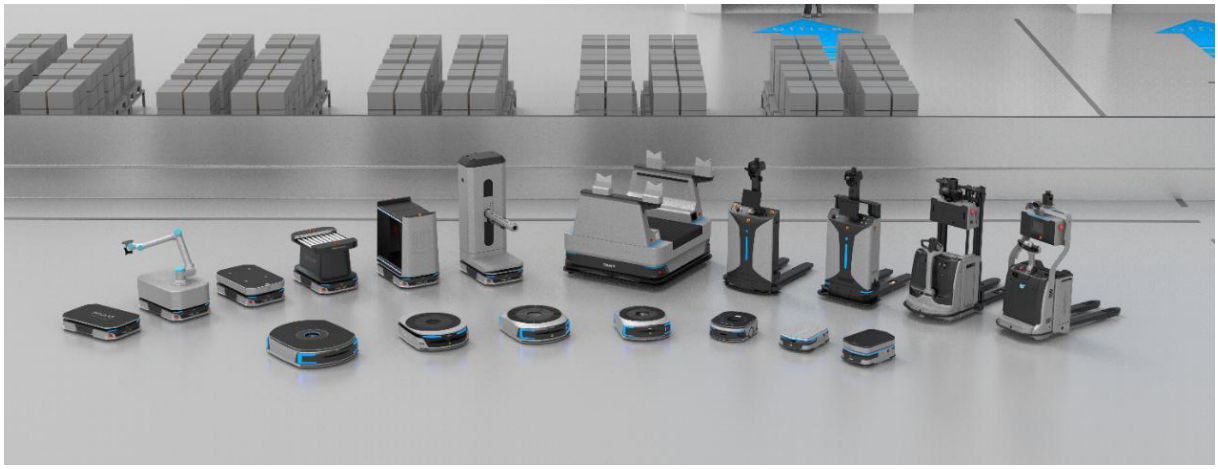
Material-Handling AGV - Elevator Interface Manual

Controlled Document



Doc No.			Version	V1.0
Department	Moving team		Application scope	☑Internal / ☑External
Prepared by	Jian Pang		External NDA	☐Yes / ☑No
Audited by	Reviewed by	Approved	Release date	Effective date
Moving team			20250904	20250904

Revision records

The revision records accumulate the descriptions of each update. The latest document issue contains all the changes in previous issues.

Release date	Version	Revised content	Revised by
2025-09-10	V1.0	**First official release** Created	Jian Pang

Original instructions

Copyright statement

Contents

1 Description of this manual

1.1 Audience and purpose

This manual is primarily intended for Geekplus implementation personnel, testing personnel, external partners, planners, and on-site maintenance staff.

Ensure that relevant personnel understand the AGV cross-floor business scenarios, interface signals, modification solution, and system configurations, ultimately enabling AGV cross-floor moving operations.

1.2 Personnel requirements

This manual is designed and written specifically for qualified professional technicians. Operating without proper training and lacking suitable tools and equipment may cause harm to yourself or others.

1.3 Application scope

This manual applies to cross-floor operations of M and F series robots.

1.4 Glossary

No.	Item	Description
1	Elevator	Generally refers to freight elevators in a factory.
2	Robot	Generally refers to F/M/P series moving robots.
3	Elevator waiting point	The location in front of the elevator where robots call the elevator.
4	Elevator station	The position inside the elevator where the robot

		stops.
6	Geek+ compact PLC	Acts as an intermediary between the RMS scheduling system and the elevator control module. It handles specific logic processing that the RMS and elevator controller cannot execute directly. This setup has low configuration requirements.
7	Elevator control module	Elevator control PLC
8	DMP	Device Management Platform. It primarily functions to convert the system's HTTP protocol to Modbus TCP protocol.

2 Basic concepts

2.1 Robot cross-floor operation

Scenario description: The robot performs tasks such as material moving, charging, and going to rest areas across floors. The elevator control is integrated with the RMS.

Integration method: RMS → compact PLC → elevator control module. All elevator control commands must be issued exclusively through the RMS.

2.2 Robot intra-floor operation

Scenario description: The robot performs tasks only on the same floor and does not operate across different floors. When materials need to be handled between floors, the robot will deliver them to the elevator on the current floor and then leave independently. Subsequently, the upstream business system takes over elevator control to enable cross-floor dispatch. After the elevator reaches the target floor, the upstream system assigns a new task, dispatching a cart on that floor for transfer.

Integration method: The upstream business systems integrate with both the Geek+ system and the elevator control module: integrating with the Geek+ system to dispatch AGV moving tasks, and integrating with the elevator control module for cross-floor elevator operation.

2.3 Elevator specifications

No.	Item	Specifications	Notes
1	Elevator weight capacity	Calculated using the formula	Must exceed the combined weight of the AGV and cargo (including tray).

2	Elevator size	⩾ 1400 mm (width)	Using a tray width of 1000 mm as a reference. If the tray is wider than 1000 mm, the elevator door width should be increased accordingly. The elevator depth should be ≥ AGV + shelf size with a minimum clearance of 200mm on one side.
3	Elevator crevice	< 20 mm	No height difference; no settlement after elevator loading.
4	AP distribution	Network AP for robot use	If a robot needs to operate across multiple floors, APs must be installed in the elevator to ensure communication.

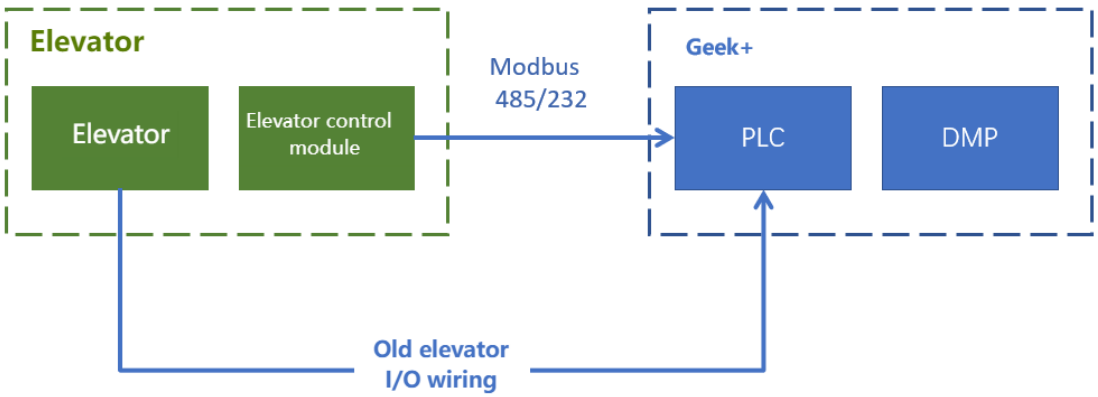
3 Interface process overview

3.1 Interface protocol

3.1.1 Elevator control interface protocol

Elevator control module interface

No.	Interface method	Interface protocol
1	The elevator has its own controller and supports external interface protocols.	Modbus TCP/UDP RS485/RS232
2	The elevator implements the interface protocol by configuring an elevator controller.	
3	The elevator only supports I/O (old elevator).	I/O wiring If the building has more than 5 floors, the cost of the elevator control module and the development effort will increase.



3.1.2 Elevator signal acquisition

Elevator signal acquisition

No.	Signal acquisition
1	Current floor
2	Target floor
3	Doors fully open status
4	Doors fully closed status
5	Operation status

3.1.3 Command signal

Command signal

No.	Command signal	Signal description
1	Elevator mode	AGV mode / mixed traffic mode
2	Origin floor	Hall call 1~N
3	Target floor	Car call 1~N
4	Open doors (front/rear)	Configurable for elevators with dual doors.
5	Close doors	Sent after AGV has entered the elevator.
6	AGV exits the elevator	Elevator can automatically switch to manual mode after AGV departure.

3.1.4 Lessons Learned

RMS: The scheduling system employs fixed logic, working in conjunction with a

DMP to translate and dispatch commands to different floors.

Elevator control module: A component of the elevator control system. Typically interfaces with proprietary modules (e.g., Mitsubishi, KONE, Hitachi) designed for specific elevator systems, or manufacturers using integrated solutions like Inovance elevator controllers. Features fixed, unmodifiable logic.

Compact PLC: For the parts of the RMS and elevator control module where the inherent logic cannot be modified, the compact PLC performs logical processing to facilitate protocol interface between both sides.

