking. When waitwobj instruction is successfully executed, the conveyor belt work object is associated with the ARL program. When the ARL program has been associated with work object on the conveyor belt, the robot motion instruction can refer to the work object coordinate system. When referenced to the work object coordinate system associated with the conveyor, all motion is relative to work object on the conveyor.

> Tip
>
> By moving the association between the work object coordinate system and work object, you can switch between different work objects on the conveyor belt, but you can only associate one conveyor belt work object at a time.

Disassociate from work object

When one of the following situations occurs, the association between the work object coordinate system and work object will be disconnected:

- Work object exceeds the maximum distance defined by the conveyor belt
- Control cabinet restart
- Program reset

Work object queue

The work object queue is a group of work objects within a specified distance relative to the conveyor belt and photoelectric switch, as shown in Figure 1-2. For descriptions of letters and serial numbers in the figure, see Table 1-2 and Table 1-3.

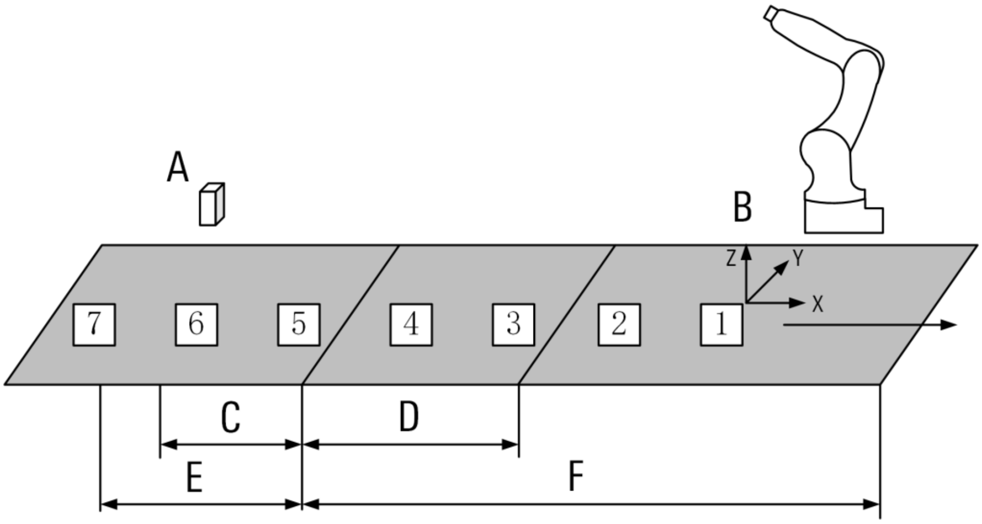


Figure 1-2 Work object queue diagram

Table 1-2 Serial number description

Serial number	Illustrate
A	Photoelectric switch
B	Move the work object coordinate system
C	Waiting area for getting on (that is, the distance between the zero position of the getting on area and the photoelectric switch)
D	Getting area
E	Maximum distance in negative direction
F	Processing interval, that is, the robot working area
1~7	Work objects 1~7, the related status description is shown in Table 1 3.
Tip	Only when work object is located between the maximum negative distance and the maximum positive distance can the work object coordinate system be associated with it. The maximum negative distance can be greater or smaller than the zero offset distance. If the conveyor belt stops and moves backwards, when work object crosses the negative maximum distance, the moving work object coordinate system associated with it will be released.

Table 1-3 Serial number description

Serial number	Illustrate
1	Work object 1 is in the associated state of moving the work object coordinate system, and the work object coordinate system is bound to the motion of work object 1;
2	Work object 2 has exceeded the getting on area and will be skipped;
3	The movements of work object 3 and work object 4 are tracked in the getting on area. When work object

Serial number	Illustrate
4	1 is released, work object 3 will give priority to establishing the work object coordinate system association; If work object 3 is associated and then released, while work object 4 is still in the getting on section, work object 4 will establish the association first; if work object 3 is still in the getting on section when work object 4 is released, work object 3 will still establish the association first; (this case applied to reciprocating processing of two work objects, if A work object is associated and B work object is waiting for association, after A is released, even if its position data is still retained in the work object queue, the work object that will be associated first will still be B)
5	Work object 5 and work object 6 have entered the waiting area for getting on, but have not yet entered the getting on area. Their movements are recorded. When work objects 1~4 are released, work object 5 will establish the association first, but will not process immediately after the association is established until work object 5 enters the getting on area. If work object 5 is associated and then released, and work object 6 is also released after being associated, and work object 5 has not yet passed the getting on area, the association will still be established first;
6	
7	Work object 7 has not yet passed the photoelectric switch, and its movement will not be recorded yet.

Coordinate System

Figure 1-3 shows the main coordinate systems used in conveyor tracking. The serial number description in the figure is as shown in Table 1-4.

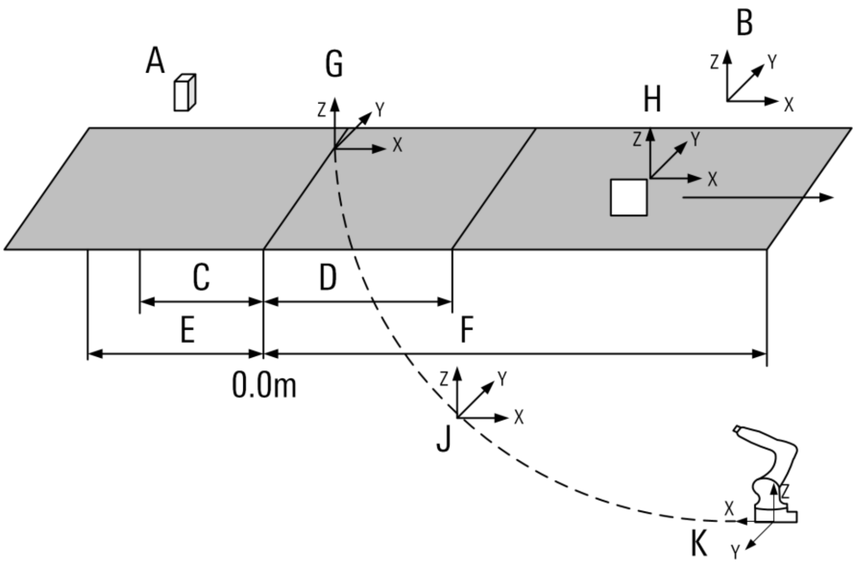


Figure 1-3 Diagram of the main coordinate systems used in conveyor belt tracking

Table 1-4 Serial number description

Serial number	Illustrate
A	Photoelectric switch
G	Conveyor belt base coordinate system (fixed to the zero position of the conveyor belt, that is, the zero point of the processing starting area)
H	Conveyor belt user coordinate system (follows the movement of the conveyor belt, its specific position depends on the work object currently being associated)
B	Move work object coordinate system (associated with move work object)

Serial number	Illustrate
C	Waiting area for getting on (that is, the distance between the zero position of the getting on area and the photoelectric switch)
D	Getting on area
E	Maximum distance in negative direction
F	Processing interval
J	World coordinate system
K	Robot base coordinate system

Tip

- The conveyor base coordinate system is defined based on the world coordinate system.
- If the linear conveyor belt is regarded as a mechanical unit with 1 mobile axis, then the conveyor belt user coordinate system is equivalent to the "flange coordinate system" of the mechanical unit. The position of the mobile axis of the mechanical unit depends on the currently associated work object. When work object passes through the zero point of the conveyor belt, the conveyor belt user coordinate system coincides with the conveyor belt base coordinate system. When work object moves forward, the conveyor belt user coordinate system also moves accordingly. When the associated work object is switched, the conveyor belt user coordinate system jumps. Change to the location of the new associated work object.
- The position of the conveyor belt user coordinate system is equivalent to the axis position of the conveyor belt mechanical unit, and the axis position can be displayed in teach pendant.
- The mobile work object coordinate system is defined based on the conveyor belt user coordinate system. Since the conveyor belt user coordinate system coincides with the conveyor belt base coordinate system when work object is at the conveyor belt zero position, the work object coordinate system calibration can be performed when work object is at the conveyor belt zero position.
- For a straight conveyor belt, the x direction of the conveyor belt base coordinate system is along the positive direction of movement of the conveyor belt, and the z direction coincides with the z direction of the world coordinate system upward by default.
- The conveyor belt can be arranged in any position and transmission direction within the processing range of the robot.

2 Hardware coupled

2.1 Coupled method of conveyor belt

2.1.1 Coupled motor and servo drive

inCube20/22 control cabinet coupled method

Figure 2-1 takes the AE5115 series AC servo driver as an example to illustrate the connection method of the servo driver with the inCube20/22 control cabinet and multiple motors. For each interface coupled mode and interface description of the AE5115 series servo driver, please refer to the "AE5115 AC Servo Driver User Manual".

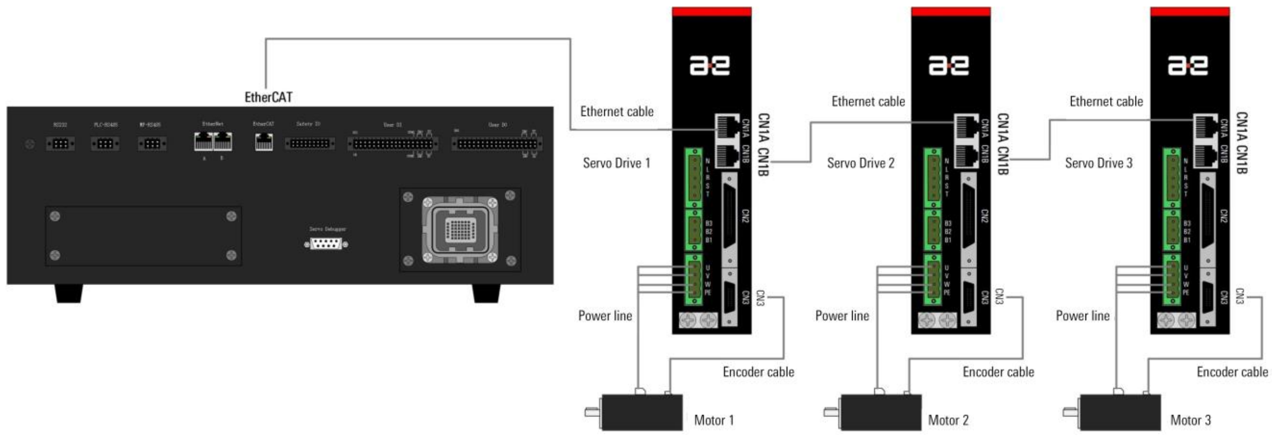


Figure 2-1 Diagram of inCube20/21 connecting three motors

Servo drive and motor holding brake coupled method

The connection method between the servo drive and the motor holding brake is as shown in Figure 2-2.

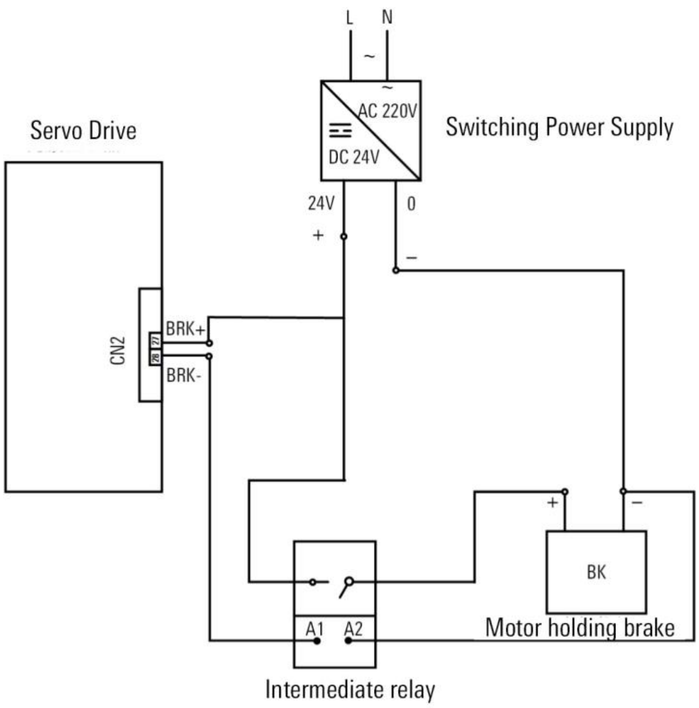


Figure 2-2 Diagram of the connection between the servo driver and the motor holding brake

Debugging mode

After completing the device coupled, the parameters of the servo drive need to be debugged according to the actual load. For the parameter debugging method of the AE5115 series servo driver, please refer to the "AE5115 AC Servo Driver User Manual".

2.2 Coupled method of camera and photoelectric switch

Both the photoelectric switch and the camera communicate with the control cabinet through I/O terminal modules. The photoelectric switch requires DI, 24V and GND of the coupled user I/O terminal module. The camera needs to be connected to the DO, 24V, and GND of the user's I/O terminal modules.

IO configuration instructions

The user I/O coupled terminal block of the inCube20/22 control cabinet is an optional product. This option uses a 5m wire length to lead the user I/O signals on the control cabinet panel to the terminal block for user convenience.

For information about the material number of the user I/O connection terminal block of the inCube20/22 control cabinet, see Table 2-1.

Table 2-1 User I/O connection terminal block material number of inCube20/22 control cabinet

Name	Material number
inCube20_User I/O coupled terminal block	PC5100000096

For details on the configuration of the user I/O connection terminal block of the inCube20/22 control cabinet, see Table 2-2.

Table 2-2 Main configuration table of user I/O connection terminal block of inCube20/22 control cabinet

Serial number	Name	Specification	Adaptation control cabinet	Material number	Construct dosage
1	inCube20-User DI terminal module cable	5m	inCube20/22	P04082001304	1
2	inCube20-User DO terminal module cable	5m		P04082001305	1

Pin definition

Tip

For the user I/O definition of inCube20/22 control cabinet, refer to Table 2-4 For more information, please refer to our company's "inCube20 Control Cabinet Manual" and "inCube22 Control Cabinet Manual".

For the diagram and description of the user I/O connection terminal block, please refer to Figure 2-3 and Table 2-3 respectively.

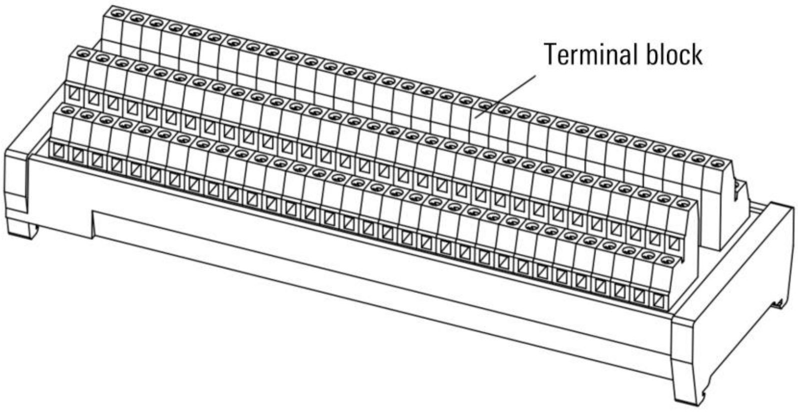


Figure 2-3 inCube20/22 user I/O connection terminal block diagram

Table 2-3 User I/O connection terminal block composition description

Name	Illustrate
Terminals	For pin definitions, see Table 2 4
Weidmüller quick plug terminal coupled	For pin definitions, see Table 2 4

Table 2-4 User I/O definition for inCube20/22 control cabinet

User DI		User DO	
Quick plug terminal coupled device pin	Definition	Quick plug terminal coupled device pin	Definition
A1	OPERATED_DI1	B1	OPERATED_DO1
A2	OPERATED_DI14	B2	OPERATED_DO14
A3	OPERATED_DI2	B3	OPERATED_DO2
A4	OPERATED_DI15	B4	OPERATED_DO15
A5	OPERATED_DI3	B5	OPERATED_DO3
A6	OPERATED_DI16	B6	OPERATED_DO16
A7	OPERATED_DI4	B7	OPERATED_DO4
A8	OPERATED_DI17	B8	OPERATED_DO17
A9	OPERATED_DI5	B9	OPERATED_DO5
A10	OPERATED_DI18	B10	OPERATED_DO18
A11	OPERATED_DI6	B11	OPERATED_DO6
A12	OPERATED_DI19	B12	OPERATED_DO19
A13	OPERATED_DI7	B13	OPERATED_DO7
A14	OPERATED_DI20	B14	OPERATED_DO20
A15	OPERATED_DI8	B15	OPERATED_DO8
A16	OPERATED_DI21	B16	OPERATED_DO21
A17	OPERATED_DI9	B17	OPERATED_DO9

User DI		User DO	
Quick plug terminal coupled device pin	Definition	Quick plug terminal coupled device pin	Definition
A18	OPERATED_DI22	B18	OPERATED_DO22
A19	OPERATED_DI10	B19	OPERATED_DO10
A20	OPERATED_DI23	B20	OPERATED_DO23
A21	OPERATED_DI11	B21	OPERATED_DO11
A22	OPERATED_DI24	B22	OPERATED_DO24
A23	OPERATED_DI12	B23	OPERATED_DO12
A24	OPERATED_DI25	B24	OPERATED_DO25
A25	OPERATED_DI13	B25	OPERATED_DO13
A26	OPERATED_DI26	B26	OPERATED_DO26
A27	DI_COM_1	B27	Not Connected
A28	DI_COM_2	B28	Not Connected
A29	D+24V_EX	B29	D+24V_EX
A30	D+24V_EX	B30	D+24V_EX
A31	D+24V_EX	B31	D+24V_EX
A32	D+24V_EX	B32	D+24V_EX
A33	GND_EX	B33	GND_EX
A34	GND_EX	B34	GND_EX
A35	GND_EX	B35	GND_EX
A36	GND_EX	B36	GND_EX

3 Permission level for system parameter configuration

When using AIR-TP teach pendant for the first time, the user interface when logging in for the first time will be prompted. The user can choose:

- Teacher: Permission 4

  You can write robot work programs and other operations, and have permission to modify some parameters. The initial login password is: PEACE.

- Operator: Permission 5

  You can simply check the operation status of the robot's position parameters without permission to modify the program or parameters. The initial login password is: LOVE.

Tip	Ordinary users can only log in to teach pendant with teacher or operator permissions.

4 (Optional) Calibration of tool coordinate system

Before the belt calibration, the calibration of the tool coordinate system can be completed first, which will improve the accuracy of the belt calibration. Generally, the tool coordinate system is calibrated using the conventional 4-point method.

The calibration process of the tool coordinate system is as follows:

Step1. Click [Run/Calibrate/Coordinate cali] on the upper right side of the main interface of the teach pendant (as shown in Figure 4-1) to enter the [Coor Sys measure] interface.

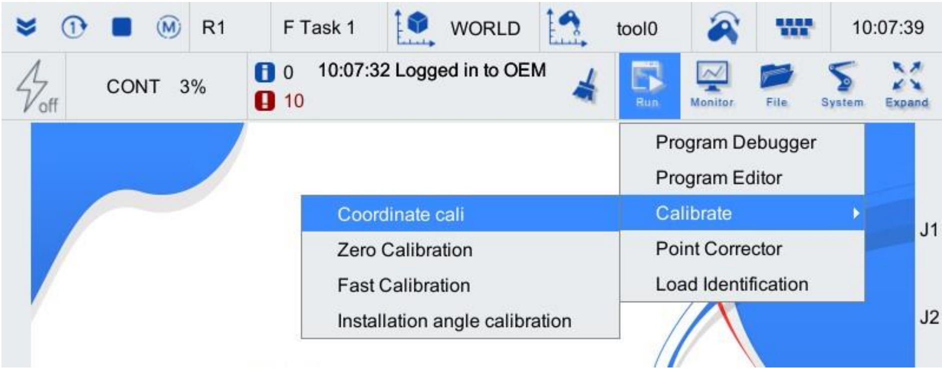


Figure 4-1 [Coor Sys measure] interface

Step2. Select "Tool" in [Coordinate System Type], click to select the row of the tool coordinate system to be calibrated, as shown in Figure 4-2.

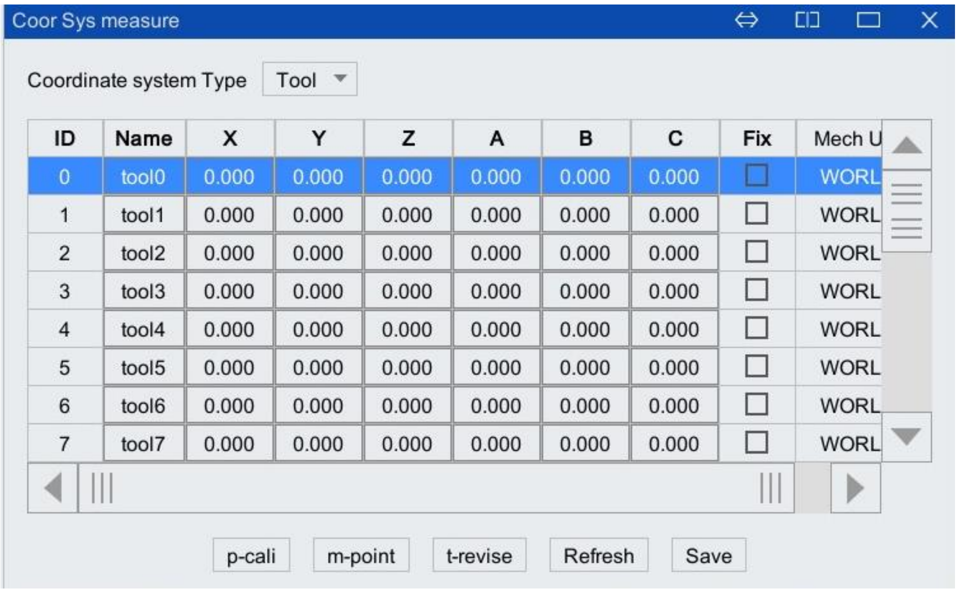


Figure 4-2 Select the row containing the tool coordinate system

Step3. Click <m-piont >, and the configuration interface will pop up as shown in Figure 4-3.

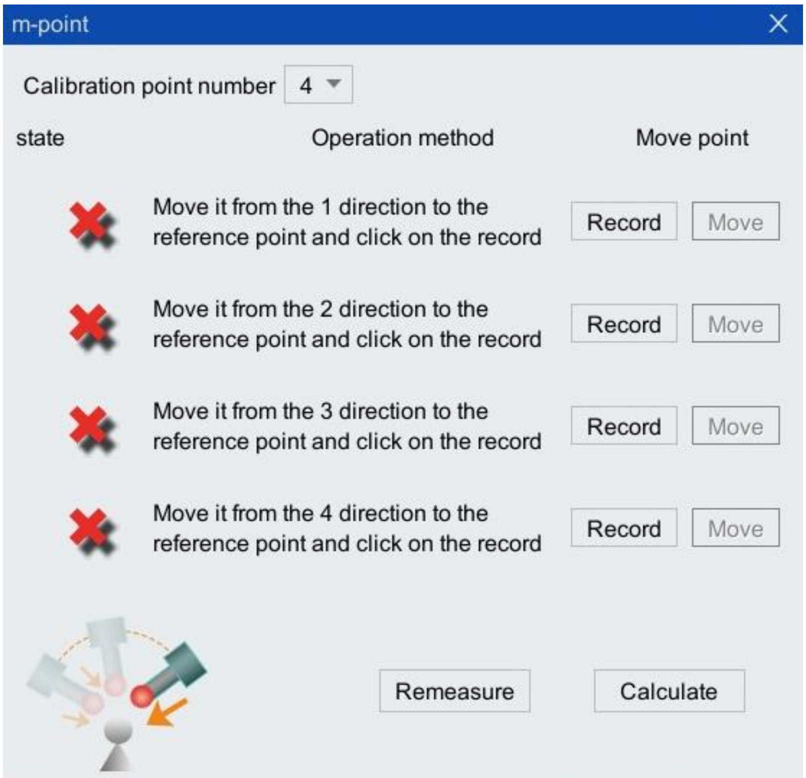


Figure 4-3 "Four-point method" calibration interface

Step4. Just follow the operating instructions in the figure for calibration. After each point is successfully calibrated, the × in front will change to √, as shown in Figure 4-4.

