ROBUSTMOTION

PRODUCT USER MANUAL

RM-MFGB-ITG

MULTI-FUNCTION LONG STROKE GRIPPER

Please read this MANUAL carefully before using the product.

DOCCODE: MFGBPUM2509

Product Disclaimer Statement

Dear Customer,

First and foremost, we extend our heartfelt gratitude for choosing products from Foshan Augmented Intelligence Technology Co., Ltd. (hereinafter referred to as "we" or "our company"). This disclaimer aims to clarify the potential risks and responsibilities associated with the use of our products, ensuring that the rights and interests of both parties are effectively protected. Please read the following carefully and thoroughly understand the content.

1. Product Usage Risk Warning
   Our products are designed and manufactured in strict accordance with industry standards. Nevertheless, it is inevitable that any product may carry certain risks associated with its use. We strongly advise you to strictly follow operating procedures and safety manuallines when using our products to minimize potential risks to the greatest extent. Please note that you should bear the relevant risks associated with the use of our products, including but not limited to product performance, accuracy, and applicability.

2. Disclaimer
   We do not assume any liability for any losses or damages caused by the following situations:
   1)Improper operation, misuse, unauthorized modification, or use beyond the scope of the product.
   2)The product cannot meet all specific purposes of the user. It is recommended that the user assess whether the product meets their specific needs before use.
   3)Indirect damages, special damages, incidental damages, or consequential damages caused by product failures, delays, or defects resulting from the use of third-party maintenance services not authorized by our company.
   4)Any liability arising from the combination of third-party products or services with our products.

3. Copyright and Intellectual Property Statement
   All products, services, and trademarks provided by our company are protected by intellectual property laws and other relevant laws and belong to our company. Without our explicit permission, no unit or individual may modify, rent, lease, sell, share, or distribute our products and services in any way.

4. Product Quality Guarantee
   Our product quality guarantee is limited to manufacturing defects inherent to the product itself. This guarantee does not apply to products that have been improperly stored, assembled, used, or placed in outdoor or humid environments. It does not cover normal wear and tear, cuts, and scratches, or damages caused by impact or accidents.

5. Applicable Law and Jurisdiction
   This disclaimer is governed by and interpreted in accordance with the laws of the People's Republic of China. In the event of any dispute, both parties should first attempt to resolve it through friendly negotiation; if negotiation fails, either party has the right to submit the dispute to the People's Court located where our company is based.

6. Modifications and Updates
   We reserve the right to modify, update, upgrade, or discontinue the product at any time. For any modifications to this disclaimer, we will publish updates on our company's website or in the product manual, which will take effect immediately upon publication.

7. Other Terms
   This disclaimer does not replace any other contractual relationships between the parties. If there are other contract terms, please comply with them as well.

We look forward to continuing our cooperation with you and are committed to providing you with high-quality products and services.

PREFACE

OVERVIEW

RM-MFGB-ITG multi-functional long-stroke gripper is a mature and widely adopted intelligent electric gripper, featuring distinctive advantages including high rigidity, long stroke range, high precision, exceptional reliability, and simplified control. It comes with a built-in waterproof connector and achieves IP55 protection rating.RM-MFGB-ITG multi-functional long-stroke gripper is compatible with industrial robots, collaborative robots, and single or multi-axis equipment modules from various domestic and international brands, capable of performing various industrial operations such as clamping, transporting, assembling, palletizing, handling, and inspecting items of different sizes, shapes, and weights.

This manual provides comprehensive information on the product, including an overview, installation considerations, Commands, application examples, communication and control methods, software debugging tools, troubleshooting, and maintenance. For first-time use, please be sure to read this manual carefully. If you have any questions or doubts regarding the content of the manual, please feel free to consult our engineers or technical staff for professional guidance.

APPLICABLE MODELS

- This manual is applicable to the entire range of RM-MFGB-ITG (Multi-Function Long Stroke Gripper) series products.

FEATURES

- Large Stroke
- Large Gripping Force
- Adaptive Gripping
- IP55 Level
- Waterproof Connector
- Supports Internal Gripping/External Gripping
- Speedy Gripping
- High-Speed Response
- Stable Output
- Multi-point Position Control
- High Repeatability Accuracy
- Automatic Drop Detection
- Supports Automatic Measurement Of Inner Diameter/Outer Diameter
- Industrial-Grade Durability

APPLICATION

- 3C Electronics Manufacturing
- Automated Production & Assembly
- Battery Manufacturing
- Automation Equipment
- Cosmetics Production
- Other Industries

PRECAUTIONS

1. This manual serves as a general manual for a series of products. The illustrations within are provided as examples and may differ from the product you have ordered.
2. We are committed to the continuous improvement of our products.RobustMotion reserves the rightto amend, upgrade, or modify the appearance and performance at any time without prior notice. Please refer to the latest information on our official website.
3. Should you encounter any issues during use, please contact our after-sales technical engineers for assistance.

Contents

1 Product Introduction

1.1 Product Description

RM-MFGB-ITG (Multi-Functional Long-Stroke Gripper)

The RM-MFGB-ITG multi-functional long-stroke grippers are powered by one servo motor, which simplifies operation and control and can automatically adjust to fit the shape of the object being gripped.

The RM-MFGB-ITG has two main working modes to adapt to different clamping needs:

1. External Gripping: Suitable for clamping objects within the gripper's stroke band, it does not limit the shape of the object, offering a wide band of applicability.
2. Internal Gripping: Suitable for clamping objects with irregular shapes or those with outer surfaces that cannot be squeezed, ensuring effective clamping of special-shaped objects.

The design of the RM-MFGB-ITG multi-functional long-stroke grippers not only enables the grippers to adapt to objects of various shapes and sizes, but also eliminates the need for frequent replacement of end effectors. With simple gripping command actions, they can perform multifunctional gripping tasks, greatly enhancing their application efficiency in various industries, such as 3C electronics, automotive manufacturing, logistics, and food processing.

Compared to pneumatic grippers, the RM-MFGB-ITG multi-functional long-stroke grippers support adaptive flexible gripping, eliminating the need for frequent replacement of end effectors. As long as it is within the effective stroke band of the actuator, they can quickly, effectively, and stably complete the gripping of workpieces. With simple gripping command actions, they can achieve multifunctional and compound motion grasping tasks, greatly enhancing the application efficiency in various industries, such as 3C electronics, automotive manufacturing, logistics, food processing, and other fields.

The standard product is the ITG models with integrated drive and control system. Please refer to the following table for detailed specifications.

1.2 ITG Integrated Models (with Built-in Controller)

Item		Product Model and Specifications
Model		RM-MFGB-08-80-1-ITG
Size Code		08
Stroke (mm)		80
Type Code		150
Max.gripping Force (N)		3
Max. load Mass (kg)		±0.03
Repeat Positioning Accuracy (mm)		0.7/0.7
Max.opening/closing Time (s)		MR: 5.5 , MP: 3.7 , MY: 3.1
Permissible Load Torque (N.m)		MR:8.0,MP:5.9,MY:5.9
Controller		Controller Built-in
Control Method	Bus	Modbus RTU
	Interface	I/O
I/O Interface		Input 3 Points(NPN),Output 3 Points(NPN)
Rated Voltage (V)		DC24±10%
Rated Current (A)		4
Weight (kg)		1
Usage Environment		0~40℃、＜85%RH (Without condensation)
Protection Class IP		IP55

1.3 Packing List

Please check the "Sales Dispatch Note" in the packaging box to confirm whether it corresponds with the product model and quantity received.

Sales Dispatch Note

Customer Name: xxx Co., Ltd.
Attn: Zhang Xiaoming
Phone: xxx xxxx xxxx
Address: Beijing City xxx xxx xxxx

Shipping Date: 2022-08-08
Document Number: xxxx xxxx
Remarks: xxx

No.	Item	Unit	Quantity	Remark
1	RM-MFGB-08-80-1-ITG Gripper	pcs	78	
2	USB to RS485 Adapter	pcs	78	
3				
4				

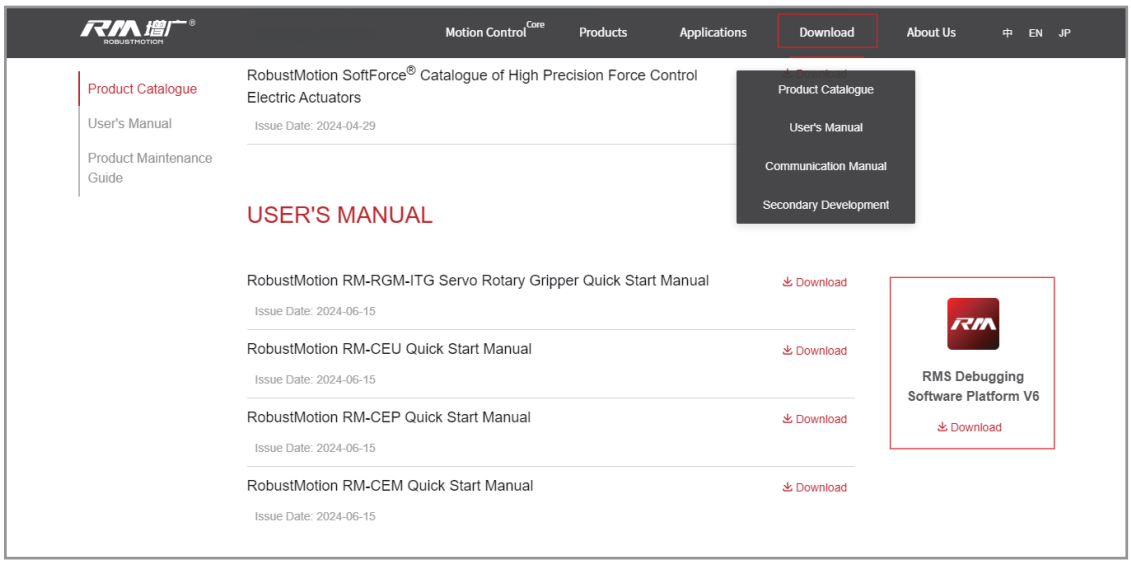
1.4 Extra Items Prepared by User

1. Output power supply: DC24V±10% . Please refer to the controller label for rated current.
2. A computer or laptop.

	Minimum System Requirements of the PC
Processor	Intel® or AMD Processor with 64-bit Support
Operating System	Windows 10 (64-bit) Version or Above
RAM	2GB

1.5 RMS Software

Please visit the official website of RobustMotion (www.rmaxis.com/en) Download page to download the software, or contact our after-sales engineers to obtain the corresponding version of the software package.



2 Wiring of the Actuator

- Please perform wiring on the actuator while it is powered off. Do not turn on the power before the wiring is complete, as plugging in with power can damage the actuator or the controller.
- The operating environment for the electric gripper should be within 0-40° C and below 85% RH (without condensation). Try to meet the operating conditions of the electric gripper to prevent any malfunction.

2.1 Design and Installation of the Tooling Fixture

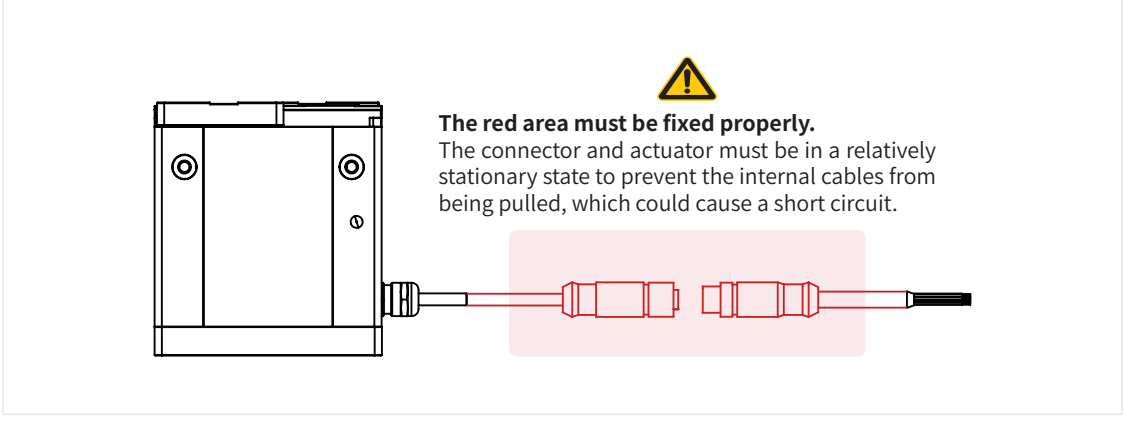
The front end of the gripper can be customized or replaced according to specific application requirements and the size of the object to be gripped to achieve the best gripping effect.
When gripping by friction, it is recommended that the gripping force should be at least 10 to 20 times the weight of the object to ensure a strong grip even when the contact surface is small and not aligned properly.

During the installation process, the stability of the gripper should be ensured by following these steps:

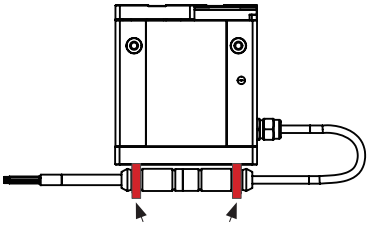
1. Utilize all available bolt holes to secure the gripper body when fixing it in place.
2. Ensure that all bolts are tightened properly to prevent any loosening.
3. Make sure that the jig you design aligns precisely with the gripper's slider, and the corresponding holes should match. If necessary, use adapter devices to adjust the alignment.
4. During use, you can adjust the action of the gripper to ensure efficient clamping.

2.1.1 Recommended Cable-tying and Fixing Method

To ensure the stability of cable connections during the use of the product, it is recommended to use flexible fixing devices to secure the cables. This approach allows for the appropriate band of motion for the cables and ensures they have sufficient elasticity to handle bending and stretching. It helps prevent accidental detachment or disconnection due to unstable contact between the cables and the electric gripper connection points during operation.

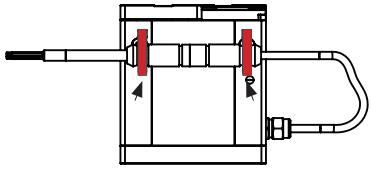


Bottom Fixing



Suggested fixing position

Middle Fixing

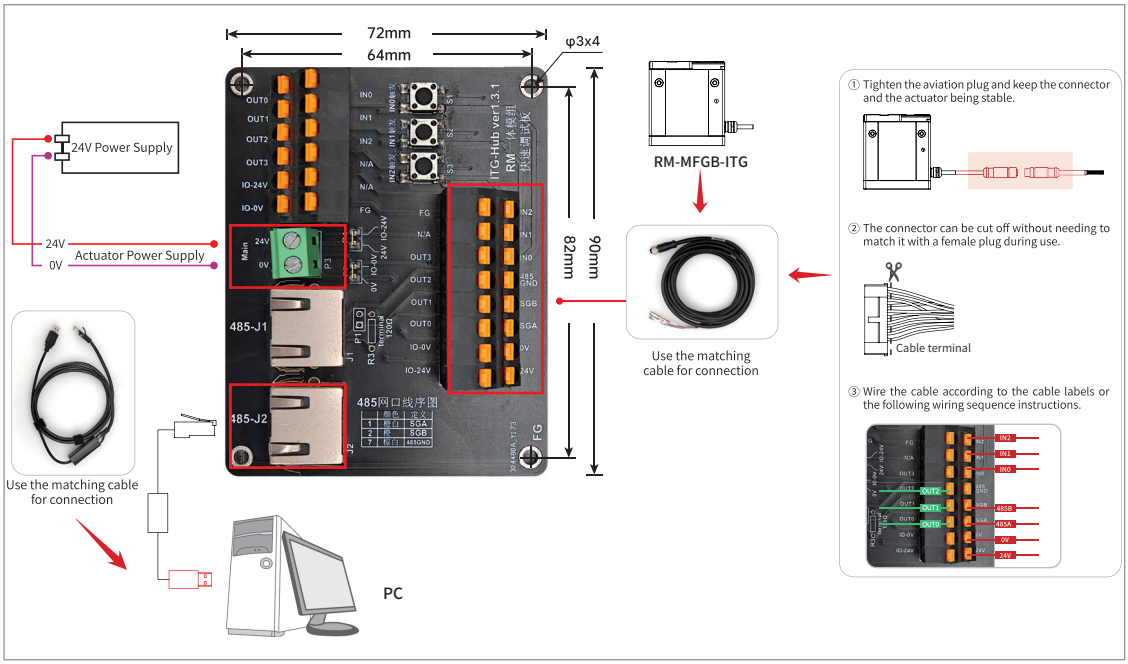


Suggested fixing position

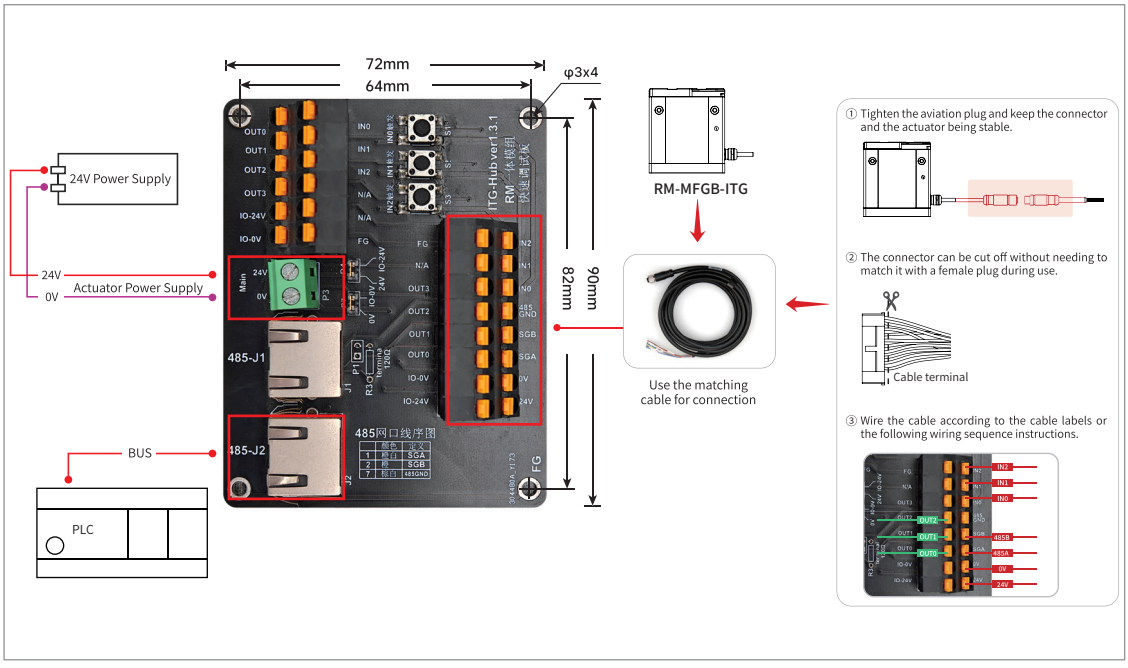
2.2 Integrated Connection Panel Wiring Instructions

The "integrated connection panel" is intended for quick debugging by first-time users and is generally not required for regular use.