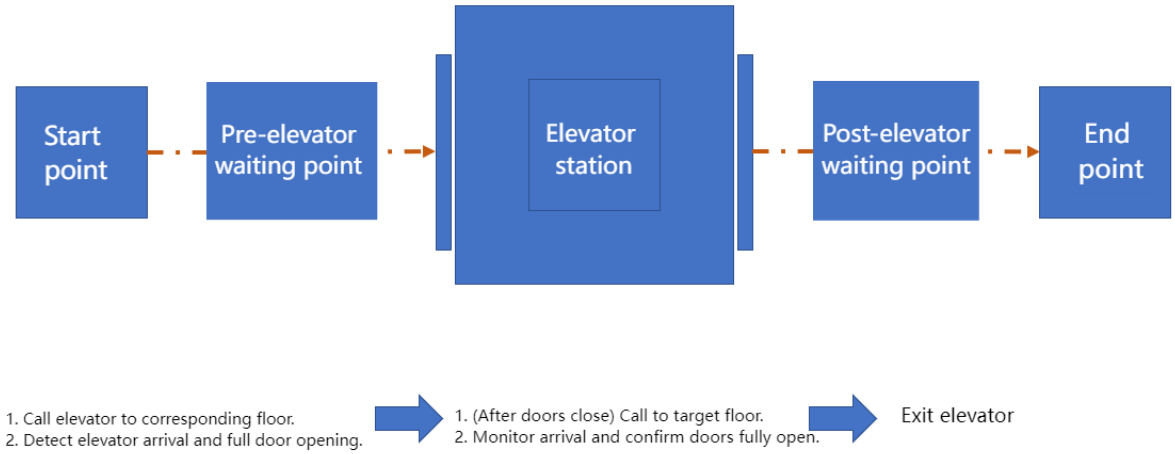
Interface summary: If the elevator control module supports Ethernet ports, the Modbus TCP protocol can be used for interface following RMS logic, eliminating the need for an additional compact PLC and reducing Geek+ hardware costs.

3.2 Interface process

Geek+ elevator calling logic

No.	Action and logic explanation	AGV signal	Underlying logic	Signal flow
1	1. AGV makes hall call. 2. The elevator reaches the corresponding floor. 3. Elevator doors fully open.	Hall call	The robot operates across multiple floors and calls the elevator from the designated waiting point outside.	AGV sends to PLC AGV → PLC
		Enter permission	After the robot initiates the call, the elevator reaches the corresponding floor and the doors open.	AGV reads from PLC PLC→AGV
2	1. The AGV enters the elevator. 2. The AGV calls the elevator to the	Car call	Upon entering the elevator, the robot calls the target floor.	AGV sends to PLC AGV→PLC
		Exit	After the robot's car call,	AGV reads from

	target floor. 3. The elevator arrives at the target floor. 4. Elevator doors fully open.	permission	the elevator arrives at the target floor and the doors open completely.	PLC PLC→AGV
3	1. AGV leaves the elevator	AGV departure	The robot has exited the elevator upon reaching the target floor.	AGV sends to PLC AGV→PLC

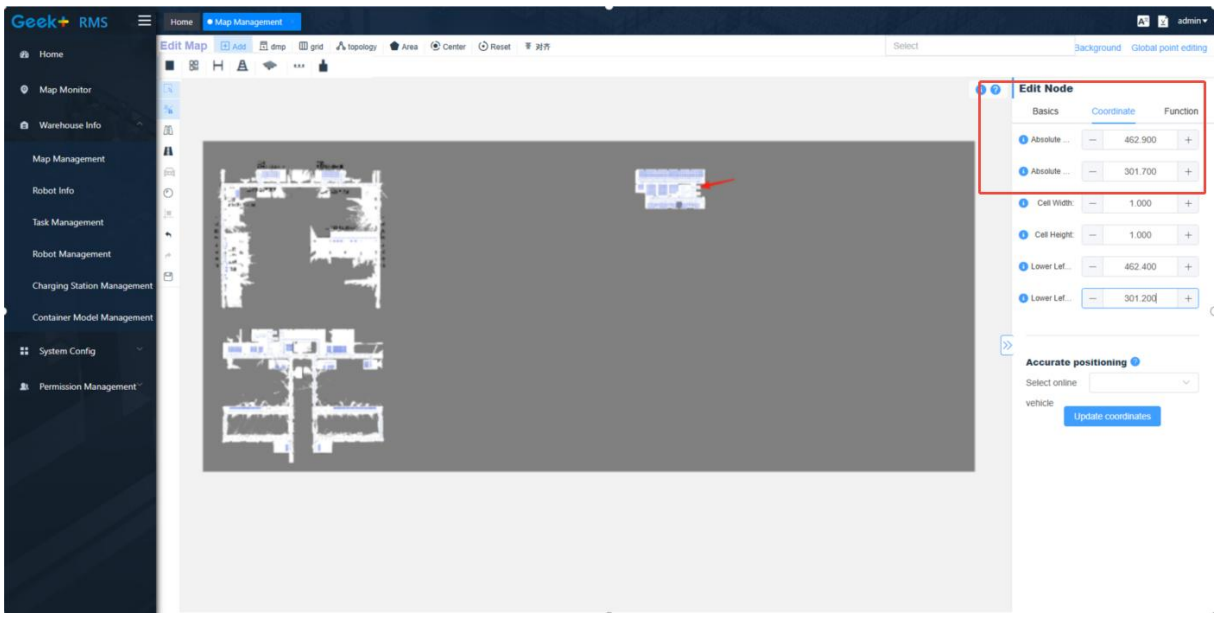


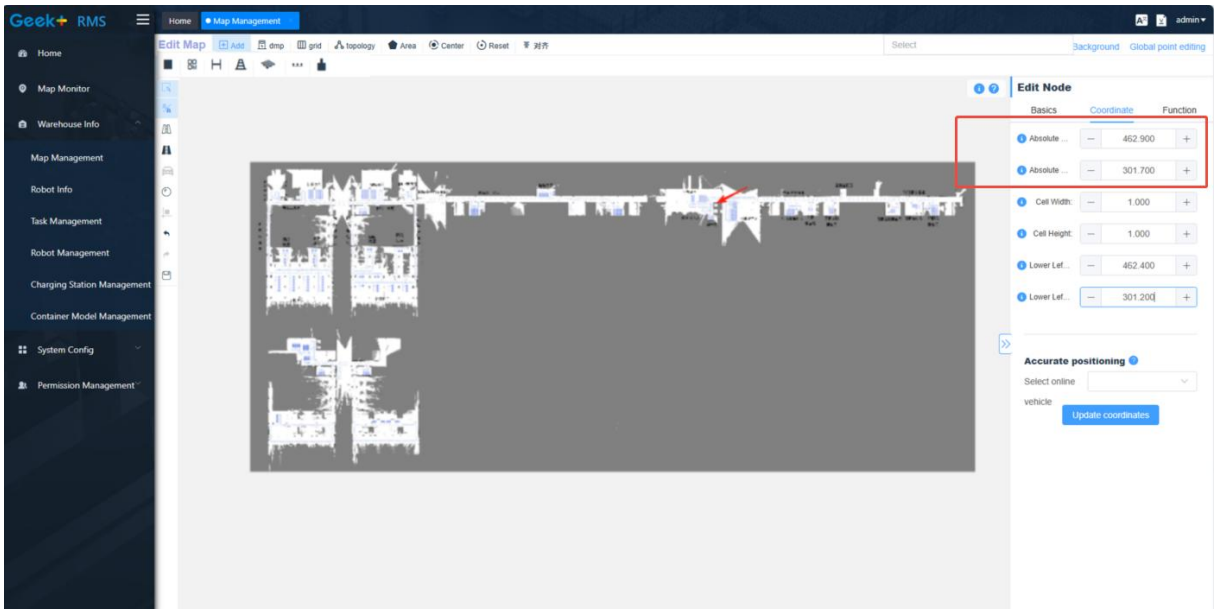
4 System-related configuration

4.1 RMS configuration

4.1.1 Base map configuration

Normal mapping operation, but the elevator coordinates must remain consistent across different floor maps. Therefore, the base maps for different floors need to be expanded. Graphic tools such as Photoshop should be used to modify the base maps, ensuring the elevator shaft positions are consistent across multiple base maps and that the coordinates in the RMS align for different floors, as shown in the following figure:

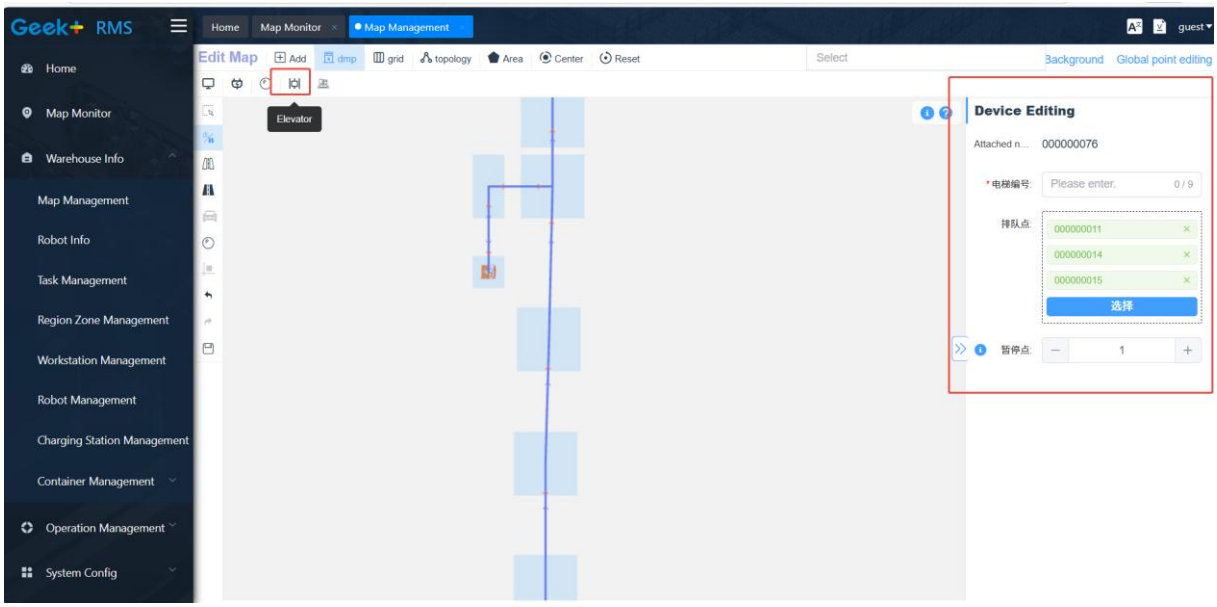


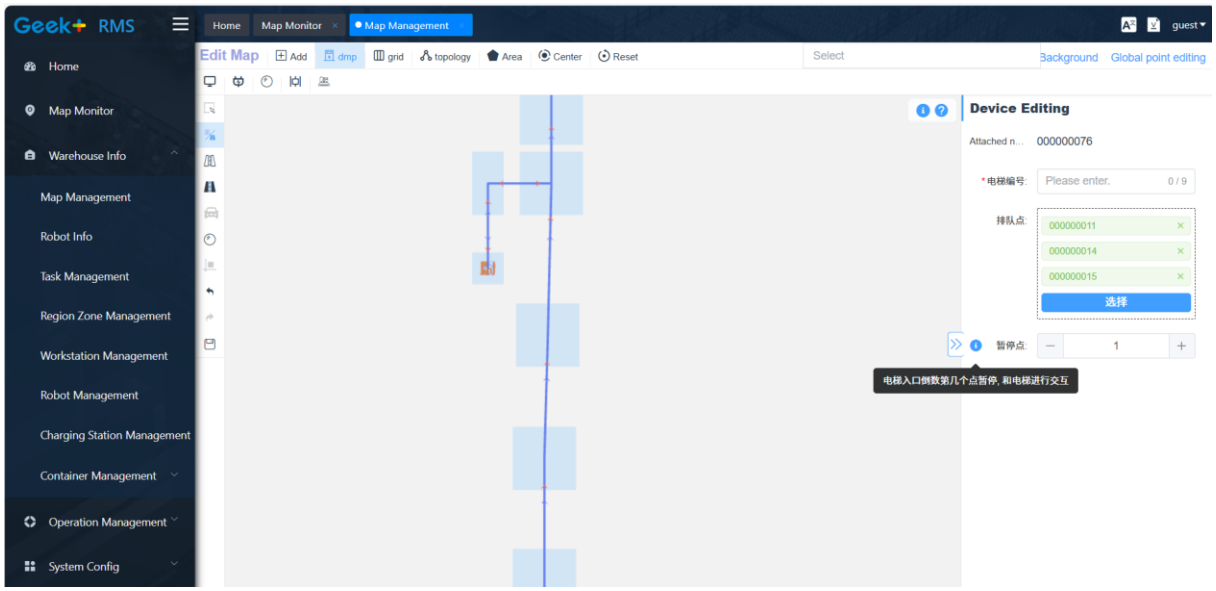


4.2 RMS configuration

Add elevator functionality to the existing points on different floors, ensuring the elevator numbers are consistent across all floors.

- Queuing point: If multiple AGVs require access, they will enter the queuing point to wait. It is recommended to position this point away from the elevator doorway to avoid congestion.
- Staging point: The interaction point between the AGV and the elevator. It is advised to locate the staging point away from both the elevator and the AGV exit path to prevent blocking traffic. It is the "elevator waiting point" shown in the figure.





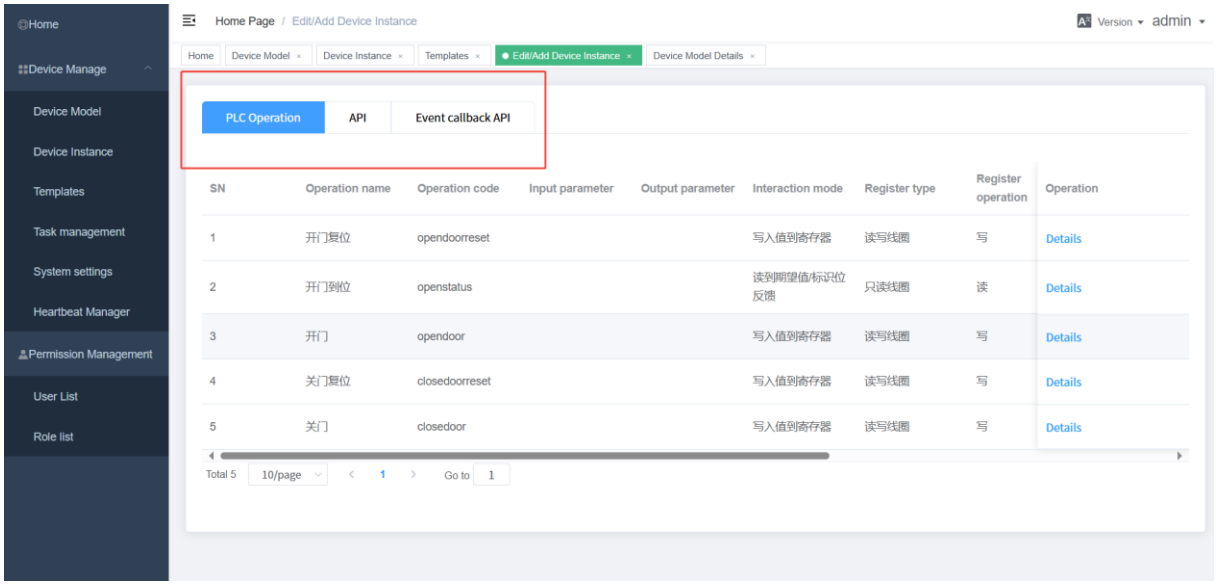
4.3 Practical configuration case study

4.3.1 DMP command

PLC operation: Refers to reading from or writing to register bits or I/O points during interfacing with the PLC.

API: Combines multiple PLC operations into a single API. One API may include multiple PLC operations (such as reading from/writing to multiple register bits or I/O points) to enable interfacing with the PLC. The application executes the PLC

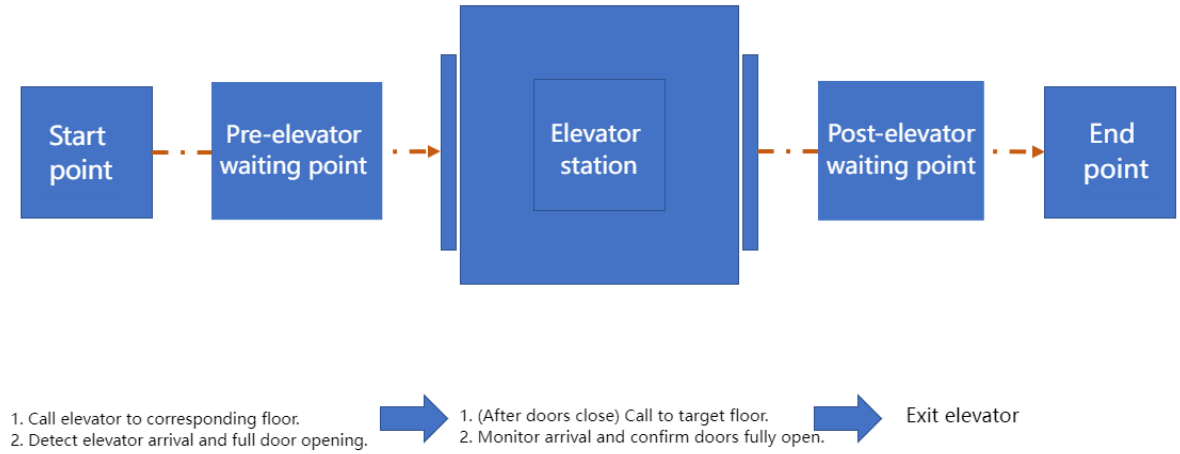
operation in the order they are added.