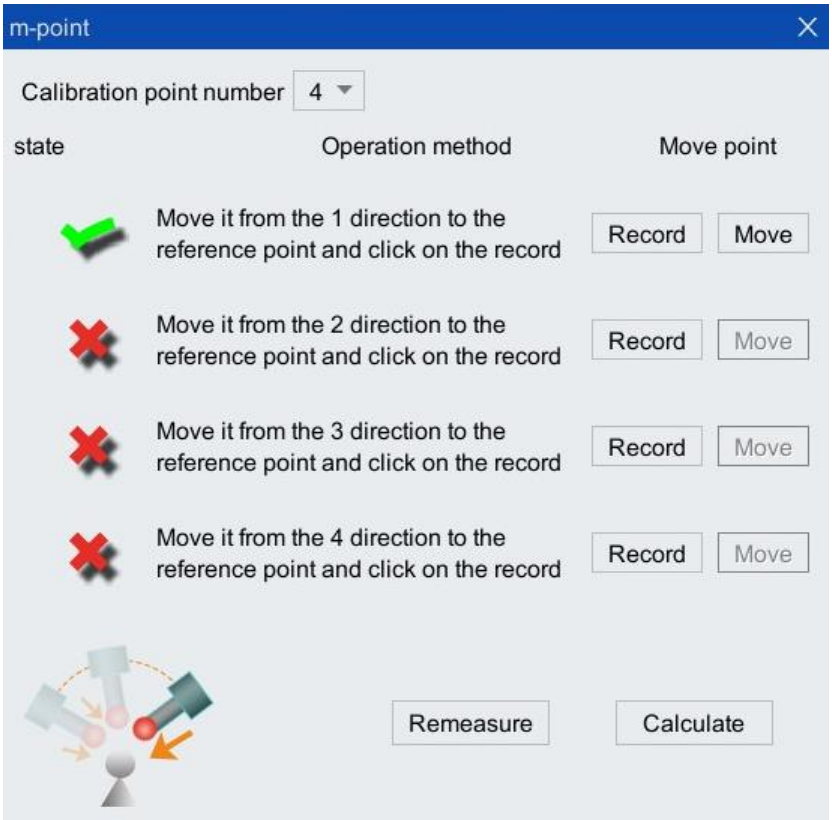


Figure 4-4 Indicates the successful 1 point interface state

Step5. After all calibrations are successful, all × will become √, as shown in Figure 4-5. Click the <Calculate> button to calculate the error. If it exceeds the error range, it needs to be re-calibrated. If it is within the allowable error range, the calibration is completed.

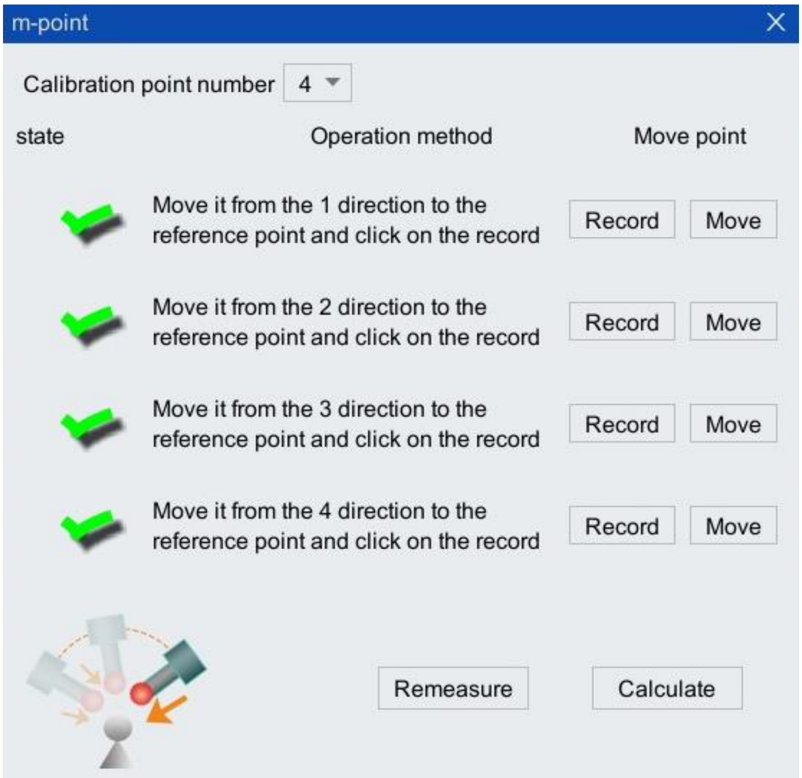


Figure 4-5 Interface status after all 4 points have been calibrated

Step6. After the calibration of the tool coordinate system is completed, click <Save>, and click <Yes> in the pop-up prompt dialog box to save the configuration.

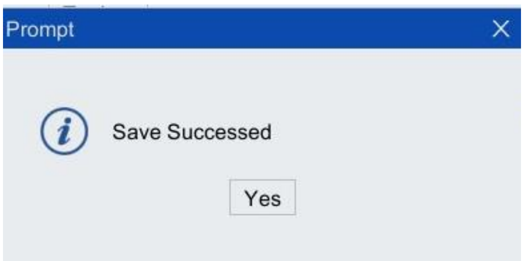


Figure 4-6 Save dialog box

5 Conveyor belt parameter configuration and calibration

Before using the conveyor belt dynamic tracking function, please complete the relevant parameter configuration of the system.

5.1 Activate conveyor belt

The configuration steps are as follows:

Step1. Click [Expand/Conveyor] in the upper right corner, as shown in Figure 5-1.

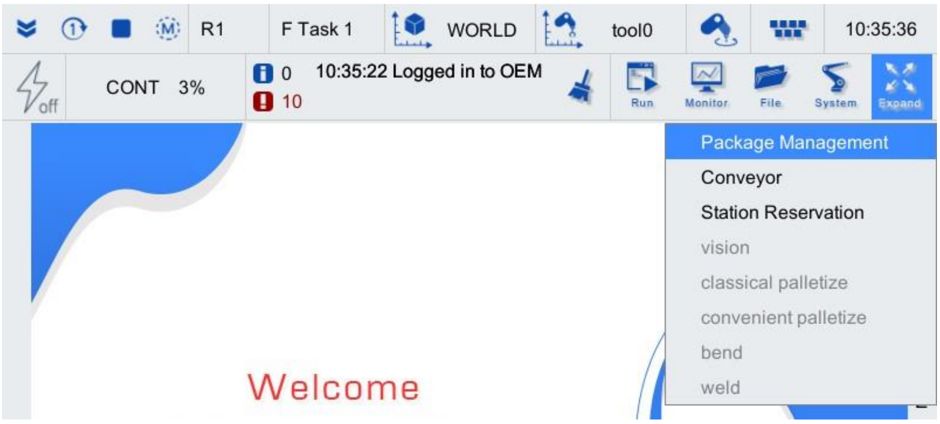


Figure 5-1 Teach pendant main interface diagram

Step2. Enter the [Conveyor setting] interface, and click <Activate> at the upper left corner of the interface, as shown in Figure 5-2.

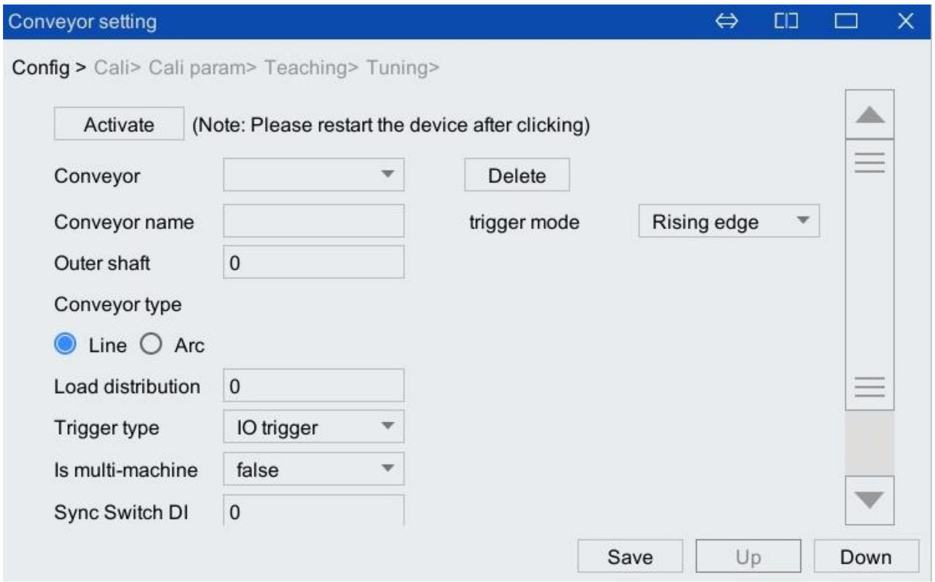


Figure 5-2 Activate the conveyor belt operation interface

Step3. Click <Save> to restart the control cabinet and take effect.

5.2 Configure conveyor belt

5.2.1 Photoelectric mode

Basic parameter configuration

Step1. Enter the [Conveyor setting] interface again. The basic parameters in the configuration interface are as shown in Figure 5-3, and the parameter description is as shown in Table 5-1. After the configuration is complete, click <Save> and restart the control cabinet for the configuration to take effect.

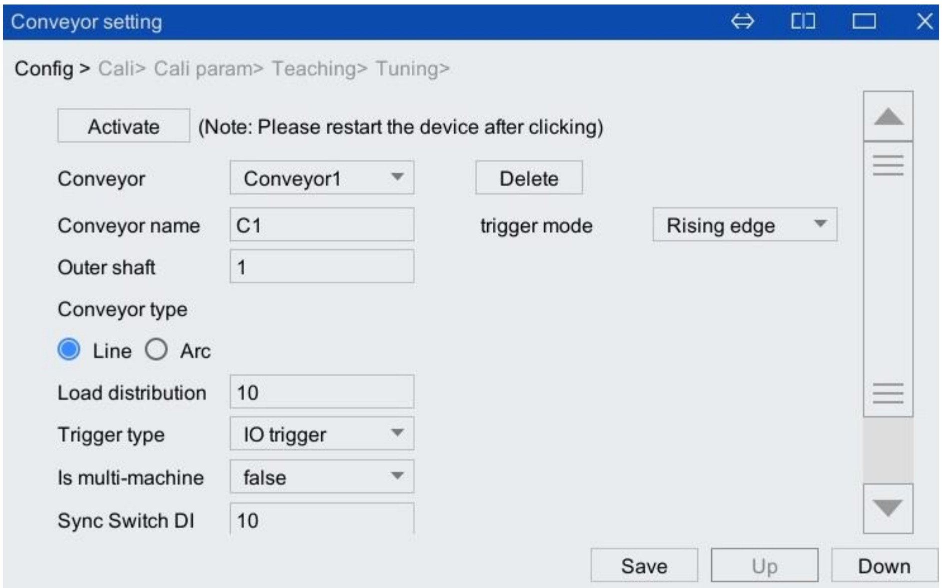


Figure 5-3 Activate the basic parameter configuration interface of the conveyor belt

Table 5-1 Parameter description

Parameter	Illustrate
Conveyor	Select the conveyor belt that needs to be configured.
Conveyor name	User can set the name of the conveyor belt
Outer shaft	The external axis serial number corresponding to the conveyor belt, which corresponds to the first external axis
Conveyor type	The type of conveyor belt used, select "line" here
Trigger type	Conveyor belt work object trigger mode, select "IO trigger" here
Is multi-machine	Whether to use multi-machine conveyor belts, select "false" here (multi-machine conveyor belts are not supported)
Syn Switch DI	Photoelectric switch DI. This value is the port number for communication between the actual photoelectric switch and the robot DI.
Trigger mode	Conveyor belt DI trigger mode, the values are as follows: ■ Rising edge trigger (the following uses rising edge trigger as an example to illustrate the configuration) ■ Falling edge trigger
<Delete>	Delete the currently configured conveyor belt.
<Save>	After the configuration is complete, click <Save>. The configuration takes effect after the restart

Base coordinate system calibration

Step2. Click <Down> to enter the [Cali] interface, and click the <Calibrate> button in the upper part of [Base coordinate system calibration]. As shown in Figure 5-4.

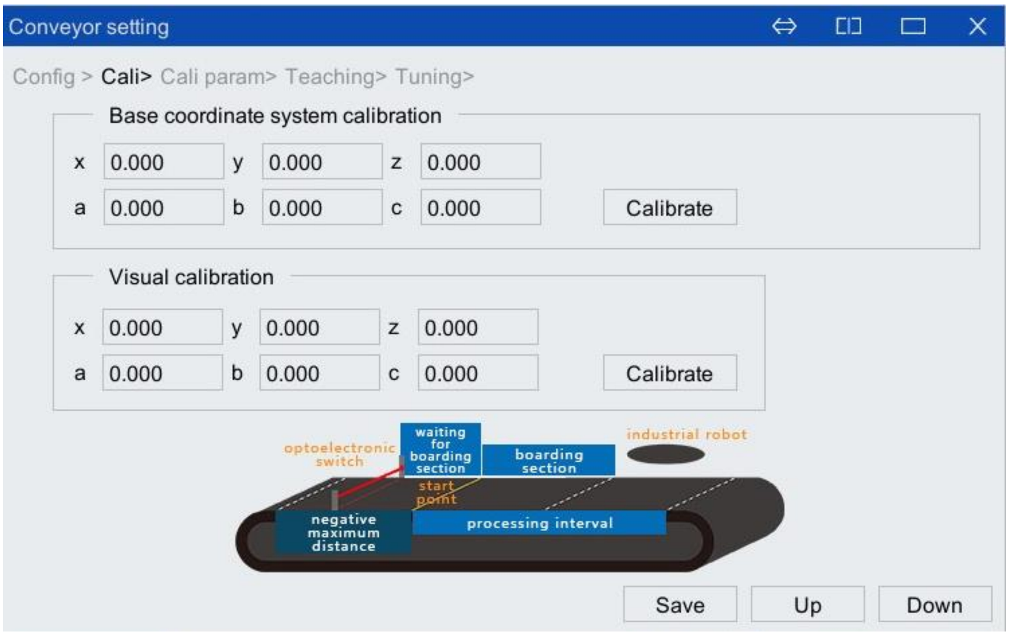


Figure 5-4 Conveyor belt calibration interface

Step3. Enter the [Calibration of base coordinate system of conveyor] interface, as shown in Figure 5-5. The calibration steps are as follows:

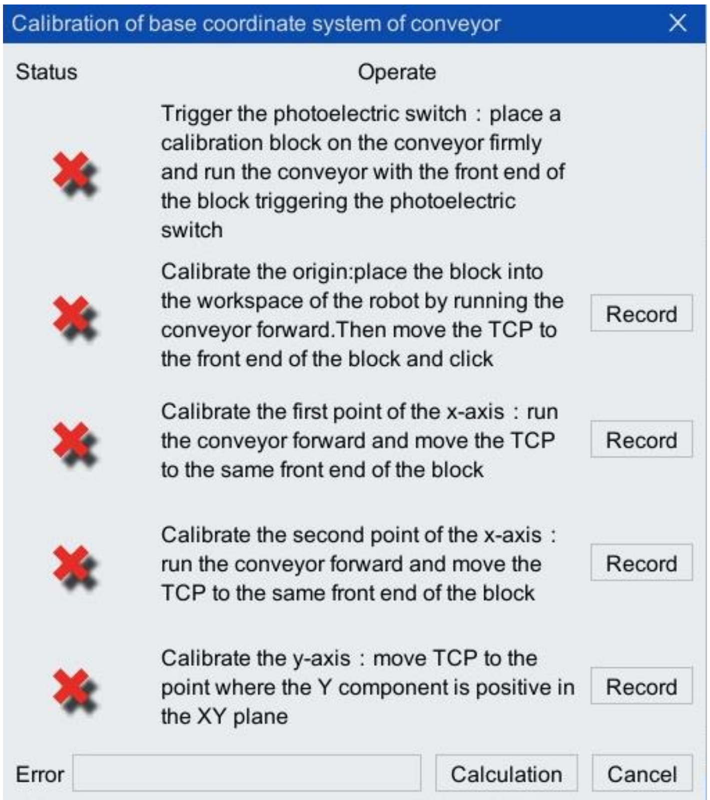


Figure 5-5 Conveyor belt base coordinate system calibration interface

Step4. The interface needs to record five points to complete the calibration. Follow the operating instructions in Figure 5-5 to complete the calibration of the base coordinate system. For the scene of the work object movement during the calibration process, please refer to Figure 5-6. After successful recording, the × in front will change to √.

> Tip
>
> It is recommended to select the origin on the edge of the conveyor belt and specify that the positive direction of X is consistent with the rotation direction of the conveyor belt.

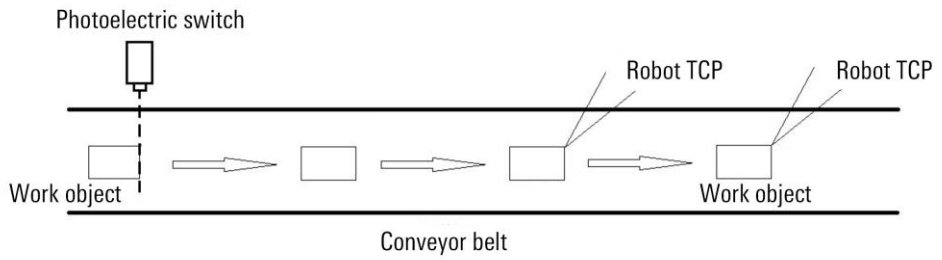


Figure 5-6 Diagram of work object movement on the conveyor belt

Step5. After all calibration is successful, click the <Calculation> button to calculate the error. If it exceeds the error range, it needs to be re-calibrated. If it is within the allowable error range, the calibration is completed.

(Optional) Calibration of mobile work object coordinate system

When you need to change the direction of the coordinate system of moving work object, please calibrate the coordinate system of moving work object. The calibration steps are as follows:

Step6. Place a work object on the conveyor belt and rotate the conveyor belt so that work object passes the photoelectric switch and stops the conveyor belt within the robot's motion range.

Step7. Customize a coordinate system on work object. Usually the point at the front of work object that triggers the photoelectric switch is used as the origin, and the direction of the coordinate system is specified. It is recommended that the direction of the coordinate system be consistent with the direction of the base coordinate system, and be determined one point in X axis and Y axis respectively.

Step8. Click [Run/Calibrate/Coordinate Cali] on the main interface of teach pendant to enter [Coor Sys measure].

Step9. Use the "3-point" to calibrate the moving work object coordinate system. In the [Coor Sys measure] interface in Figure 5-7, select "Wobj " in [Coordinate system Type].

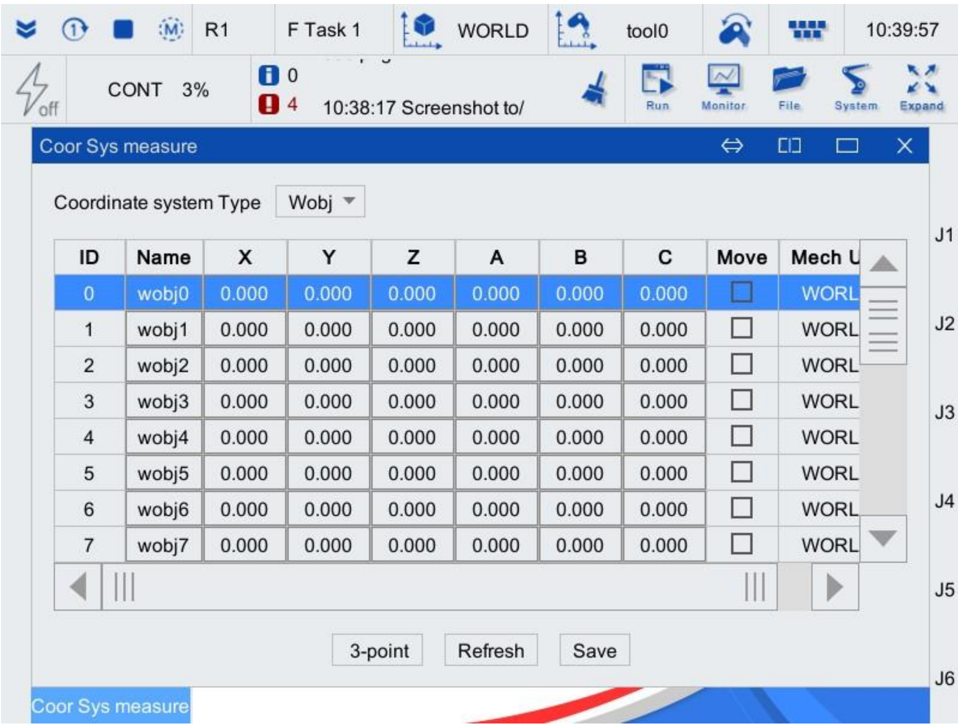


Figure 5-7 Workpiece coordinate system calibration interface

Step10. Click to select the [Name] of the work object coordinate system to be calibrated, and modify the name of the [Mechanical Unit], for example, to "C1" (as shown in Figure 5-8).

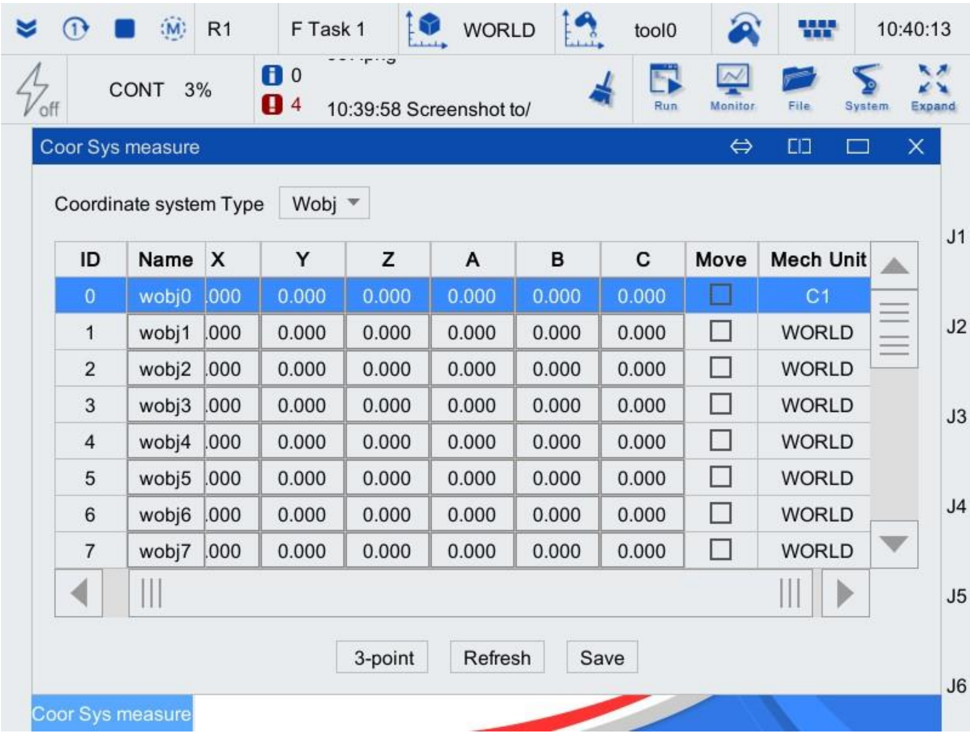


Figure 5-8 Select mechanical unit

Step11. After clicking <3-point>, the prompt interface shown in Figure 5-9 will pop up: Follow the prompts to complete the operation.

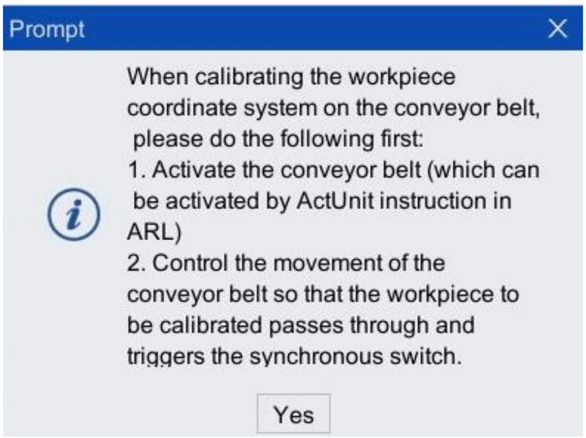


Figure 5-9 "Prompt" interface

Step12. Click <Yes>, and the 3-point calibration method interface shown in Figure 5-10 will pop up. [Tool] select the name of the previously calibrated tool coordinate system.