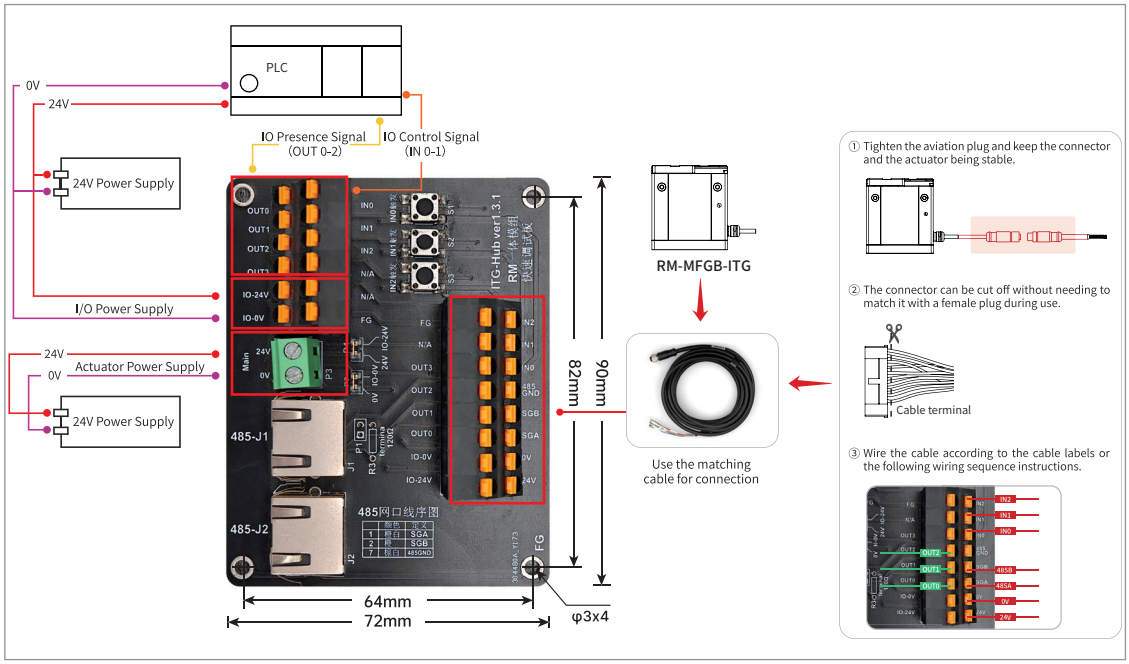
2.2.1 Upper Computer Software Debugging Wiring Method



2.2.2 Bus Control Wiring Method



2.2.3 I/O Control Wiring Method



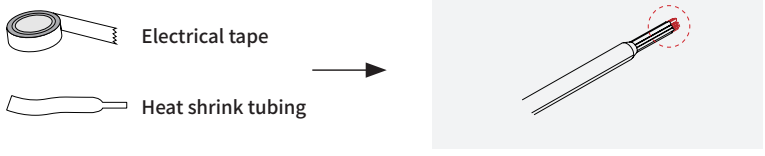
2.2.4 Actuator Wire Sequence Description

Group	Color	Definition	Description
Main Power Supply	Purple and White	24V	Actuator Power 24V
	Purple	0V	Actuator Power 0V
CAN	Red	CAN_L	CANopen Communication L
	Black	CAN_H	CANopen Communication H
Shielded Wire	Silver White	FG	Shell Ground FG
I/O Input	Yellow	IN0	Input 0
	Yellow White	IN1	Input 1
	Grey	IN2	Input 2
I/O Output	Brown	OUT0	Output 0
	Brownish and White	OUT1	Output 1
	Blue	OUT2	Output 2
485	Green	485-A	485 Communication A
	Oband	485-B	485 Communication B

Different batches of cables may cause slight differences in the color of the wire core. Please refer to the actual color of the cable for details.

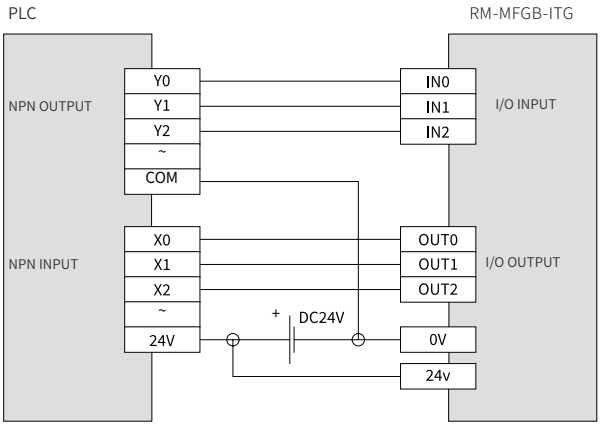
2.2.5 Insulation Protection of Loose Wire

After completing the required wiring, for any unused loose wires, it is imperative to use insulating electrical tape or insulating heat shrink tubing to provide insulation protection for the loose wires to prevent accidental contact that could cause a short circuit.

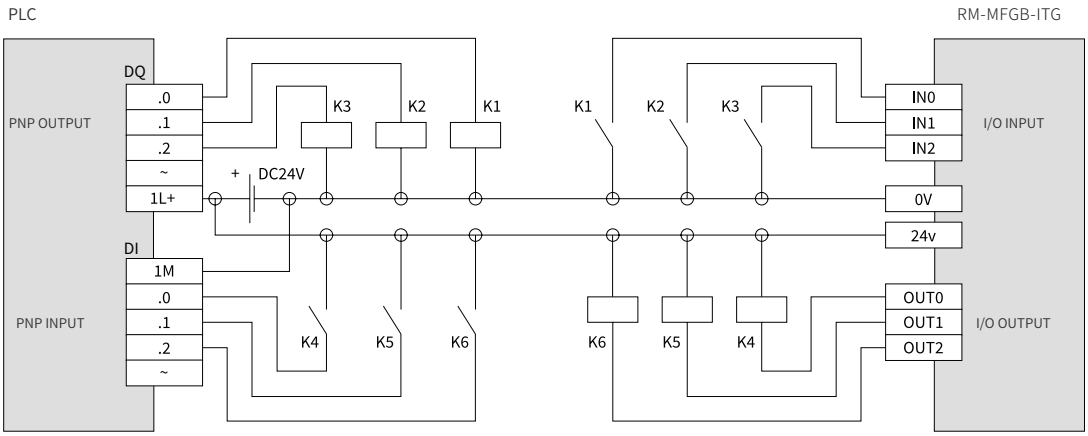


2.2.6 Circuit Diagram Wiring Illustration

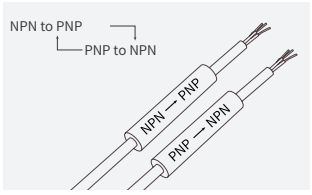
1. The RM-MFGB-ITG natively supports NPN. When the PLC I/O type is NPN, the wiring method is as follows:



2. When the PLC I/O type is PNP, indirect control can be achieved by using a relay wiring method as follows:



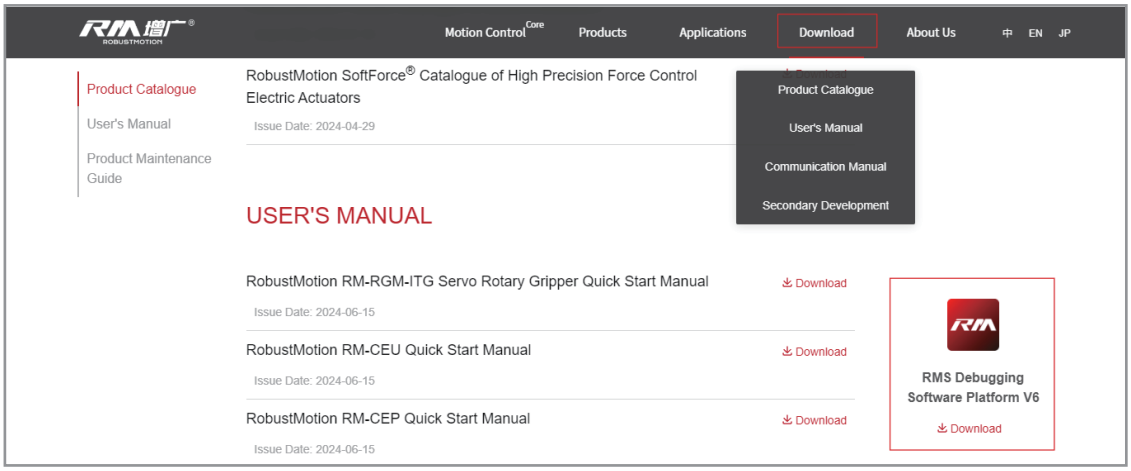
It is also possible to use a PNP to NPN converter or an NPN to PNP converter (as shown in the following figure) to achieve a high-to-low or low-to-high level conversion.



The PNP to NPN converter, or NPN to PNP converter, should be wired strictly following the wiring method provided by the cable manufacturer.

3 RMS Software Debugging Platform Usage

Please visit the official website of RobustMotion (www.rmaxis.com/en) and download the software from the Download page, or contact our after-sales engineer to obtain the RMS debugging software package. Through the RMS software debugging platform, users can set motion commands, modify parameters, and monitor control according to actual process requirements. The RMS software debugging platform has a simple, friendly, and feature-rich interface. For example, by simply setting point parameters, you can quickly complete the motion control settings of the actuator.

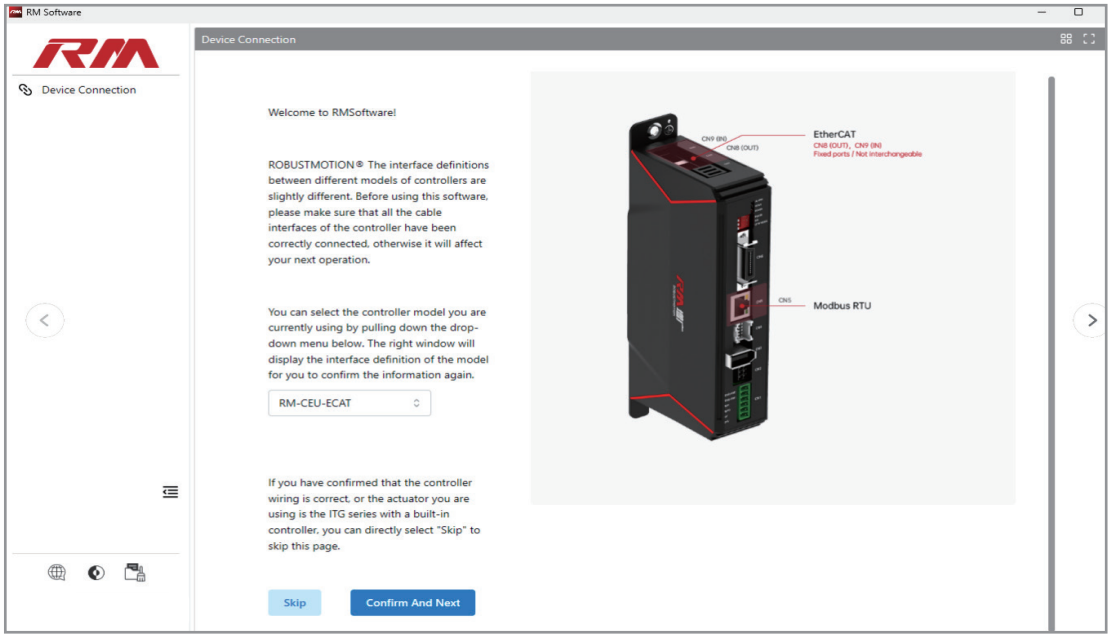


3.1 Software Operation

If you are unable to launch the software or encounter operational issues like unexpected crashes, this could be a result of incompatible computer specifications. For further assistance, please reach out to your sales representative or our technical support team.

3.2 Confirmation Interface of Controller Ports Wiring

This page serves as a reminder for users to verify the correctness of the controller's port wiring to prevent any impact on subsequent debugging processes. If you have confirmed that the wiring is correct, or if the actuator is equipped with an integrated ITG controller model, you may proceed by selecting [Skip].



3.3 Device Connection

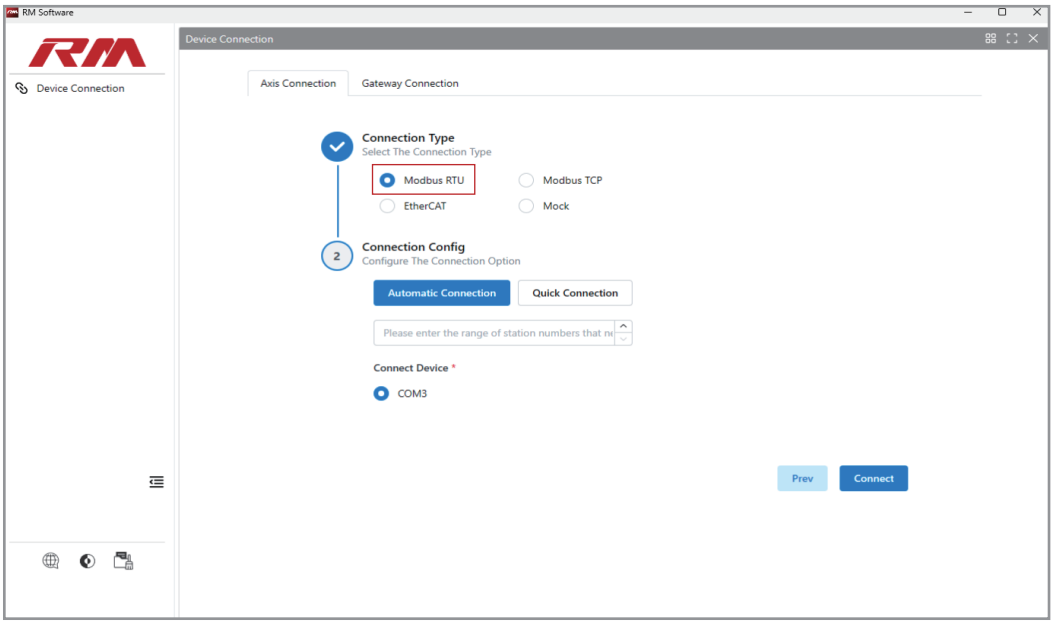
For electric actuator debugging, the Modbus RTU communication protocol is typically selected due to its straightforward mechanism for monitoring actuator movement and facilitating initial diagnostics. Ensure that the USB-to-485 adapter for debugging is properly connected to both the controller and the PC.

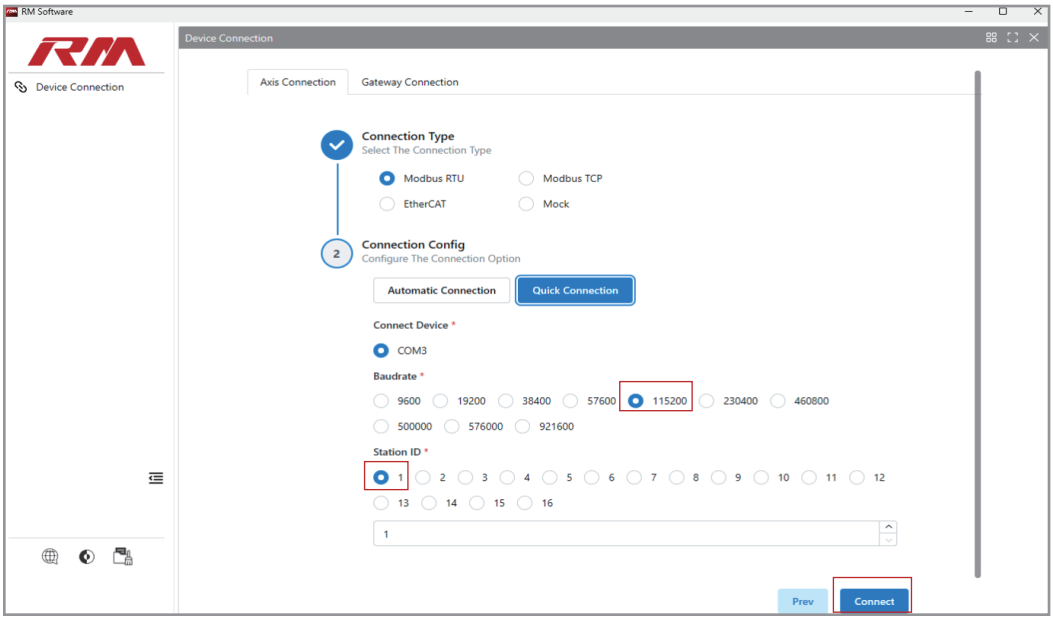
For integrated models, consult the [2.2 Integrated Connection Panel Wiring Instructions].

The specific connection methods are detailed as follows:

3.3.1 Modbus RTU Connection Type

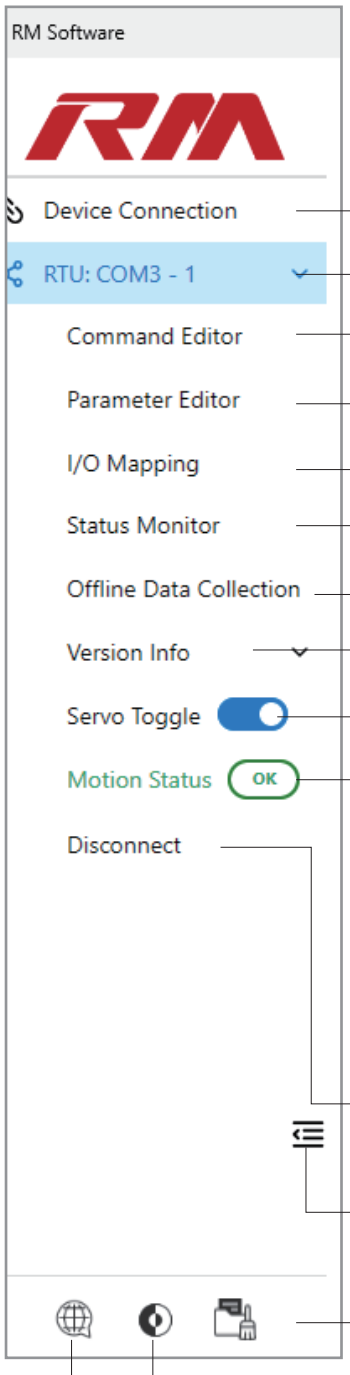
1. [Connection Type] , Select "Modbus RTU".
2. [Connection Config], choose the baudrate "115200" (factory default); station ID select "1" (factory default).
3. Click [Connect].





3.3.2 Overview of Main Interface Functions

The presence of the navigation bar on the left side of the interface, as depicted in the figure below, signifies that the software has established a successful connection with the actuator/controller. Upon each connection, the software automatically retrieves the current parameters from the controller.



[Device Connection] Click to connect to additional actuators/controller devices.