4.3.2 Protocol interface instructions

- For elevator without AGV mode, signal translation is handled by a compact PLC. For elevators with AGV mode, the compact PLC pre-configures mode switching and parameter read/write operations, eliminating the need for system-level parameter configuration.
- The elevator allows only one robot to enter at a time.
- AGVs share pathways with pedestrians, eliminating the need for manual transportation and saving space inside the elevator.

Interface process:



`

Interface signal:

- API
  - Hall call: 1) The AGV, located outside the elevator, calls the elevator to the corresponding floor. 2) Reads confirmation that the elevator has arrived at the corresponding floor and the door is fully open.
  - Car call: 1) The AGV, located inside the elevator, calls the elevator to the target floor. 2) Reads confirmation that the elevator has arrived at the target floor and door opening is complete.
  - Exit the elevator: 1) The AGV has completely exited the elevator.
  - Exception reset: 1) Clears all elevator commands output by the AGV when an interface exception occurs.
- Elevator interface execution sequence

No.	API	PLC operations included
1	Hall call	1) Call the elevator to the corresponding floor

		2) Reads confirmation that the elevator has arrived at the corresponding floor and door opening is complete.
2	Car call	1) Call the elevator to the target floor 2) Reads confirmation that the elevator has arrived at the target floor and door opening is complete.
3	Exit elevator	1) The AGV has completely exited the elevator.

■ Register bit interface logic

Register bit address	Address description	Status description	Description
5	**Hall call**	Default value: 0 Write 1: Call 1st floor Write 8: Call 2nd floor Write 32: Call 3rd floor	To call elevator from outside, AGV writes value to register 5: 1 = 1st floor 8 = 2nd floor 32 = 3rd floor
6	**Car call**	Default value: 0 Write 1: Call 1st floor Write 2: Call 2nd floor Write 4: Call 3rd floor	To call target floor: AGV writes value to register 6: 1 = 1st floor 2 = 2nd floor 4 = 3rd floor
7	AGV allowed to enter elevator	Default value: 0 Read 1: Allows AGV to enter the elevator.	Hall call: After the elevator arrives at the floor called by the AGV, the register bit value changes to 1. Upon reading this value, the AGV enters the elevator.
7	AGV allowed to exit elevator	Default value: 0 Read 2: Allows AGV to enter the elevator.	Car call: After the elevator arrives at the floor called by the AGV, the register bit value changes to 1. Upon reading this value, the

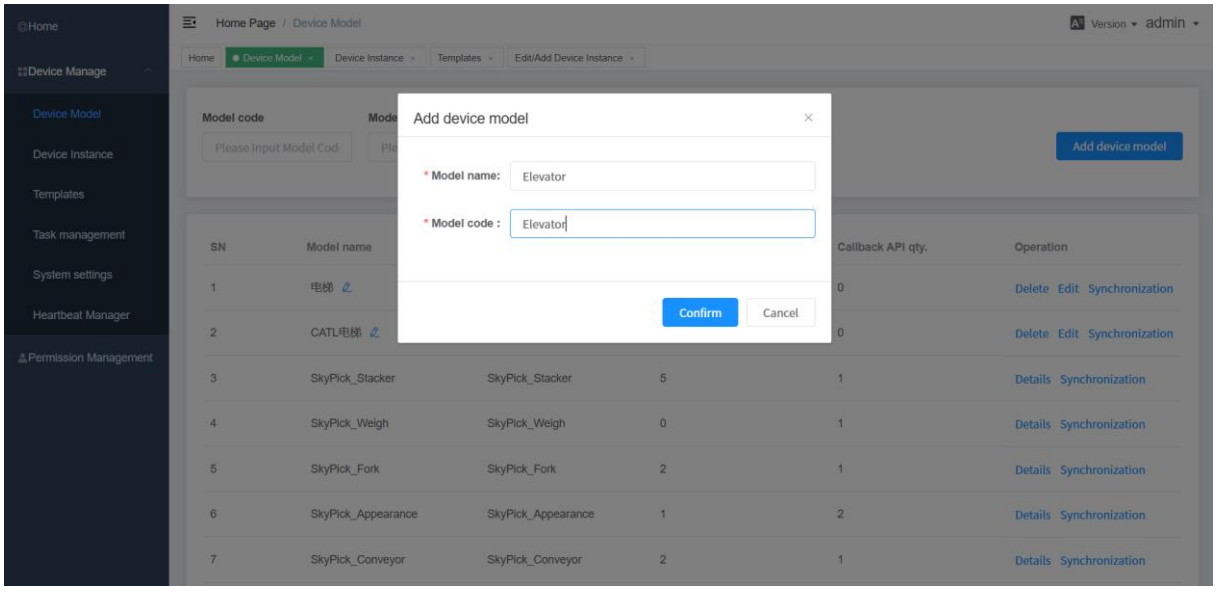
			AGV exits the elevator.
7	AGV exits the elevator	Default value: 0 Write 3: Indicates the AGV has exited.	The AGV exits the elevator, completing the entire cross-floor task interface. AGV writes value to register 7. Write 3: Indicates the AGV has exited the elevator and the interface task is completed.
8	Functional reset (for system freeze)	Default value: 0 1: System reset	Standby, used for initialization, primarily for handling abnormal tasks, with initial historical values written to the PLC.

4.3.3 DMP configuration

4.3.3.1 Device model

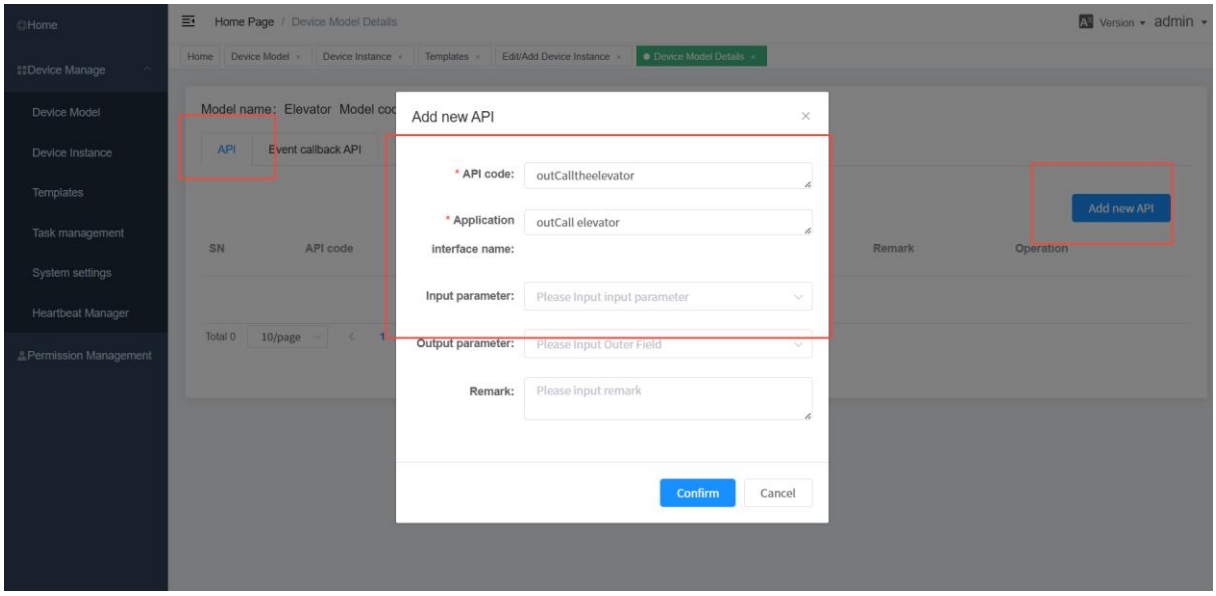
Create a device type

Create an API (command description) in the Device Model.