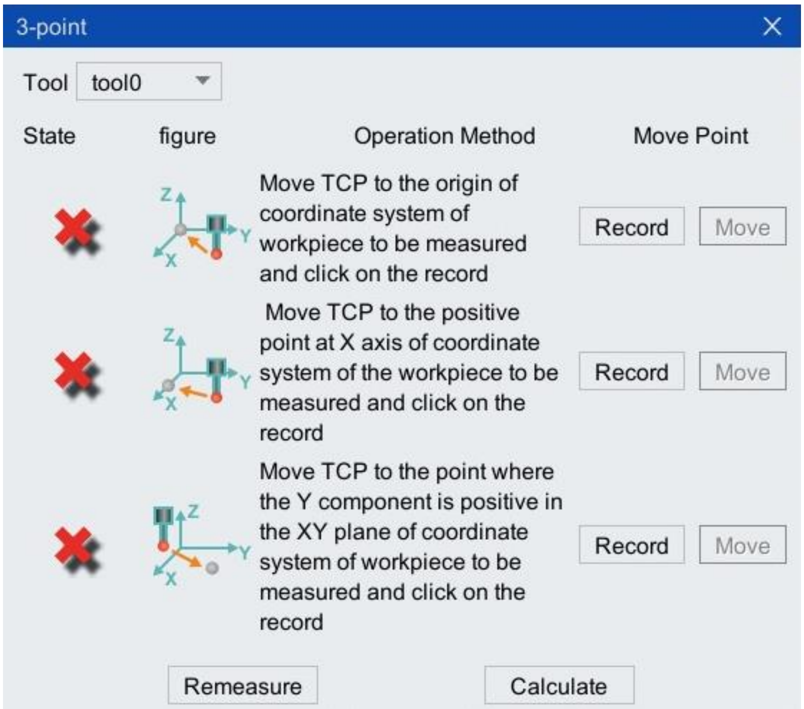


Figure 5-10 "Three-point method" calibration interface

Step13. Next, just follow the instructions in the picture for calibration. After each point is successfully calibrated, the × in front will change to √. After all calibration is successful (such as shown in Figure 5-11), click <Calculate> button to calculate the error. If it exceeds the error range, it needs to be re-calibrated. If it is within the allowable error range, the calibration is completed.

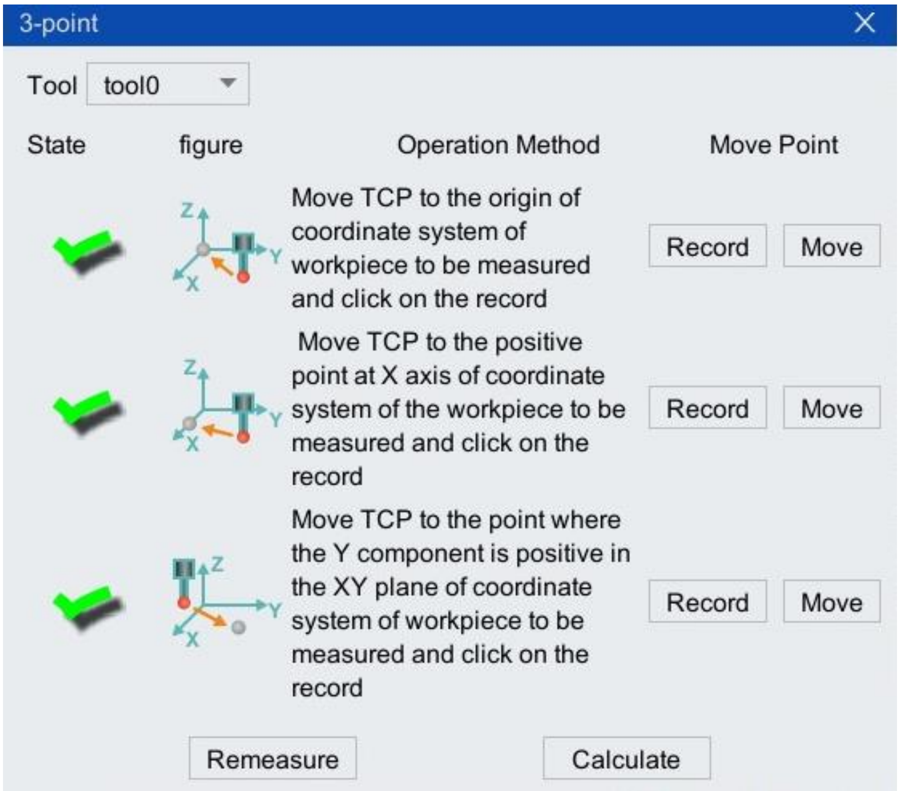


Figure 5-11 Interface status after all 3 points have been calibrated

Step14. Click <Save> at the bottom of the screen to save the configuration.

Calibration parameter settings

Step15. In the conveyor belt configuration interface, click <Down> to enter the [Cali param] interface, where you can adjust the calibration parameters, as shown in Figure 5-12, and the parameter description is as shown in Table 5-2.

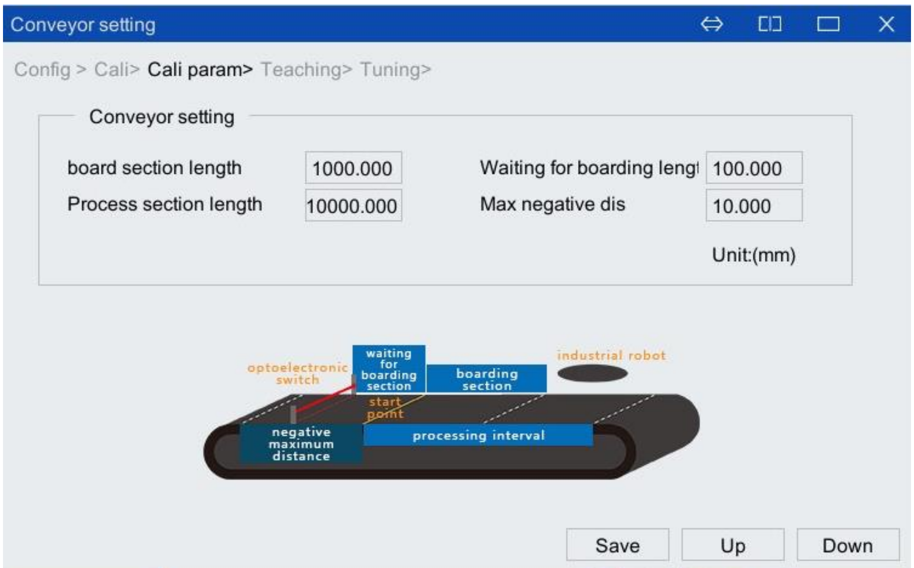
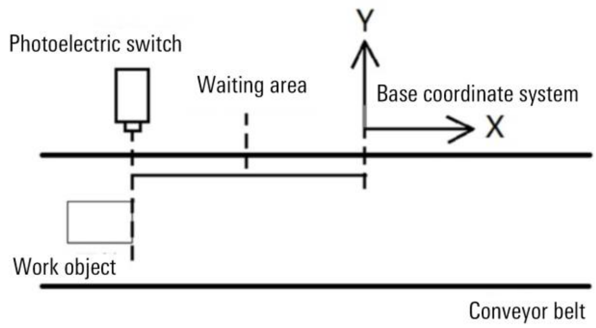


Figure 5-12 Conveyor belt calibration parameter interface

Table 5-2 Parameter description

Parameter	Illustrate
board section length	After work object in the work object queue enters the getting on area with the conveyor belt, they will start waiting to be tracked in turn.
Process section length	The processing area is the distance set along the conveyor moving direction from the zero point of the conveyor base coordinate system. If work object is not actively released after being associated, it will be passively released after reaching this position. See Figure 1 2 for detailed description.
Waiting for boarding length	The waiting area indicates the distance between the origin of the specified conveyor belt base coordinate system and the position of the photoelectric switch. Please refer to the figure below.  The "waiting area for getting on" and the "conveyor base coordinate system" are calibrated in the same series of actions, so these two quantities correspond to the same <Calibration> button. See Figure 1 2 for detailed description.
Max negative dist	The maximum distance in the negative direction is the distance set from the zero point of the conveyor base coordinate system in the opposite direction of the conveyor movement. If work object is not actively released after being associated, it will be passively released after reaching this position. See Figure 1 2 for detailed description.

Conveyor belt teaching

Step16. Click <Down> to enter the "Teaching" interface, as shown in Figure 5-13. The teaching steps are as follows:

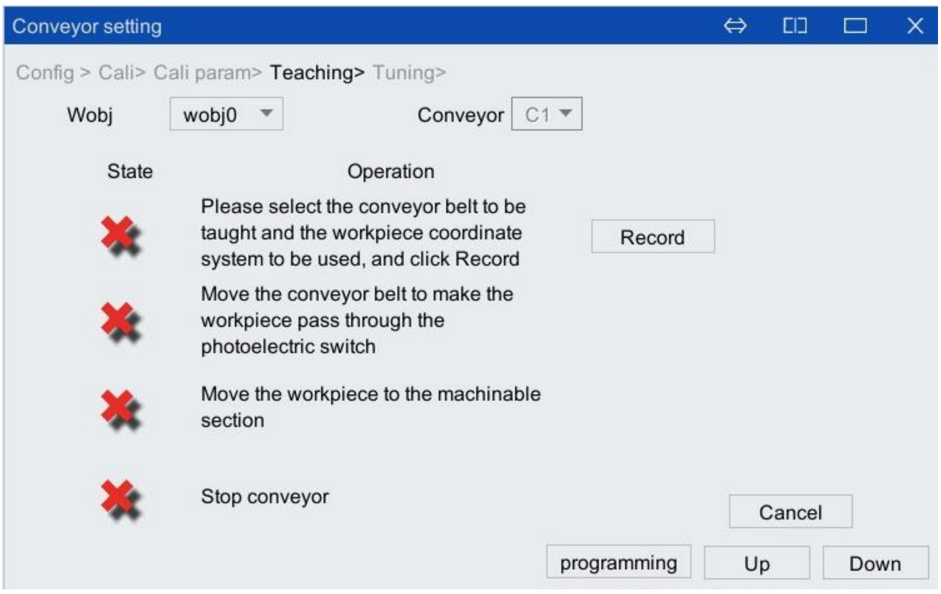


Figure 5-13 Conveyor belt teaching interface

1 In the upper left corner of the interface, select the work object coordinate system you want to use and click <Record>.

2 Move the conveyor belt so that work object passes the photoelectric switch

3 Continue to move the conveyor belt until the × in the third line of the interface changes to √

4 Stop conveyor belt

5 Move the TCP to the processing point on work object that needs to be processed, and click < Get Location > on the relevant point in ARL

Parameter fine-tuning

Step17. Click <Down> to enter the [Tuning] interface, where parameters can be adjusted according to specific tracking conditions, as shown in Figure 5-14. The number description is shown in Table 5-3.

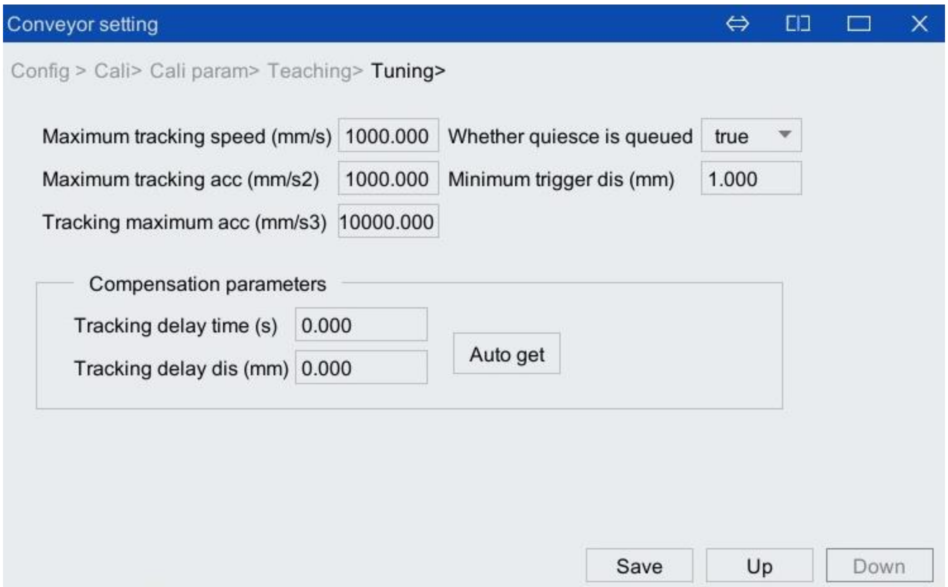
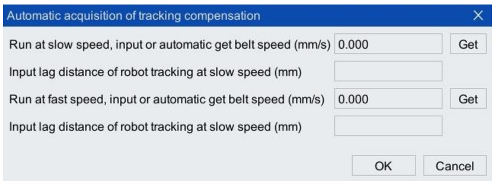


Figure 5-14 Conveyor belt parameter fine-tuning interface

Table 5-3 Parameter description

Parameter	Illustrate
Maximum tracking speed	System default value, maximum tracking speed of motion trajectory
Maximum tracking acc	System default value, maximum acceleration of motion trajectory tracking
Tracking maximum acc	System default value, maximum acceleration of motion trajectory tracking
Minimum trigger dis	The minimum distance between two work objects. If it is less than this distance, the latter work object will not get on.
Whether quiesce is queued	Indicates whether work object is allowed to enter the queue when the conveyor belt is stationary. The value is as follows: ■ true: Indicates that work object is allowed to enter the queue when the conveyor belt is stationary. ■ false: Indicates that work object is not allowed to enter the queue when the conveyor belt is stationary.

Parameter	Illustrate
Tracking delay time (s)	When the robot tracking lags, it needs to compensate for the delay time. The unit is seconds.
Tracking delay dis (mm)	When robot tracking lags, the delay distance needs to be compensated. The unit is m.
< Auto get>	Click the <Auto get> button to pop up the configuration interface of [Automatic acquisition of tracking compensation] as shown in the figure below. You can manually enter or click <Get> to automatically calculate compensation parameters. 

Step18. Click <Save> to save and complete the configuration.

5.2.2 Visual mode

Basic parameter configuration

Step1. Enter the [Conveyor setting] interface again. The basic parameters in the configuration interface are as shown in Figure 5-15, and the parameter description is as shown in Table 5-4. After the configuration is complete, click <Save> and restart the control cabinet for the configuration to take effect.

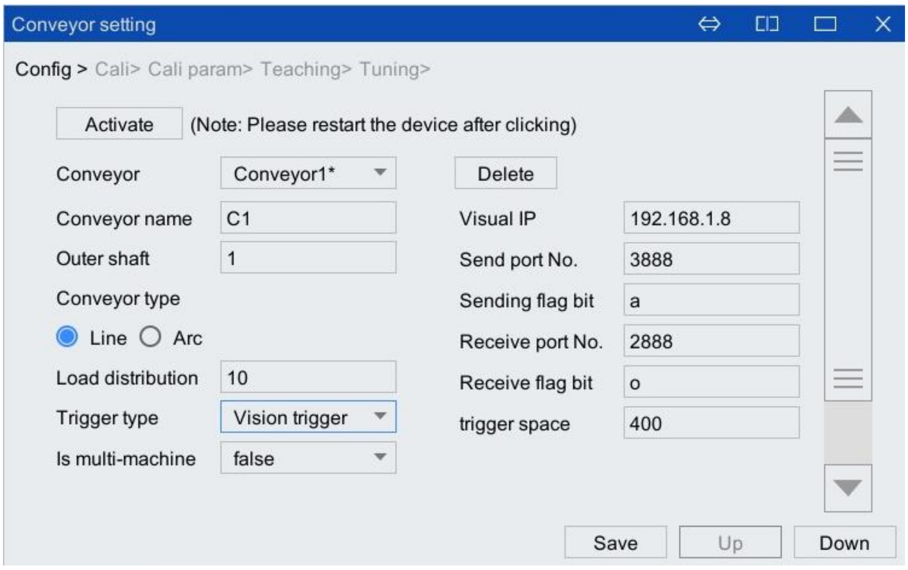


Figure 5-15 Activate the basic parameter configuration interface of the conveyor belt

Table 5-4 Parameter Description

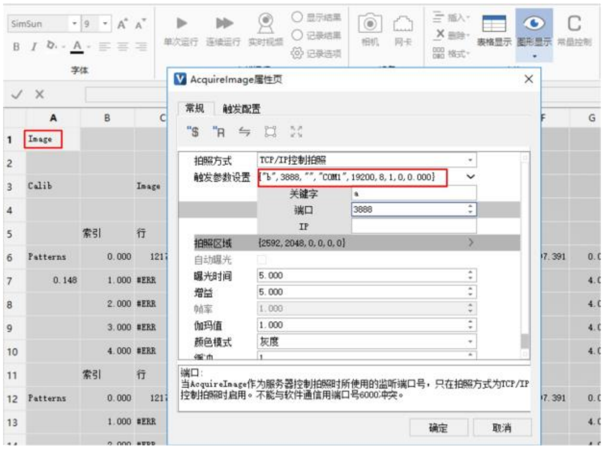
Parameter	Illustrate
Conveyor	Select the conveyor belt that needs to be configured.
Conveyor name	User can set the name of the conveyor belt
Outer shaft	The external axis serial number corresponding to the conveyor belt, which corresponds to the first external axis
Conveyor type	The type of conveyor belt used, select "line" here

Parameter	Illustrate
Trigger type	Conveyor belt work object trigger method, select "visual trigger" here
Is multi-machine	Whether to use multi-machine conveyor belts, select "false" here (multi-machine conveyor belts are not supported)
Visual IP	The IP address of the vision software should be in the same network segment as the robot network port IP.
Send port No.	Send a shooting instruction from the port with the specified serial number
Sending flag bit	Send the content of the shooting instruction. Photography instructions agreed with vision, such as Figure 5 15 shows the robot sending "a" through port 3888, and the camera takes a picture after receiving it.
Receive port No.	Receive the captured information from the port with the specified serial number
Receive flag bit	The received message ends with the specified character
trigger space	Specify the distance to trigger the camera parameters, that is, the robot will send a camera instruction every time the conveyor belt moves this distance, unit mm
<Delete>	Delete the currently configured conveyor belt.
<Save>	After the configuration is complete, click <Save>. The configuration takes effect after the restart.

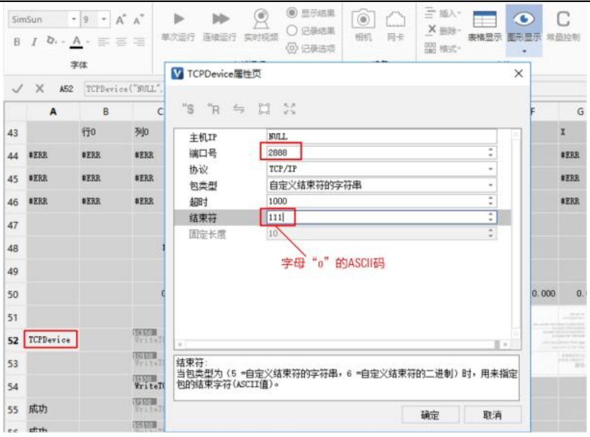
Tip

The sending port number, sending flag bit, receiving port number, and receiving flag bit parameters must be set and saved consistently on both the robot and vision ends, otherwise normal communication will not be possible.

- Vision software sender configuration. Taking Peitian vision software (AEIV_studio) as an example, in the vision software, click [Image] in the project file, and the [AcquireImage property page] as shown below will pop up. Fill in the sender parameters in [Trigger parameter settings].



- Vision software receiver configuration. Take Peitian vision software (AEIV_studio) as an example. In the vision software, click [TCPDevice] in the project file, and the [TCPDevice property page] as shown in the figure below will pop up. Fill in the sender parameters.

Parameter	Illustrate
	

Base coordinate system calibration

Step2. Click <Down> to enter the [Cali] interface, and click the <Calibrate> button in the upper part of [Base coordinate system calibration]. As shown in Figure 5-16.

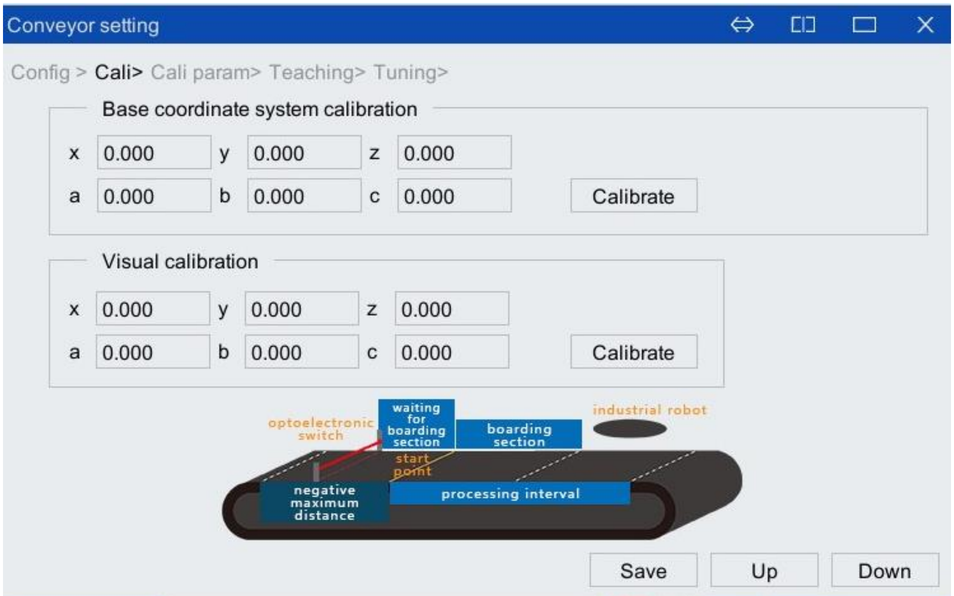


Figure 5-16 Conveyor belt calibration interface

Step3. Enter the [Calibration of base coordinate system of conveyor] interface, as shown in Figure 5-17. The calibration steps as follows:

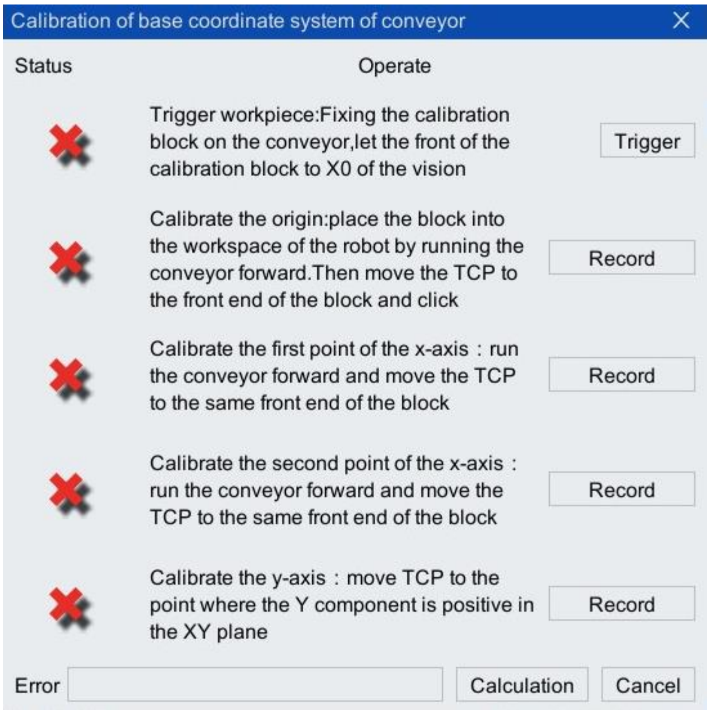


Figure 5-17 Conveyor belt base coordinate system calibration interface

Step4. The interface needs to record five points to complete the calibration. Follow the instructions in Figure 5-17 to complete the calibration of the base coordinate system. The × in front will change to √.