[Device Name] The connected device will be named according to its connection method and COM port, such as the connected device name "RTU:COM5-1" above, which indicates COM port: 5, and station ID: 1.

[Command Editor] You can set one or more motion commands by entering parameter values, control the movement of the electric actuator, and monitor the real-time data of the actuator. This is one of the most commonly used and main features of this debugging software version.

[Parameter Editor] Allows you to adjust some internal parameters of the controller, such as baudrate, station number, configuration of I/O input and output functions, pulse control numbers, etc.

[I/O Mapping] Allows you to view and configure the controller's I/O input and output functions.

[Status Monitor] Allows you to monitor the status signals of the electric actuator, including whether it has received the motion command signal correctly, whether it has correctly reflected the completion of the motion command signal, and whether it has received the external I/O signals normally.

[Offline Data Collection] Can collect data from the electric actuator in real-time and generate corresponding curves and Excel spreadsheets. It is commonly used to observe whether the actuator's movement is normal and to optimize control actions to help improve process quality.

[Version Info] Allows you to view the information of the currently connected actuator.

[Servo Toggle] Controls the state of the electric actuator, ON for being able to control the actuator's movement, OFF for the actuator being powered off.

[Motion Status] Normal operation is indicated by a green 'OK', while an abnormal condition triggers an alarm displaying red 'ERROR'. Click to view the alarm information, and please take a screenshot to provide to the after-sales engineer for quicker issue diagnosis and resolution."

[Disconnect] Once the debugging process is complete and the Commands have been successfully saved to the controller, click the [Disconnect] button to terminate the connection with the software.

[Show/Hide] The navigation bar can be expanded or collapsed.

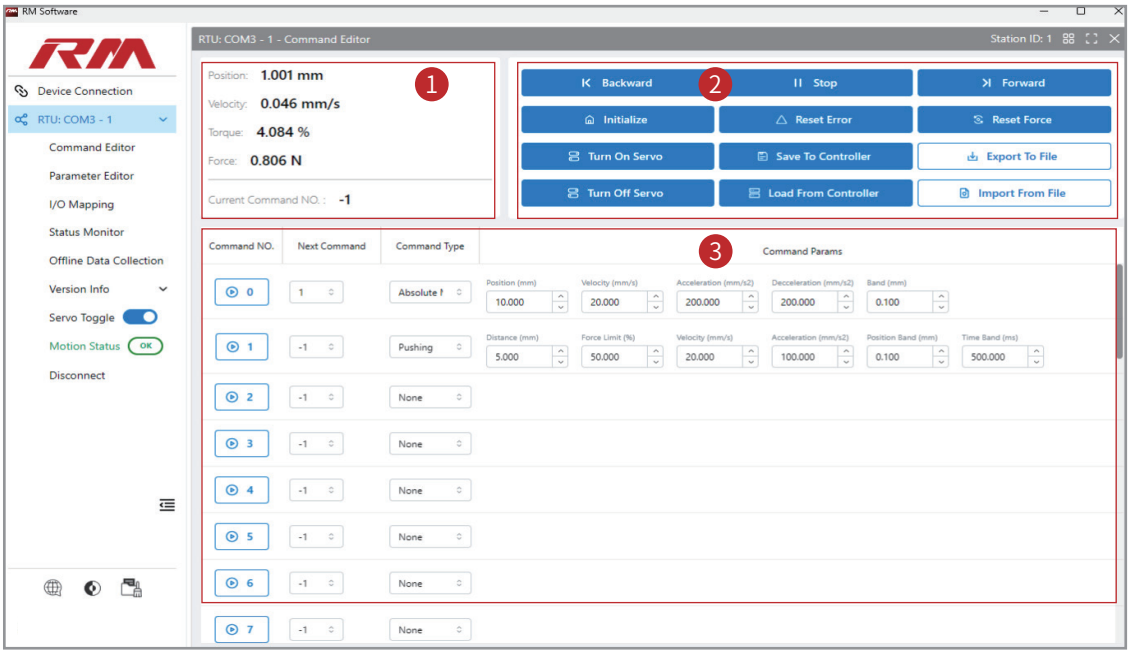
[Clear Cache] Clear the software parameter cache.

[Day/Night Mode] Click the button to switch between Day Mode and Night Mode.

[Chinese/English] Click the button to switch between Chinese and English.

3.4 Command Editor

Select 'Command Editor' from the navigation bar to access the interface below. This interface is the primary tool for actuator control, command configuration, and motion status display, and is one of the most frequently utilized features in the system.



3.4.1 Interface Feature Introduction

① Status Bar

The Status Bar provides real-time readouts and displays for the electric actuator's current position, velocity, torque, and force (for precision force control series only) of the electric actuator, as well as the Command number currently being executed. You can observe the data from this interface to determine whether the actuator's movement is normal and adjust the actuator's movement in a timely manner.

Status Items	Feature Introduction
[Position]	The current position of the actuator (mm).
[Velocity]	The current velocity of the actuator (mm/s).
[Torque]	The current torque of the actuator (standard actuator products display the percentage of output force (%)).※
[Force]	The applied force of the actuator (standard actuator products did not display the applied force, SoftForce® actuator products display the current applied force (N)).
[Current Command NO.]	The command number currently being executed by the actuator (corresponds to the [Command NO] in the Command Editor below, defaults to -1 in the stopped state).

When the actuator performs a pushing, the displayed force percentage (%) represents the actual allowable output force percentage (%) ; the actual allowable output force percentage (%) = safety factor × set output force percentage (%) .

② Command Bar

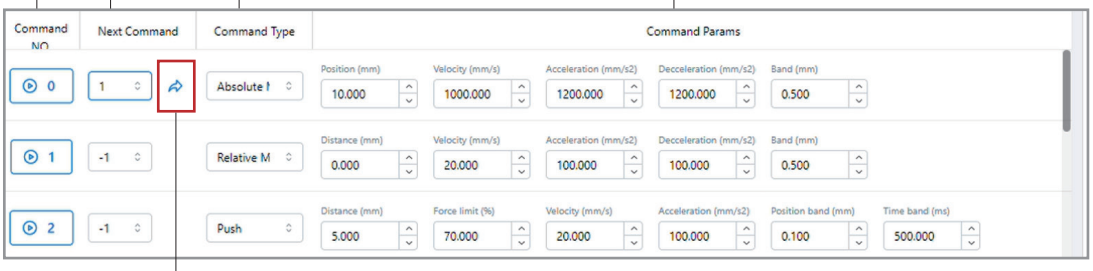
You can use this interface to control the movement of the electric actuator, including backward and forward, initialize, reset errors, and switching the servo on and off. You can also import point configurations from other controllers into the connected controller or export the current point configurations through this interface.

Command Items	Feature Introduction
[Backward/ Forward]	It is the JOG movement mode of the actuator, used when fine-tuning the position of the actuator is needed. [Backward] is for JOG-, [Forward] is for JOG+.
[Stop]	Used to stop the actuator's instructed movement.
[Initialize]	The Initialize action is a must-do operation after the actuator is powered on or restarted after power off. Click [Initialize] and wait for the Initialize to complete before performing other operations. When the "Current Command Number" in the Status Bar changes from a dynamic display to "-1", you can perform other operations.
[Reset Error]	It is used to clear the alarm when the actuator is alarmed. Click on the operation status in the left navigation bar to view the alarm information. Note that before clearing the alarm, you should first check the alarm information for troubleshooting by the after-sales technical engineer.
[Reset Force]	It is used to zero the force sensor of the actuator and is only applicable when debugging precision force control type electric actuators.
[Turn On/Off Servo]	It is used to open or close the electric actuator servo enable. It can also control the opening or closing of the servo through the Servo Toggle in the left navigation bar. When the switch is blue, the servo is in the open state, and vice versa when closed.
[Save To Controller]	Every time you create or modify Commands, you need to click [Save to Controller] to take effect; you can also click [Load from Controller] to verify if the current Commands have been saved to the controller.
[Load From Controller]	It can read Commands from the controller to verify if the current Commands have been saved.
[Export To File]	Save parameter files externally.
[Import To File]	Import parameter files from an external source.

③ Command Editor

Used to edit point commands, each point command represents an action, and the rising edge signal can trigger it. Control is simple, and you can modify parameter values for configuration as needed.

No.	Introduction
[Command NO]	The Command number recognized by the system. Click the "triangle symbol" to trigger the motion of the current Command.
[Next Command]	The user-defined execution order for the associated jump of the Command, that is, after the current number Command is completed, it automatically continues to execute the next Command with another number. For example, if you want the action to start with Command 0 and continue with Command 1, then fill in "1" in the "Next Step"; the default is "-1", which means it ends after executing the Command number.
[Command Type]	Set the Command type according to the actual process requirements. Common Command types and their uses are detailed in section [4.4.2 Detailed Explanation of Command Types].
[Command Params]	The parameters that can be set vary depending on the Command type and should be set according to the actual situation. For specific parameter definitions, please refer to section[4.4.2 Detailed Explanation of Command Types]. (Note: After changing the Command parameters, you need to click the command [Save Commands to Controller] to take effect).

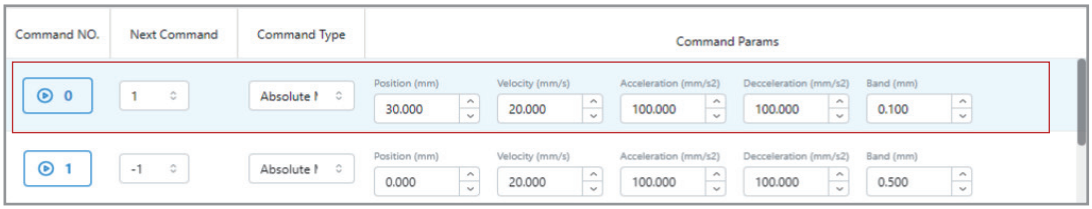


Clicking this button allows you to quickly navigate to the position of the "Next Step" Command number.

3.4.2 Detailed Explanation of Command Types

1. [Absolute Move] Command

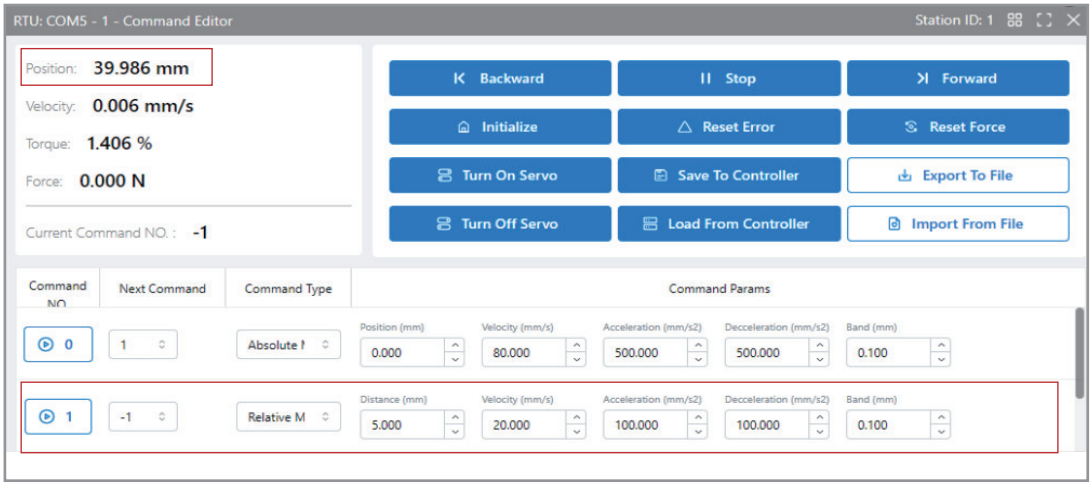
The Absolute Move Command is a motion command for the actuator to move to a set position using the origin as a reference point.



Command Parameters	Parameter Description
Position (mm)	The target position for "Absolute Move", set the value to be less than the "Maximum Stroke Value" of the corresponding product model parameter.
Velocity (mm/s)	The velocity at which to move to the target position. Set the effective value band to be less than the "Maximum Velocity Value" of the corresponding product model parameter.
Acceleration (mm/s^2)	The acceleration required to move to the target position. The default setting value is 500 mm/s^2.
Deceleration (mm/s^2)	The deceleration required to move to the target position. The default setting value is 500 mm/s^2.
Positioning Band (mm)	Used to set the band for the positioning signal. The default value is 0.1 mm. If the positioning band is set to ±0.1 mm, when the actuator reaches the target position and the actual position is within ±0.1 mm of the target position, the controller will generate a positioning completion signal for the current Command. For example, in "Command 0" on the diagram, the "Positioning Band" is set to 0.1mm, and the "Position" is set to 30mm. When the actuator moves within the absolute position band of 29.9-30.1mm, the controller will output the completion signal for "Command 0." Note: The "Positioning Band" is only used to set the band for issuing the positioning signal and does not affect the final set position that the actuator moves to.

2. [Relative Move] Command

The Relative Move Command is a motion command for the actuator to move to a set position using the current position as a reference point.



Command Parameters	Parameter Description
Distance (mm)	The distance that needs to be moved relative to the current position.
Velocity (mm/s)	The velocity at which to move to the target distance, with the set value band being less than the "Maximum Velocity Value" of the corresponding product model parameter.
Acceleration (mm/s^2)	The acceleration required to move to the target distance, with the default setting value being 500 mm/^{2}.
Deceleration (mm/s^2)	The deceleration required to move to the target distance, with the default setting value being 500 mm/s^2.
Positioning Band (mm)	Used to set the band for the positioning signal, with the default value being 0.1 mm. If the positioning band is set to ±0.1 mm, when the actuator reaches the target position and the actual position is within ±0.1 mm of the target position, the controller will generate a positioning completion signal for the current Command. For example, in "Command 1" on the diagram, the actuator's current position is "2mm", the "Positioning Band" is set to 0.1mm, and the "Distance" is set to 5mm. Therefore, when the actuator moves to the actual position within the band of 6.9-7.1mm, the controller will output the completion signal for "Command 1". Note: The positioning band is only used to set the band for issuing the positioning signal and does not affect the final set position that the actuator moves to.

3. [Push] Command

The Push Command refers to starting from the current position, setting a movement at a rated output (current percentage) for a certain distance until the force reaches the set value and then maintaining it.

- For electric grippers, this is an important Command to achieve adaptive gripping. By setting the Absolute Move + Push Commands, the "quick approach and flexible gripping" action can be realized.

Command Parameters	Parameter Description
Distance (mm)	The distance that needs to be moved relative to the current position. The set value should be greater than the actual distance from the target position to the current position. When the set value is greater than the maximum stroke value of the corresponding actuator model, the actuator can achieve full-stroke "Push".
Force limit (%)	The "Push" at the set output percentage (current percentage).
Velocity (mm/s)	The velocity at which to move to the target distance. The set value band is less than the "Maximum Speed Value" of the corresponding product model parameter. The recommended value is 20 mm/s.
Acceleration (mm/s^2)	The acceleration required to move to the target distance, with the default setting value being 100 mm/s.
Position Band (mm)	Used to set the band for the positioning signal, with the default value being 0.1 mm. If the positioning band is set to ±0.1 mm, when the actuator reaches the target position and the actual position is within ±0.1 mm of the target position, the controller will generate a positioning completion signal for the current instruction. For example, in "Command 1" on the diagram, the "Position Band" is set to "0.1mm", and the "Distance" is set to "10mm". Therefore, when the actuator moves to 9.9mm, it outputs the "Command 1" arrival signal. Note: The position band is solely used to define the scope for issuing the arrival signal and does not affect the final set destination of the actuator's movement.
Time Band (ms)	It determines the time band value for the force to be stably in place. In the diagram for command 1, the time band is set to 500ms with an output force of 50%. Once the actuator's output force reaches 50% and is maintained for 500ms, it is judged to be properly positioned in terms of force, and the arrival signal for command 1 is output simultaneously.

3.5 Command Editing Examples

3.5.1 Rapid Positioning (Commonly used for grippers to quickly position to the open or closed position)

1. **Example One: [Absolute Move]**

For instance, adjustments are needed for the RM-MFGB-08-80-1-ITG gripper (opening and closing stroke: 80mm on both sides, 40mm on one side). Currently, the electric gripper is at the 0mm position (i.e., the gripper fingers are at the outermost position), as shown in Figure 1; execute the [Absolute Move] Command to bring both sides of the fingers as close as possible, that is, the gripper needs to move to the 60mm position, as shown in Figure 2. The specific Command setting steps are as follows:

Firstly, determine the distance for [Absolute Move]. Since the gripper fingers need to move to 60mm, the "Position" value for [Absolute Move] is set to "60mm"; "Velocity" is set to the gripper's fastest velocity "80mm/s"; "Acceleration/Deceleration" is set to the recommended value "500mm/s²"; "Band" is set to the recommended value "0.1mm". After completing the Command settings, click [Save Params To Controller] and the gripper state after running the Command is shown in Figure 2.

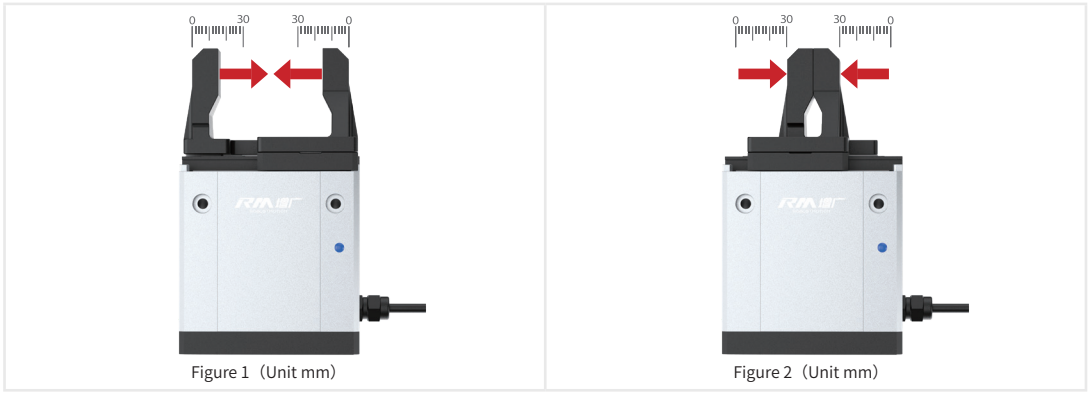


Figure 1（Unit mm）

Figure 2（Unit mm）

Conversely, when the electric gripper is at the 60mm position (i.e., the gripper fingers are at the innermost), as shown in Figure 3; execute the [Absolute Move] Command to separate both sides of the fingers as much as possible, i.e., the gripper needs to move to the lower limit position of 0mm, as shown in Figure 4. The specific Command setting steps are as follows:

First, determine the distance for [Absolute Move]. Since the lower limit position of the gripper fingers is 0mm, the "Position" value for [Absolute Move] is set to "0mm"; "Velocity" is set to the gripper's fastest velocity "80mm/s"; "Acceleration/Deceleration" is set to the recommended value "500mm/s²"; "Band" is set to the recommended value "0.1mm". After completing the Command settings, click [Save Command to Controller], and the gripper state after running the Command is shown in Figure 4.