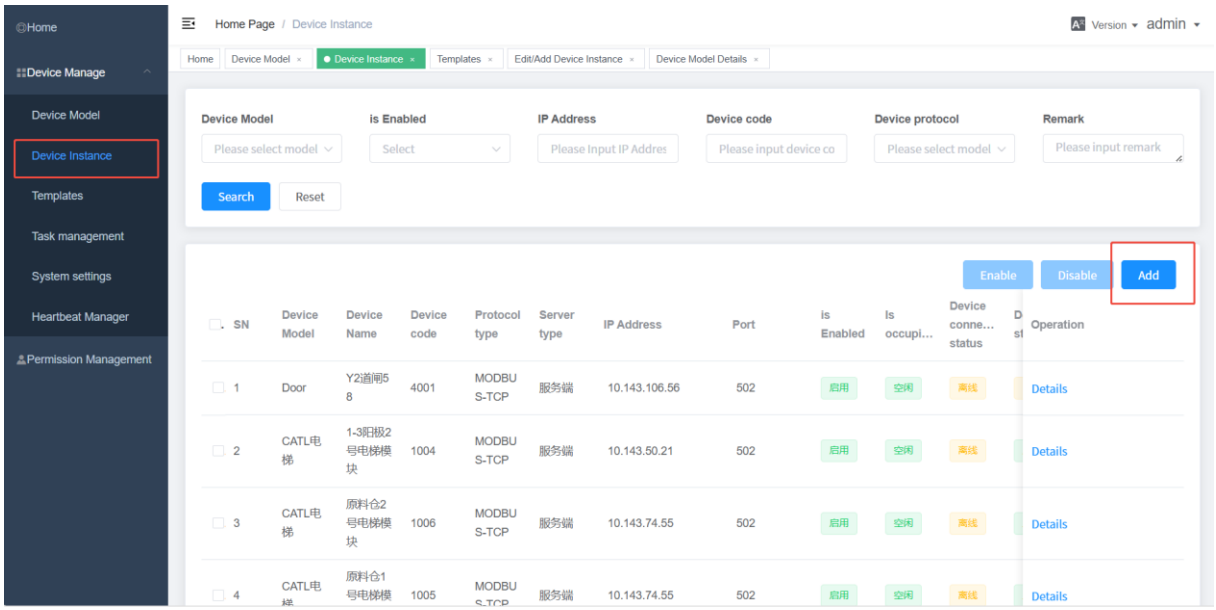


Create a new API

- Input corresponding names and numbers to distinguish between different commands.
- The input field requires the parameter "from".
- **For hall call parameters, use the field "from"; for car call and elevator exit-related parameters, use the field "to".**



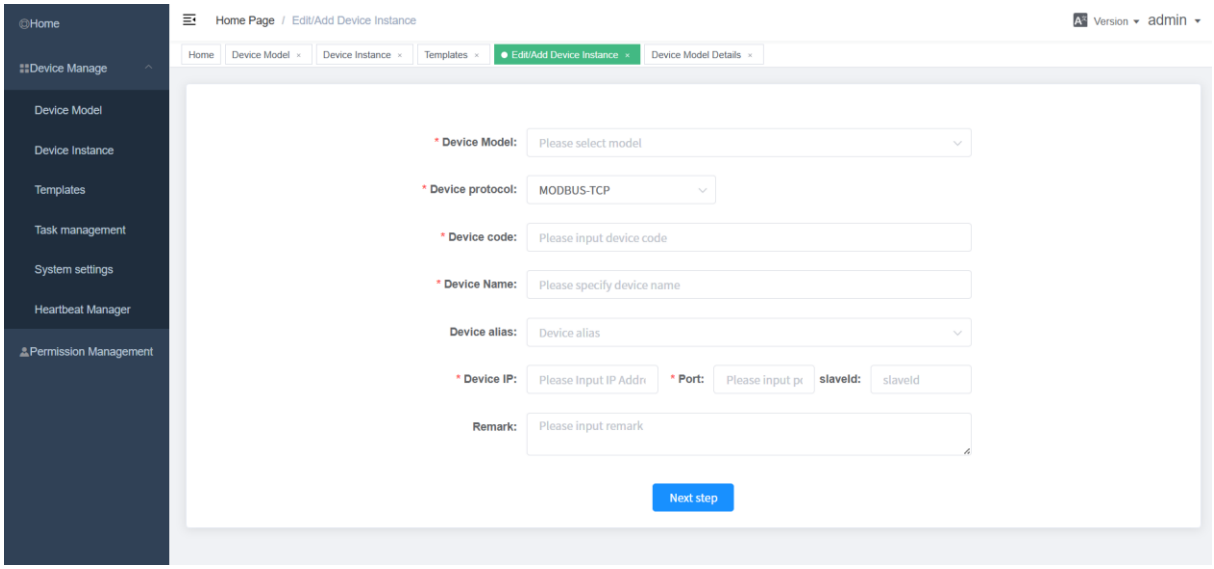
4.3.3.2 Device instance



Boundary device instance

- Device model: Select the created device model to access the API.
- Protocol: MOUBUS-TCP

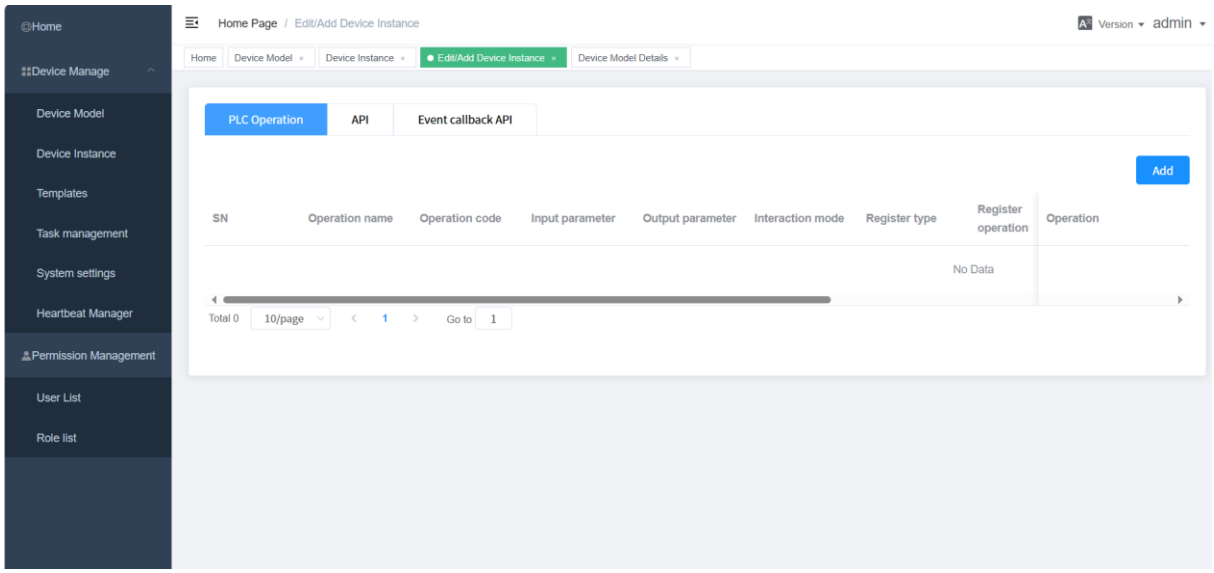
- Device code: Used to distinguish between different elevators.
- Device name: Used to distinguish between different elevators.
- Device alias: Must use "LIFT" + "ID". The ID must match the device ID in the RMS.
- IP + port ID: Fill in as provided at the working site.



4.3.3.3 PLC operation

Once created, new PLC operation can be added.