> Tip
>
> It is recommended to select the origin on the edge of the conveyor belt and specify that the positive direction of X is consistent with the rotation direction of the conveyor belt.

Step5. Click the <Calculation> button to calculate the error. If it exceeds the error range, it needs to be re-calibrated. If it is within the allowable error range, the calibration is completed.

Camera coordinate system calibration

The steps for visual coordinate system calibration are as follows:

Step6. Place the chessboard calibration board on the conveyor belt (as shown in Figure 5-18). When placing it, try to make the long side of the chessboard calibration board parallel to the edge of the conveyor belt. The plane of the calibration plate is at the same height as the upper surface of work object.

> Tip
>
> The calibration plate should be firmly fixed on the conveyor belt and should not slide. The conveyor belt moves in the positive direction to directly below the visual field (moving in the positive direction to directly below the visual field is to avoid backlash in the movement of the conveyor belt).

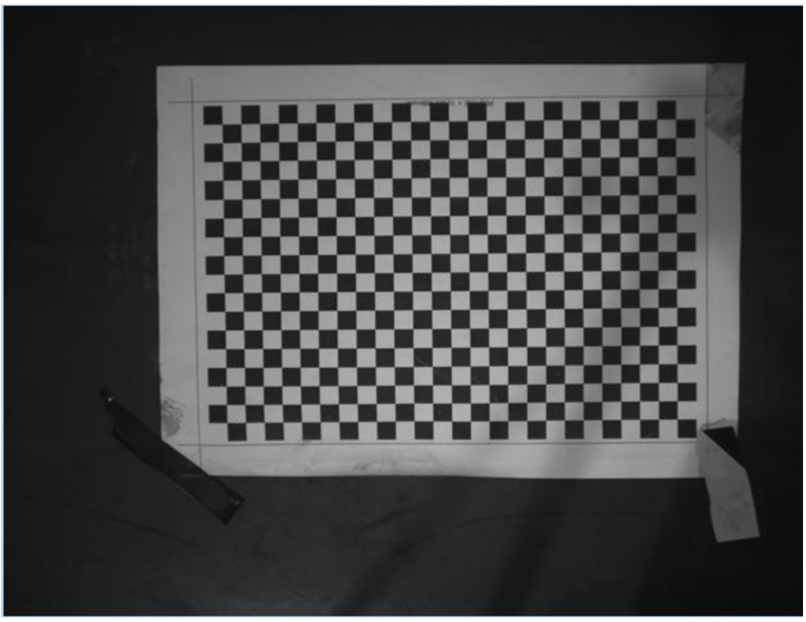


Figure 5-18 Chessboard calibration board

Step7. Open the visual software. Here we take Peitian's AEIV_studio software as an example for explanation. The main interface is shown in Figure 5-19. Click <Camera> to check whether the calibration board is in the center of the field of view on the main interface.

> **Tip**
>
> - When calibrating, you need to enter the accurate chessboard size to ensure that the visually recognized points are consistent with the actual work object size.
> - The positive directions of X and Y axis of the visual coordinate system are consistent with the positive directions of X and Y axis of the conveyor belt base coordinate system. For the definition of the base coordinate system X and Y axis, see step 2 of the calibration of the conveyor belt base coordinate system.

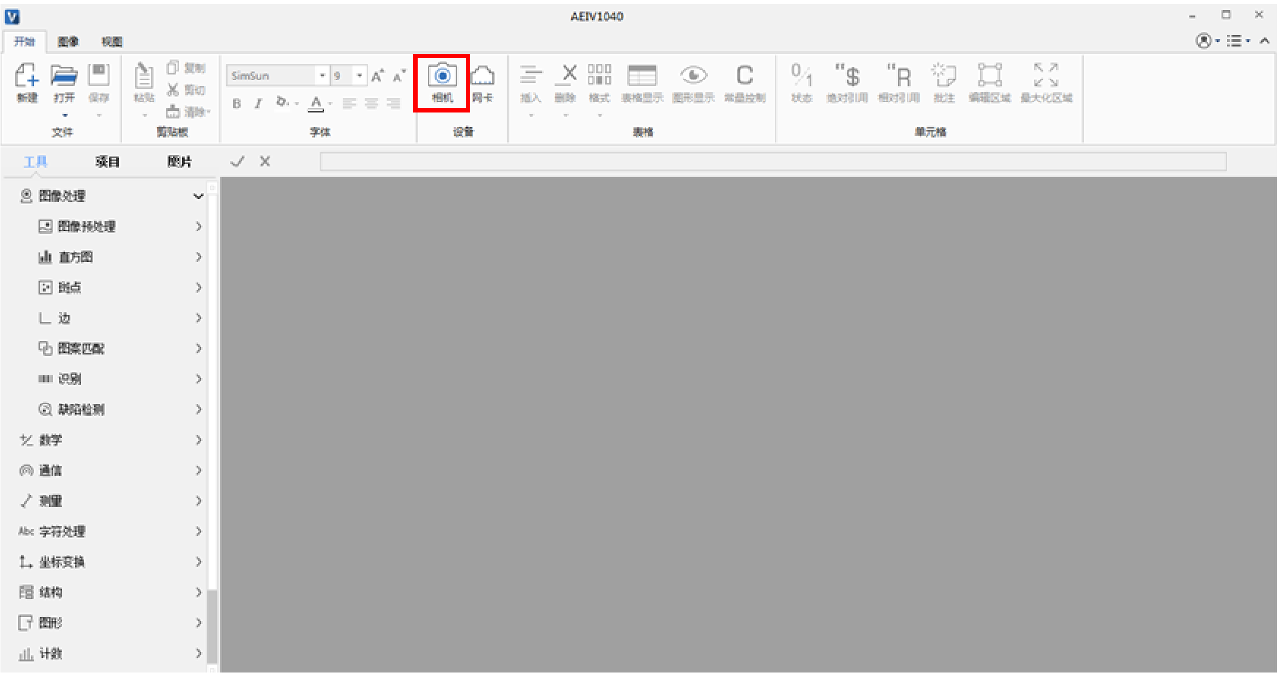


Figure 5-19 AEIV_studio software interface

Step8. Click <New> to create a new project. Or click <Open> to import the previously created project. As shown in Figure 5-20.

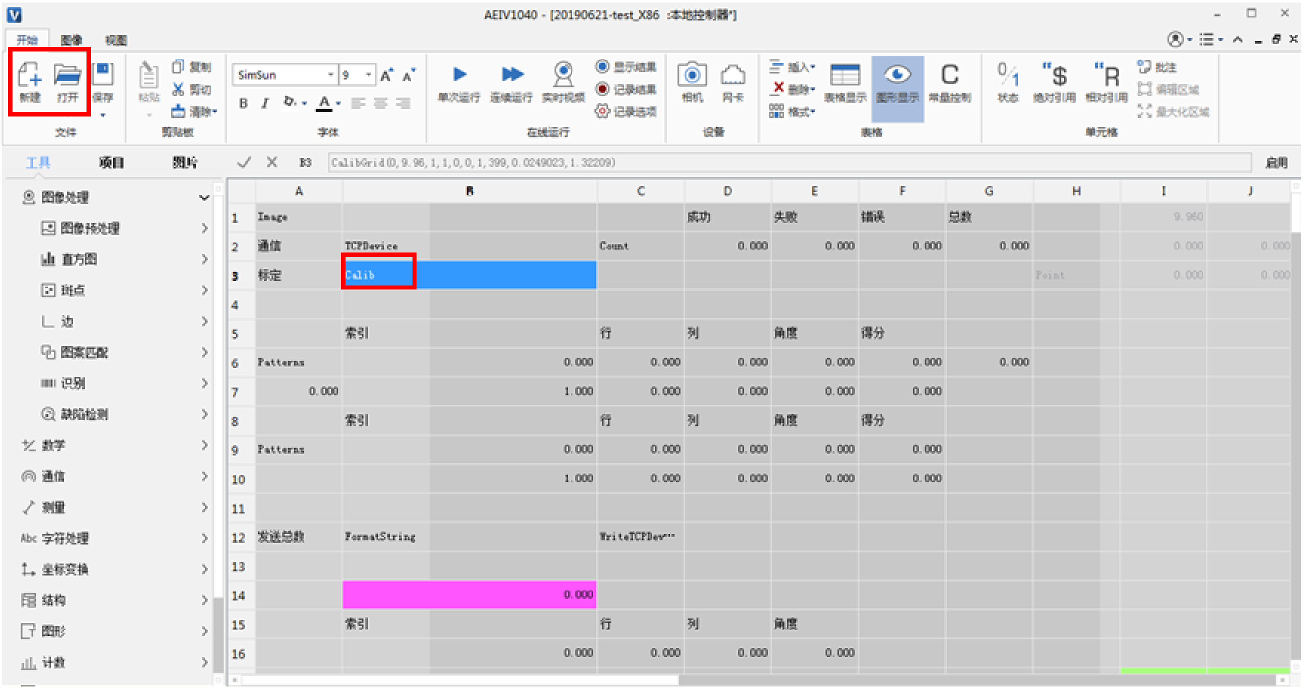


Figure 5-20 AEIV_studio new or import project interface

Step9. Double-click [Calib Cell], and the [Chessboard Calibration] interface will appear, as shown in Figure 5-21.

Figure 5-21 Chessboard calibration interface

Step10. Click <Trigger> in the interface of pose 1, and the three keys <Select origin>, <Select x axis> and <Select Y axis> in the network axis will become highlighted, as shown in Figure 5-22.

Figure 5-22 Chessboard calibration interface

Step11. Click <Select Origin>, the chessboard calibration chart appears as shown in Figure 5-23. Select one of the green points and double-click it as the origin of the visual coordinate system (it turns red after double-clicking).

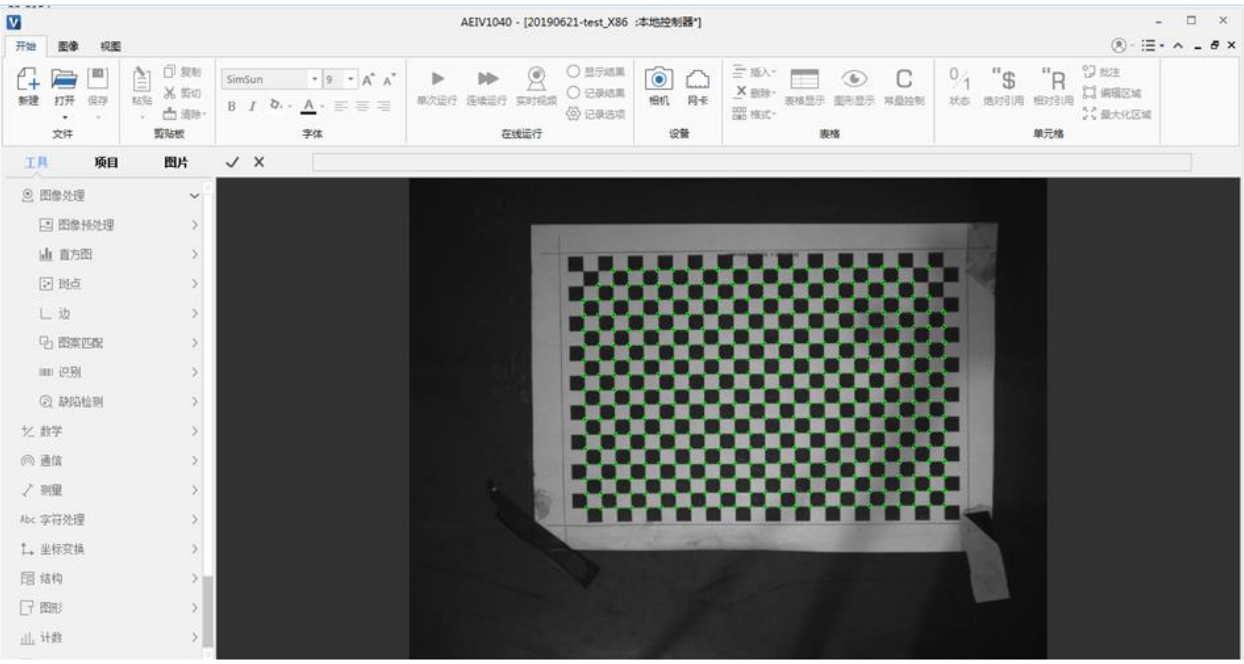


Figure 5-23 Chessboard calibration chart origin calibration

Step12. Continue to click <Select X-axis>, the chessboard calibration chart appears (as shown in Figure 5-24). Select one of the four green points as the positive direction of the visual coordinate system X axis (it is recommended to choose the same positive direction as the conveyor belt drive). Continue to click <Select Y-axis> and select the direction of a green dot as the positive direction of the Y-axis of the visual coordinate system. Then click <Calibrate> and <OK> to complete the calibration.

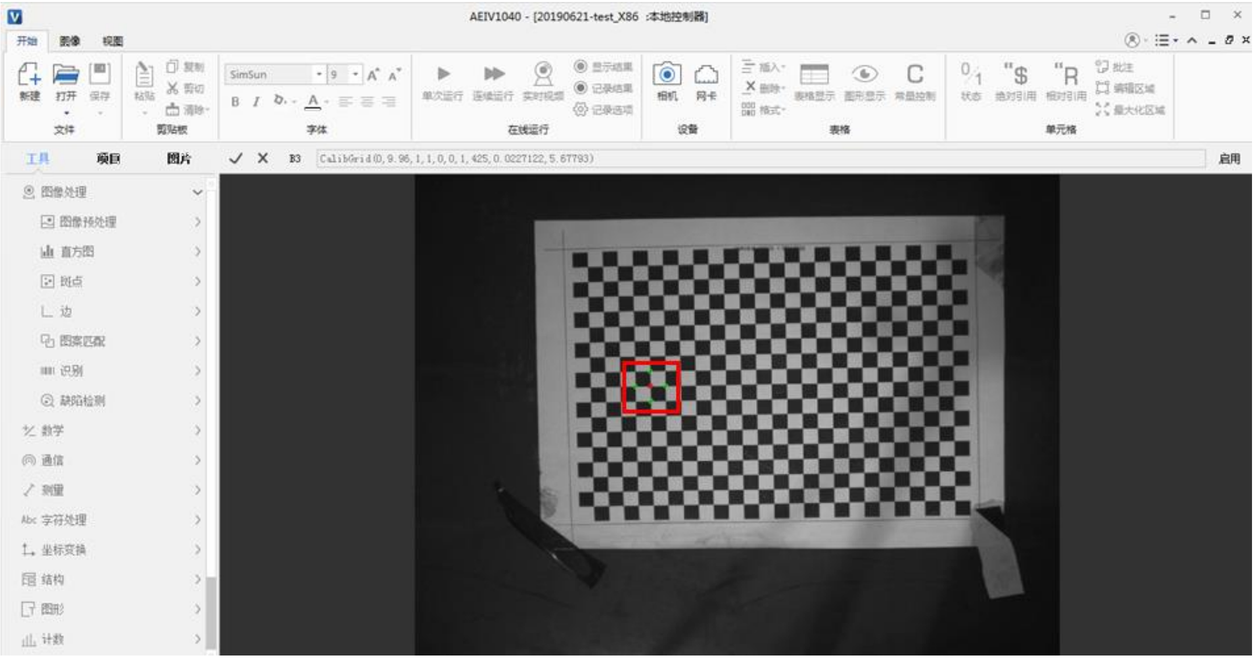


Figure 5-24 Chessboard calibration chart X-axis calibration

Step13. After the calibration is complete, click the <Save> button in the upper left corner of the vision software to save the visual project.

Visual coordinate system calibration

Step14. Enter the [Cali] interface of the [Conveyor setting] interface again, and click the <Calibrate> button in the [Visual calibration] interface. As shown in Figure 5-25.

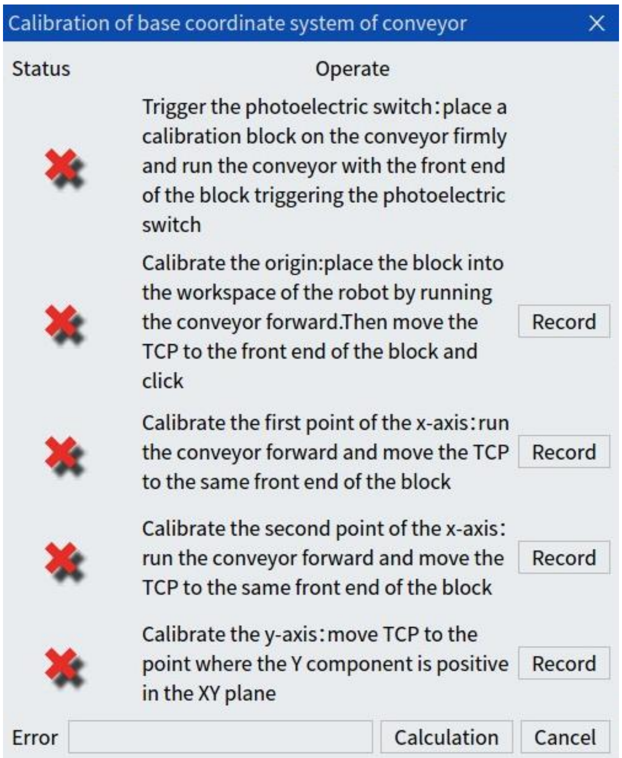


Figure 5-25 Visual coordinate system calibration interface

1 Fix the calibration plate on the conveyor belt, the calibration plate is located in the camera field of view, after completing the calibration of the camera coordinate system, click the first <Record>

2 Move the calibration plate to the robot's range of motion, pause the conveyor belt, and click the second <Record>

3 Holding the conveyor still, move the robot so that the TCP points to the origin of the camera coordinate system and click the third <Record>

4 Keep the conveyor belt still, move the robot so that the TCP points a little in the positive direction of the origin of the camera coordinate system X axis, and click the fourth <Record>

5 Keep the conveyor belt still, move the robot so that the TCP points a point in the positive direction of the origin of the camera coordinate system Y axis, and click the fifth <Record>

6 Click <Calculate> and <Save>

Tip	When calibrating, the distance between the positive point of X axis and the positive point of Y axis should be as far as possible from the origin.

Tip	■ What the robot needs to confirm with the vision terminal is the number of work objects photographed in one row. ■ Each time work object is visually recognized, the sequence of points sent must always be consistent.

Calibration parameter settings

Step15. Click <Down> to enter the interface of [Cali param], where calibration parameters can be adjusted, as shown in Figure 5-26, and parameter description is shown in Table 5-5.

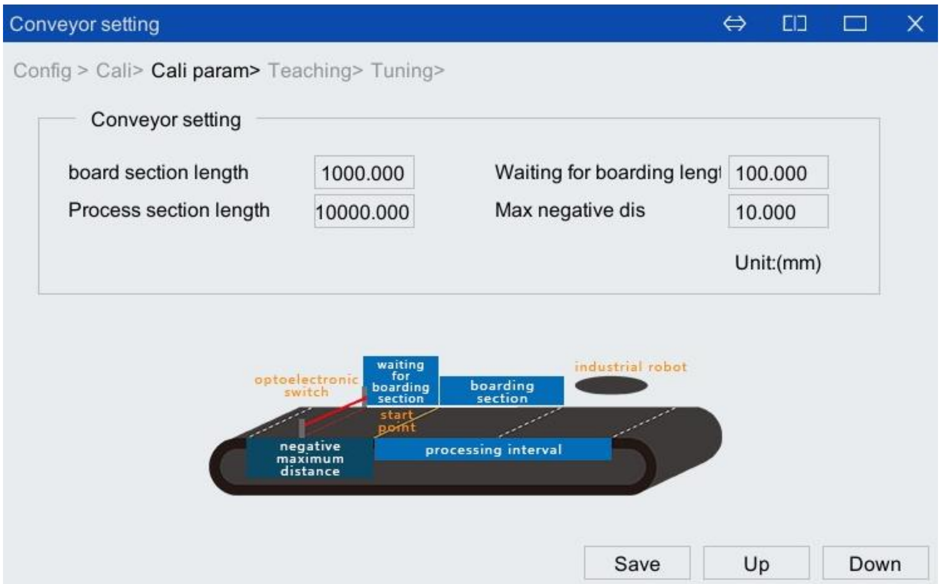


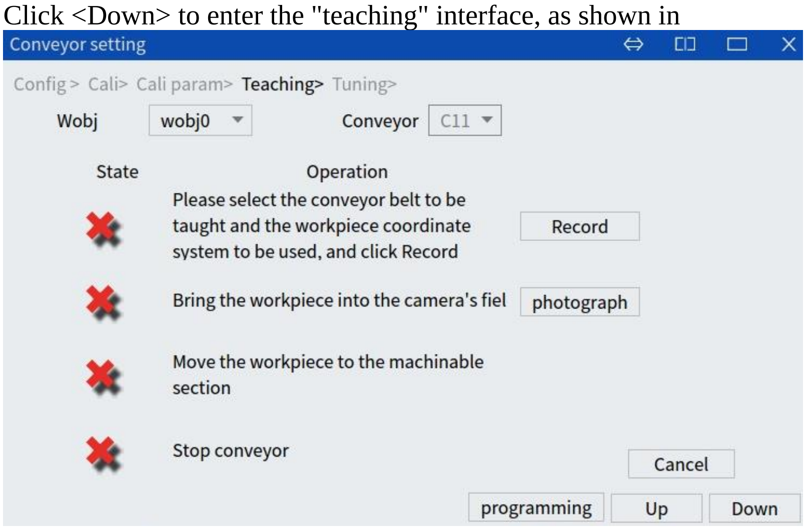
Figure 5-26 Conveyor belt calibration parameter interface

Table 5-5 Parameter description

Parameter	Illustrate
board section length	After work object in the work object queue enters the getting on area with the conveyor belt, they will start waiting to be tracked in turn.
Process section length	The processing area is the distance set along the moving direction of the conveyor belt from the end of the waiting area of the conveyor belt. If work object is not actively released (dropwobj) after being associated, it will be passively released after reaching this position. See Figure 1 2 for detailed description.
Waiting for boarding length	The length of the waiting area during visual triggering is calibrated by the joint calibration of the vision and robot, and is the distance between the origin of the camera coordinate system and the origin of the visual coordinate system.
Max negative dist	The maximum distance in the negative direction is the distance set from the zero point of the conveyor base coordinate system in the opposite direction of the conveyor movement. If work object is not actively released after being associated, it will be passively released after reaching this position. See Figure 1 2 for detailed description.

Conveyor belt teaching

Click <Down> to enter the "teaching" interface, as shown in



Step16. Figure 5-27. The teaching steps are as follows:

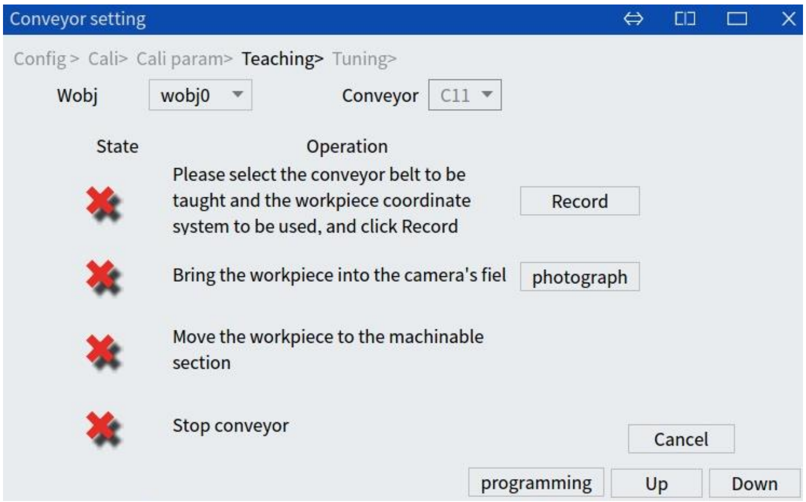


Figure 5-27 Conveyor belt teaching interface

1 In the upper left corner of the interface, select the work object coordinate system you want to use and click <Record>.