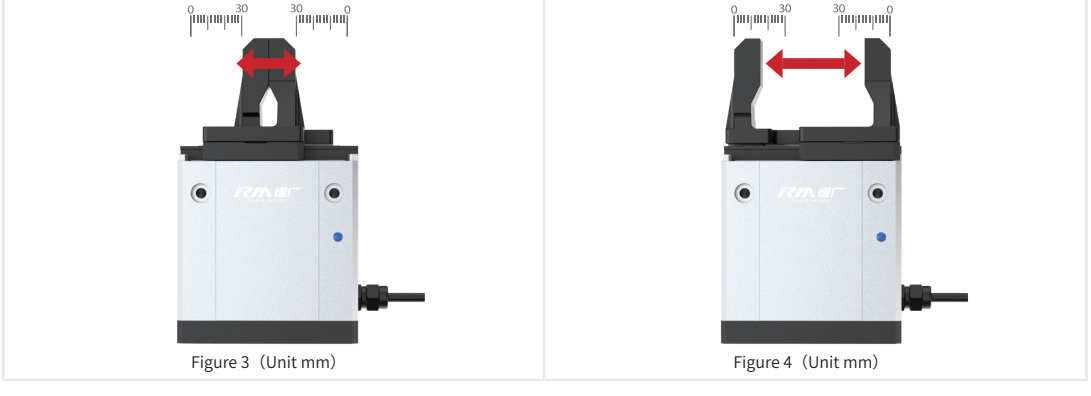


Figure 3（Unit mm）

Figure 4（Unit mm）

2. Example Two: [Relative Move]

For instance, adjustments are needed for the RM-MFGB-08-80-1-ITG gripper. Currently, the electric gripper is at the 0mm position (i.e., the gripper fingers are at the outermost), as shown in Figure 5; execute the [Relative Move] Command to bring both sides of the fingers as close as possible, i.e., the gripper needs to move to the 60mm position, as shown in Figure 6. The specific Command setting steps are as follows:

First, determine the distance for [Relative Move]. Since the current position of the gripper fingers is 0mm and the target position is 60mm, the displacement increases by 60mm (60mm-0mm=60mm), so the "Distance" value for [Relative Move] is set to "60mm"; "Velocity" is set to the gripper's fastest velocity "80mm/s"; "Acceleration/Deceleration" is set to the recommended value "500mm/s²"; "Band" is set to the recommended value "0.1mm". After completing the Command settings, click [Save Params To Controller] and the gripper state after running the Command is shown in Figure 6.

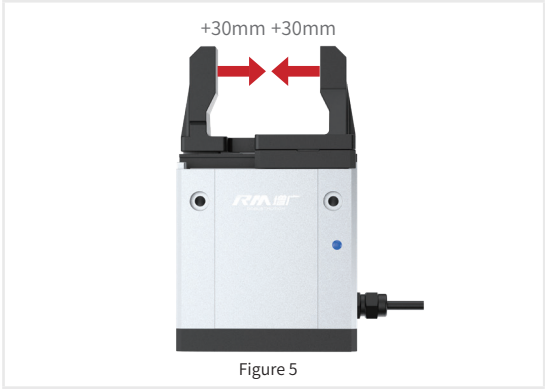


Figure 5

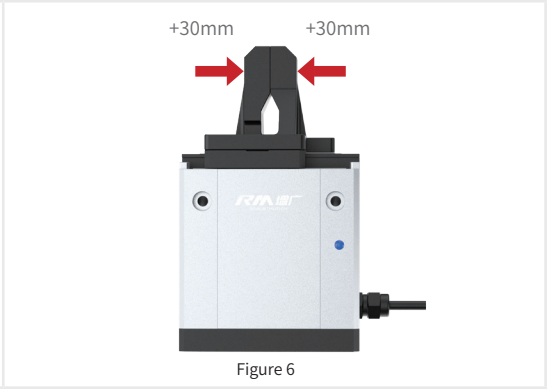
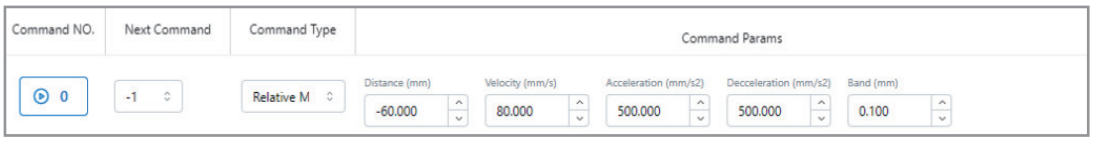


Figure 6

Conversely, when the electric gripper is at the 60mm position (i.e., the gripper fingers are at the innermost), as shown in Figure 7; execute the [Relative Move] Command to separate both sides of the fingers as much as possible, i.e., the gripper needs to move to the lower limit position of 0mm, as shown in Figure 8. The specific Command setting steps are as follows:

First, determine the distance for [Relative Move]. Since the current position of the gripper fingers is 60mm and the target position is 0mm, the displacement decreases by 60mm (0mm-60mm=-60mm); therefore, the "Distance" value for [Relative Move] is set to "-60mm"; "Velocity" is set to the gripper's fastest velocity "80mm/s"; "Acceleration/Deceleration" is set to the recommended value "500mm/s²"; "Band" is set to the recommended value "0.1mm". After completing the Command settings, click [Save Params To Controller] and the gripper state after running the Command is shown in Figure 8.



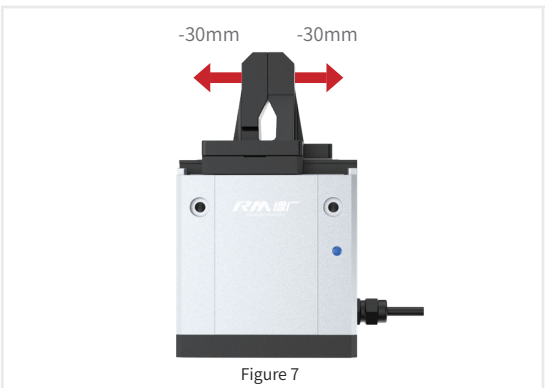


Figure 7

Figure 8

3.5.2 Rapid Flexible Pushing and Pressing

(Commonly used for grippers to quickly and flexibly grip or support workpieces)

> The electric gripper's motion cannot be set with only a single [Absolute Move]/[Relative Move] Command to grip the workpiece; otherwise, an alarm will occur.

1. **Example One: [Absolute Move] + [Push]**

For instance, use the RM-MFGB-08-80-1-ITG (opening and closing stroke: 80mm on both sides, 40mm on one side) to grip and inwardly grasp a workpiece with an outer diameter of about 20mm. Currently, the gripper is at position 0mm, and the distance between one side of the gripper finger and the workpiece is about 7mm, as shown in Figure 9.

Figure 9

Figure 10

Figure 11

The specific operation steps are as follows:

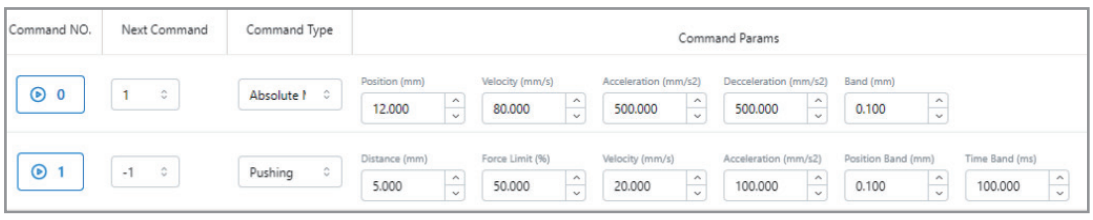
① Set the [Absolute Move] Command

First, determine the distance for [Absolute Move]. Since the distance between one gripper finger and the workpiece is 7mm, it is necessary to bring the fingers as close to the workpiece as possible. Therefore, the movement distance for both fingers should be less than and close to 14mm, so the "Position" value is set to "12mm" (0mm+12mm=12mm); "Velocity" is set to the gripper's fastest velocity "80mm/s"; "Acceleration/Deceleration" is set to the recommended value "500mm/s²"; "Band" is set to the recommended value "0.1mm". After completing the Command settings, click [Save Params To Controller] to complete the rapid approach action set by the [Absolute Move] Command. The gripper state after running this Command is shown in Figure 10.

② Set the [Push] Command

Now, the distance between one side of the gripper finger and the workpiece is about 1mm. At this time, set the next Command [Push]. The "Distance" value for the gripper to push must be greater than the total distance of 2mm between the fingers and the workpiece after the rapid approach. So, it is recommended to add an extra 3mm to this "Distance" value (inward pushing is a positive value) to ensure that even if there are slight changes in the size or position of the workpiece, the fingers can still properly push onto the workpiece. Therefore, the "Distance" value for the Pushing is set to "5mm" (2mm+3mm=5mm); "Force Limit" is set to "50%" of the maximum output; "Velocity" is set to the recommended velocity "20mm/s"; "Acceleration" is set to the recommended value "100mm/s²"; "Position Band" is set to the recommended value "0.1mm"; "Time Band" is set to the recommended value "100ms". After completing the Command settings, click [Save Params To Controller] to complete the flexible gripping action set by the [Push] Command. The gripper state after running this Command is shown in Figure 11.

If you need the gripper to automatically perform [Push] after completing [Absolute Move], you can set the [Next Command] parameter value of [Absolute Move] to the sequence number where [Push] is located. After completing the Command settings, click [Save Params To Controller] to complete the consecutive motion of the two Commands. The final complete Command is shown in the following figure.



2. Example Two: [Absolute Move] + Reverse [Push]

For instance, using the RM-MFGB-08-80-1-ITG (opening and closing stroke: 80mm on both sides, 40mm on one side) to externally support and grasp a workpiece with an inner diameter of approximately 25mm. The current gripper is at the 40mm position, and the distance between one side of the gripper finger and the workpiece is about 12mm, as shown in Figure 12.

Figure 12

Figure 13

Figure 14

The specific operation steps are as follows:

① Set Motion [Absolute Move] Command

First, determine the distance for [Absolute Move]. Since the distance between one finger of the gripper and the workpiece is 12mm, it is necessary to bring the finger as close to the workpiece as possible. Therefore, the distance that both fingers of the gripper should move outward should be less than and close to 24mm. Here, take the close value of 22mm, so the "Position" parameter for [Absolute Move] is set to "18mm" (40mm-22mm=18mm); "Velocity" is set to the gripper's fastest velocity "80mm/s"; "Acceleration/Deceleration" is set to the recommended value "500mm/s²"; "Band" is set to the recommended value "0.1mm". After completing the Command settings, click [Save Params To Controller], and you can complete the rapid approach action set by the [Absolute Move] Command. The gripper status after running this Command is shown in Figure 13.

② Set Motion [Push] Command

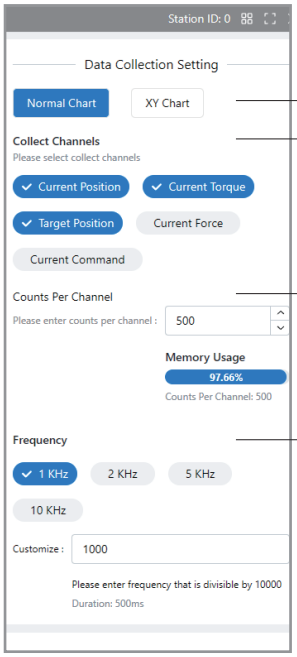
Now, the distance between one side of the gripper finger and the workpiece is about 1mm. At this time, set the next Command [Push]. The "Distance" value for the gripper to push must be greater than the total distance of 2mm between the finger and the workpiece after the rapid approach. Therefore, it is recommended to additionally extend 3mm to this "Distance" value (outward extension is a negative value) to ensure that even if there are slight changes in the size or position of the workpiece, the finger can still normally push onto the workpiece. Thus, the "Distance" value for the Pushing is set to "-5mm" (-2mm-3mm=-5mm); "Force Limit" is set to "50%" of the gripper's maximum output; "Velocity" is set to the recommended velocity "20mm/s"; "Acceleration" is set to the recommended value "100mm/s²"; "Position Band" is set to the recommended value "0.1mm"; "Time Band" is set to the recommended value "100ms". After completing the Command settings, click [Save Params To Controller] to complete the flexible gripping action set by the [Push] Command. The gripper status after running this Command is shown in Figure 14.

If you need the gripper to automatically perform [Push] after completing [Absolute Move], you can set the [Next Step] parameter value of [Absolute Move] to the sequence number where [Push] is located. After completing the Command settings, click [Save Params To Controller] to complete the consecutive motion of the two Commands. The final complete Command is shown in the following figure.

3.6 Offline Data Collection Interface

The offline data collection interface can collect real-time data such as current position, current output, homing position, and current force, and generate a line graph of data and time. It also allows for the export of data to Excel for analysis.

1. Data Collection Settings



Use the [Normal Chat] for data collection settings.

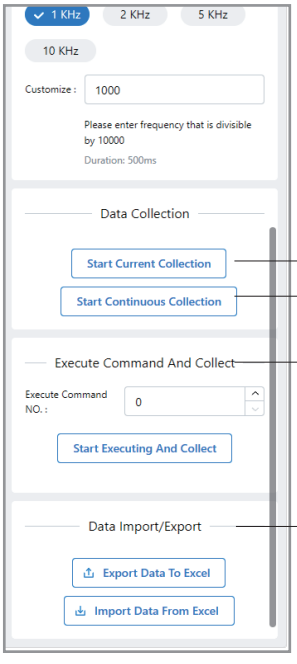
[Collect Channels]
You can select multiple collection channels (collect variables) at the same time. The most common use is when using a precision force control actuator, you can check both the [Current Position] , [Current Force] and [Target Position] items to display the force-position curve.

[Counts Per Channel]
The default collection quantity is 500, but it can also be set to 1000. The controller memory will automatically allocate the collection quantity based on the number of selected variables.

[Frequency]
The default collection frequency is 1kHz, but you can choose a higher frequency or customize it. The higher the frequency you choose, the shorter the collection time.

2. Data Collection Commands

Data collection commands allow for [Start Current Collect], [Start Persistence Collect] and collection targeting a specific Command.



[Start Current Collection]
Refers to the data collection for the current motion.

[Start Continuous Collection]
Indicates continuous data collection during the actuator's motion, collecting a segment of data each time: clicking again will cancel continuous collection.

[Execute Command and Collect]
The most commonly used collection command is after setting an action Command in the "Command Editor." If you need to observe the actuator's performance curve when executing the Command, you can fill in the "Execute Command NO." and then click "Start Execution and Collect", which will collect real-time data curves during the execution of the corresponding action.

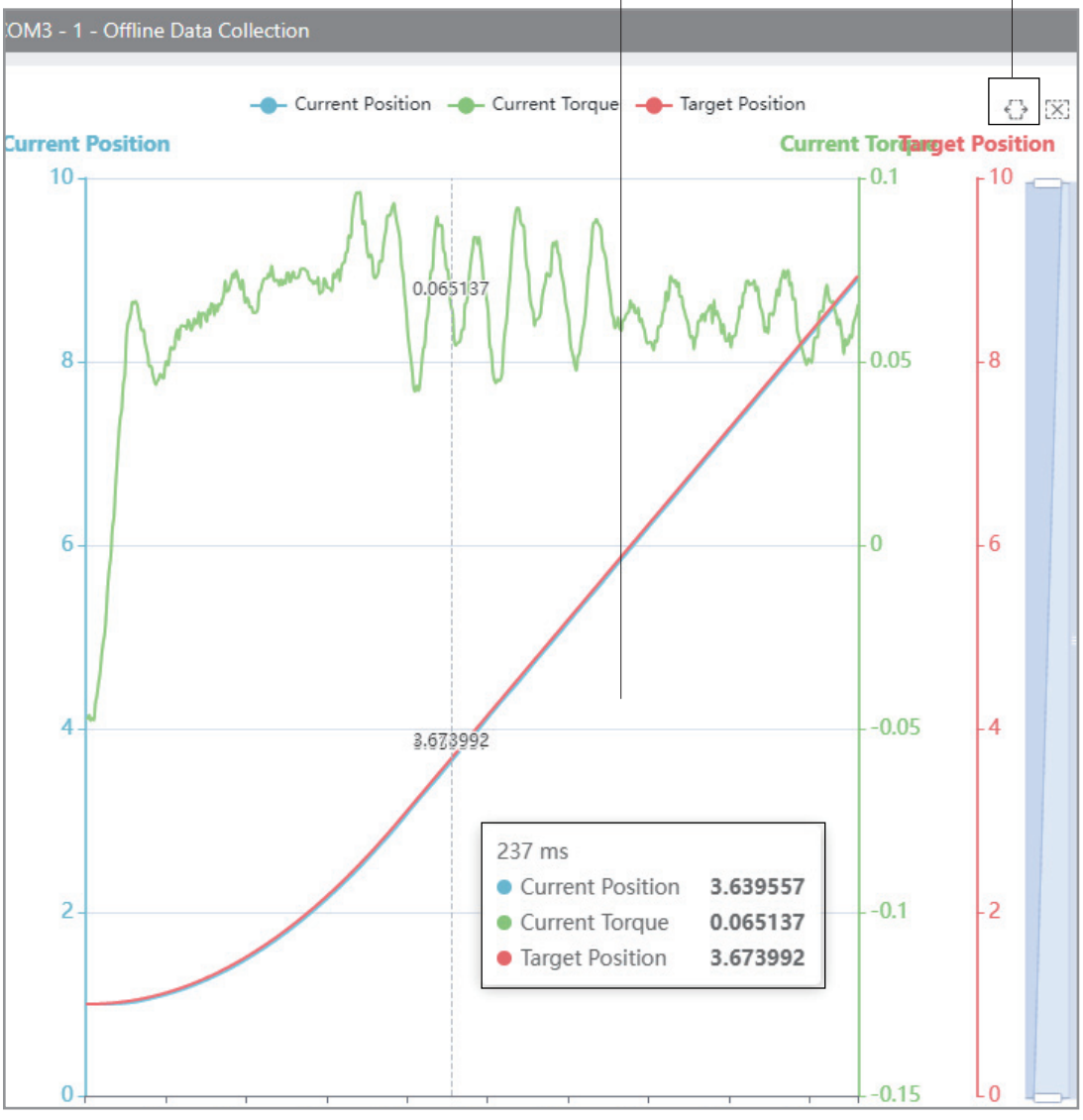
[Data Import/Export]
You can import/export data from the left curve into/out of Excel for data comparison analysis, etc.

3. Curve Data

Data collection commands allow for direct [Start current collect], [Start persistence collect], and also collection targeting a specific Command.