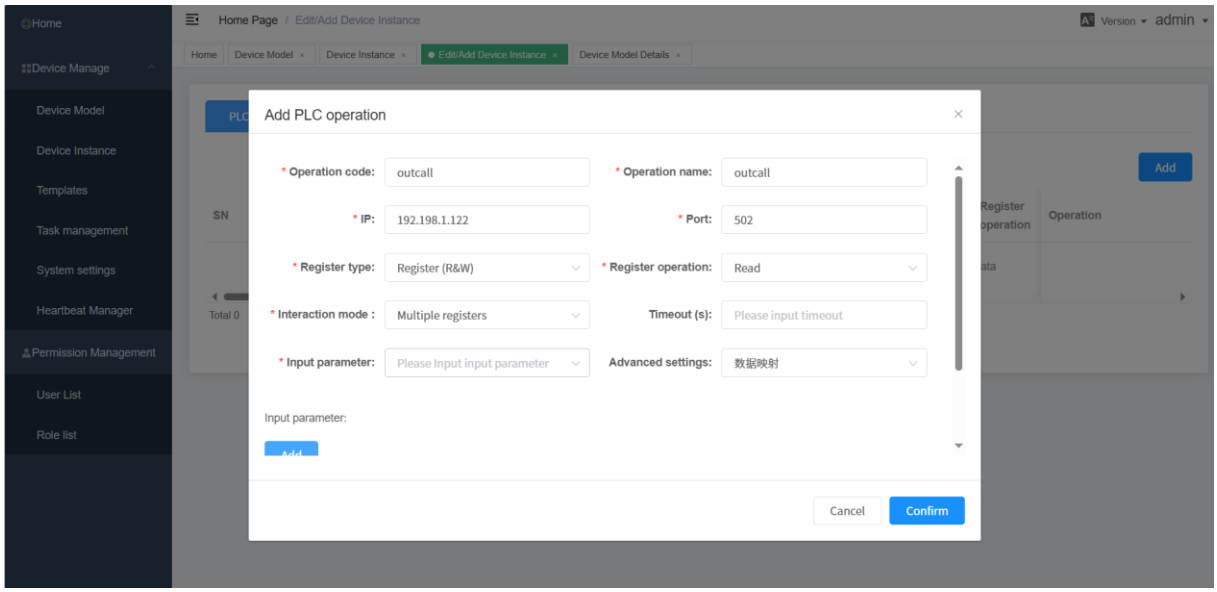
You can view the APIs created in the device model on the API sheet.



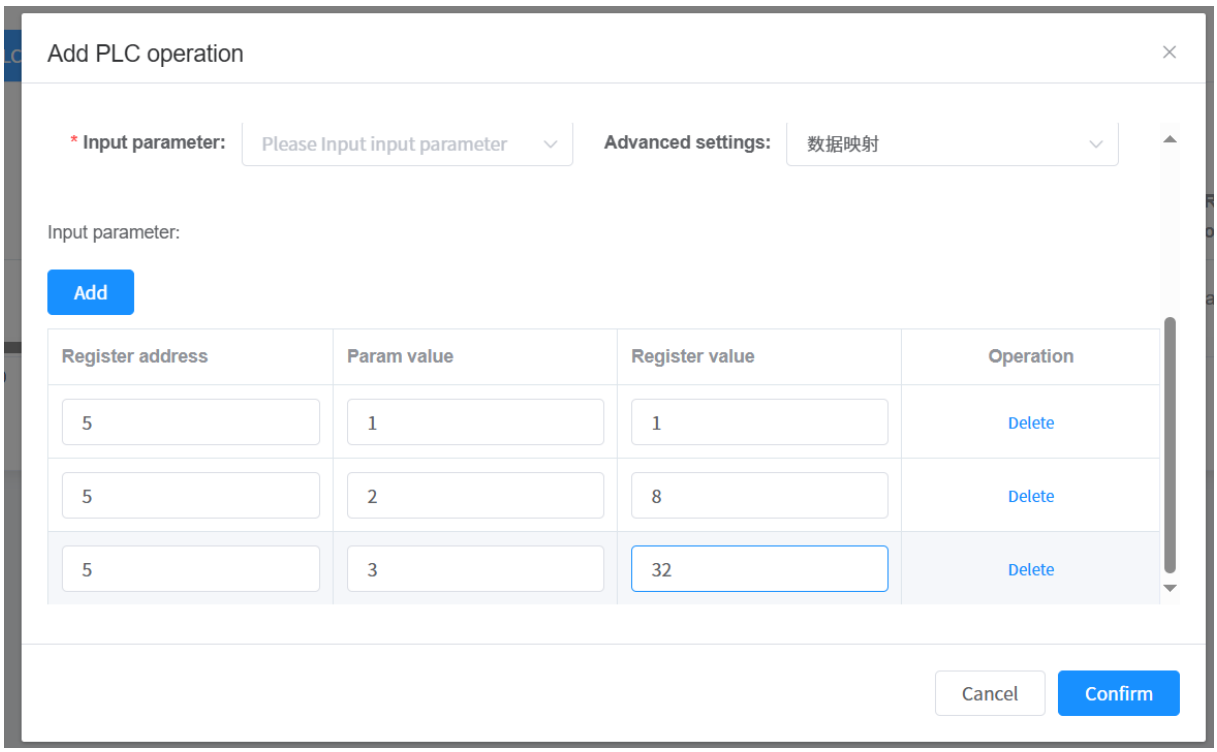
Take hall call as an example.

- Register type: Currently set to "register (R&W)", with the primary distinction being between "read" and "write".
  - Coil (R)
  - Coil (R&W)
  - Register (R)
  - **Register** (R&W)
- Register operations: Different options are available depending on the register type; for example:
  - Read register: Read
  - Register (R&W): Read/write
- Data processing method: currently using register set
  - Write value to register: Write a fixed value to a fixed register address.
  - Upstream value written to register: references an external value; writes the input parameter field (parameter) to a fixed register address.
  - Register set: Writing different values to a single register
- Input parameter field: Commands for creating APIs, mainly mapping the RMS floor fields.
- Advanced configuration: here uses data mapping.
  - Data extraction: Mainly extracts input parameter fields and writes them to different registers.
  - Data mapping: Based on multiple pieces of information in the input parameter field, responds to different information and writes to different registers.

The hall call-related details are shown in the figure below:



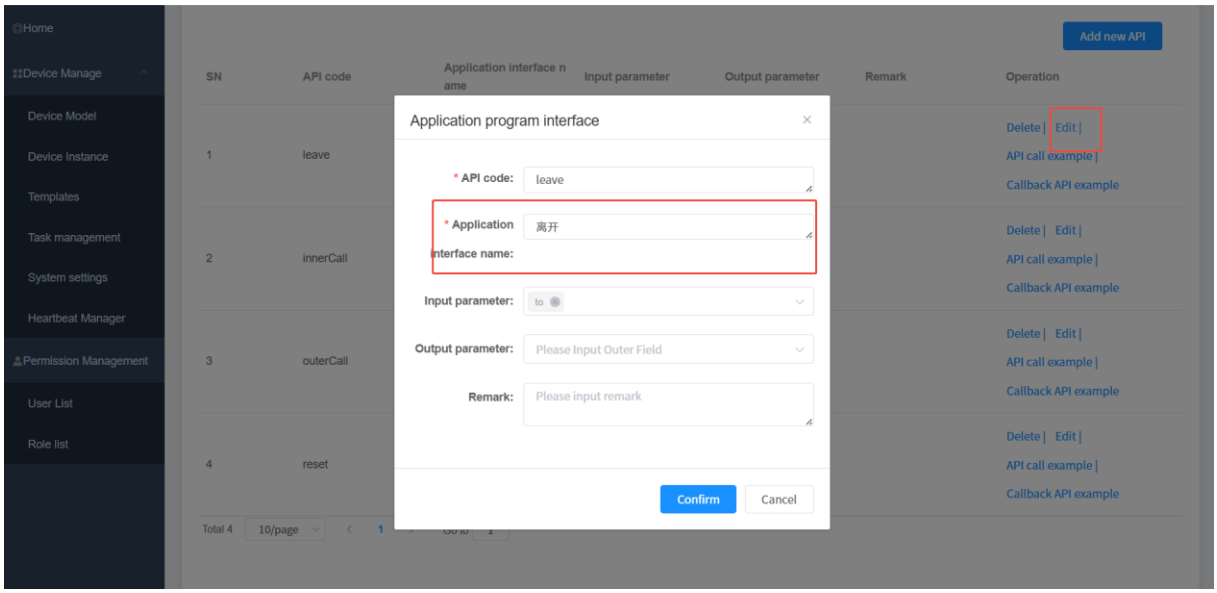
Enter the relevant register bits and values based on previous discussions.