2 Move the conveyor belt so that work object enters the field of view of the camera and click <Photograph>.

3 Continue to move the conveyor belt and move work object to the processable area

4 stop conveyor belt

5 Move the TCP to the processing point on work object that needs to be processed, and click <Get Location> on the relevant point in ARL

Parameter fine-tuning

Step17. Click <Down> to enter the "Tuning" interface. Parameters can be adjusted according to the specific tracking situation, as shown in Figure 5-28. The number description is shown in Table 5-6.

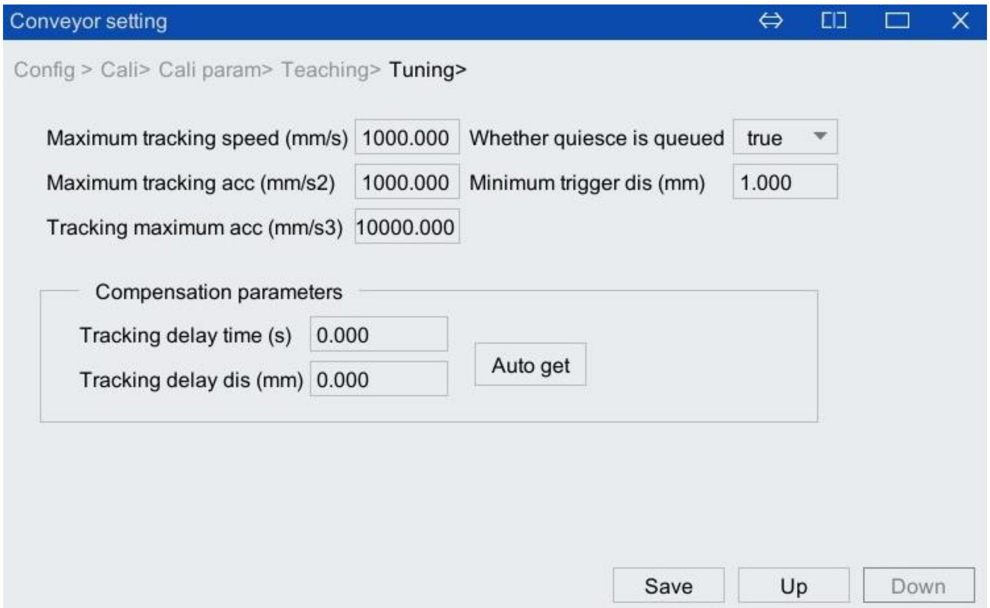
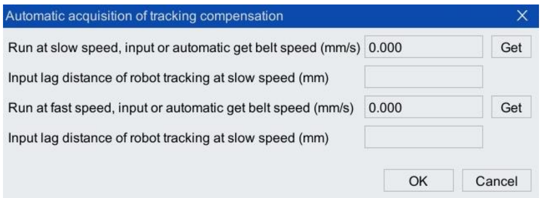


Figure 5-28 Conveyor belt parameter fine-tuning interface

Table 5-6 Parameter Description

Parameter	Illustrate
Maximum tracking speed	System default value, maximum tracking speed of motion trajectory
Maximum tracking acc	System default value, maximum acceleration of motion trajectory tracking
Tracking maximum acc	System default value, maximum jerk of motion trajectory tracking
Minimum trigger dis	The minimum distance between two work objects. If it is less than this distance, the latter work object will not get on.
Whether quiesce is queued	Indicates whether work object is allowed to enter the queue when the conveyor belt is stationary. The value is as follows: ■ true: Indicates that work object is allowed to enter the queue when the conveyor belt is stationary. ■ false: Indicates that work object is not allowed to enter the queue when the

Parameter	Illustrate
	conveyor belt is stationary.
Tracking delay time (s)	When the robot tracking lags, it needs to compensate for the delay time. The unit is seconds.
Tracking delay dis (mm)	When robot tracking lags, the delay distance needs to be compensated. The unit is m.
<Auto get>	Click the button to pop up the configuration interface of "Automatic acquisition of tracking compensation" as shown in the figure below. You can manually enter or click <Get> to automatically calculate compensation parameters. 

Step18. Click <Save> to save and complete the configuration.

5.2.3 Photoelectric + visual mode

Basic parameter configuration

Step1. Enter the [Conveyor Belt Configuration] interface again. The basic parameters in the configuration interface are as shown in Figure 5-29, and the parameter description is as shown in Table 5-7. After the configuration is complete, click <Save> and restart the control cabinet for the configuration to take effect.

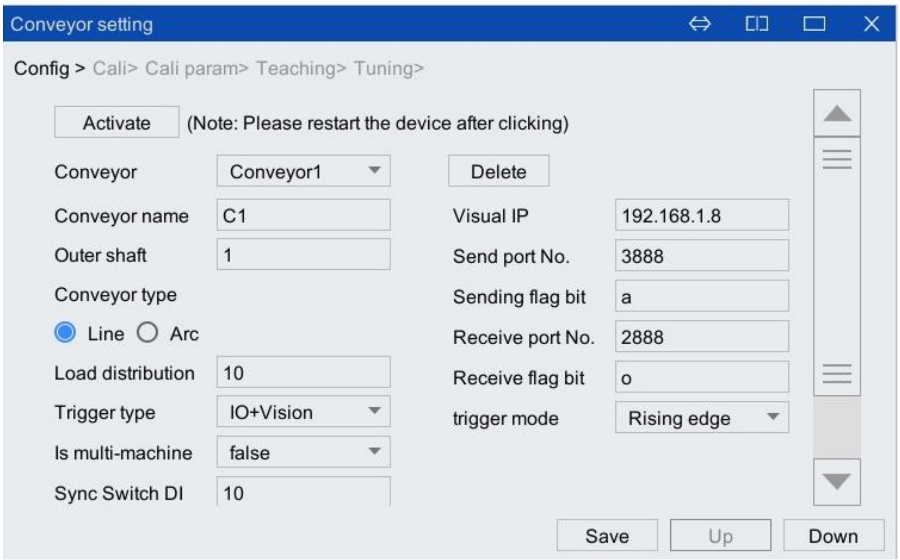
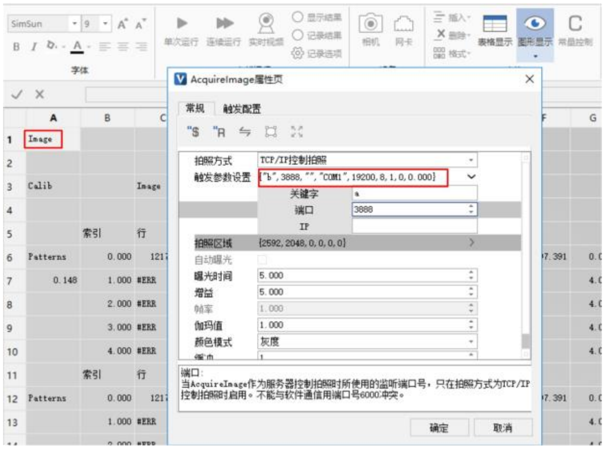
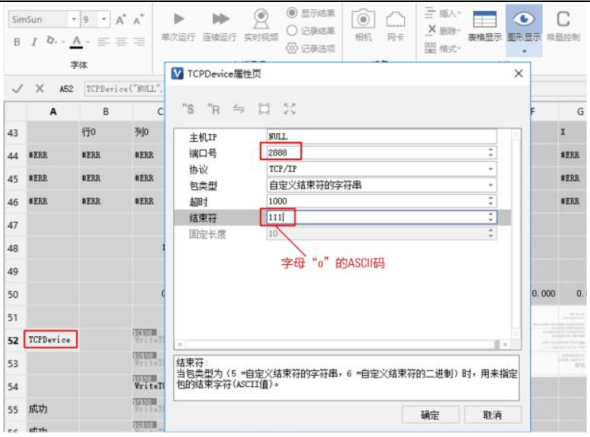


Figure 5-29 Activate the basic parameter configuration interface of the conveyor belt

Table 5-7 Parameter Description

Parameter	Illustrate
Conveyor	Select the conveyor belt that needs to be configured.
Conveyor name	User can set the name of the conveyor belt
Outer shaft	The external axis serial number corresponding to the conveyor belt, which corresponds to the first external axis

Parameter	Illustrate
Conveyor type	The type of conveyor belt used, select "line" here
Trigger type	Conveyor belt work object trigger mode, select "IO+Visual" here
Is multi-machine	Whether to use multi-machine conveyor belt, select "false" here (multi-machine is not supported)
Syn switch DI	Photoelectric switch DI. This value is the port number for communication between the actual photoelectric switch and the robot DI.
Visual IP	The IP address of the vision software should be in the same network segment as the robot network port IP.
Send port No.	Send a shooting instruction from the port with the specified serial number
Sending flag bit	Send the content of the shooting instruction. Photography instructions agreed with vision, such as Figure 5 15 shows the robot sending "a" through port 3888, and the camera takes a picture after receiving it.
Receive port No.	Receive the captured information from the port with the specified serial number
Receive flag bit	The received message ends with the specified character
Trigger mode	Conveyor belt DI trigger mode, the values are as follows: ■ Rising edge trigger (the following uses rising edge trigger as an example to illustrate the configuration) ■ Falling edge trigger
<Delete>	Delete the currently configured conveyor belt.
<Save>	After the configuration is complete, click <Save>. The configuration takes effect after the restart Tip: The sending port number, sending flag bit, receiving port number, and receiving flag bit parameters must be set and saved consistently on both the robot and vision ends, otherwise normal communication will not be possible. ■ Vision software sender configuration. Taking Peitian vision software (AEIV_studio) as an example, in the vision software, click [Image] in the project file, and the [AcquireImage property page] as shown below will pop up. Fill in the sender parameters in [Trigger parameter settings].  ■ Vision software receiver configuration. Take Peitian vision software (AEIV_studio) as an example. In the vision software, click [TCPDevice] in the project file, and the [TCPDevice property page] as shown in the figure below will pop up. Fill in the sender parameters.

Parameter	Illustrate
	

Base coordinate system calibration

Step2. Click <Down> to enter the [Cali] interface, and click the <Calibrate> button in the upper part of [Base coordinate system calibration]. As shown in Figure 5-30.

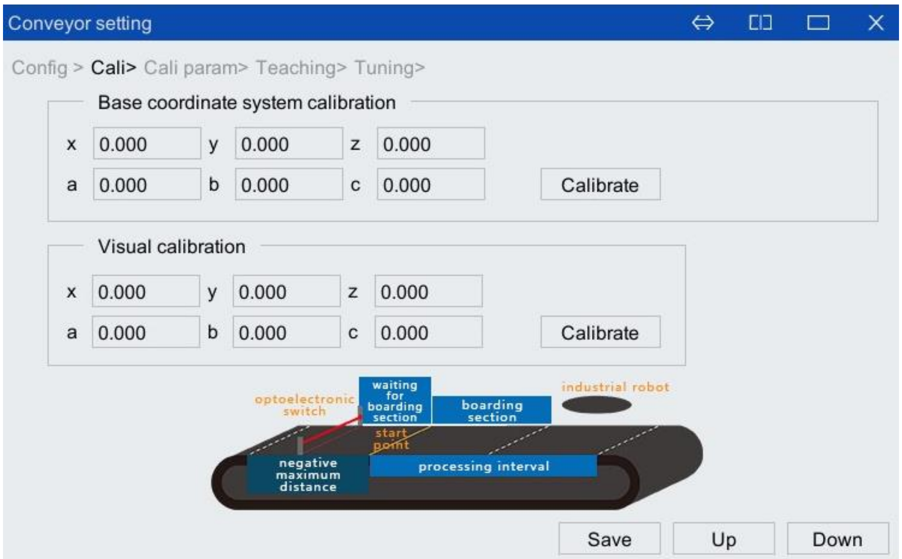


Figure 5-30 Conveyor belt calibration interface

Step3. Enter the [Calibration of base coordinate system of conveyor] interface, as shown in Figure 5-31. The calibration steps are as follows:

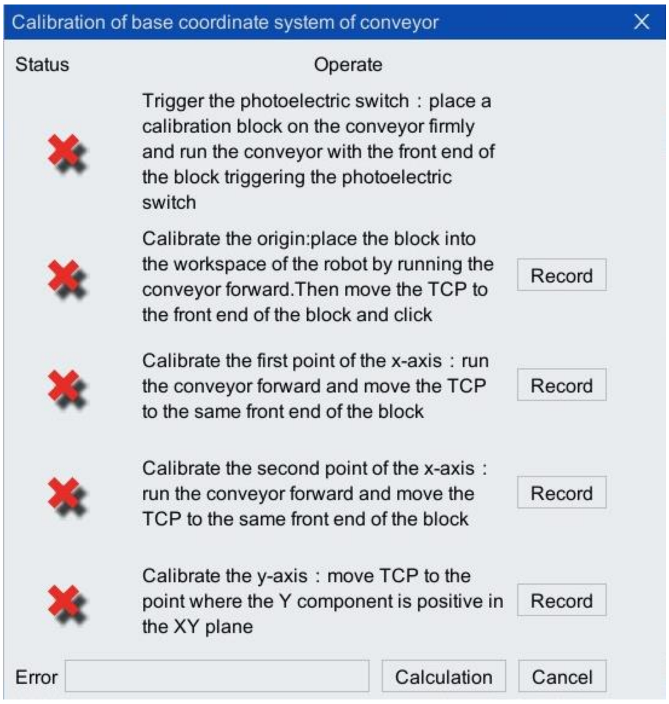


Figure 5-31 Conveyor belt base coordinate system calibration interface

Step4. The interface needs to record five points to complete the calibration. Follow the operating instructions in Figure 5-31 to complete the calibration of the base coordinate system. For the scene of the work object movement during the calibration process, please refer to Figure 5-32. After successful recording, the × in front will change to √.

> Tip
>
> It is recommended to select the origin on the edge of the conveyor belt and specify that the positive direction of X is consistent with the rotation direction of the conveyor belt.

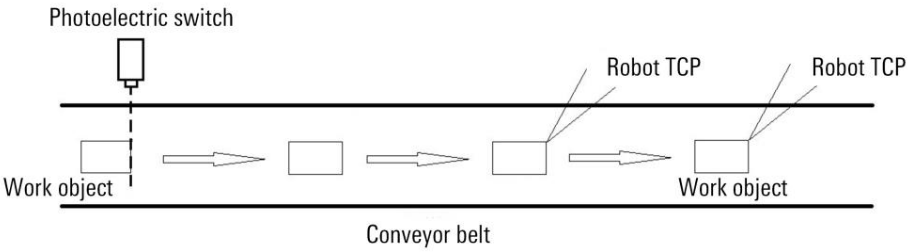


Figure 5-32 Diagram of work object movement on the conveyor belt

Step5. Click the <Calculation> button to calculate the error. If it exceeds the error range, it needs to be re-calibrated. If it is within the allowable error range, the calibration is completed.

Camera coordinate system calibration

The steps for visual coordinate system calibration are as follows:

Step6. Place the chessboard calibration board on the conveyor belt (as shown in Figure 5-33). When placing it, try to make the long side of the chessboard calibration board parallel to the edge of the conveyor belt. The plane of the calibration plate is at the same height as the upper surface of work object.

> Tip
>
> The calibration plate should be firmly fixed on the conveyor belt and should not slide. The conveyor belt moves in the positive direction to directly below the visual field (moving in the positive direction to directly below the visual field is to avoid backlash in the movement of the conveyor belt).

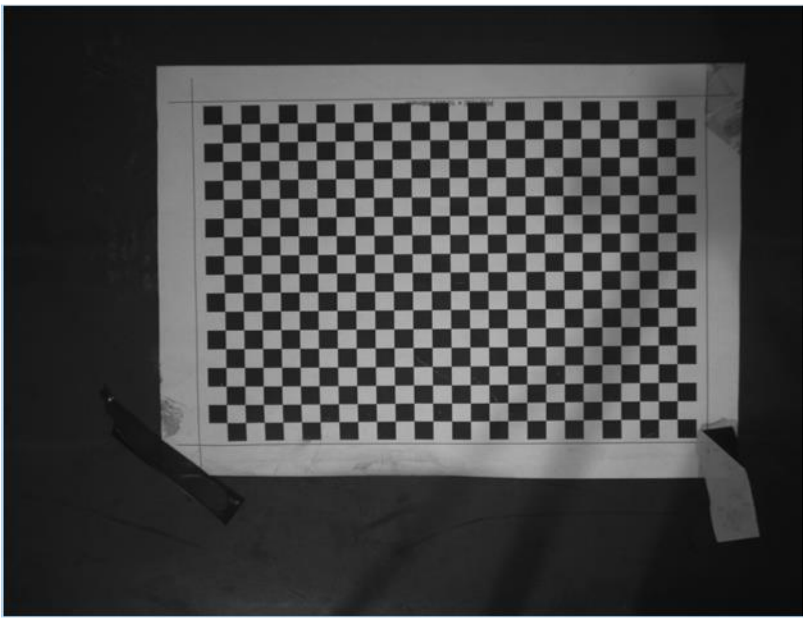


Figure 5-33 Chessboard calibration board

Step7. Open the visual software. Here we take Peitian's AEIV_studio software as an example for explanation. The main interface is as shown in Figure 5-34. Click the <Camera> button to check whether the calibration plate is in the center of the field of view on the main interface.

> Tip
>
> ■ When calibrating, you need to enter the accurate chessboard size to ensure that the visually recognized points are consistent with the actual work object size.
>
> ■ The positive directions of X and Y axis of the visual coordinate system are consistent with the positive directions of X and Y axis of the conveyor belt base coordinate system. For the definition of base coordinate system X and Y axis, see step 2 of base coordinate system calibration.

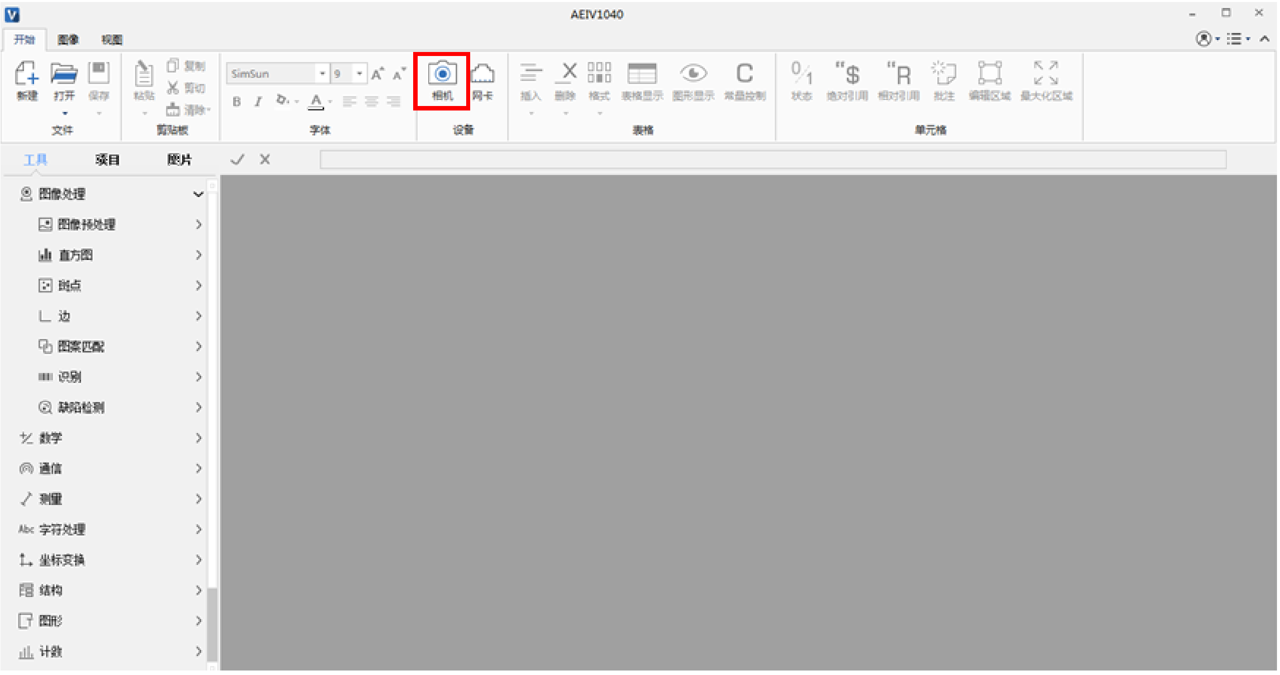


Figure 5-34 AEIV_studio software interface

Step8. Click <New> to create a new project. Or click <Open> to import the previously created project. As shown in Figure 5-35.

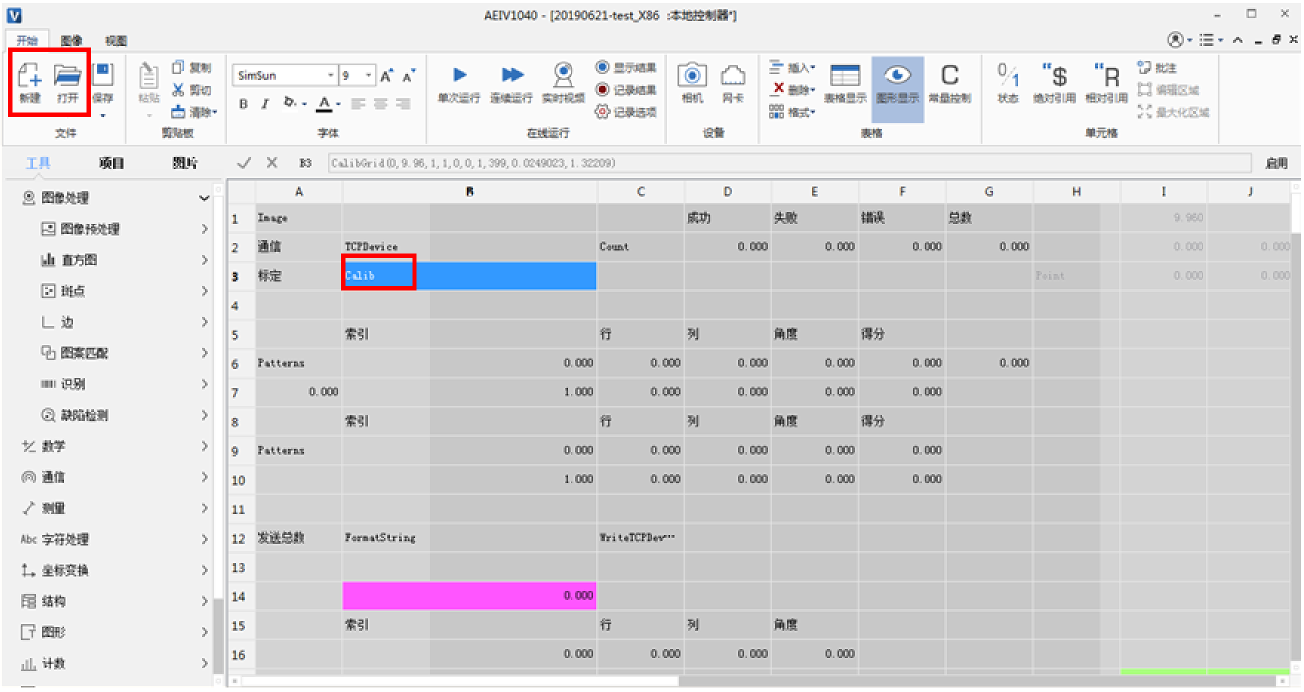


Figure 5-35 AEIV_studio new or import project interface

Step9. Double-click [Calib Cell], and the [Chessboard Calibration] interface will appear, as shown in Figure 5-36.

Figure 5-36 Chessboard calibration interface

Step10. Click <Trigger> in the interface of pose 1, and the three keys <Select origin>, <Select X axis> and <Select Y axis> in the network axis will become highlighted, as shown in Figure 5-37.

Figure 5-37 Chessboard calibration interface

Step11. Click <Select Origin>, the chessboard calibration chart appears as shown in Figure 5-38. Select one of the green points and double-click it as the origin of the visual coordinate system (it turns red after double-clicking).