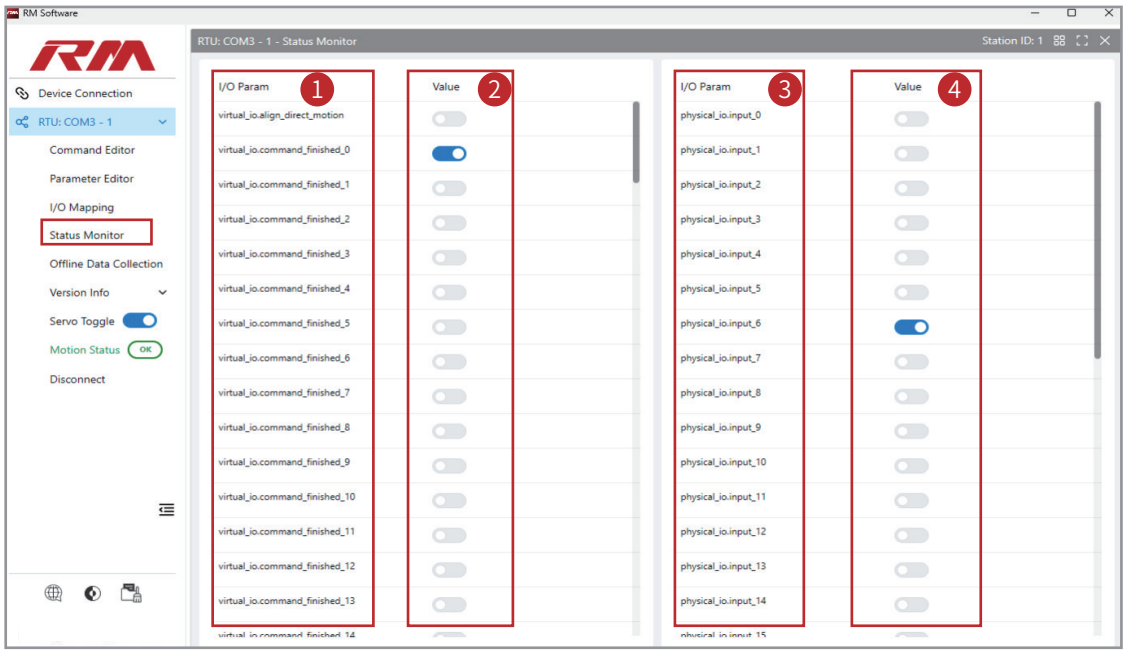
By moving the mouse, you can view the changes and values of the current position and the current force experienced.

Select the icon, and after clicking, you can choose a period on the curve for more specific data analysis. Click another icon to cancel the selection.



3.7 Status Monitor Interface

You can observe the current motor actuator's action execution status (Boolean quantity) and the input/output status of external I/O in the [Status Monitor] interface.



3.7.1 Left Side Status Bar

The left side Status Bar shows the current action execution status of the motor, with ① as the status parameter name, and ② as the current status.

Command completion signal status description:

1. When "Command Editor" sets the position Command 0 as [Absolute Move].
   - This signal will be turned ON after the actuator completes the action Command and the current position is within the positioning band of the target position.

2. When "Command Editor" sets the position Command 0 as [Push] / [Precision Push].
   - When the actuator completes the motion and the current position is within the target position's band, this signal will be set to ON, and simultaneously, the "Position Reached" signal in the status monitoring will also be set to ON; users can determine from these two signals whether the current action is an empty grip/push.
   - When the actuator completes the motion, the motor's output reaches the set output value, and the current position is not within the target position's band, this signal will be set to ON, and at the same time, the "Position Reached" signal in the status monitoring will be set to OFF; users can determine from these two signals whether the current action is gripping/pressing onto the workpiece.

3.7.2 Right Side Status Bar

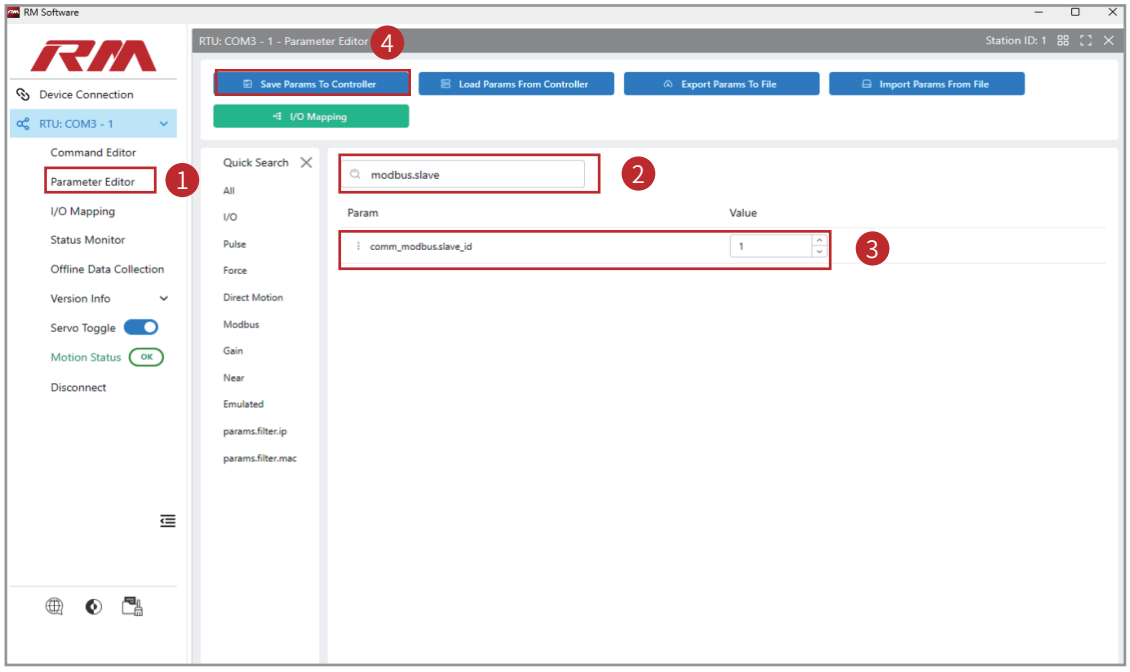
The right side Status Bar is for external I/O status, with ③ as the external status parameter name, and ④ as the current external status.

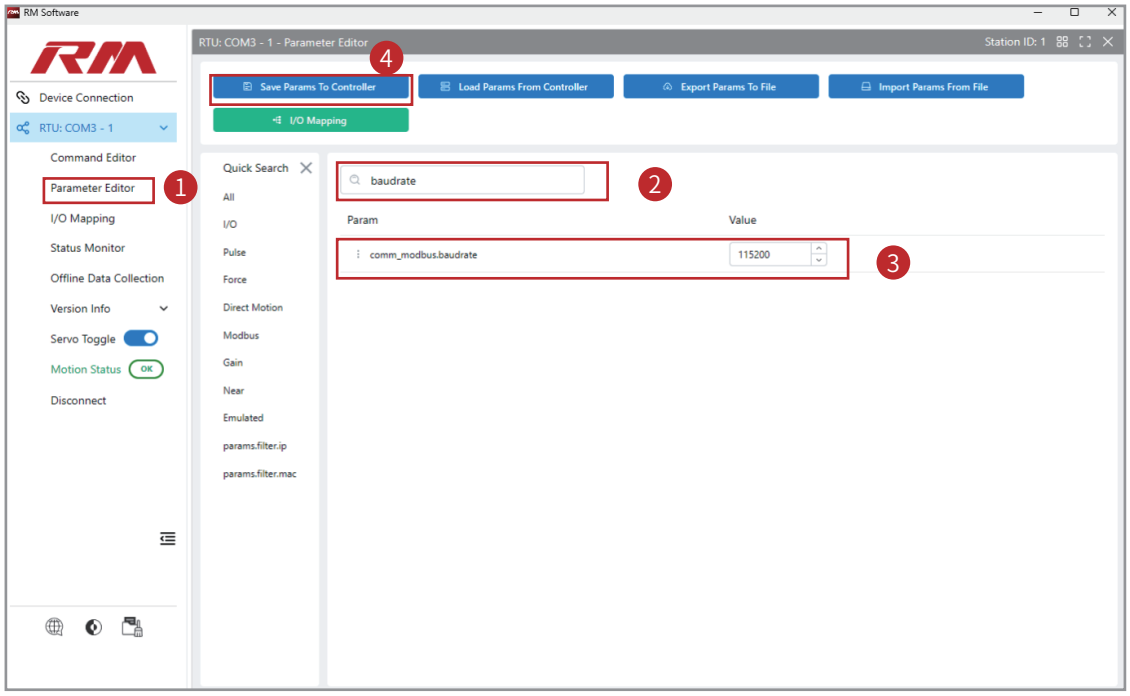
- When using I/O control, you can observe whether there is an external I/O input signal or whether the I/O signal is normally given through status monitoring, which can help troubleshoot problems that occur during I/O control.
- When an external input signal IN0 is received, the external I/O input 0 will be set to ON. When the configured I/O output OUT0 is mapped to an ON state, the external I/O output 0 will be set to ON.

3.8 Parameter Editor Interface

3.8.1 Change Station Number & Baudrate

Firstly establish a connection with the controller via the Modbus RTU. Upon successful connection, access the [Parameter Editor] interface. Within the Parameter Editor, navigate to the "modbus.slave" setting to modify the controller's station address, ensuring it falls within the permissible range of 1 to 255. Subsequently, locate and adjust the "baudrate" parameter to a preferred value, commonly selected from standard rates such as 9600, 19200, 38400, 57600, or 115200. Once the desired settings are applied, proceed to click [Save Params To Controller]. The updated parameters will be effective upon the subsequent power-up of the actuator/controller, as illustrated in the accompanying diagram.

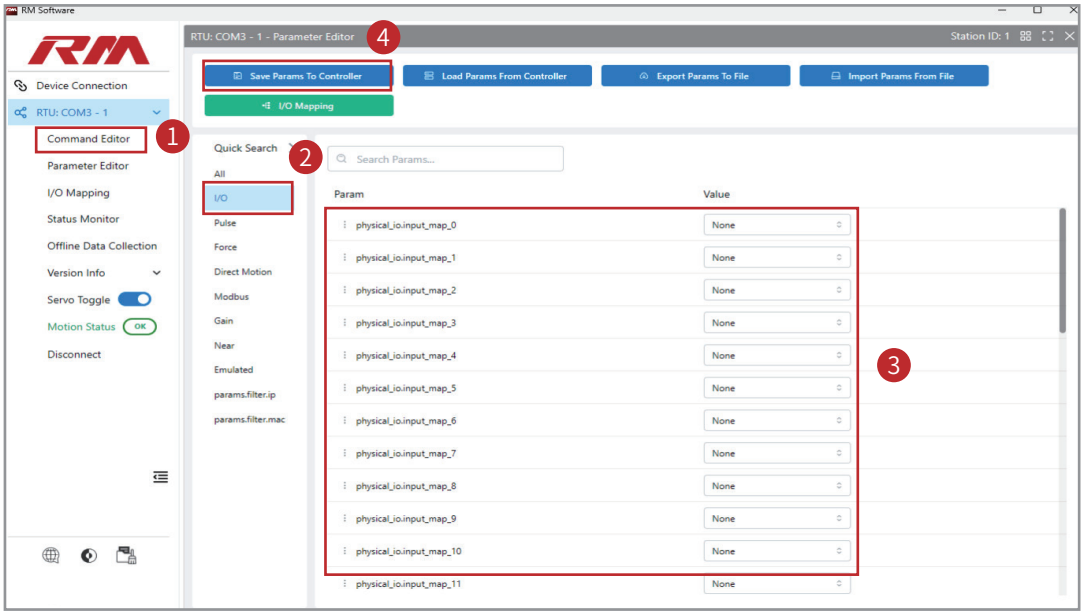




3.8.2 External I/O Input and Output Configuration

When using I/O control, if you need to configure external I/O mapping within the controller, first connect the software using Modbus RTU or other methods. In [Command Editor], search for "I/O" to find [physical_io.input_map_0] and [physical_io.output_map_0]. [physical_io.input_map_0] corresponds to IN0 in the actual I/O wiring of the actuator, and [physical_io.output_map_0] corresponds to OUT0 in the actual I/O wiring of the actuator. You can configure the corresponding input and output signals of I/O mapping according to actual needs.

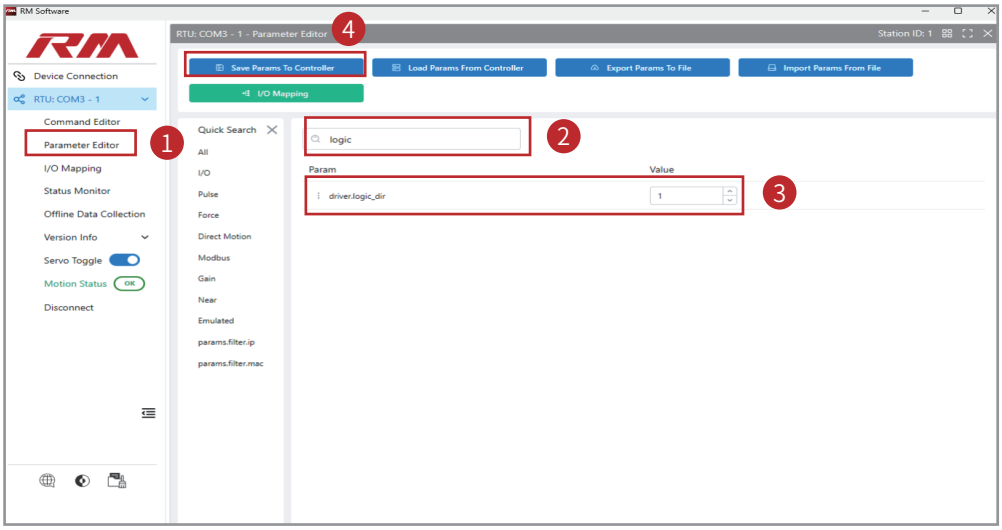
For example: If a user needs to use the external I/O input mapping 0 (corresponding to the actuator I/O port IN0) to trigger the point Command 0 in [Command Editor], they only need to set the parameter of "physical_io.input_map_0" to "virtual_io.command_start_0". After completing the change, click [Save Params To Controller]. The actuator/controller will take effect after being powered on again.



3.8.3 Home Position Direction Reversal

If you need to change the direction of the home position, first connect the software using Modbus RTU or other methods. In [Parameter Editor] , search for "logic" and find [driver.logic_dir].

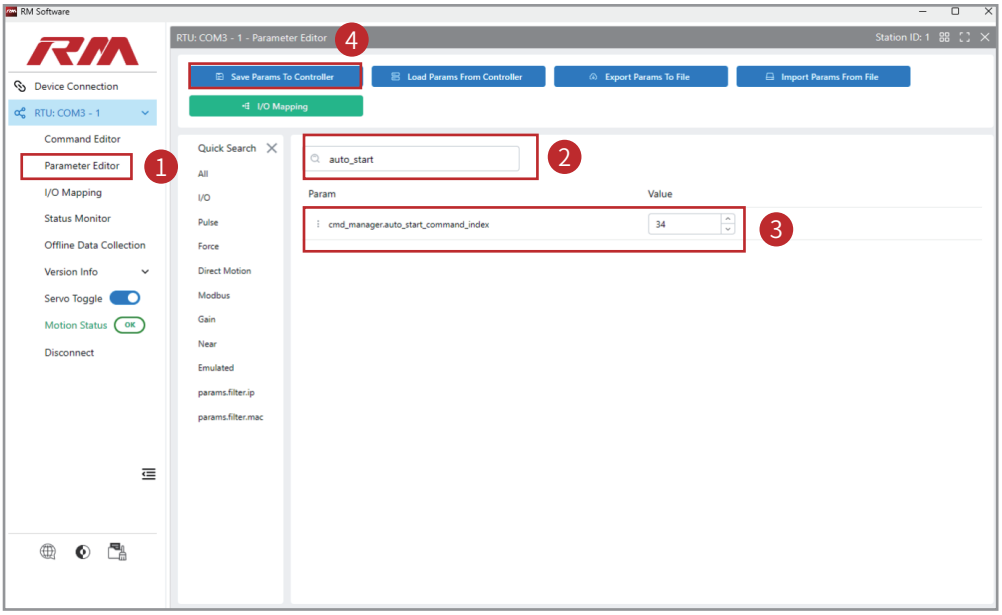
The valid values for [driver.logic_dir] are "1" and "-1". If the current default value is "1," change the value to "-1" to reverse the home position direction. Conversely, if the current default value is "-1," change the value to "1" to reverse the home position direction. After completing the change, click [Save Params To Controller]. The actuator/controller will take effect after being powered on again.



3.8.4 Power-Up Home Position Setting

After the actuator performs the "Push" (fingers open), do not use the "Initialize" command to open it. Instead, set an "Absolute Move" to "0mm" to achieve "returning to the origin" or move to the desired position.

The actuator is set to automatically return to the home position by default before leaving the factory, and manual operation is generally not required. If the electric actuator needs to "enable" or "cancel" the automatic return to the home position upon power-up, first connect the software using Modbus RTU or other methods. In [Parameter Editor], search for "auto" and find [cmd_manager.auto_start_command_index]. When the parameter [cmd_manager.auto_start_command_index] is set to "34," the actuator enables the automatic execution of the home position action upon power-up; when this parameter is set to "-1," the actuator cancels the automatic execution of the home position action upon power-up. After completing the change, click [Save Params To Controller]. The actuator/controller will take effect after being powered on again.

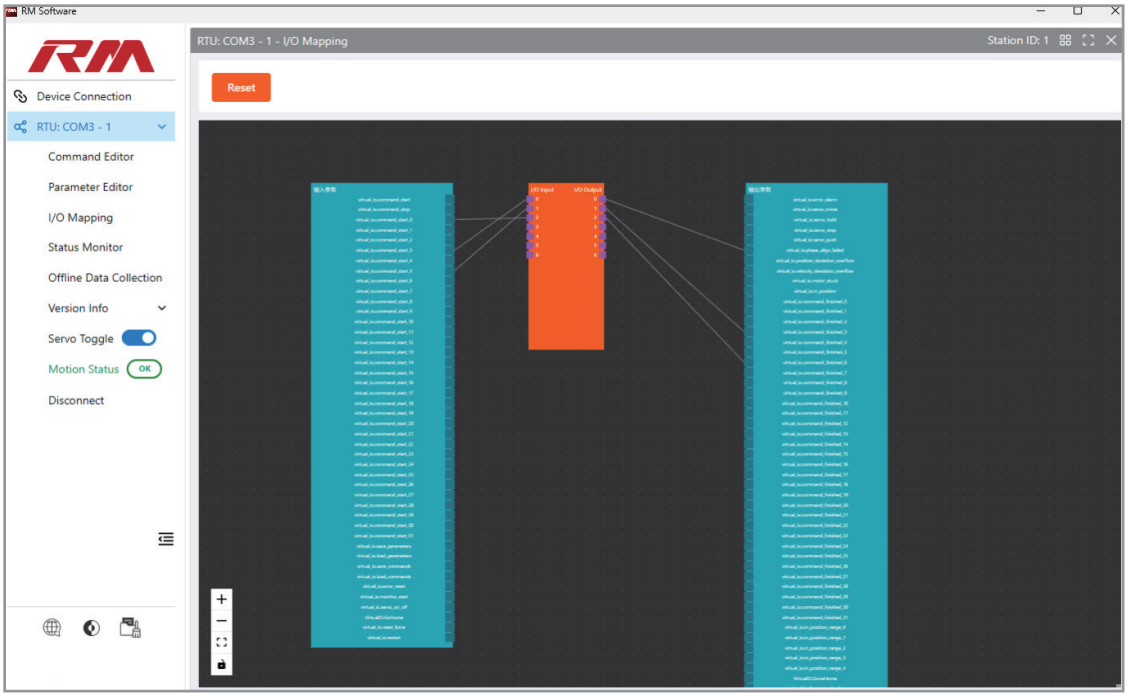


3.9 I/O Mapping Interface

[I/O Mapping] is another convenient method for configuring I/O input and output. Click on [I/O Mapping] in the left navigation bar, and by connecting the input and output parameters on the left and right with the I/O interface in the center, you can associate the I/O interface with the required functions to achieve I/O customization.