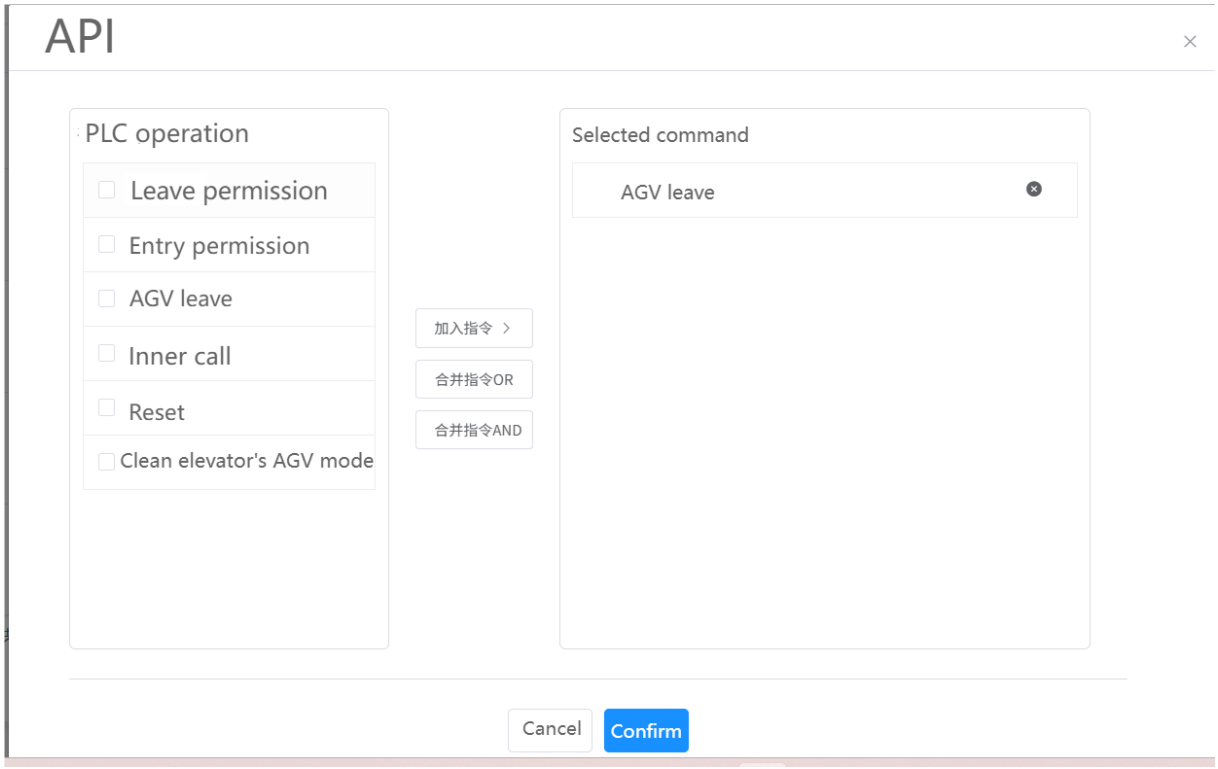
4.3.3.4 Application programming interface (API)

API: Combines PLC operations

API: When creating a device model, it is necessary to define an API. On this page, multiple PLC operations are integrated into one API instruction to communicate with the PLC and complete the interaction.



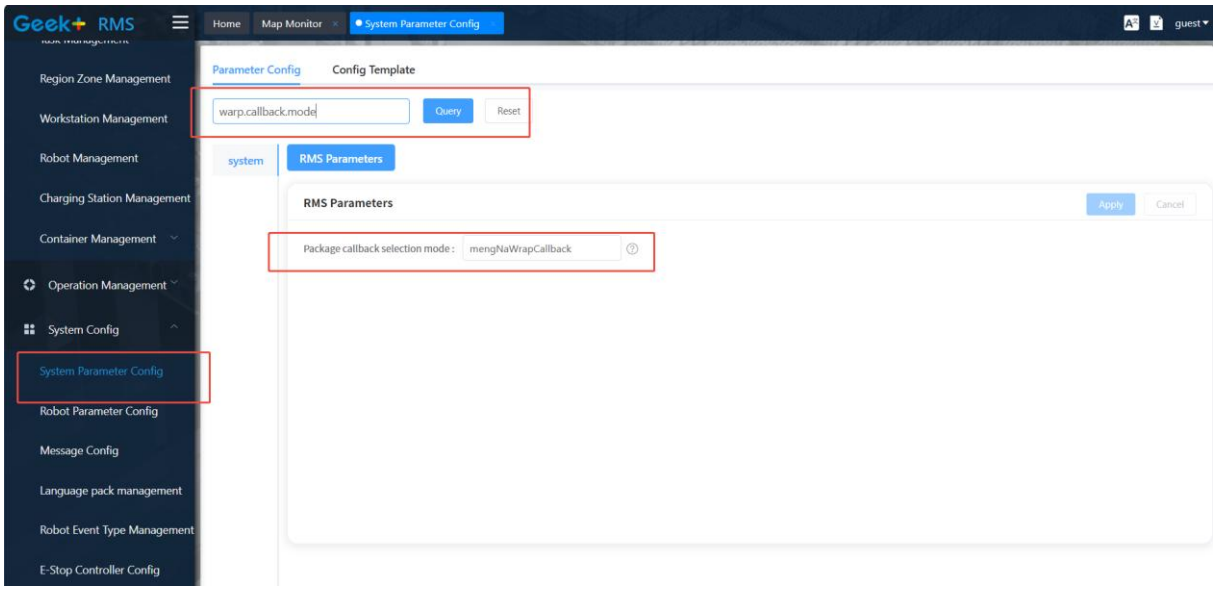
Merge the completed PLC operations and save them.



4.3.4 RMS parameter configuration

RMS system parameter configuration

Elevator interaction: warp.callback.mode=mengNaWrapCallback



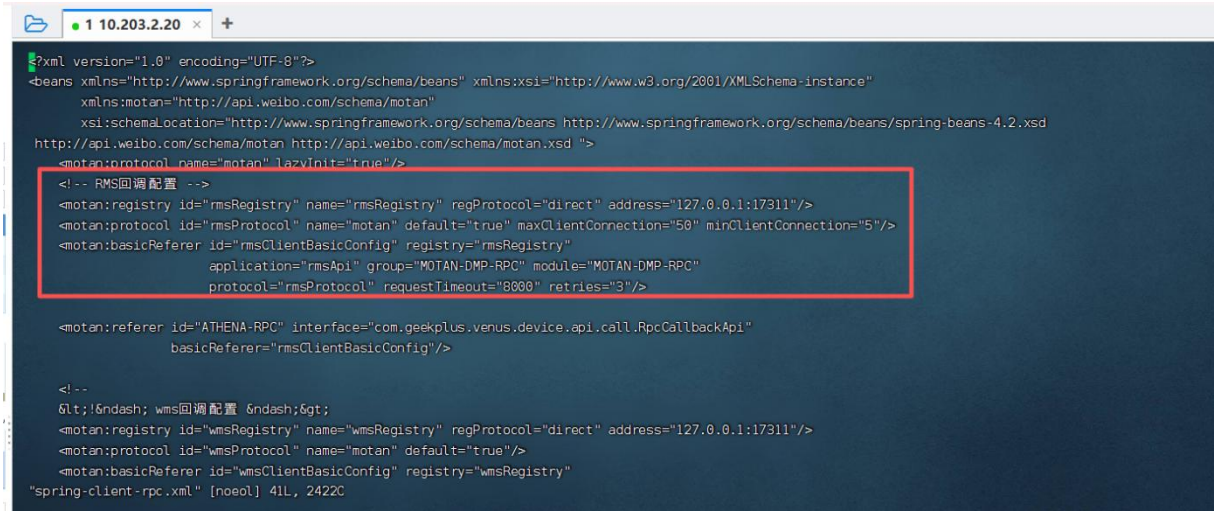
4.3.5 Configuration file modifications

The RMS and DMP systems need to enable mutual callback aisles, which are disabled by default.