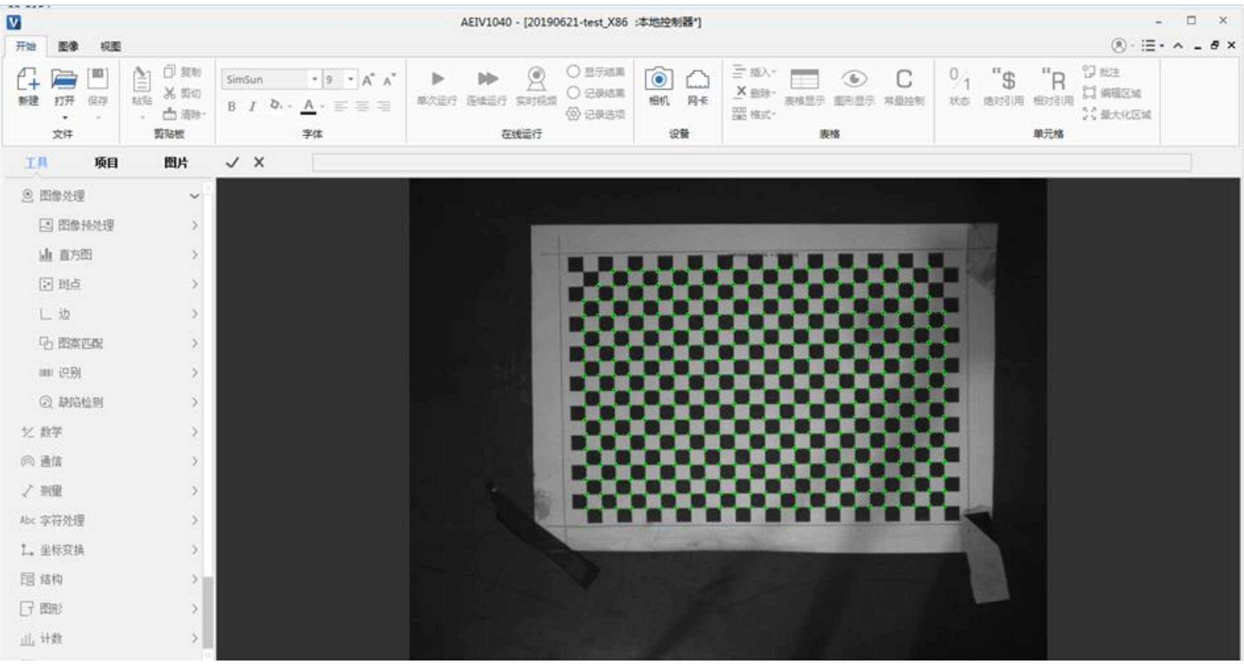


Figure 5-38 Chessboard calibration diagram origin calibration

Step12. Continue to click <Select X-axis>, the chessboard calibration chart appears (as shown in Figure 5-39). Select one of the four green points as the positive direction of the visual coordinate system X axis (it is recommended to choose the same positive direction as the conveyor belt drive). Continue to click <Select Y-axis> and select the direction of a green dot as the positive direction of the Y-axis of the visual coordinate system. Then click <Calibrate> and <OK> to complete the calibration.

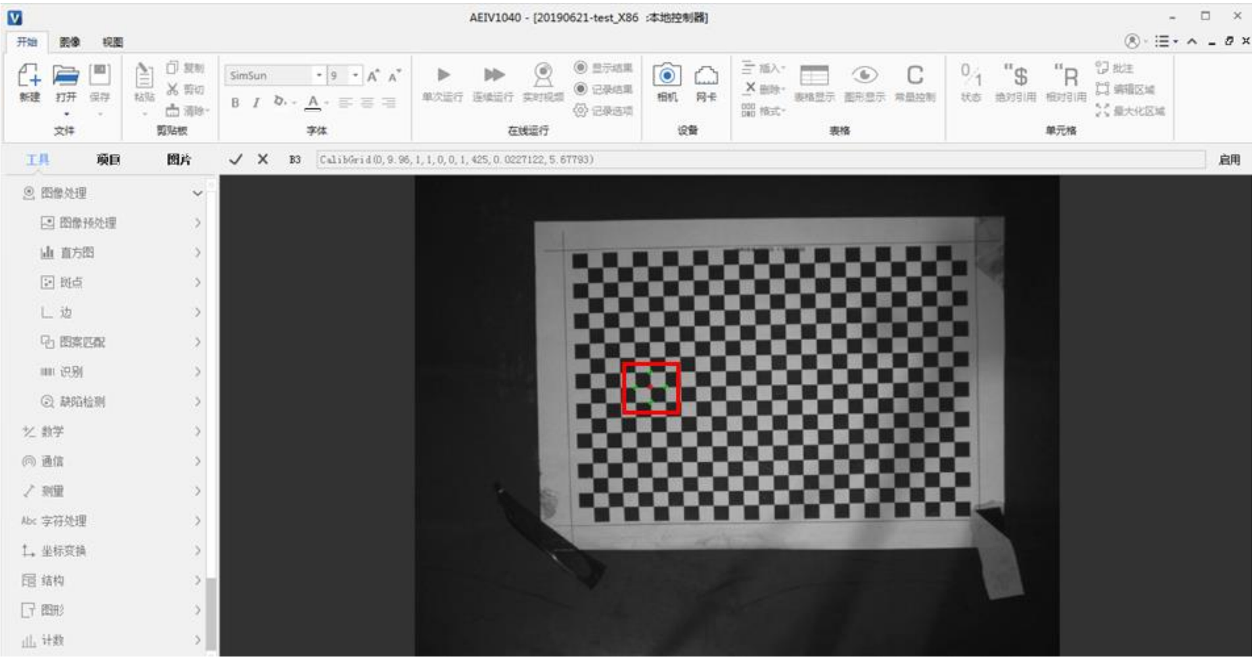


Figure 5-39 Chessboard calibration chart X-axis calibration

Step13. After the calibration is complete, click the <Save> button in the upper left corner of the vision software to save the visual project.

Visual coordinate system calibration

Step14. Enter the [Cali] interface of the [Conveyor setting] interface again, and click the <Calibrate> button in the [Visual calibration] interface. As shown in Figure 5-40.

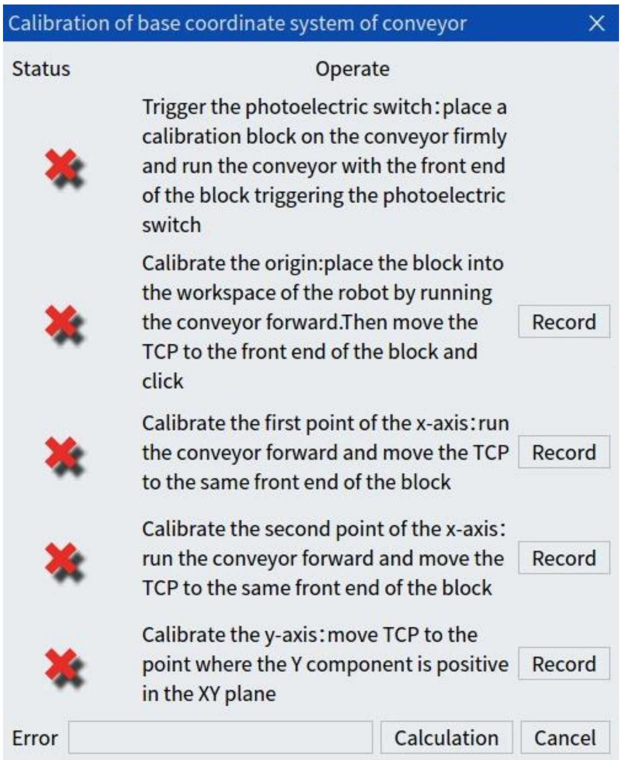


Figure 5-40 Visual coordinate system calibration interface

1 Fix the calibration plate on the conveyor belt so that the calibration plate is within the camera's field of view. After completing the calibration of the camera coordinate system, click the first <Record>.

2 Move the calibration plate to the robot's range of motion, pause the conveyor belt, and click the second <Record>

3 Holding the conveyor still, move the robot so that the TCP points to the origin of the camera coordinate system and click the third <Record>

4 Keep the conveyor belt still, move the robot so that the TCP points a point in the positive direction of the origin of the camera coordinate system X axis, and click the fuorth <Record>

5 Keep the conveyor belt still, move the robot so that the TCP points a point in the positive direction of the origin of the camera coordinate system Y axis, and click the fifth <Record>

6 Click <Calculate> and <Save>

> Tip
>
> When calibrating, the distance between the positive point of X axis and the positive point of Y axis should be as far as possible from the origin.

Tip

- What the robot needs to confirm with the vision terminal is the number of work objects photographed in one row.
- Each time work object is visually recognized, the sequence of points sent must always be consistent.

Calibration parameter settings

Step15. Click <Down> to enter the interface of [Cali param], where calibration parameters can be adjusted, as shown in Figure 5-41, and parameter description is shown in Table 5-8.

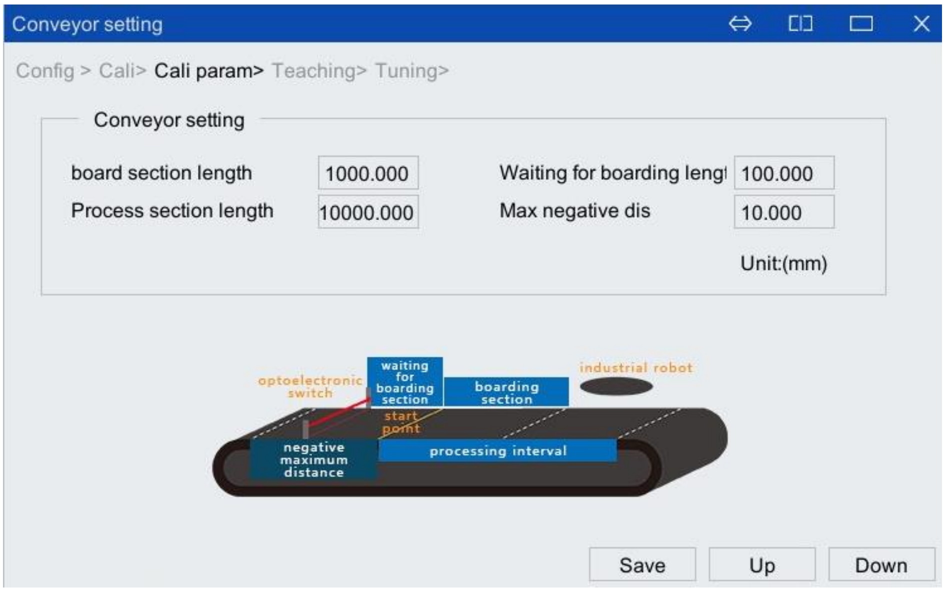


Figure 5-41 Conveyor belt calibration parameter interface

Table 5-8 Parameter description

Parameter	Illustrate
board section length	After work object in the work object queue enters the getting on area with the conveyor belt, they will start waiting to be tracked in turn.
Process section length	The processing area is the distance set along the conveyor moving direction from the end of the conveyor waiting area. If work object is not actively released (dropwobj) after being associated, it will be passively released after reaching this position. See Figure 1-2 for detailed description.
Waiting for boarding length	The length of the waiting area during visual triggering is calibrated by the joint calibration of the vision and robot, and is the distance between the origin of the camera coordinate system and the origin of the visual coordinate system.
Max negative dist	The maximum distance in the negative direction is the distance set from the zero point of the conveyor base coordinate system in the opposite direction of the conveyor movement. If work object is not actively released (dropwobj) after being associated, it will be passively released after reaching this position. See Figure 1-2 for detailed description.

Conveyor belt teaching

Step16. Click <Down> to enter the "Teaching" interface, as shown in Figure 5-42. The teaching steps are as follows:

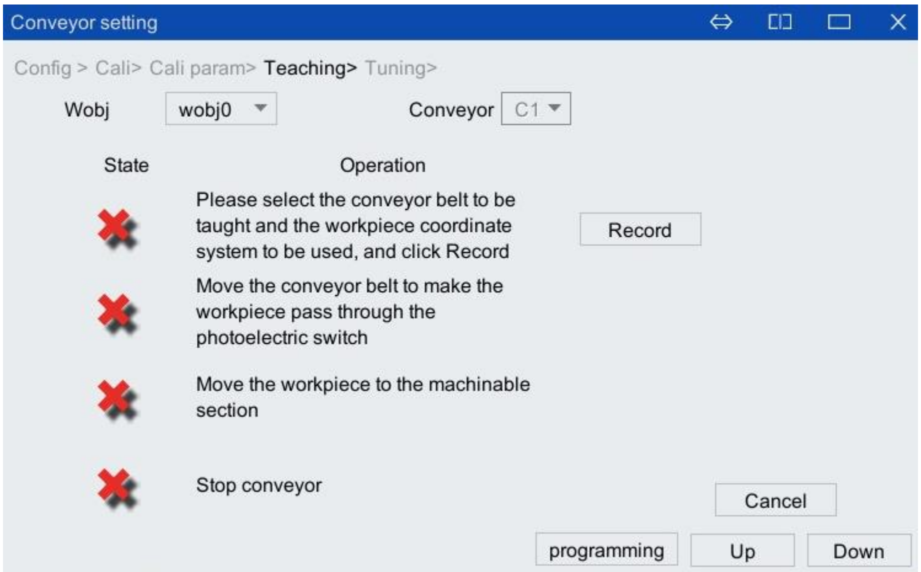


Figure 5-42 Conveyor belt teaching interface

1 In the upper left corner of the interface, select the work object coordinate system you want to use and click <Record>.

2 Move the conveyor belt so that work object passes the photoelectric switch

3 Continue to move the conveyor belt until the × in the third line of the interface changes to √

4 Stop conveyor belt

5 Move the TCP to the processing point on work object that needs to be processed, and click <Get Location> on the relevant point in ARL

Parameter fine-tuning

Step17. Click <Down> to enter the "Tuning" interface. Parameters can be adjusted according to the specific tracking situation, as shown in Figure 5-43. The parameter description is shown in Table 5-9.

Step18. Click <Save> to save and complete the configuration.

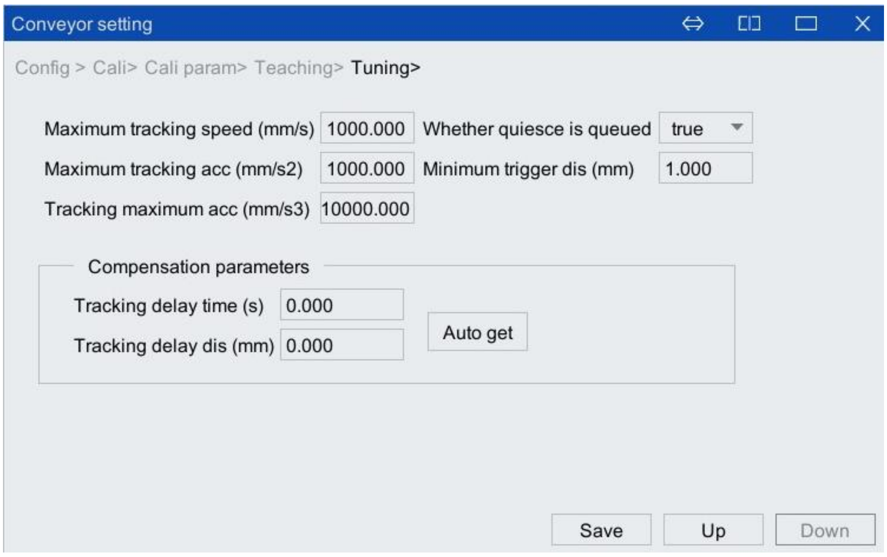
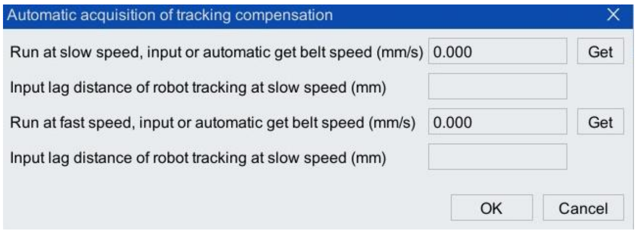


Figure 5-43 Conveyor belt parameter fine-tuning interface

Table 5-9 Parameter description

Parameter	Illustrate
Maximum tracking speed	System default value, maximum tracking speed of motion trajectory
Maximum tracking acc	System default value, maximum acceleration of motion trajectory tracking
Tracking maximum acc	System default value, maximum jerk of motion trajectory tracking
Minimum trigger dis	The minimum distance between two work objects. If it is less than this distance, the latter work object will not get on.
Whether quiesce is queued	Indicates whether work object is allowed to enter the queue when the conveyor belt is stationary. The value is as follows: ■ true: Indicates that work object is allowed to enter the queue when the conveyor belt is stationary. ■ false: Indicates that work object is not allowed to enter the queue when the conveyor belt is stationary.
Tracking delay time (s)	When the robot tracking lags, it needs to compensate for the delay time. The unit is seconds.
Tracking delay dis (mm)	When robot tracking lags, the delay distance needs to be compensated. The unit is m.
<Auto get>	Click the button to pop up the configuration interface of "Automatically obtain tracking Supplement" as shown in the figure below. You can manually enter or click < Obtain > to automatically calculate compensation parameters. 

Step18. Click <Save> to save and complete the configuration.

6 Check and modify system parameter configuration

Activating the conveyor function will automatically configure the system parameters, and no additional operations are required by the user during normal use. If you encounter problems during debugging or need to make custom modifications, you can check or change the relevant parameters according to this section.

The following uses "one body R1 (AIR6L_V2) + one linear conveyor belt C1 (corresponding to the first external axis)" as an example to illustrate the method of checking and modifying the configuration.

Server slave SERVO_NUM

The steps to view and modify parameters are as follows:

Step1. In the [System/Parameter Configuration/Global] interface, you can see that the [SERVO_NUM (Servo Number)] parameter value is "7". As shown in Figure 6-1.

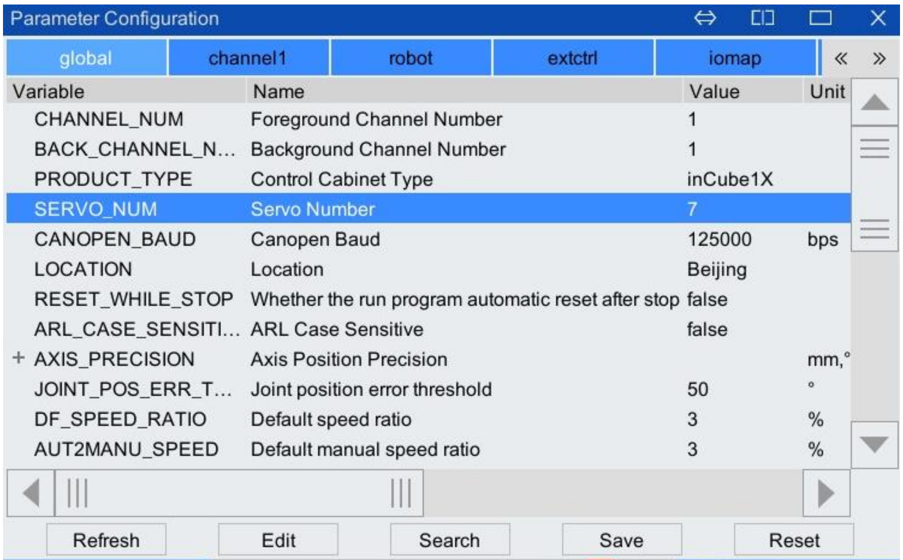


Figure 6-1 [System/Parameter Configuration/Global] configuration interface

Step2. To modify the configuration, click on the line where [SERVO_NUM (Servo Number)] is located, click <Edit>, and the dialog box as shown in Figure 6-2 will pop up. You can modify the parameters in [Value], and click <Yes> after the modification is completed. See Table 6-1 for parameter description.

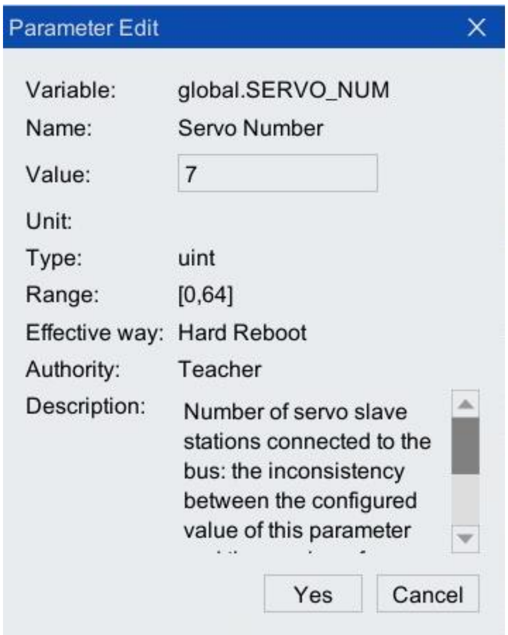


Figure 6-2 Server slave number configuration interface

Table 6-1 Parameter description

Parameter	Illustrate
Value	The number of coupled servo slaves on the bus, the 6 servo slaves of the include manipulator and the external servo slaves. You can configure 1 to 6 external servo slave stations, and the value is 6+n (n is an integer, and the value range is 1 to 6). Tip The parameter configuration does not match the actual number of slave stations, which may cause bus coupled to fail when starting the system.

External axis quantity EX_JOINT_NUM

The steps to view and modify parameters are as follows:

Step1. Select [System/Parameter Configuration/Channel] in the upper right corner of the teach pendant main interface, and you can see that the value of the [EX_JOINT_NUM] parameter is "1". As shown in Figure 6-3.

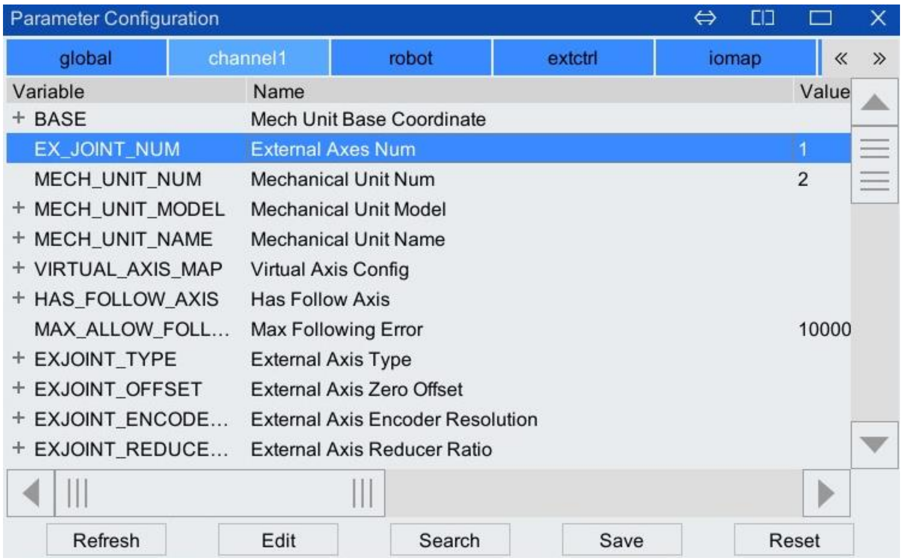


Figure 6-3 External axis number (EX_JOINT_NUM) parameter configuration interface

Step2. If you need to modify the configuration, click the line where the parameter [EX_JOINT_NUM] is located, click the <Edit> button, and the dialog box as shown in Figure 6-4 will pop up. The parameters in [Value] can be modified. The parameter description is shown in Table 6-2.

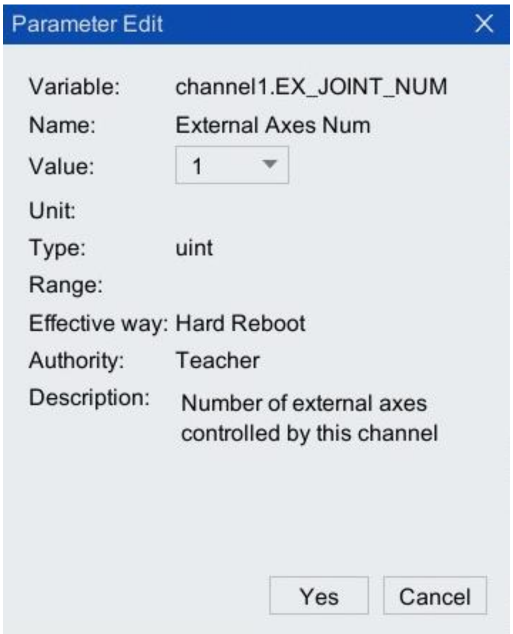


Figure 6-4 External axis quantity dialog box

Table 6-2 Parameter description

Parameter	Illustrate
Value	The value of external axis quantity (EX_JOINT_NUM) can define the number of external axes controlled by this channel. Integer form, value range is 0~6.