Select the connection and press the DELETE key on the keyboard to delete the connection.

This function is consistent with the function of "[4.8.4 External I/O Input and Output Configuration]". The related configuration changes will be synchronized and updated.



4 Modbus RTU Communication Guide

When using Modbus RTU communication, the corresponding function codes and addresses are required to control the motion of the electric gripper and to modify the parameters of the electric gripper.
Note: The Modbus addresses used in the examples are in decimal format.

4.1 Function Code Address Explanation

4.1.1 02H Function Code

The 02H function code is utilized for reading input statuses, specifically the states of digital quantity inputs (DIs). It allows for the retrieval of current statuses of the electric actuator, such as the error alarm signal (address: 0) and the Initialize completion signal (address: 1037). Additionally, it can read the completion signals of user-defined positions (addresses: 1000-1015), which can be employed for making automated logical judgments.

The Initialize completion signal is a constant ON signal, which remains ON after the initial power-up and Initialize. If you need to change the signal type, please contact our company's engineers for modification.

Name	Function Code	Address (Decimal)	Number of Registers	Data Type	Function
Error Alarm	02H	0	1	bool	Read the alarm signal.
Position Deviation Alarm		1	1		Read the position deviation alarm signal.
Velocity Deviation Alarm		2	1		Read the velocity deviation alarm signal.
Motor Stall		3	1		Read the motor stall alarm signal.
Target Position Reached		8	1		Read the target position reached signal.
Arrived Signal 0		1000	1		Controller has arrived at position 0.
Arrived Signal n		1000+n	1		Controller has arrived at position n.
Arrived Signal 15		1015	1		Controller has arrived at position 15.
Initialize Complete		1037	1		Initialize complete signal (home position has been returned to).

4.1.2 03H / 10H Function Code

The 03H function code is for reading holding registers, used to read one or more 16-bit values from the holding registers of the slave device.
The 10H function code is for presetting multiple registers, used to write desired values into the registers of the slave device.
You can use the 03H function code to read the current torque (address: 2154), which occupies two registers.

Name	Function Code	Address (Decimal)	Number of Registers	Data Type	Function
Current Torque %	03H	2154	2	real	Read the current torque.

03H Function Code:
The 03H function code can be used to read the actual values (values from the controller's internal registers) of position (address: 2284), velocity (address: 2286), acceleration (address: 2288), torque (address: 2290), and torque switch (address: 2282) in the positioning mode. This can be used to compare whether they are consistent with the values written by the host computer. The position, velocity, acceleration, and torque each occupy two registers, while the torque mode switch occupies one register.

10H Function Code:
The 10H function code can be used to write values to the addresses corresponding to the positioning mode in the following sequence: 1) Set the torque, 2) Set the target acceleration, 3) Set the target velocity, 4) Set the target position. After setting the target position, the system can directly move to the corresponding location according to the configured values.

1. To use the electric gripper to hold a workpiece, the torque mode switch must be turned on; only then can gripping be performed, otherwise the electric gripper will generate an error. The point mode and positioning mode are two distinct modes and do not interfere with each other. After initiating a point action, the current position must be assigned to the position register; otherwise, the value in the position register will not change automatically. If the same position is set again, the system will default to the position as unchanged, and the electric gripper will not trigger action.
2. The Command type and the next-step command modbus address data type is a double integer.

Positioning Mode					
Name	Function Code	Address (Decimal)	Number of Registers	Data Type	Function
Set Target Position	Read 03H/ Write 10H	2284	2	real	Set the target position. (mm)
Set Target Velocity		2286	2	real	Set the target velocity. (mm/s)
Set Target Acceleration		2288	2	real	Set the target acceleration. (mm/s^2)
Set Torque		2290	2	real	Set the torque (%); Torque Mode: When the torque setting is at "1", it represents absolute movement; when less than "1", it indicates "Push".
Torque Mode Switch		2282	1	int	Switch to set the torque mode. (To enable the function: 1, to disable the function: 0)

In point mode, the point parameters can also be read using the 03H function code and written using the 10H function code to the corresponding parameters, with the specific addresses as shown in the figure.

15-Point Editor Parameter Modification										
No.	Type	Next Step Command	Command Parameter Addresses are arbandd in the order of parameters related to each Command type						Function Code	Number of Registers
0	5000	5002	5004	5006	5008	5010	5012	5014	Write 10H Read 03H	2
1	5016	5018	5020	5022	5024	5026	5028	5030		
2	5032	5034	5036	5038	5040	5042	5044	5046		
3	5048	5050	5052	5054	5056	5058	5060	5062		
4	5064	5066	5068	5070	5072	5074	5076	5078		
5	5080	5082	5084	5086	5088	5090	5092	5094		
6	5096	5098	5100	5102	5104	5106	5108	5110		
7	5112	5114	5116	5118	5120	5122	5124	5126		
8	5128	5130	5132	5134	5136	5138	5140	5142		
9	5144	5146	5148	5150	5152	5154	5156	5158		
10	5160	5162	5164	5166	5168	5170	5172	5174		
11	5176	5178	5180	5182	5184	5186	5188	5190		
12	5192	5194	5196	5198	5200	5202	5204	5206		
13	5208	5210	5212	5214	5216	5218	5220	5222		
14	5224	5226	5228	5230	5232	5234	5236	5238		
15	5240	5242	5244	5246	5248	5250	5252	5254		

Example: Command Sequence Number 0							
Absolute Move	Type	Next Step Command	Position	Velocity	Acceleration	Decceleration	Band
Address	5000	5002	5004	5006	5008	5010	5012

Example: Command Sequence Number 0								
Push	Type	Next Step Command	Distance	Velocity	Acceleration	Force limit %	Position Band	Time Band
Address	5000	5002	5004	5006	5008	5010	5012	5014

Command Type Sequence Number Explanation		
Command Type	Number	Taking Command Sequence Number 0 as an example: the Modbus address for the Command type is 5000. When 5000 equals 1, the Command type is for setting the home position. When 5000 equals 3, the Command type is for "Absolute Move". The data type for the Command type is a double integer.
None	0	
Set Home	1	
Delay	2	
Absolute Move	3	
Push	4	
Relative Move	5	
Precise Push	6	
Force Reset	7	
Stop	8	
Execute and Collect Data	9	

Command Type Description		
Command Type	Command Parameters	Data Type
Set Home	Home Position Offset (mm)	Floating Point Number
Delay	Time (ms)	Double Integer
Absolute Move	Position (mm)	Floating Point Number
	Velocity (mm/s)	
	Acceleration (mm/s^2)	
	Decceleration (mm/s^2)	
	Band (mm)	
Push	Distance (mm)	Floating Point Number
	Velocity (mm/s)	
	Acceleration (mm/s^2)	
	Force limit (%)	
	Position Band (mm)	
	Time Band (ms)	
Relative Move	Position (mm)	Floating Point Number
	Velocity (mm/s)	
	Acceleration (mm/s^2)	
	Decceleration (mm/s^2)	
	Band (mm)	
Precise Push	Distance (mm)	Floating Point Number
	Force (N)	
	Velocity Factor	
	Impact Factor	
	Force Positioning Band (N)	
	Stabilization Time (ms)	
Execute and Collect Data	Acquisition Frequency (khz)	Double Integer
	Acquisition Quantity	
	Number of Acquisition Channels	
	Channel 0	
	Channel n	

4.1.3 04H Function Code

The 04H function code is for reading input registers, used to read one or more 16-bit values from the input registers of the slave device. The 04H function code can be used to read the current position (address: 0), velocity (address: 2), and force sensor readings (address: 16), with each data point occupying two registers. This function allows for real-time reading of the electric gripper's position, velocity, and sensor force parameters, facilitating real-time observation of the electric actuator's status or making conditional judgments in automated processes.

Name	Function Code	Address (Decimal)	Number of Registers	Data Type	Function
Current Position	04H	0	2	real	Read the current position of the motor.
Current Velocity		2	2		Read the current velocity of the motor.
Sensor Current Reading (N)		16	2		Read the sensor readings.

4.1.4 05H Function Code

The function of the 05H function code is to force a single coil, that is, to turn a specific Digital Output (DO) contact ON or OFF. The 05H function code can be used to trigger actions where the data type is a boolean, as shown in the figure.

Name	Function Code	Address (Decimal)	Number of Registers	Data Type	Function
Reset Error	05H	0	1	bool	Trigger the controller to reset errors on the rising edge.
Servo Toggle		1	1		Set the Servo Toggle state. (Write 0 to disable, write 1 to enable)
Start Command		2	1		Trigger the controller to start Commands on the rising edge. (Use with a specified sequence number; it is recommended to use the direct execution of position sequence numbers below.)
Stop Command		3	1		Trigger the controller to stop Commands on the rising edge.
Save Parameters		9	1		Trigger the controller to save parameters on the rising edge. (Save the actuator's operating parameters, for debugging use only.)
Save Positioning Command		11	1		Trigger the controller to save all Commands from the position editor on the rising edge. (Save modified target positions, velocitys, accelerations, and other Command parameters.)
Reset Force		16	1		Trigger the controller to reset the force value on the rising edge.
Initialize		17	1		Trigger the controller for Initialize (return to home position) on the rising edge.
Execute Position 0		1000	1		Trigger the controller to execute position 0 on the rising edge.
Execute Position n		1000+n	1		Trigger the controller to execute position n on the rising edge.
Execute Position 15		1015	1		Trigger the controller to execute position 15 on the rising edge.

Except for the Servo Toggle command, which needs to be continuously set to ON, all other command triggers are on the rising edge, with the trigger method being to first write 0 and then write 1. If the value of 1 is repeatedly written, the action will not be properly triggered.

4.2 Modbus Communication Message Example

Modbus RTU (Remote Port Unit) communication message format adheres to a strict binary format, suitable for serial communication and particularly common in device communication within industrial automation environments. Below are the general components of a Modbus RTU message:

Name	Function
Device Address	A byte, ranging from 0x00 to 0x7F (0 to 247 in decimal), with the 0x00 address typically used for broadcasting, and other addresses used to specify a particular device.※
Function Code	A byte that identifies the specific action requested, such as reading coil status (0x01), reading discrete input status (0x02), reading holding registers (0x03), writing a single holding register (0x06), and so on.
Data Field	Depending on the function code, several bytes follow to carry the necessary data, such as register addresses, the number of registers, and the data values to be read or written.
Checksum	A two-byte Cyclic Redundancy Check (CRC) value is used to detect if there are any errors that occurred during the transmission of the message.

※Our products can all control all electric actuators in the network to move synchronously via broadcast mode, and at the same time, use a polling method to obtain the status of each actuator.

A typical example of a Modbus RTU message is shown below.

[Device Address]	[Function Code]	[Data Format]	[CRC High Byte]	[CRC Low Byte]
8bit	8bit	N*8bit	8bit	8bit

The actual CRC value is calculated from the entire message (excluding the CRC itself) using a specific algorithm. CRC checks are performed during both transmission and reception to confirm the integrity of the message. Additionally, there are no extra padding characters or spaces between messages; adjacent messages are distinguished by the shortest pause time.

4.2.1 Read Current Position / Velocity / Torque

01 04 00 00 00 02 71 CB （Read Current Position）

- **01** represents the slave device address, indicating that the message is sent to the device with the station number 1.
- **04** represents the function code, indicating the reading of values from the input registers, which are registers that store the digital quantity of external input signals.
- **00 00** represents the address in hexadecimal, indicating the starting address to be read from, with 00 00 corresponding to address 0.
- **00 02** represents the data length, indicating that two registers are to be read.
- **71** represents the CRC check low byte.
- **CB** represents the CRC check high byte.

This message indicates the use of function code 04 to read from the registers of slave station 1, starting from address 0, reading two registers. According to the address table, we can understand that the purpose of this message is to read the current position of slave station 1.

Similarly, to read parameters such as Velocity or torque, simply change the function code and address accordingly.

4.2.2 Read Current Alarm Signal/Action Completion Signal

01 02 00 00 00 01 B9 CA（Read Current Alarm Status）

- **01** represents the slave device address, indicating that the message is sent to the device with station number 1.
- **02** represents the function code, indicating the reading of input status, which means reading a digital input quantity.
- **00 00** represents the address in hexadecimal, indicating the starting address to be read from, with 00 00 corresponding to address 0.
- **00 01** represents the data length, indicating that one input status is to be read.
- **B9** represents the CRC check low byte.
- **CA** represents the CRC check high byte.

This message indicates the use of function code 02 to read the input status of slave station 1, starting from address 0, reading one input status. According to the address table, we can understand that the purpose of this message is to read the current error alarm status of slave station 1.

Similarly, to read parameters such as position completion signals or home return completion signals, simply change the address.

4.2.3 Read Current Torque/Positioning Parameter Information

01 03 08 6A 00 02 E6 77（Read Current Torque）

- **01** represents the slave station address, indicating that the message is being sent to the device with station number 1.
- **03** represents the function code, indicating the reading of holding registers, which are registers whose values are not changed by external input signals.
- **08 6A** represents the address in hexadecimal, indicating the starting address to be read from, with 08 6A corresponding to address 2154.
- **00 02** represents the data length, indicating that two holding registers are to be read.
- **E6** represents the CRC check low byte.
- **77** represents the CRC check high byte.

This message indicates the use of function code 03 to read the holding registers of slave station 1, starting from address 2154, reading two holding registers. According to the address map, it is known that the purpose of this message is to read the current torque of slave station 1.

Similarly, to read parameters in point mode or the values in the holding registers of positioning mode, simply change the address.

4.2.4 Set Positioning Parameters/Positioning Mode Parameters

When writing parameters, it is necessary to perform operations for converting floating-point numbers to hexadecimal and for endian conversion. Endian conversion is to address the differences in the order of data storage between various computer systems. The main reasons include:

1. System Architecture Differences: Different systems may use little-endian (low byte first) or big-endian (high byte first) byte order.
2. Network Communication: Network protocols often specify a unified byte order to ensure that data is correctly transmitted between different systems.
3. Data Consistency: Ensuring the correctness and consistency of data in cross-platform applications.
4. Performance Optimization: Optimizing data access according to the characteristics of the processor to improve efficiency.
5. Compatibility: Maintaining compatibility with existing software libraries and data formats.

Therefore, when writing parameters for point mode or positioning mode, it is necessary to first convert the floating-point numbers into hexadecimal, then perform endian conversion before writing into the controller.

If you need to write the floating-point number 20 into the controller, first convert 20 into a hexadecimal number. The hexadecimal equivalent of the floating-point number 20 is 41 A0 00 00, and after endian conversion, it becomes 00 00 41 A0.

Thus, the message to write the floating-point number 20 into the controller is: **01 10 08 EC 00 02 04 00 00 41 A0 AA 5A**

- **01** represents the slave station address, indicating that the message is being sent to the device with station number 1.
- **10** represents the function code, where 10 in hexadecimal is used for presetting multiple registers.
- **08 EC** represents the address in hexadecimal, indicating the starting address to be set, with 08 EC corresponding to address 2284.
- **00 02** represents the number of registers to be written.
- **04** represents the number of bytes of the value to be written.
- **00 00 41 A0** represents the value to be written, which is the hexadecimal conversion of the floating-point number 20 followed by an endian conversion.
- **AA** represents the CRC check low byte.
- **5A** represents the CRC check high byte.

This message indicates the use of function code 10 to preset registers in slave station 1, starting from address 2284 and writing two registers with the value of a 4-byte floating-point number 20. According to the address map, it is known that the purpose of this message is to write the floating-point number 20 into the positioning mode's position register.

Similarly, to set parameters for point mode or set parameter values for positioning mode, simply change the address and the value being written.

4.2.5 Trigger Error Reset/Servo Toggle/Command Stop/Force Reset(Precision Torque Control)/Initialize/Execute Positioning Actions

01 05 00 00 FF 00 8C 3A （Reset Error）
01 05 00 00 00 00 CD CA （Reset Error Acknowledgment）

- **01** represents the slave station address, indicating that the message is being sent to the device with station number 1
- **05** represents the function code, which is used to force a single coil, effectively setting a specific Digital Output (DO) point to ON or OFF.
- **00 00** represents the address in hexadecimal, indicating the starting address to be set, with 00 00 corresponding to address 0.
- **FF 00** represents the value to be written, meaning ON.
- **8C** represents the CRC check low byte.
- **3A** represents the CRC check high byte.

This message indicates the use of function code 05 to force a coil in slave station 1 to be ON, with the address set to 0. According to the address map, it is known that the purpose of this message is to force the triggering of the error reset command.

Similarly, commands for stopping Commands, resetting force (precision torque control), Initialize, and executing positioning actions can be achieved by changing the address and the value being written. The Servo Toggle must be kept ON to operate normally.

When using the 05 function code to trigger an action, you need to first write 0 and then write 1. The controller captures a rising edge to trigger the corresponding function. If the value is continuously set to 1, it will prevent the action from being continuously triggered (the "Servo Toggle" function is an exception. When this register is set to 1, it maintains the enabled state; when set to 0, it is disabled).