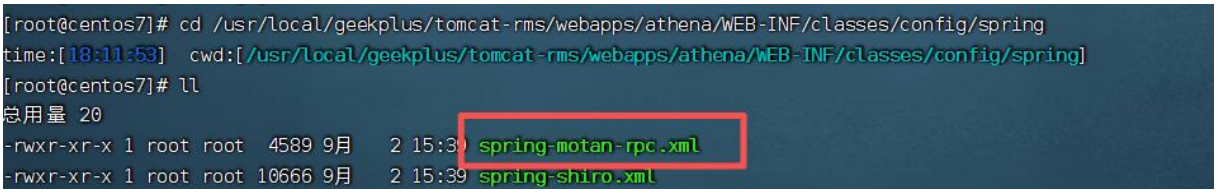
- DMP configuration file path: /usr/local/geekplus/tomcatvenus/webapps/venus/WEB-INF/classes/config/spring
- Configuration file name: spring-client-rpc.xml

```
time:[18:05:24]  cwd:[/usr/local/geekplus/tomcat-venus/webapps/venus/WEB-INF/classes/config/spring]
[root@centos7]# ll
总用量 20
-rwxr-xr-x 1 root root  2422 9月   2 15:39 spring-client-rpc.xml
-rwxr-xr-x 1 root root  1902 9月   2 11:09 spring-rpc.xml
-rwxr-xr-x 1 root root 10002 9月   2 15:39 spring-shiro.xml
time:[18:05:26]  cwd:[/usr/local/geekplus/tomcat-venus/webapps/venus/WEB-INF/classes/config/spring]
```

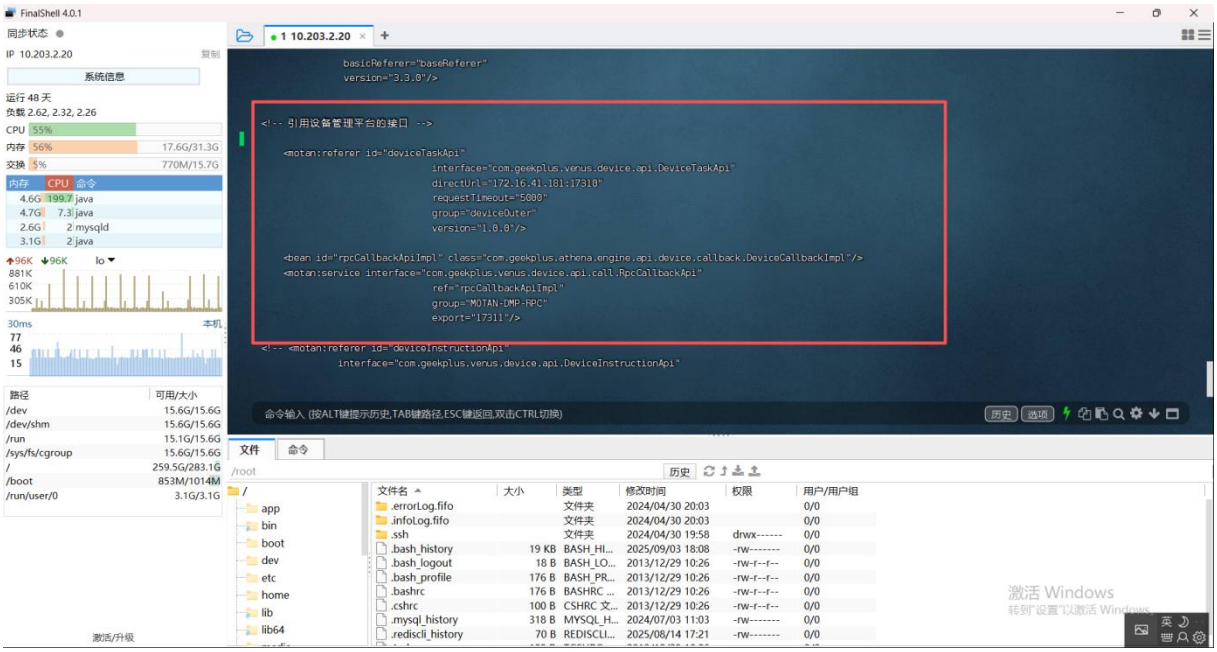
- Edit the current file: use vi to delete the corresponding comments and wq to save the file.



- RMS configuration file path: /usr/local/geekplus/tomcat-rms/webapps/athena/WEB-INF/classes/config/spring
- Configuration file name: spring-motan-rpc.xml



- Edit the current file: use vi to delete the corresponding comments and wq to save the file.



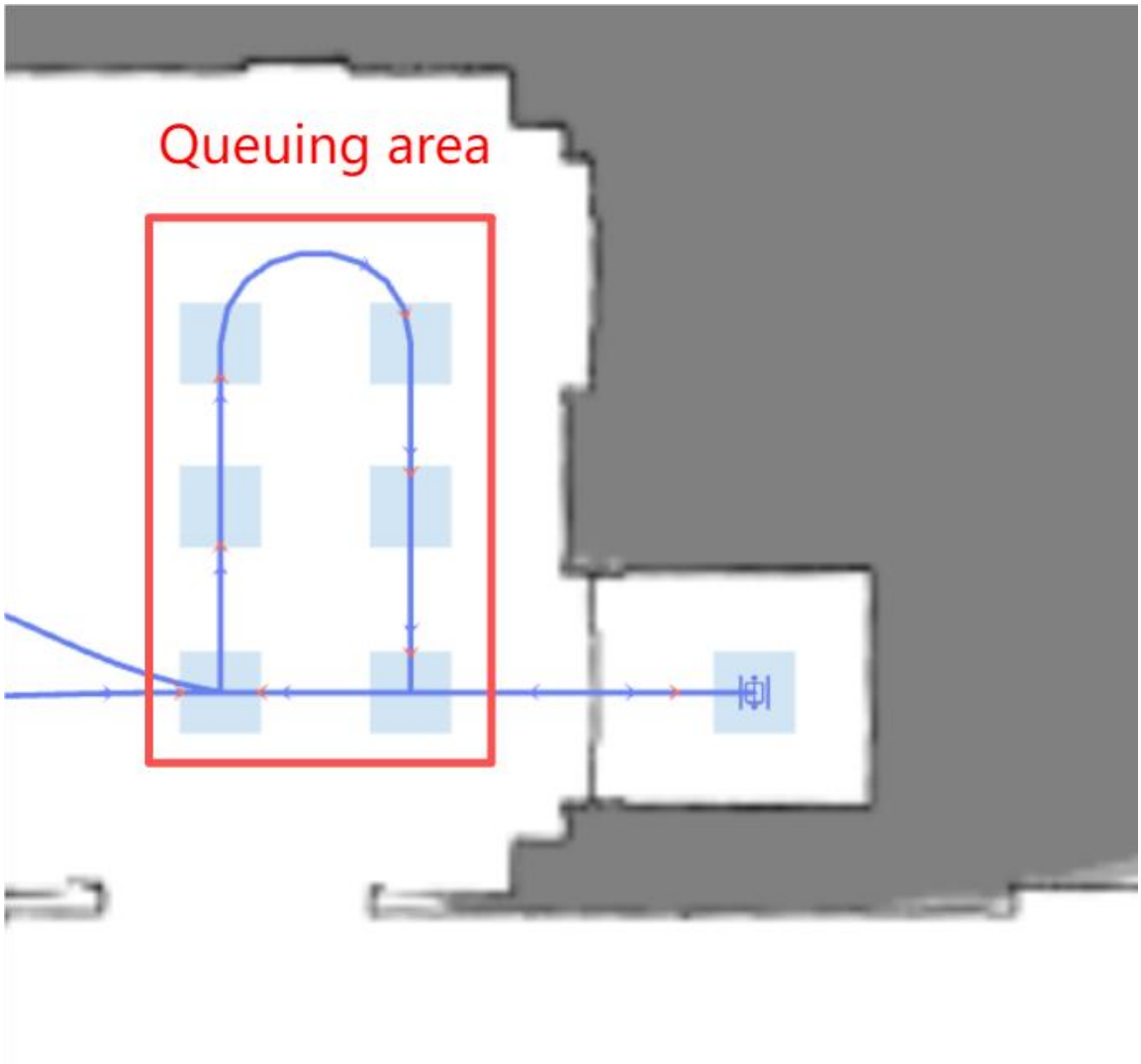
After modifying the configuration file, restart the RMS and DMP services.

5 Common processing logic for compact PLCs

- Switching the elevator to AGV mode: Some elevators feature an AGV mode. To minimize DMP-related configuration tasks, when the PLC receives a "hall call request", it must integrate with the elevator control module to switch the elevator to "AGV mode". Once the AGV issues a departure command, the elevator reverts to "manual mode".
- Exception handling: During task execution, the AGV may need to process commands sent by the elevator or utilize the elevator's "one-click restore" feature if manually removed at various integrate stages.
- Clear: The AGV output command requires clearing the historical hall call commands and corresponding floor commands after the AGV leaves the elevator. This ensures the elevator interface status returns to a "zero" state for the next operation.

6 Key planning considerations

- The path is passable. Generally, elevators operate bidirectionally, with separate directions for entering and exiting. Therefore, space must be reserved at the elevator entrance to allow the AGV to exit the path. Elevator interaction points and queuing points should be positioned away from the elevator entrance.
- The path in front of the elevator requires reserved queuing and interaction areas.



- The recommended integrate time for the elevator within 3 floors is 80 seconds.
- Multiple vehicles moving between floors require increased queuing time based on the logistics cycle.

Geek+

MOVING THE WORLD INTELLIGENTLY

Geekplus Technology Co., Ltd.

www.geekplus.com

400-045-0010

sales@geekplus.com