Step3. After viewing or modifying parameters, click <Yes>.

Number of mechanical units MECH_UNIT_NUM

The steps to view and modify parameters are as follows:

Step1. Select [System/Parameter Configuration/Channel] in the upper right corner of the teach pendant main interface, and you can see that the [MECH_UNIT_NUM] parameter value is "2". As shown in Figure 6-5.

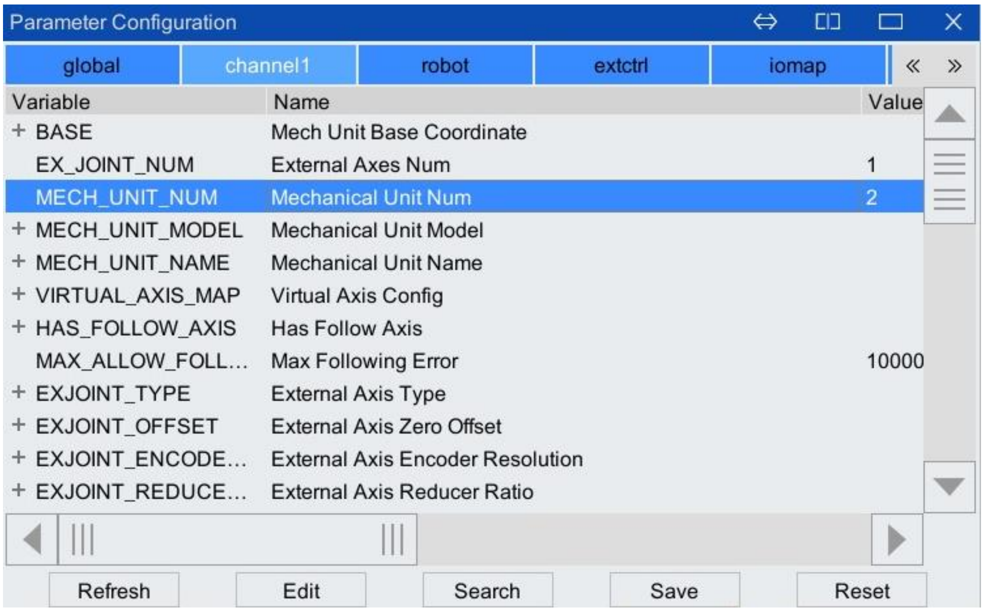


Figure 6-5 Mechanical unit number (MECH_UNIT_NUM) parameter configuration interface

Step2. If you need to modify the configuration, click the line where the parameter [MECH_UNIT_NUM] is located, click <Edit> button, and the dialog box as shown in Figure 6-6 will pop up. The parameters in [Value] can be modified. The parameter description is shown in Table 6-3.

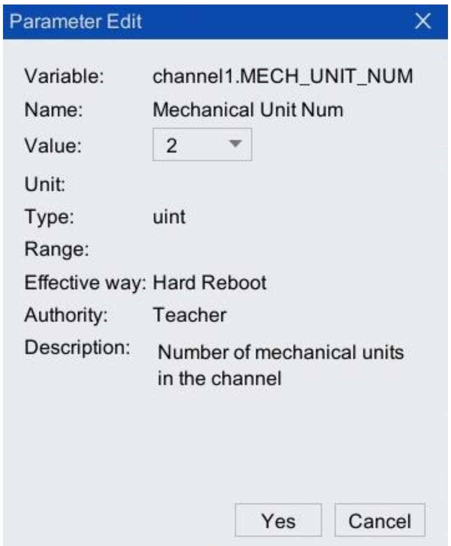


Figure 6-6 Mechanical unit quantity dialog box

Table 6-3 Parameter description

Parameter	Illustrate
value	The value of the mechanical unit number (MECH_UNIT_NUM) parameter can define the number of mechanical units in the channel. Integer form, the value range is 1~3. One conveyor belt occupies one mechanical unit.

Step3. After viewing or modifying the parameters, click <Yes>.

Mechanical unit model MECH_UNIT_MODEL

The steps to view and modify parameters are as follows:

Step1. Select [System/Parameter Configuration/Channel] in the upper right corner of the teach pendant main interface, click [+] on the left side of the [MECH_UNIT_MODEL] parameter, and check that the values of sub-items [1] and [2] in the expanded sub-items are " AIR6L_V2" and "Conveyor". As shown in Figure 6-7.

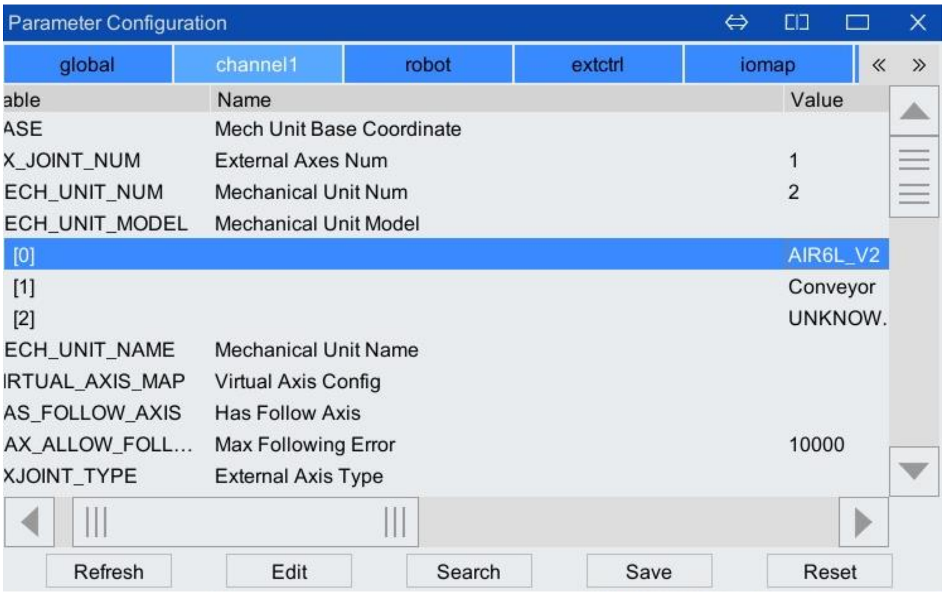


Figure 6-7 Mechanical unit model [MECH_UNIT_MODEL] parameter configuration interface

Step2. If you need to modify the configuration, click the line where the [0] subitem is located, click <Edit> button, and the dialog box as shown in Figure 6-8 will pop up. The parameters in [Value] can be modified. The parameter description is shown in Table 6-4. After viewing or modifying, click <Yes>.

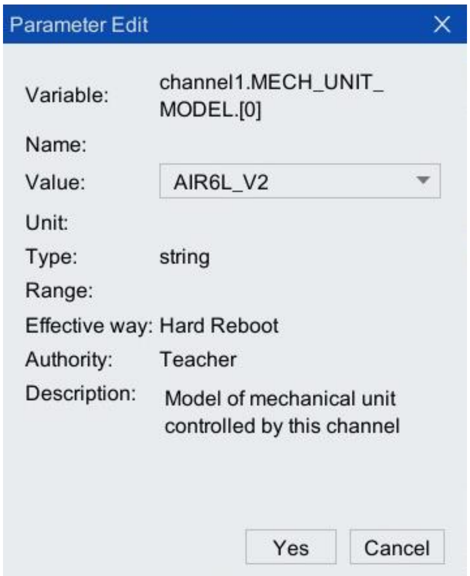


Figure 6-8 Mechanical unit model dialog box

Step3. Click the line where the [1] subitem is located and click the <Edit> button, and the dialog box as shown in Figure 6-9 will pop up. The parameters in [Value] can be modified. The parameter description is shown in Table 6-4.

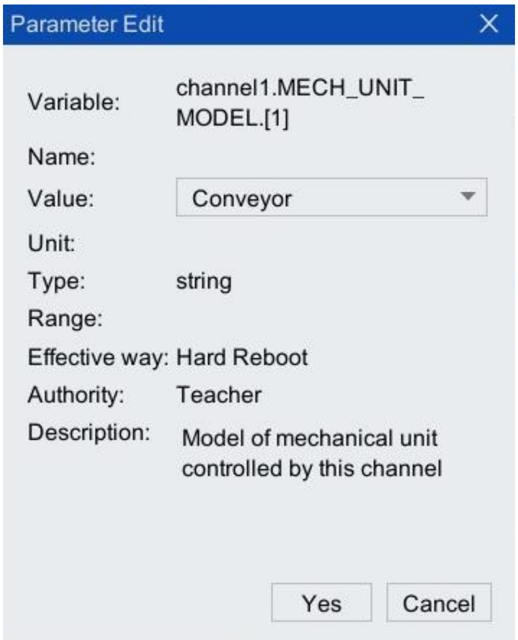


Figure 6-9 Mechanical unit model dialog box

Table 6-4 Parameter description

Parameter	Illustrate
Value	The value of the mechanical unit model [MECH_UNIT_MODEL] parameter (each sub-item) can select the mechanical unit model controlled by this channel. When the mechanical unit is a conveyor belt, the value is Conveyor.

Step4. After viewing or modifying the parameters, click <Yes>.

Mechanical unit name MECH_UNIT_NAME

The steps to view and modify parameters are as follows:

Step1. Select [System/Parameter Configuration/Channel] in the upper right corner of the teach pendant main interface, click [+] on the left side of the [MECH_UNIT_NAME] parameter, and check that the values of sub-items [1] and [2] in the expanded sub-items are " R1" and "C1". As shown in Figure 6-10.

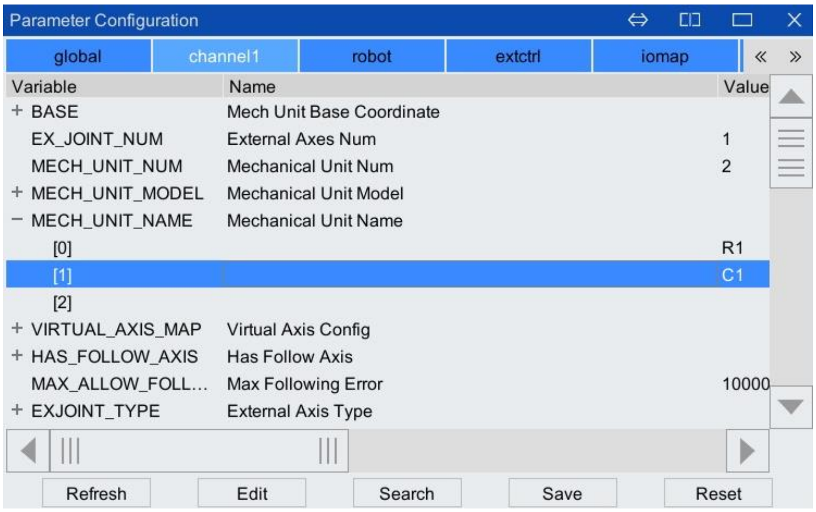


Figure 6-10 Mechanical unit name [MECH_UNIT_NAME] parameter configuration interface

Step2. To modify the configuration, click the line where the [0] subitem is located, click <Edit> button, and the dialog box as shown in Fig.6 -11 will pop up. The parameters in [Value] can be modified. The parameter description is shown in Table 6 5. Click <Yes> when the modification is complete.

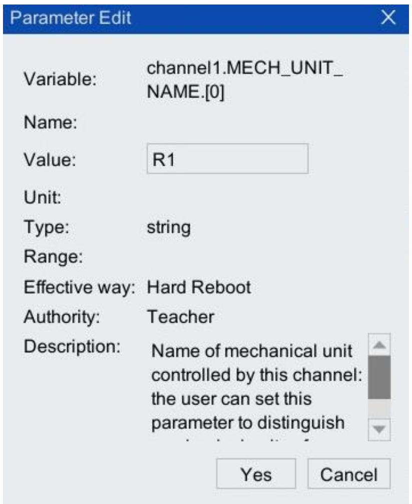


Figure 6-11 Mechanical unit name dialog box

Step3. If you need to modify the configuration, click the line where the [1] subitem is located, click the <Edit> button, and the dialog box shown in Figure 6-12 will pop up. The parameters in [Value] can be modified. The parameter description is shown in Table 6-5.

Figure 6-12 Mechanical unit name dialog box

Table 6-5 parameter description

Parameter	Illustrate
Value	The value of the mechanical unit name (MECH_UNIT_NAME) parameter (each sub-item) can distinguish the mechanical units of different channels. String format, which can be composed of letters, numbers, spaces, "." or "_" construct.

Step4. After viewing or modifying the parameters, click <Yes>.

Whether external axis uses external control EXJOINT_EXT_CONTROL

The steps to view and modify parameters are as follows:

Step1. Select [System/Parameter Configuration/Channel] in the upper right corner of the teach pendant main interface, click [+] on the left side of the [EXJOINT_EXT_CONTROL] parameter, and check that the value of the [0] sub-item is "true" in the expanded sub-items. As shown in Figure 6-13.

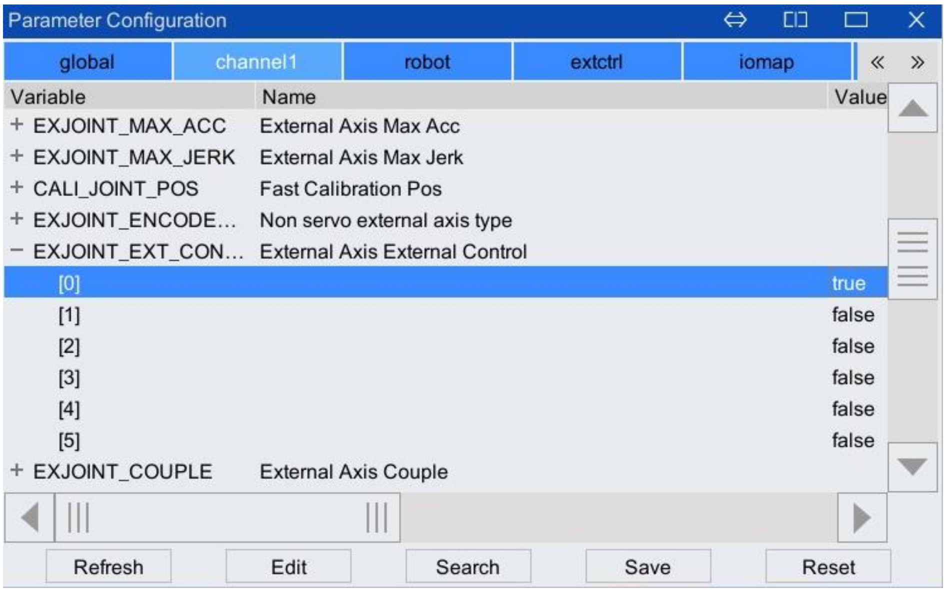


Figure 6-13 Whether the external axis uses external control (EXJOINT_EXT_CONTROL) parameter configuration interface

Step2. If you need to modify the configuration, click to select the row of the [0] sub-item in the [EXJOINT_EXT_CONTROL] parameter, click the <Edit> button, and a dialog box will pop up as shown in Figure 6-14. The parameters in [Value] can be modified. The parameter description is shown in Table 6-6.

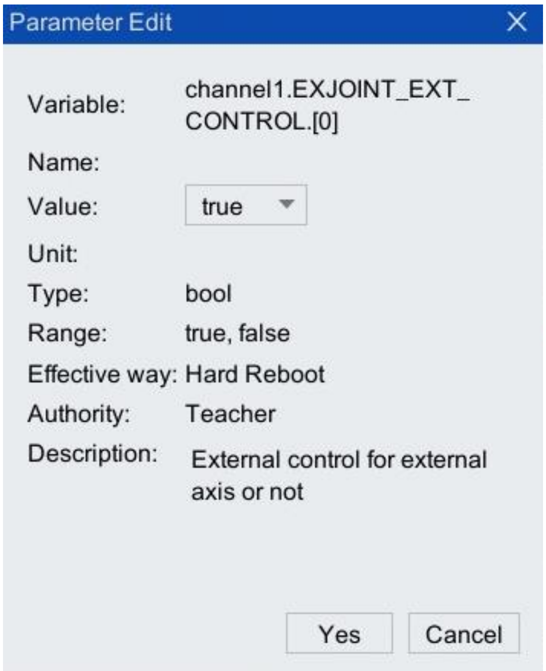


Figure 6-14 Whether to use external control dialog box for external axis

Table 6-6 Parameter Description

Parameter	Illustrate
Value	Whether external axis uses external control (EXJOINT_EXT_CONTROL) parameter (each sub-item)

Parameter	Illustrate
	value can define whether external axis uses external control. When the mechanical unit is a conveyor belt, the value must be true.

Step3. After viewing or modifying the parameters, click <Yes>.

External axis type EXJOINT_TYPE

The steps to view and modify parameters are as follows:

Step1. Select [System/Parameter Configuration/Channel] in the upper right corner of the teach pendant main interface, click [+] on the left side of the [EXJOINT_TYPE] parameter, and see that the value of the [0] sub-item is "1" in the expanded sub-items. As shown in Figure 6-15.

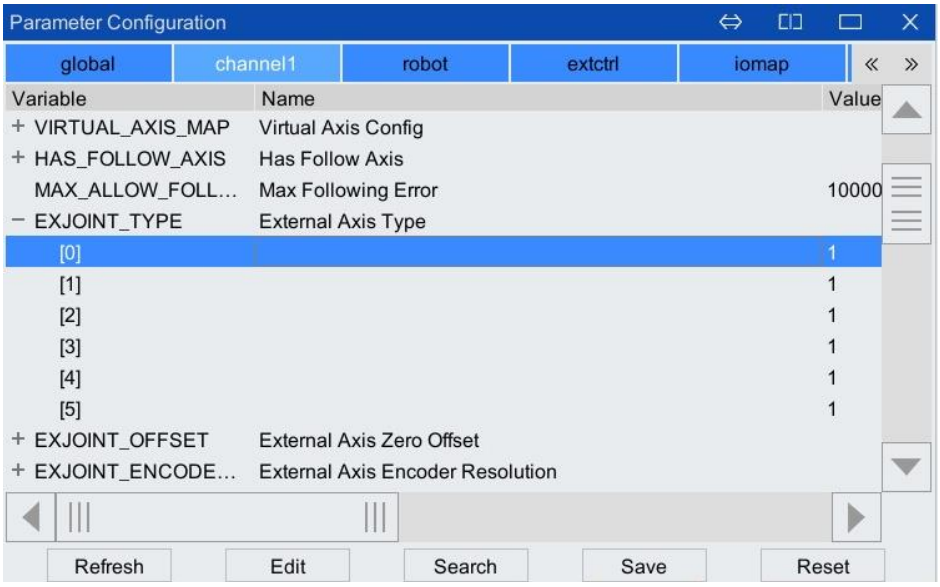


Figure 6-15 External axis type (EXJOINT_TYPE) parameter configuration interface

Step2. If you need to modify the configuration, click to select the row of the [0] sub-item in the [EXJOINT_TYPE] parameter, click the <Edit> button, and a dialog box will pop up as shown in Figure 6-16. The parameters in [Value] can be modified. The parameter description is shown in Table 6-7.

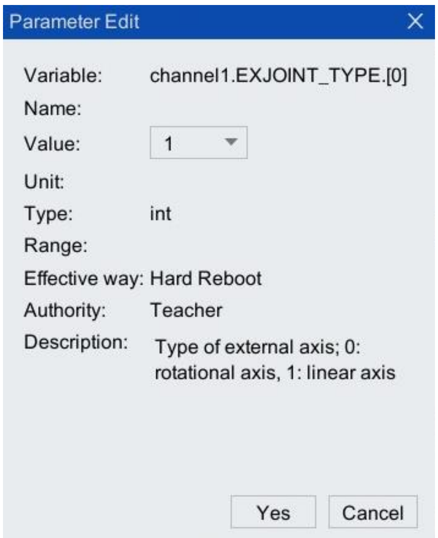


Figure 6-16 External axis type dialog box

Table 6-7 Parameter Description

Parameter	Illustrate
Value	By configuring the value of the external axis type (EXJOINT_TYPE) parameter (each sub-item), you can select the external axis type. The values are as follows: ■ 1: Straight line axis (straight line conveyor belt)

Step3. After viewing or modifying the parameters, click < OK >.

External axis encoder single-turn pulse number EXJOINT_ENCODER_RESO

The steps to view and modify parameters are as follows:

Step1. Select [System/Parameter Configuration/Channel] in the upper right corner of the teach pendant main interface, click [+] on the left side of the [EXJOINT_ENCODER_RESO] parameter, and in the expanded sub-items, you can see that the value of the [0] sub-item is "131072". As shown in Figure 6-17.

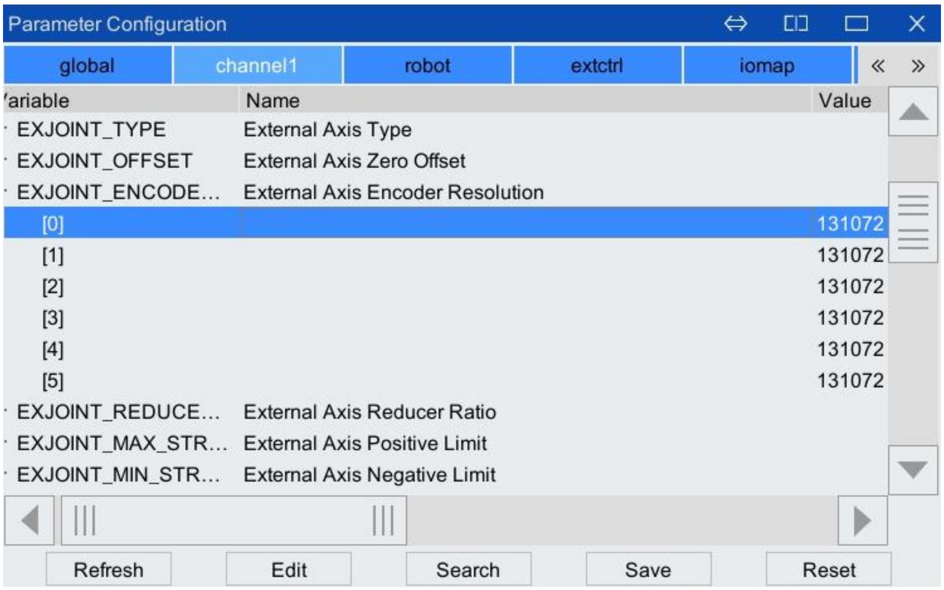


Figure 6-17 External axis type (EXJOINT_TYPE) parameter configuration interface

Step2. If you need to modify the configuration, click and select the row of the [0] sub-item in the [EXJOINT_ENCODER_RESO] parameter, click the <Edit> button, and a dialog box will pop up as shown in Figure 6-18. Modify the parameters in [Value]. The parameter description is shown in Table 6-8.

Figure 6-18 External axis type dialog box

Table 6-8 Parameter description

Parameter	Illustrate
Value	external axis encoder single-turn pulse number. ■ If you are using an "incremental encoder", the number of single-turn pulses is configured as 10000. ■ If you are using "Absolute encoder", the default configuration 131072 is used.

Step3. After viewing or modifying the parameters, click <Yes>.

Conveyor parameters

Click [System/Parameter Configuration] on the main interface of teach pendant. The newly created [Conveyor] tab is added to the [Parameter Configuration] interface. Click this tab to enter the interface shown to view and modify parameters. Select the row containing the parameter that needs to be modified and click <Edit> to modify the parameter.