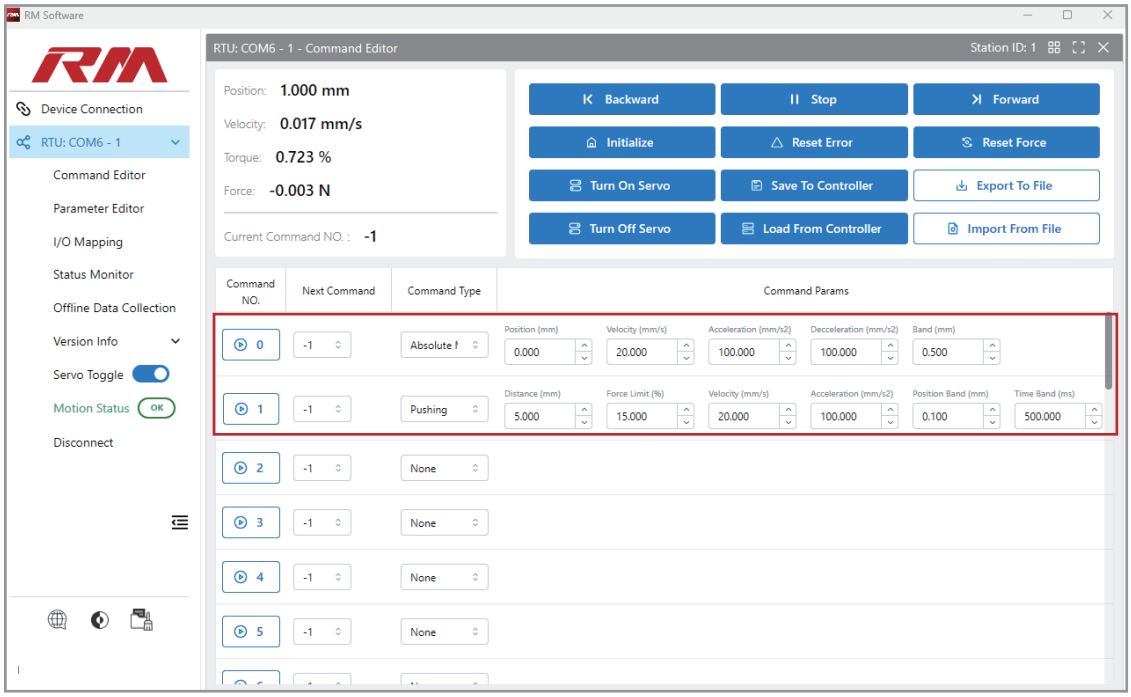
4.3 Point Mode Usage Instruction

4.3.1 Point Mode Introduction

In point mode, the parameters of the point command must be programmed first. This involves writing the position, force, distance, acceleration, velocity, etc., into the point sequence number n and saving these parameters to the controller or actuator. Once the parameters are saved, the point command number n can be triggered, and the actuator will execute the action according to the set point command. If the parameters are written but not saved, the actuator will not execute the newly written action.

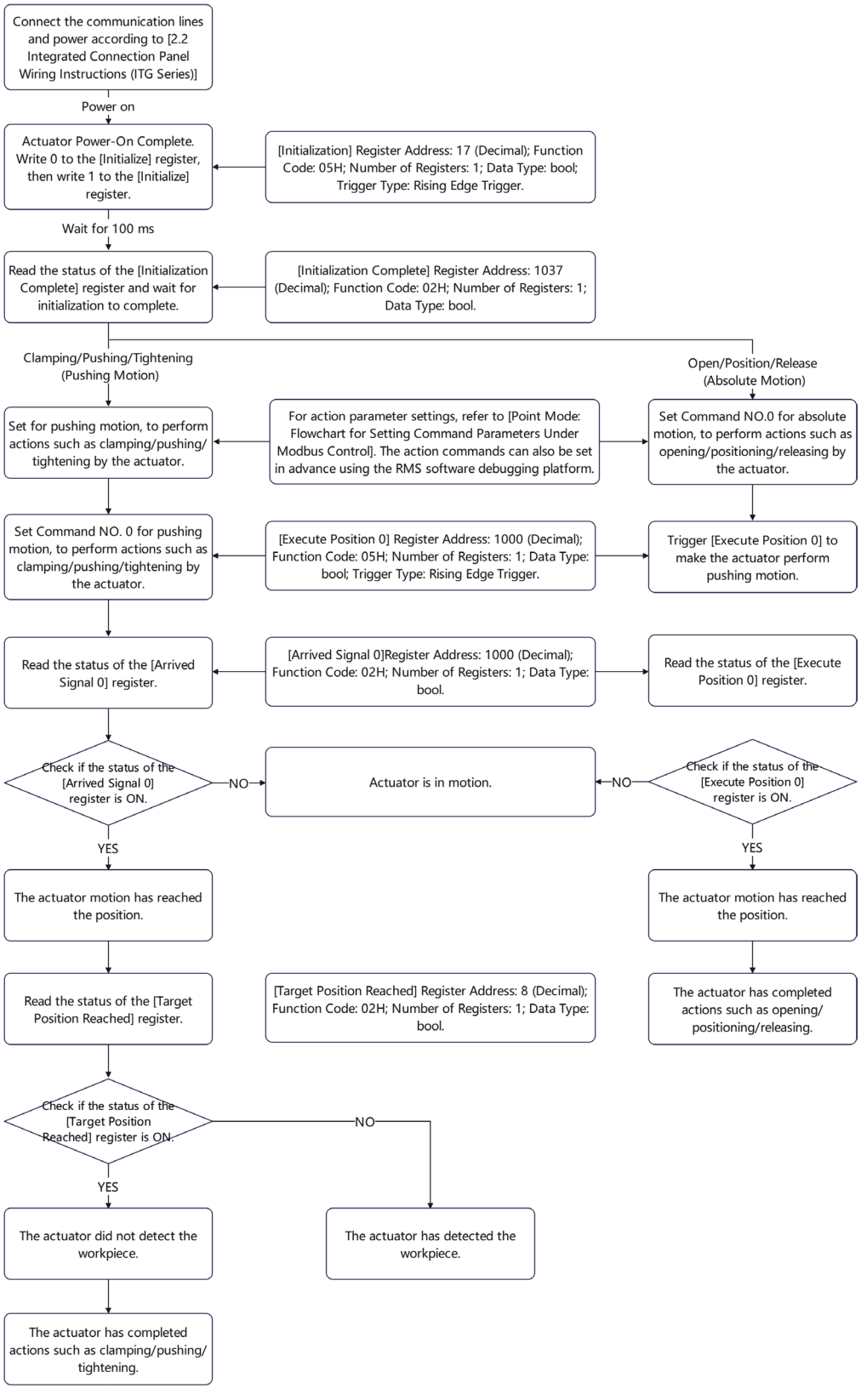
Point parameters in point mode can also be written using the RM software debugging platform (as shown in the figure). Access the [Command Editor] section in the RM software debugging platform, enter the command parameters for the point sequence number, and after completion, click [Save Params To Controller] to save the parameters.

In point mode, valid point sequence numbers range from 0 to 15, corresponding to trigger register addresses from 1000 to 1015. For example, when trigger register address 1000 is activated, the actuator will execute the action for point sequence number 0.

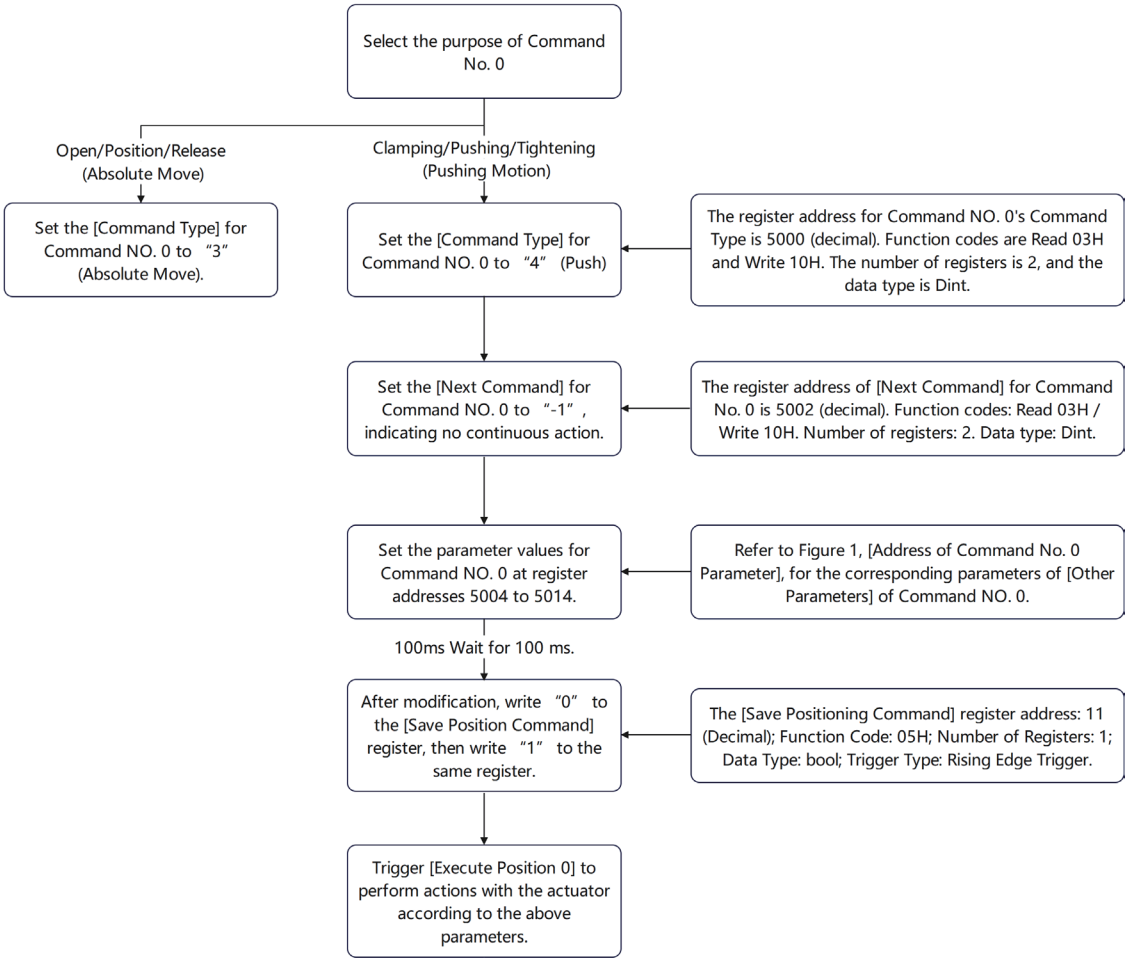


4.3.2 Modbus Control Flow Chart (Point Mode)

1. Point Mode: Flowchart for Triggering Actions Under Modbus Control



2. Point Mode: Flowchart for Setting Command Parameters Under Modbus Control



Address of Command NO. 0 Parameter		
Register Address	When the [Command Type] value is 3.	When the [Command Type] value is 4.
5004	Position (mm)	Distance (mm)
5006	Velocity (mm/s)	Velocity (mm/s)
5008	Acceleration (mm/s^2)	Acceleration (mm/s^2)
5010	Deceleration (mm/s^2)	Force Limit (%)
5012	Band (mm)	Position Band (mm)
5014	—	Time Band (ms)
Function Code: Read 03H/Write 10H; Number of Registers: 2; Data Type: REAL (Floating Point). Note: When the value of [Command Type] is different, the meanings of the above registers will vary.		

4.3.3 Modbus RTU Example (Point Mode)

1. Set command number 0 to [Absolute Move] and trigger (Typically used for actions such as jaw opening, positioning, and releasing).

Target Action Parameters						
Command Type	Next Command	Target Position (mm)	Velocity (mm/s)	Acceleration (mm/s^2)	Deceleration (mm/s^2)	Position Band (mm)
3 (Absolute Move)	-1	5	80	500	500	0.1

- **Set the [Command Type] of Command NO. 0 to Absolute Move.**
  Send: 01 10 13 88 00 02 04 00 03 00 00 D3 59
  Return: 01 10 13 88 00 02 C5 66

- **Set the [Next Command] of Command NO. 0 to -1 (Next action is not executed).**
  Send: 01 10 13 8A 00 02 04 FF FF FF FF A3 14
  Return: 01 10 13 8A 00 02 64 A6

- **Set [Position] of Command NO.0 to 5 (Target position of Absolute Move).**
  Send: 01 10 13 8C 00 02 04 00 00 40 A0 13 12
  Return: 01 10 13 8C 00 02 84 A7

- **Set the [Velocity] of Command NO.0 to 80 (Target velocity for Absolute Move).**
  Send: 01 10 13 8E 00 02 04 00 00 42 A0 93 AB
  Return: 01 10 13 8E 00 02 25 67

- **Set the [Acceleration] of Command NO. 0 to 500 (Target acceleration for Absolute Move).**
  Send: 01 10 13 90 00 02 04 00 00 43 FA 92 80
  Return: 01 10 13 90 00 02 45 61

- **Set the [Deceleration] of Command NO. 0 to 500 (Target deceleration for Absolute Move).**
  Send: 01 10 13 92 00 02 04 00 00 43 FA 13 59
  Return: 01 10 13 92 00 02 E4 A1

- **Set the [Positioning Range] of Command NO. 0 to 0.1 (Determination range for absolute move to reach the target position).**
  Send: 01 10 13 94 00 02 04 CC CD 3D CC 9C 6A
  Return: 01 10 13 94 00 02 04 A0

- **Use the rising edge to trigger the [Save Positioning Command] and save the parameters of the above settings.**
  Set the [Save Positioning Command] Register to 0.
  Send: 01 05 00 0B 00 00 BC 08
  Return: 01 05 00 0B 00 00 BC 08
  Set the [Save Positioning Command] Register to 1.
  Send: 01 05 00 0B FF 00 FD F8
  Return: 01 05 00 0B FF 00 FD F8
  ※If the above point parameters have been written and saved in advance by the RM Debugging Software Platform, there is no need to re-write them.

- **Trigger [Command NO. 0] on the rising edge. The actuator will start moving.**
  Set the [Command NO. 0] Register to 0.
  Send: 01 05 03 E8 00 00 4D BA
  Return: 01 05 03 E8 00 00 4D BA
  Set the [Command NO. 0] Register to 1.
  Send: 01 05 03 E8 FF 00 0C 4A
  Return: 01 05 03 E8 FF 00 0C 4A

Judging whether the electric actuator has reached the target position

- **Read [Arrived Signal 0] (the arrived signal corresponding to [Command NO. 0]).**
  Send: 01 02 03 E8 00 01 39 BA
  Case ① Return: 01 02 01 00 A1 88 (Actuator is in motion.)
  Case ② Return: 01 02 01 01 60 48 (Actuator has reached the target position in absolute move.)

2. Set command number 0 to [Push] and trigger (Typically used for actions such as gripping, pushing and tightening).

Target Action Parameters							
Command Type	Next Command	Target Position (mm)	Velocity (mm/s)	Acceleration (mm/s²)	Force Limit (%)	Position Band (mm)	Time Band (ms)
4（Push）	-1	5	20	100	50	0.1	100

- **Set the [Command Type] of Command NO.0 to Push.**
  Send: 01 10 13 88 00 02 04 00 04 00 00 62 98
  Return: 01 10 13 88 00 02 C5 66

- **Set the [Next Command] of Command NO. 0 to -1 (Next action is not executed).**
  Send: 01 10 13 8A 00 02 04 FF FF FF FF A3 14
  Return: 01 10 13 8A 00 02 64 A6

- **Set the [Distance] of Command NO. 0 to 5 (the pushing distance for the push motion).**
  Send: 01 10 13 8C 00 02 04 00 00 40 A0 13 12
  Return: 01 10 13 8C 00 02 84 A7

- **Set the [Velocity] of Command NO.0 to 80 (Target velocity for Push).**
  Send: 01 10 13 8E 00 02 04 00 00 41 A0 93 5B
  Return: 01 10 13 8E 00 02 25 67

- **Set the [Acceleration] of Command NO. 0 to 100 (Target acceleration for Push).**
  Send: 01 10 13 90 00 02 04 00 00 42 C8 12 C5
  Return: 01 10 13 90 00 02 45 61

- **Set the Force Limit% of Command NO. 0 to 0.5 (Force Limit% for the push, where 0.5 corresponds to 50% output).**
  Send: 01 10 13 92 00 02 04 00 00 3F 00 B3 DA
  Return: 01 10 13 92 00 02 E4 A1

- **Set the [Positioning Range] of Command NO. 0 to 0.1 (Determination range for Push motion to reach the target position).**
  Send: 01 10 13 94 00 02 04 CC CD 3D CC 9C 6A
  Return: 01 10 13 94 00 02 04 A0

- **Set the Time Band of Command NO. 0 to 100 ms (Time range for determining when the pushing force reaches the set value).**
  Send: 01 10 13 96 00 02 04 00 00 42 C8 92 EF
  Return: 01 10 13 96 00 02 A5 60

- **Use the rising edge to trigger the [Save Positioning Command] and save the parameters of the above settings.**
  Set the [Save Positioning Command] Register to 0.
  Send: 01 05 00 0B 00 00 BC 08
  Return: 01 05 00 0B 00 00 BC 08
  Set the [Save Positioning Command] Register to 1.
  Send: 01 05 00 0B FF 00 FD F8
  Return: 01 05 00 0B FF 00 FD F8
  ※If the above point parameters have been written and saved in advance by the RM Debugging Software Platform, there is no need to re-write them.

- **Trigger [Command NO. 0] on the rising edge. The actuator will start moving.**
  Set the [Command NO. 0] Register to 0.
  Send: 01 05 03 E8 00 00 4D BA
  Return: 01 05 03 E8 00 00 4D BA
  Set the [Command NO. 0] Register to 1.
  Send: 01 05 03 E8 FF 00 0C 4A
  Return: 01 05 03 E8 FF 00 0C 4A

To judge whether the electric actuator has clamped / pushed / tightened the workpiece:

- **Read [Arrived Signal 0] (The arrived signal corresponding to [Command NO. 0]).**
  Send: 01 02 03 E8 00 01 39 BA
  Case ① Return：01 02 01 00 A1 88 (Actuator is in motion.)
  Case ② Return：01 02 01 01 60 48 (Actuator has reached the target position in absolute move.)
  (Wait for 100ms)

- **Read [Target Position Reached]**
  Send: 01 02 00 08 00 01 38 08
  Case ① Return: 01 02 01 01 60 48 (Return value is 1, indicating no grip / no push / not tightened)
  Case ② Return: 01 02 01 00 A1 88 (Return value is 0, indicating clamped / pushed / workpiece tightened)

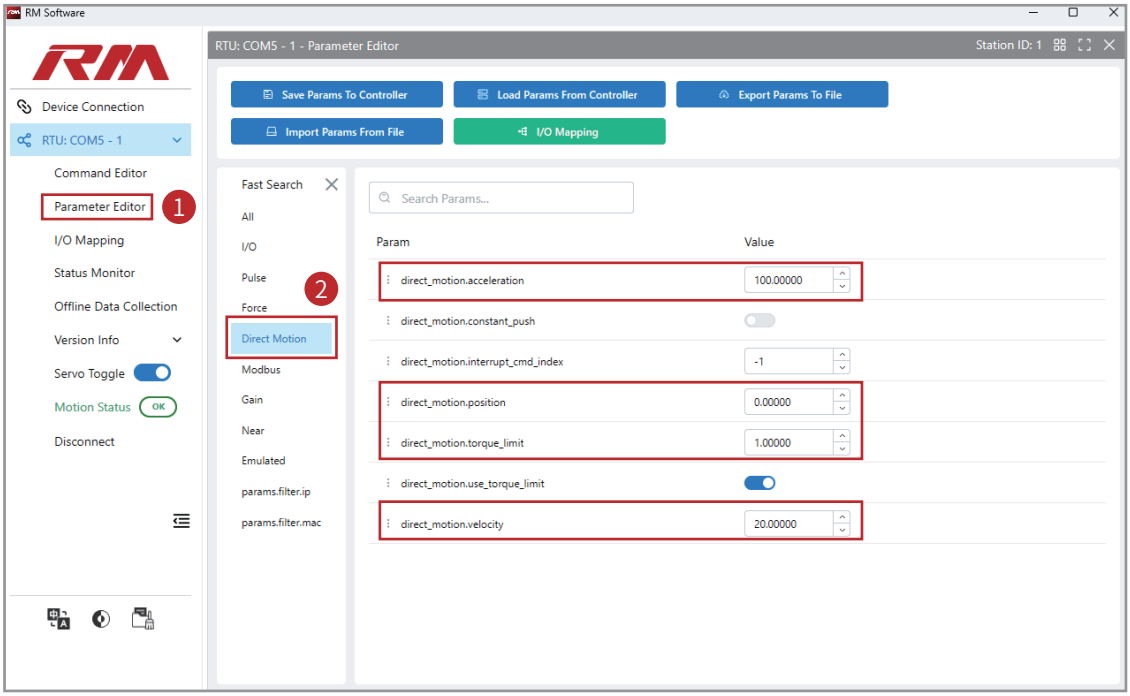
4.4 Positioning Mode User Guide

4.4.1 Introduction to Positioning Mode

Positioning Mode, also known as Position Mode, requires the input of parameters such as distance, acceleration, and Velocity before the target position parameter is written. Once the target position parameter is written, the actuator will immediately execute the action without the need for a trigger signal; if only the target position parameter is written without the other parameters, the actuator will not perform any action. When the torque value set is "1", the electric actuator will perform "Absolute Move"; when the torque value is set to less than "1", the electric actuator will perform pressing motion, which is the torque mode.