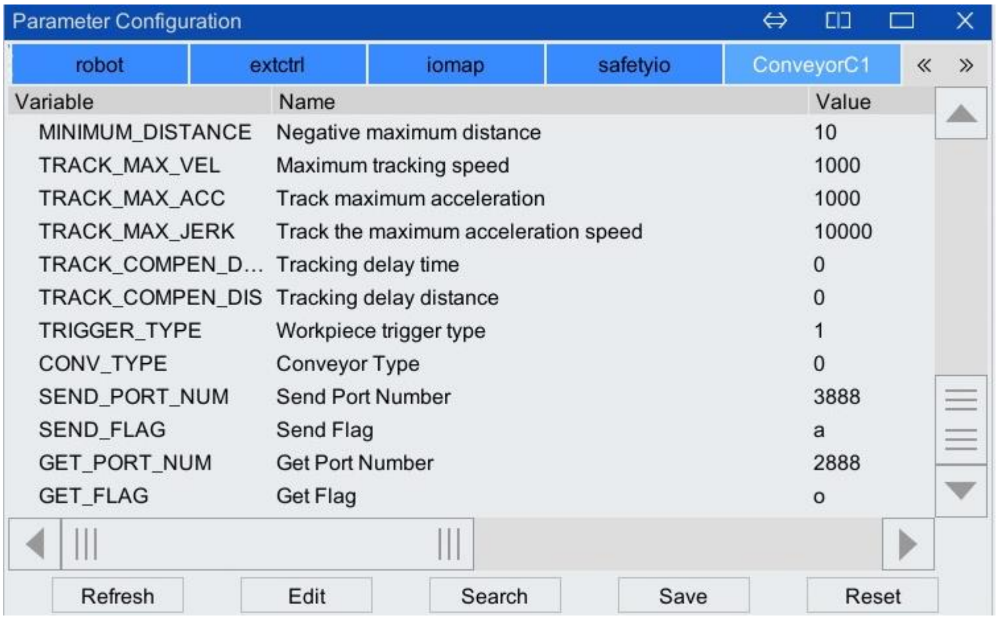


Figure 6-19 Conveyor belt parameter fine-tuning interface

Table 6-9 Parameter description

Parameter	Illustrate
MM_PER_PULSE	The distance represented by each conveyor belt pulse, in mm
GET_ON_OR_OFF_ACC	Jerk of getting on and off
ADJUST_SPEED	Adjust speed
VISUAL_WOBJ	Conveyor belt visual coordinate system for visual tracking
EXAIS_NO	External axis corresponding to the conveyor belt
EDGE_TYPE	Trigger mode, the values are as follows: ■ 0: rising edge trigger ■ 1: Falling edge trigger
LAST_TRIGGER_PULSE	The number of pulses the conveyor last triggered
CONV_ID	ID of the conveyor belt
VISION_IP	Vision IP, the network port IP of coupled on the industrial computer side. This network port is connected to the network port on the control cabinet side.
ENABLE_ADDWOBJ_WHEN_STOP	Whether to allow work object to queue when inactive
WAITTING_VISION_FLAG	Whether waiting for vision to send data.
REL_DIS	Robot processing delayed start distance
TRIGGER_SEPRATION	Conveyor belt trigger distance
MAX_WAITTING_TIME	The maximum waiting time for work object, if exceeded, an alarm will be issued.
MIN_SEPRATION	Minimum trigger distance between work object
MAXMUM_DISTANCE	Processing interval length
START_WINDOW	Getting on area length
PENDING_WINDOW	Conveyor belt waiting distance

Parameter	Illustrate
MINIMUM_DISTANCE	Conveyor belt tracking negative maximum distance
TRACK_MAX_VEL	Track maximum speed
TRACK_MAX_ACC	Track maximum acceleration
TRACK_MAX_JERK	Tracking maximum jerk
TRACK_COMPEN_DELAY	Tracking delay time
TRACK_COMPEN_DIS	Tracking delay distance
CONV_TYPE	Conveyor belt type is linear
SEND_PORT_NUM	Vision send port number
SEND_FLAG	Send the photo-taking instructions agreed with vision to vision
GET_PORT_NUM	Visual information receiving port number
GET_FLAG	Receive visual signal of end of sending message

Parameter saving

After the above parameters are configured, click <Save> at the bottom of the interface, and the interface as shown in Figure 6-20 will pop up. After selecting "Save global", click <Yes>. The parameters take effect after a power outage and restart.

Figure 6-20 Save interface

7 Conveyor belt related ARL instruction

7.1 Structure type

7.1.4 Cwobj

Description

The Cwobj structure is used to represent several parameters of a certain work object contain.

Format

```
struct Cwobj

{

double pulse

wobj visual_wobj

intid

int property

}
```

Member

For details about the members of the Cwobj structure type, see Table 7-1.

Table 7-1 Members of the Cwobj structure type

Name	Type of data	Illustrate
pulse	double	Conveyor belt pulse when work object getting on
visual_wobj	wobj	Coordinate system established on work object
id	int	work object id (automatically assigned by ARCS)
property	int	Customer-set work object features

Example

```
Cwobjw1//Declare variable w1

w1.pulse=getconvpulse(1,"C1")//acquire  current encoder pulse number

w1.property=2;//Set the work object  feature to 2
```

7.2 Basic instruction

7.2.1 waitwobj (Wait for work object)

Description

The waitwobj instruction is used to associate the robot with the work object that is to be tracked on the conveyor belt.

Format

waitwobj w

Parameter

Table 7-2 Parameters of waitwobj instruction

Name	Type of data	Illustrate
w	wobj structure	This parameter specifies the work object coordinate system waiting to be associated. The member variable unitname needs to be added to the wobj structure to represent the name of the mechanical unit to which the work object coordinate system refers (the reference coordinate system of wobj is the user coordinate system of the mechanical unit). When watiwobj instruction is executed, the tracked conveyor belt will be identified based on the mechanical unit name in parameter w.

Example

waitwobj $wobj1

7.2.2 waittime (Delayed waiting)

Description

The waittime instruction is used to specify how long the program waits. When executing this instruction, the advance planning of motion instruction will be automatically suspended. When performing conveyor belt tracking, execute this instruction, and the robot TCP and the conveyor belt remain relatively stationary.

Format

waittime time:

Parameter

Table 7-3 Parameters of waittime instruction

Name	Type of data	Illustrate
time	double	This parameter specifies the waiting time, in seconds.

Example

waittimetime:5//Indicates the waiting time of 5 seconds

7.3 Conveyor belt related functions

7.3.3 RemoveCwobj (Remove work object)

Description

Removes the specified work object from the queue of the conveyor belt work object with the specified name.

Format

RemoveCwobj (conv_name,c_wobj)

Parameter

Table 7-4 Parameters of the RemoveCwobj function

Name	Type of data	Illustrate
conv_name	string	Conveyor belt name
c_wobj	wobj	Conveyor belt work object to be removed

Example

RemoveCwobj("C1", w1)//Remove w1 work object in the conveyor belt named C1.

7.3.4 ClearCwobj (Clear work object)

Description

Clear all work object in the conveyor belt work object queue with the specified name.

Format

ClearCwobj (conv_name)

Parameter

Table 7-5 Parameters of ClearCwobj function

Name	Type of data	Illustrate
conv_name	string	Conveyor belt name

Example

ClearCwob("C1") //Clear all work object in the conveyor belt work object queue named C1.

7.3.5 GetCwobjSize (Acquire work object number)

Description

Gets the number of work objects in the conveyor work object queue for the specified name.

Format

GetCwobjSize(conv_name)

Parameter

Table 7-6 Parameters of GetCwobjSize function

Name	Type of data	Illustrate
conv_name	string	Conveyor belt name

Return value

After acquire is successful, the number of work objects in the current queue is returned.

The return value type is: int.

Example

inta=GetCwobjSize("C1")

print a//Print out the number of work object in the work object queue of conveyor belt C1

7.3.6 DropCwobj (Stop monitoring work object)

Description

The DropCwobj function stops monitoring whether the current work object exceeds the maximum distance between the negative and positive directions.

Format

DropCwobj (index , conv_name)

Parameter

Table 7-7 Parameters of the DropCwobj instruction

name	type of data	illustrate
index	int	Channel number
conv_name	string	Conveyor belt name

Example

DropCwobj(1,"C1")//Stop detecting the work object being associated on conveyor belt C1 in channel 1

7.3.7 GetIntFromConveyorDB (Acquire int type conveyor belt parameter)

Description

Read the int type data item of the conveyor belt with the specified name from the database and assign its value to a variable.

Format

GetIntFromConveyorDB(conv_name,"A",a)

Parameter

Table 7-8 Parameters of the GetIntFromConveyorDB instruction

Name	Type of data	Illustrate
conv_name	string	Conveyor belt name
A	string	Data item name
a	int	The read parameters are assigned to the variable.

Example

int trigger_DI

GetIntFromConveyorDB("C1","CONV_SYNC_TRIGGER_DI",trigger_DI)//Read the data of CONV_SYNC_TRIGGER_DI on the C1 conveyor belt and store it in the variable trigger_DI.

7.3.8 GetFromConveyorDB (Acquire non-int type conveyor belt parameter)

Description

Reads the conveyor belt data item with the specified name from the database and assigns its value to the specified variable.

Format

GetFromConveyorDB(conv_name,"A",a)

Parameter

Table 7-9 Parameters of the GetFromConveyorDB instruction

Name	Type of data	Illustrate
conv_name	string	Conveyor belt name
A	string	Data item name
a	double\bool\string	The read parameters are assigned to the variable.

Example

double MM_per_pulse

GetFromConveyorDB("C1","MM_PER_PULSE",MM_per_pulse)// Assigns the MM_PER_PULSE data item from conveyor C1 to the variable MM_per_pulse.

7.3.9 getconvpulse (Acquire current encoder pulse number)

Description

Acquire specifies the current encoder pulse number of the conveyor belt.

Format

getconvpulse (index , conv_name)

Parameter

Table 7-10 Parameters of getconvpulse instruction

Name	Type of data	Illustrate
index	int	Channel number
conv_name	string	Conveyor belt name

Return value

Type: double

Returns the current pulse number of the encoder

Example

doublep_p=getconvpulse(1,"C1")

printp_p//Print the pulse number of the current encoder of conveyor belt C1 in channel 1

7.3.10 AddCwobj (Add work object)

Description

Add the specified work object to the queue of the specified conveyor belt.

Format

AddCwobj(conv_name,c_wobj)

Parameter

Table 7-11 Parameters of the AddCwobj instruction

Name	Type of data	Illustrate
conv_name	string	Conveyor belt name
c_wobj	wobj	Work object parameters

Example

AddCwobj("C1",w1)//Add work object w1 to the conveyor belt C1 queue.

7.3.11 GetCwobj (Acquire first work object)

Description

Gets the data contained in the Cwobj struct of the first work object in the conveyor work object queue with the specified name.

Format

GetCwobj(conv_name)

Parameter

Table 7-12 Parameters of the GetCwobj instruction

Name	Type of data	Illustrate
conv_name	string	Conveyor belt name

Return value

Type: Cwobj structure

After acquire is successful, the information of the first work object in the queue is returned.

Example

Cwobjw2

w2=GetCwobj("C1")// Stores the data body of the first work object in the C1 queue on the conveyor belt into w2.

7.3.12 setinterpercent (Program running multiplication function)

Description

Set the program running multiplication. When it is necessary to set the robot and the conveyor belt to be relatively stationary through external signals, the DI trigger can be set, and the program magnification is set to 0 in the trigger function, which achieves relative stationarity.

Format

setinterpercent (int interpercent)

Parameter

Table 7-13 Parameter description

Name	Type of data	Illustrate
interpercent	int	magnification value. The value range is 1~100

Example

setinterpercent(5)//Set the running speed multiplier to 5%

7.3.13 cjoint (Conveyor belt position information printing function)

Description

If you need to print the real-time position of the conveyor belt in the program, you can use the cjoint function.

Format

jointcjoint()

Return value

Type: joint

Returns the current axis position.

Example

When the conveyor belt is configured at No. 1 external axis, the real-time position of the conveyor belt can be printed through this line.

printcjoint().ej1

> **Tip**
> The actual printing result may deviate slightly from the set value due to errors (within the range of position accuracy).

7.4 Conveyor belt visual function

7.4.14 Postocwobj (Work object coordinate system conversion function)

Description

Convert the visual transmission point to the work object coordinate system.

Format

Postocwobj (pos , base, fixed, conv_name)

Parameter

Table 7-14 Parameters of the Postocwobj instruction

Name	Type of data	Illustrate
pos	carry<double>	Work object point information captured by camera
base	wobj	Base coordinate system name
false	bool	Whether the work object coordinate system after conversion is fixed, the values are as follows: ■ true: The work object coordinate system is fixed, that is, the work object coordinate system is the default value ■ false: The work object coordinate system is not fixed, that is, the values of the coordinate system need to be calibrated manually.
conv_name	string	Conveyor belt name

Example

$wobj1=PosToConvWobj(poseData,$Base[1],false,"C1")//Convert the visually captured work object point information to the work object coordinate system on the conveyor belt C1. The base coordinate system of the conveyor belt is Base[1], select the default value for the converted work object coordinate system.

7.4.15 Setconvwobjpos (Work object point information is stored in the work object queue)

Description

Store the work object point information transmitted by the camera into the work object queue.

Format

setconvwobjpos(index,conv_name,1,pos)

Parameter

Table 7-15 Parameters of the setconvwobjpos instruction

Name	Type of data	Illustrate
index	int	Channel number
conv_name	string	Conveyor belt name
1	int	Fixed to transmit one at a time
pos	carry<double>	Point information of work object

Example

setconvwobjpos(1,"C1",1,posdata)//On channel 1, send the 1 work object point information posdata captured by the vision camera to the conveyor belt C1 work object queue.

8 Tracker example

Place the following main program and subprograms in the same folder of teach pendant, load and run the main program.

Main program

```
func void main()

init()

setip("192.168.1.1","192.168.1.11","255.255.255.0","eth1")//Set the robot ip

stringcname=$mech_unit_name[1]// Obtains the name of the conveyor

sub::start(cname)

Cwobjw2


movejj:j1,vp:40%,sl:0mm,t:$FLANGE// Back to home point


while(1)


waitwobj$wobj0//Waiting for work object  to get on

print "work object  gets on "


w2=GetCwobj(cname)// Gets information about the first work object in the queue

if(sub::trigger_type! =0)$wobj0=w2.visual_wobj// Used only in visual mode

print"w2=",w2


Linp: p1, vl: 400 mm/s, sl: 0 mm, t: $FLANGE, w: $WORLD / / get on

if(w2.property==0)//Perform different tracking instruction  according to the work object property parameter

linp:p2,vl:400mm/s,sl:0mm,t:$FLANGE,w:$wobj0

waittime 1

else

//linp:p3,vl:400mm/s,sl:0mm,t:$FLANGE,w:$wobj0

waittime 4

//linp:p4,vl:400mm/s,sl:0mm,t:$FLANGE,w:$wobj0
```

```
endif

lin p:p5,vl:400mm/s,sl:0mm,t:$FLANGE,w:$WORLD  //get off


DropCwobj(1,cname)


endwhile

endfunc
```

Subroutine

```
//Declare and initialize conveyor belt variables

Cwobjw1

socket send,receive

string data,endchar

intwobjNum=0

int a,count=0

double p[60]

double posdata[6]


string cname=$mech_unit_name[1]

int trigger_type=3

int trigger_DI=0

int edge_type= 0

int exais_NO=0


intconv_id=0

int CONV_type = 2

int send_port = 0

int receive_port= 0

double MM_per_pulse = 0
```

```
double last_triggered_pulse=0
double min_sepration=99
double max_wait_time =99
double trigger_sepration=10
bool waiting_vision=true
string send_flag = "sig"
string receive_flag="endchar"
string vision_ip="192.168.0.1"
bool add_when_stop=true

funcvoidget_var()//acquire  conveyor belt variable value
GetIntFromConveyorDB(cname,"CONV_SYNC_TRIGGER_DI",trigger_DI)
print"trigger_DI=",trigger_DI
GetIntFromConveyorDB(cname,"EDGE_TYPE",edge_type)
print "edge_type=",edge_type
GetIntFromConveyorDB(cname,"EXAIS_NO",exais_NO)
print"exais_NO=",exais_NO

GetIntFromConveyorDB(cname,"CONV_ID",conv_id)
print"conv_id=",conv_id
GetIntFromConveyorDB(cname,"TRIGGER_TYPE",trigger_type)
print"trigger_type=",trigger_type
GetIntFromConveyorDB(cname,"CONV_TYPE",CONV_type)
print"CONV_type=",CONV_type
GetIntFromConveyorDB(cname,"SEND_PORT_NUM",send_port )
print "send_port=",send_port
GetIntFromConveyorDB(cname,"GET_PORT_NUM",receive_port )
print "receive_port=",receive_port

GetFromConveyorDB(cname,"MM_PER_PULSE", MM_per_pulse)
```

```
print "MM_per_pulse=",MM_per_pulse
GetFromConveyorDB(cname,"LAST_TRIGGER_PULSE", last_triggered_pulse)
print "last_triggered_pulse=",last_triggered_pulse

GetFromConveyorDB(cname,"MIN_SEPRATION",min_sepration)
print "min_sepration=",min_sepration

GetFromConveyorDB(cname,"SEND_FLAG",send_flag )
print "send_flag=",send_flag
GetFromConveyorDB(cname,"GET_FLAG",receive_flag )
print "receive_flag=",receive_flag
GetFromConveyorDB(cname,"VISION_IP",vision_ip)
print"vision_ip=",vision_ip

GetFromConveyorDB(cname,"ENABLE_ADDWOBJ_WHEN_STOP", add_when_stop)
print "add_when_stop",add_when_stop
GetFromConveyorDB(cname,"WAITTING_VISION_FLAG",waiting_vision )
print "waiting_vision=",waiting_vision
GetFromConveyorDB(cname,"TRIGGER_SEPRATION",trigger_sepration)
endfunc
funcvoidaddqueue0(stringconveyor_name)//IO mode trigger function

w1.pulse=getconvpulse(1,conveyor_name)// Get the number of real-time pulses
print"w1=",w1
count++
w1.property = count%2
//Trigger on falling edge
if (edge_type==1)
print "high_low"
waituntilgetdi(trigger_DI)==0
```

```
w1.pulse=getconvpulse(1,conveyor_name)// Get the number of pulses from the back end of the work object as it passes through the photoelectric switch
endif

addcwobj (conveyor_name, w1)
waittime 0
print" Number of current queue objects =",GetCwobjSize(conveyor_name)

endfunc

funcvoidaddqueue1(stringconveyor_name)//IO+ visual mode trigger function
if (waiting_vision==true)
print "Visual calculation is not completed, work object is ignored"
return
else
waiting_vision=true
//Trigger on falling edge
if (edge_type==1)
waituntilgetdi(trigger_di)==0
endif
waituntil write(send,send_flag)
print"a"
w1.pulse=getconvpulse(1,conveyor_name)// Get the number of pulses from the back end of the work object as it passes through the photoelectric switch
count++
w1.property=count%2

//Clear the queue
clearbuff(receive)
//Receive all position offset data of work object
```

```
waituntil readuntil(receive,data,receive_flag)

print "data=",data

scan
from:data,delimiter:",",p[0],p[1],p[2],p[3],p[4],p[5],p[6],p[7],p[8],p[9],p[10],p[11],p[12],p[13],p[14],p[15],p[16],p[17],p[18],p
[19],p[20],p[21],p[22],p[23],p[24],p[25],p[26],p[27],p[28],p[29],p[30],p[31],p[32],p[33],p[34],p[35],p[36],p[37],p[38],p[39],p
[40],p[41],p[42],p[43],p[44],p[45],p[46],p[47],p[48],p[49],p[50],p[51],p[52],p[53],p[54],p[55],p[56],p[57],p[58],p[59],endcha
r// The point at which the visual transmission is received

wobjNum=0

for(int i=0;i<10;i++)

if (p[6*i]!=0)

wobjNum++

endif

endfor

print"wobjNum=",wobjNum


for(int i=0;i<wobjNum;i++)

posdata[0]=p[i*6]

posdata[1]=p[i*6+1]

posdata[2]=p[i*6+2]

posdata[3]=p[i*6+3]

posdata[4]=p[i*6+4]

posdata[5]=p[i*6+5]

print"posdata=",posdata


setconvwobjpos(1,conveyor_name,1,posdata)


w1.visual_wobj=posToCWobj(posdata,$Base[1],false,conveyor_name)// Convert the visual sent point to the work object
coordinate system

print "wobj1=  ",w1.visual_wobj


addcwobj (conveyor_name, w1)
```

```
print "add"


endfor

waiting_vision=false

waittime 0

print" Number of current queue objects =",GetCwobjSize(conveyor_name)

endif


endfunc




funcvoidaddqueue2(stringconveyor_name)// Visual mode trigger function


doublepulse_current=getconvpulse(1,conveyor_name)// Number of pulse records


if (waiting_vision==true)

print "Visual calculation is not completed, work object  is ignored"

return

elseif (abs(last_triggered_pulse-pulse_current)*abs(MM_PER_PULSE)<trigger_sepration)

print abs(last_triggered_pulse-pulse_current)*abs(MM_PER_PULSE)

else

waiting_vision=true


waituntil write(send,send_flag)

print send_flag


w1.pulse=getconvpulse(1,conveyor_name)

last_triggered_pulse=w1.pulse

//Clear the queue

clearbuff(receive)
```

```
//Receive all position offset data of work object

waituntil readuntil(receive,data,receive_flag)
scan
from:data,delimiter:",",p[0],p[1],p[2],p[3],p[4],p[5],p[6],p[7],p[8],p[9],p[10],p[11],p[12],p[13],p[14],p[15],p[16],p[17],p[18],p
[19],p[20],p[21],p[22],p[23],p[24],p[25],p[26],p[27],p[28],p[29],p[30],p[31],p[32],p[33],p[34],p[35],p[36],p[37],p[38],p[39],p
[40],p[41],p[42],p[43],p[44],p[45],p[46],p[47],p[48],p[49],p[50],p[51],p[52],p[53],p[54],p[55],p[56],p[57],p[58],p[59],endcha
r// Receive visual transmission points

wobjNum=0
for(int i=0;i<10;i++)
if (p[6*i]!=0)
wobjNum++
endif
endfor
print"wobjNum=",wobjNum


for(int i=0;i<wobjNum;i++)
posdata[0]=p[i*6]
posdata[1]=p[i*6+1]
posdata[2]=p[i*6+2]
posdata[3]=p[i*6+3]
posdata[4]=p[i*6+4]
posdata[5]=p[i*6+5]
print"posdata=",posdata

setconvwobjpos(1,conveyor_name,1,posdata)

w1.visual_wobj=posToCWobj(posdata,$Base[1],false,conveyor_name)// Convert the visual sent point to the work object
coordinate system
```

```
print"w1=",w1
addcwobj (conveyor_name, w1)
print "add"
endfor
waiting_vision=false
waittime 0
print" Number of current queue objects =",GetCwobjSize(conveyor_name)
endif



endfunc



func void start(string conveyor_name)
get_var()
clearcwobj(conveyor_name)// Empty the queue
waiting_vision=false// Initializes the variable
for(int i=0;i<60;i++)
p[i] = 0
endfor
if(trigger_type==0)//If it is IO trigger
interrupt when:getdi(trigger_DI),do:addqueue0(cname)
//DI triggers the queue function
elseif(trigger_type==1)//If it is IO+visual trigger
waituntilconnect(send,vision_ip,send_port)// Establishes communication with vision
waituntil connect(receive, vision_ip,receive_port)
print"connected"
interruptwhen:getdi(trigger_DI),do:addqueue1(cname)//DI Triggers the queueing function
elseif(trigger_type==2)//If it is a visual trigger
waituntilconnect(send,vision_ip,send_port)// Establishes communication with vision
```

waituntil connect(receive, vision_ip,receive_port)

print"connected"

Timerinterval: 0.5, rmode: true, do: addqueue2 (cname) // timer trigger is used to implement distance photograph, into the queue

endif

endfunc

PEITIAN ROBOTICS

WeChat Official Account

Official Website

Sevice Hotline : 400-990-0909
Official Website : http://robot.peitian.com

UM-ZY0206-BJ-001 / V1.5.0 / 2024.03.04