The triggering logic in positioning mode is based on differential detection. If the written value differs from the current value of the driver, the actuator will trigger an action to match the new value; if the written value is the same as the value in the driver, the actuator will not take any action. For example: if the current register position value is 0, and the set position register value is 0.5mm, it will trigger the action.

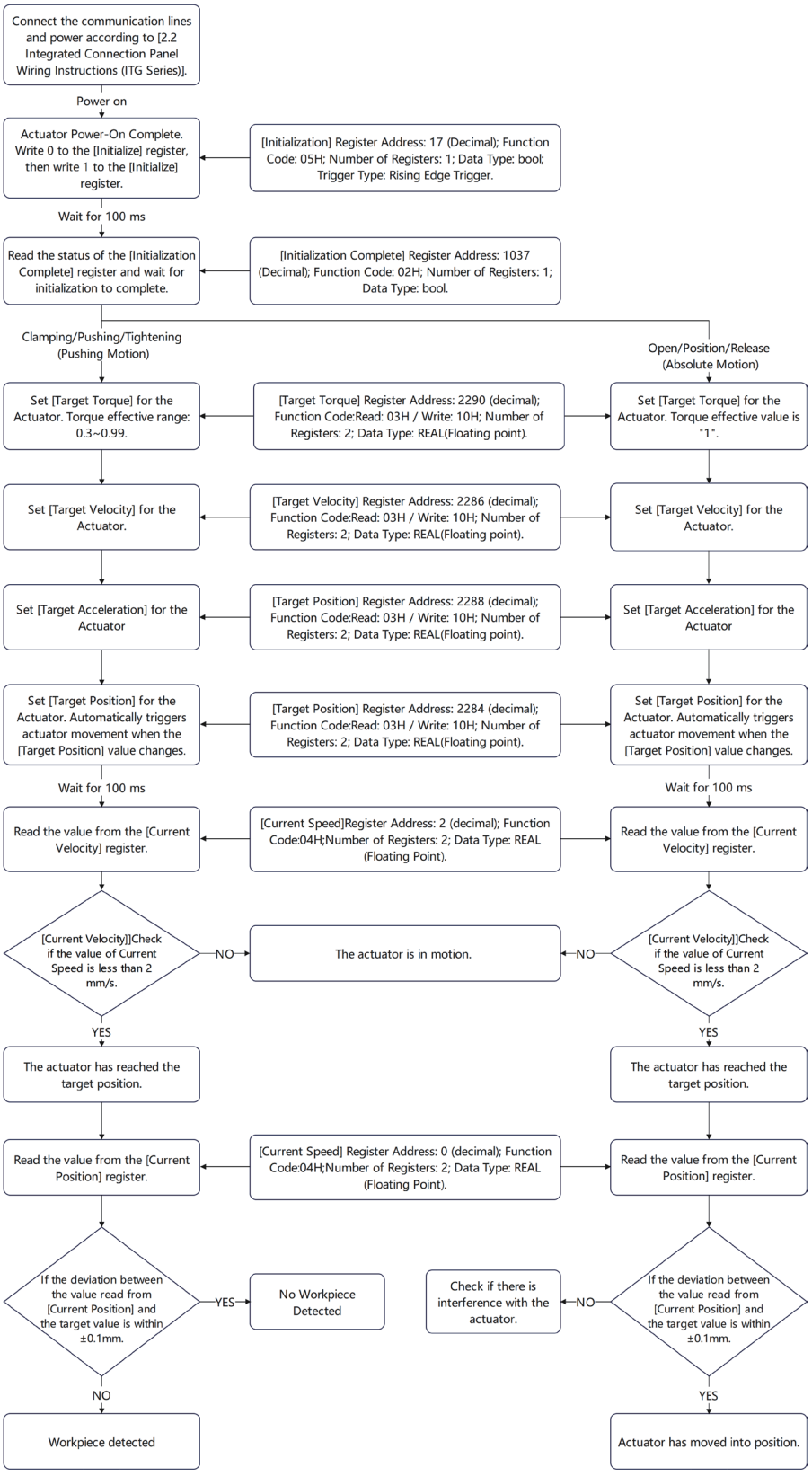
In positioning mode, the values written by the upper computer can be read through the RMS Software debugging platform. Open the [Parameter Editor] in the RMS Software debugging platform and select "Direct Motion" to read the parameter values in the positioning mode of the driver. Among them, [direct_motion.velocity], [direct_motion.acceleration], [direct_motion.torque_limit], and [direct_motion.position] correspond to the "Velocity", "Acceleration", "Torque", and "Position" in the communication address table, respectively.



When the torque switch is enabled, writing a torque value less than 1 activates the pressing mode. You can open the switch by searching for "direct_motion.use_torque_limit" in the Parameter Editor section of the RMS Software debugging platform.

4.4.2 Modbus Control Flowchart (Positioning Mode)

1. Modbus Trigger Action Flowchart



4.4.3 Modbus RTU Example (Using Positioning Mode to Perform Gripping/Opening Actions)

1. Modify Parameters and Perform "Absolute Move" (Suitable for Opening the Gripper or Rapid Approach Action) Determine if the Electric Actuator Has Reached the Target Position:

   Example: Modify the target position, Velocity, acceleration, and torque to drive the electric actuator in "Absolute Move".

Target Action Parameters			
Target Position（mm）	Velocity（mm/s）	Acceleration（mm/s²）	Torque
40	80	500	1（100%）

- **Set the Velocity to 80mm/s**
  Send：01 10 08 EE 00 02 04 00 00 42 A0 2B 73
  Return：01 10 08 EE 00 02 23 9D

- **Set the acceleration to 500mm/s²**
  Send：01 10 08 F0 00 02 04 00 00 43 FA 2A 58
  Return：01 10 08 F0 00 02 43 9B

- **Set the torque to 1 (100%)**※
  Send：01 10 08 F2 00 02 04 00 00 3F 80 0A A2
  Return：01 10 08 F2 00 02 E2 5B
  ※When the electric actuator needs to perform "Absolute Move", the torque must be set to 1 (100%).

- **Move to a target position of 40mm**※
  Send：01 10 08 EC 00 02 04 00 00 42 20 AB 0A
  Return：01 10 08 EC 00 02 82 5D
  （Movement starts）
  ※In positioning mode, it is necessary to first set the torque, acceleration, and Velocity, and then finally set the target position. If only the target position is set without other parameters, the actuator will not perform any action.

Determine if the Electric Actuator Has Reached the Target Position:

Determination Band	
Velocity	The current Velocity is below 2mm/s.
Position	The deviation between the current position and the target position is within ±0.1mm.

- **Read Current Velocity**
  Send：01 04 00 02 00 02 D0 0B
  Return：01 04 04 5A CB 3F 0B C8 95 （Convert the floating-point number 03D8 4220 to 0.5443541mm/s）

- **Read Current Position**
  Send：01 04 00 00 00 02 71 CB
  Return：01 04 04 03 D8 42 20 4A 83 （Convert the floating-point number 4220 03D8 to 40.00375mm）
  （The electric actuator has reached the target position）

2. Modify Parameters and Perform Pressing Motion (Suitable for the Gripper Tightening Inward or Supporting the Workpiece) Determine if the Electric Actuator is Holding/Pressing the Workpiece:

Example: Modify the target position, Velocity, acceleration/deceleration, and torque to drive the electric actuator in pressing motion.

Target Action Parameters			
Target Position（mm）	Velocity（mm/s）	Acceleration（mm/s²）	Torque
20	20	100	0.5（50%）

- **Set the Velocity to 20mm/s**
  Send：01 10 08 EE 00 02 04 00 00 41 A0 2B 83
  Return：01 10 08 EE 00 02 23 9D

- **Set the acceleration to 100mm/s²**
  Send：01 10 08 F0 00 02 04 00 00 42 C8 AA 1D
  Return：01 10 08 F0 00 02 43 9B

- **Set the torque to 0.5 (50%)**※
  Send：01 10 08 F2 00 02 04 00 00 3F 00 0B 02
  Return：01 10 08 F2 00 02 E2 5B
  ※When the electric actuator needs to perform pressing motion, the torque must be set within the band of 0.3 to 0.99 (30% to 99%), and the torque mode switch (register address: 2282) must be activated. If you need to change the state of the torque mode switch, you must send the save parameter command and restart the controller for the changes to take effect.

- **Move to a target position of 20mm**※
  Send：01 10 08 EC 00 02 04 00 00 41 A0 AA 5A
  Return：01 10 08 EC 00 02 82 5D
  （Movement starts）
  ※In positioning mode, it is necessary to first set the torque, acceleration, and Velocity, and then finally set the target position. If only the target position is set without other parameters, the actuator will not perform any action.

Determine if the Electric Actuator is Clamping/Pressing onto the Workpiece:

Determination Band	
Velocity	Current Velocity is below 2mm/s.
Position	Empty Grip: The deviation between the current position and the target position is within ±0.1mm.
	Holding: The deviation between the current position and the target position is outside of ±0.1mm.

- **Read Current Velocity**
  Send：01 04 00 02 00 02 D0 0B
  Return：01 04 04 5A CB 3F 0B C8 95 （Convert the floating-point number 03D8 4220 to 0.5443541mm/s）

- **Read Current Position**
  Send：01 04 00 00 00 02 71 CB
  Scenario ① Return：01 04 04 F7 20 41 9F B8 02（Convert the floating-point number 419F F720 to 19.99567mm, the electric actuator is empty gripping/pushing）
  （Movement complete, the actuator is empty gripping/pushing）
  Scenario ② Return：01 04 04 F7 A0 41 7E 79 A2 （Convert the floating-point number 417E F7A0 to 15.93546mm, the electric actuator is gripping/pressing onto the workpiece）
  （Movement complete, the actuator is gripping/pressing onto the workpiece）

4.4.4 Positioning Mode Precautions（Q&A）

Q1: What should be considered when reading and writing data?
A1: When reading and writing data, it is essential to ensure the use of the correct data types. Incorrect data types may lead to improper data parsing or abnormal actuator responses.

Q2: How can you determine if the electric actuator has reached the target position in positioning mode?
A2: To determine if the positioning mode has reached the target condition, the host computer needs to read and compare the deviation between the current position and the target position (±0.1mm), and when the current velocity is below 2mm/s, it is considered to have reached the target (the program must include the corresponding velocity judgment logic).

Q3: How can you determine if the electric actuator has gripped/pressed the workpiece in torque mode?
A3: In torque mode, when the set torque value is below "1" (0.3~0.99), and the set target position is within the product's stroke band, the judgment conditions are as follows:

- If the RMS Software debugging platform reads and compares the deviation between the current position and the target position (±0.1mm), and the current velocity is below the set threshold (e.g., 2mm/s), it is judged as an invalid operation (NG), meaning empty grip or push.
- If the position has not fully reached the target position but the current velocity is already below the set threshold (e.g., 2mm/s), it is judged as a valid operation (OK), meaning the workpiece is gripped or pressed.

Q4: Why is the read current torque percentage smaller than the set torque percentage in torque mode?
A4: The current torque percentage = safety factor × set torque percentage; this safety factor prevents users from setting the torque percentage too high, which may not match the actuator's allowable torque percentage setting, potentially damaging the actuator. The value of this safety factor varies with the product series; for inquiries, please consult our after-sales engineers.

Q5: How to deal with the problem of the electric actuator in positioning mode not responding to the re-issued command after being interrupted by the RMS Software debugging platform (such as Initialize, stop, Servo Toggle change)?
A5: For different interruption scenarios, take the following optimization measures in the program logic:

- Initialize Interrupt: If the positioning mode is interrupted by an Initialize command, the program should wait until it receives the signal that Initialize is complete (status flag set to 1), then read the current position of the electric actuator and immediately update this current coordinate to the positioning mode's position register.
- Stop or Servo Toggle Interrupt: If the positioning mode is interrupted due to a stop command or a change in the Servo Toggle state, incorporate an appropriate delay of 15-30ms in the program to ensure the actuator's state is stable, then read the current position of the electric actuator and immediately update this current coordinate to the positioning mode's position register.

5 Maintenance

5.1 Maintenance and Service General Principles

5.1.1 First Time Use

Before the initial use, please confirm whether the interval from the date of receipt to the first use exceeds half a month (reduce appropriately in winter). If it does, it is recommended to apply a small amount of WD-40 rust-preventing lubricant to the actuator's screw rod, guide rail, and other transmission components before use, and move back and forth 3-5 times to allow the lubricant to fully contact the transmission components, ensuring the actuator is in optimal condition.

5.1.2 Not Used for More Than Half a Month / Long Term Non-use

It is necessary to first apply a small amount of WD-40 rust-preventing lubricant before use, especially when accessing travel ranges that have not been utilized for a long time.

- WD-40 rust-preventing lubricant should only be used in the aforementioned situations.
- For regular daily maintenance, please use NSL grease.
- Please use lubricants that are compatible with the specified grease to avoid abnormal chemical reactions that could cause mechanical damage.

5.2 Maintenance Frequency

	Check transmission parts regularly	Regularly check the tightness of connecting screws	Regular grease replenishment
Put into service	○		
Run for 1 month	○	○	
Run for 6 month	○	○	○
Run for 1 Year	○	○	○
Later every half year	○	○	○

Note：the above is based on operation on 5 working days a week (8 hours/day).
If the actuator needs to run day and night or be used frequently, and/or the use environment is relatively harsh (such as high dust, high temperature, etc.), please shorten the inspection period relatively.

5.3 Key Maintenance Areas

	Grease replenishment cycle	Grease supply part
RM-MFGB series of servo grippers	100W times per opening and closing or half a year	Manual rails and filaments

5.4 Dust Cover Replacement

- If the dust cover shows signs of bending, notches, fractures, or other abnormal conditions, it must be replaced promptly to avoid affecting the service life of the electric actuator.
- For dust cover replacement, please contact our company's after-sales engineer.

5.5 Regular External Cleaning and Lubrication

The guiding components such as the manual fingers in this type of product are typically exposed to the air, and during the regular maintenance cycle, these parts may accumulate dust or other dark-colored impurities. To maintain the performance of the product and extend its service life, it is recommended to regularly clean and lubricate the product itself as well as its surrounding environment. When severe dirt is present on the product's surface or after a certain period of use, the following steps should be taken for cleaning, and the specific cleaning frequency should be determined based on the specific working environment.

① Cleaning

First, spray WD-40 rust-preventing lubricant into the corners of the ball slot, then let it sit for about 10 minutes, as shown in Figure 1.

Next, use a specialized brush or rag to wipe away the main dust and impurities, as depicted in Figure 2.

Finally, manually open and close the fingers back and forth to clean the guide rail multiple times, as illustrated in Figure 3.

WD-40 Rust-Preventing Lubricant

Figure 1

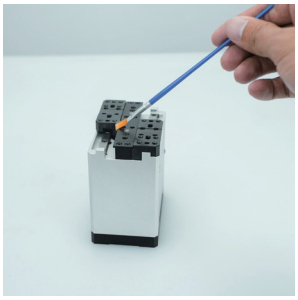


Figure 2

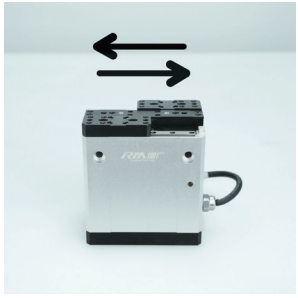


Figure 3

② Replace the Grease

After the previous step, the old lubricant should have been mostly cleaned off.

Next, move the fingers to their maximum travel and use a specialized fine brush to apply NSL grease, filling all the narrow gaps of the balls with grease, as shown in Figure 4.

NSL Grease

Figure 4

③ Remove Excess Grease

After applying the grease, the product will generally be in the condition shown in Figure 5.

To maintain the overall aesthetic of the equipment, it is recommended to wipe off the excess grease with a clean cloth.

Figure 5

④ Anti-Rust Treatment for Guide Fingers

The anti-rust capability of manual fingers is related to the presence of an oil film on their surface. Therefore, when wiping off the excess lubricating grease, you can wipe the entire surface once to ensure that a layer of oil film adheres to the surface, as shown in Figure 6.

Figure 6

5.6 Regular Self-Inspection

It is recommended to manually open and close the fingers for a complete stroke 3 to 5 times each time before powering on or changing the usage stroke. This practice helps to keep the clamps in optimal condition and prevents abnormal movement or alarms due to increased resistance introduced by the slider.

ROBUSTMOTION

USAGE GUIDE AND SERVICE SUPPORT

en.rmaxis.com/download

Scan the QR code for instant access to the Product User Manual and RMS Debugging Software.

Before use, please read the user manual carefully and properly install, debug, and use the product.

No Hot Plugging

Select the Correct Power Supply

Please Ensure Proper Wire Bundling Protection

Thank you for choosing RobustMotion!

Tel.: 0086-0757-22205682

E-mail: Overseas@rmaxis.com

ADD: 1st Floor, No. 20, Shunxiang Road, XinjiaoCommunity, Daliang Street, Shunde District, Foshan City, Guangdong Province, China

Find more at en.rmaxis.com