

Hermes

General Purpose Robot Platform

UserManual

- Small- to medium-sized robot development
- Highly adaptable and scalable
- Powerful optional functions

[Learn more >](#)



CONTENTS

1.Introduction	4
2.Introduction to Hermes Features	5
2.1 Autonomous Path-Finding	5
2.2 Robot Collaboration	5
2.3 Cross-Floor Delivery.....	5
2.4 Smart Obstacle Avoidance	5
2.5 Comprehensive Safety	5
2.6 Autonomous Recharging.....	6
3.Hermes Schematic Diagram	7
4.Hermes Charging Stand	8
5.Parameters.....	9
6. Instructions Of Hardwares.....	11
6.1 Wire Instructions.....	11
6.2 Important Interfaces of Developing Board on RK3399	13
7. Setup Instructions	13
7.1 Opening	13
7.2 Placing Charging Base	14
7.3 Power ON/OFF	15
7.4 Charging	15
7.5 Map Building and Upload	16
7.6 Host Computer Completes Startup and Loading The Map	18
8. Connection to Computer	19
8.1 Wired Connection.....	19
8.2 Wireless Connection.....	19
9. Adjusting Tools.....	19
9.1 RoboStudio	19
9.2 Web Management Backstage	20

10. Developer Tools	20
10.1 Start Using.....	20
10.2 Inter-system Adjusting Framework	20
10.3 Instructions of Each System	21
10.4 Demonstrations of Code	26
10.5 Details Of Robot API	27
11. Introduction And Use Of Elevator Control.....	29
11.1 Introduction	29
11.2 Features	29

1. Introduction

Developed by Slamtec, Hermes is a scalable and low-cost robot platform capable of meeting the needs of small-to medium-sized robot application development in areas such as smart patrol robots, container transportation robots, hotel delivery robots, food delivery robots, and more.

Autonomous Localization and Navigation

The built-in SLAMCUBE autonomous localization and navigation system kit enables path-finding and localization and navigation features to help robots figure out where they are, where they should go, and the best way to get there. Hermes is capable of performing a variety of tasks across different commercial environments.

Multi-Sensor Data Fusion

Multi-sensor data fusion technology is an effective way to improve the sensing capability of robotic systems. Hermes uses multi-sensor data fusion technology to effectively realize high degrees of autonomy and awareness in unpredictable environments. Fitted with equipment such as lidar, magnetic sensors, depth cameras, and bumper sensors, Hermes can implement autonomous mapping, localization, and navigation by flexibly responding to complex and ever-changing operational environments.



2. Introduction to Hermes Features

2.1 Autonomous Path-Finding

The built-in SLAMCUBE autonomous localization and navigation system kit enables path-finding and localization and navigation features to help robots figure out where they are, where they should go, and the best way to get there. Hermes is capable of performing a variety of tasks across different commercial environments. Hermes is capable of autonomous localization and path-finding in accordance with task target points, implementing autonomous movement.

2.2 Robot Collaboration

Hermes supports multi-robot collaboration to meet the needs of multi-task operations in complex environments during peak periods. Hermes supports local area network (LAN) and cloud platform collaborative operations, along with the dynamic adjustment of both speed and delivery routes in accordance with the environment to realize efficient, safe, and reliable multi-point delivery.

2.3 Cross-Floor Delivery

The Smart Elevator Control 3.0 system provides accurate detection of elevator statuses along with call-control functionality to provide efficient and reliable solutions that help hotel delivery robots autonomously navigate elevators in cross-floor scenarios. Through our API interfaces, Smart Elevator Control 3.0, as a relatively independent set of solutions, can also be integrated with other smart devices or third-party applications to meet the differentiated customization needs of customers.

2.4 Smart Obstacle Avoidance

The multi-sensor fusion technology of Hermes provides rapid and accurate identification of surrounding active environments, enabling smart obstacle avoidance and greatly reducing the chances of safety incidents.

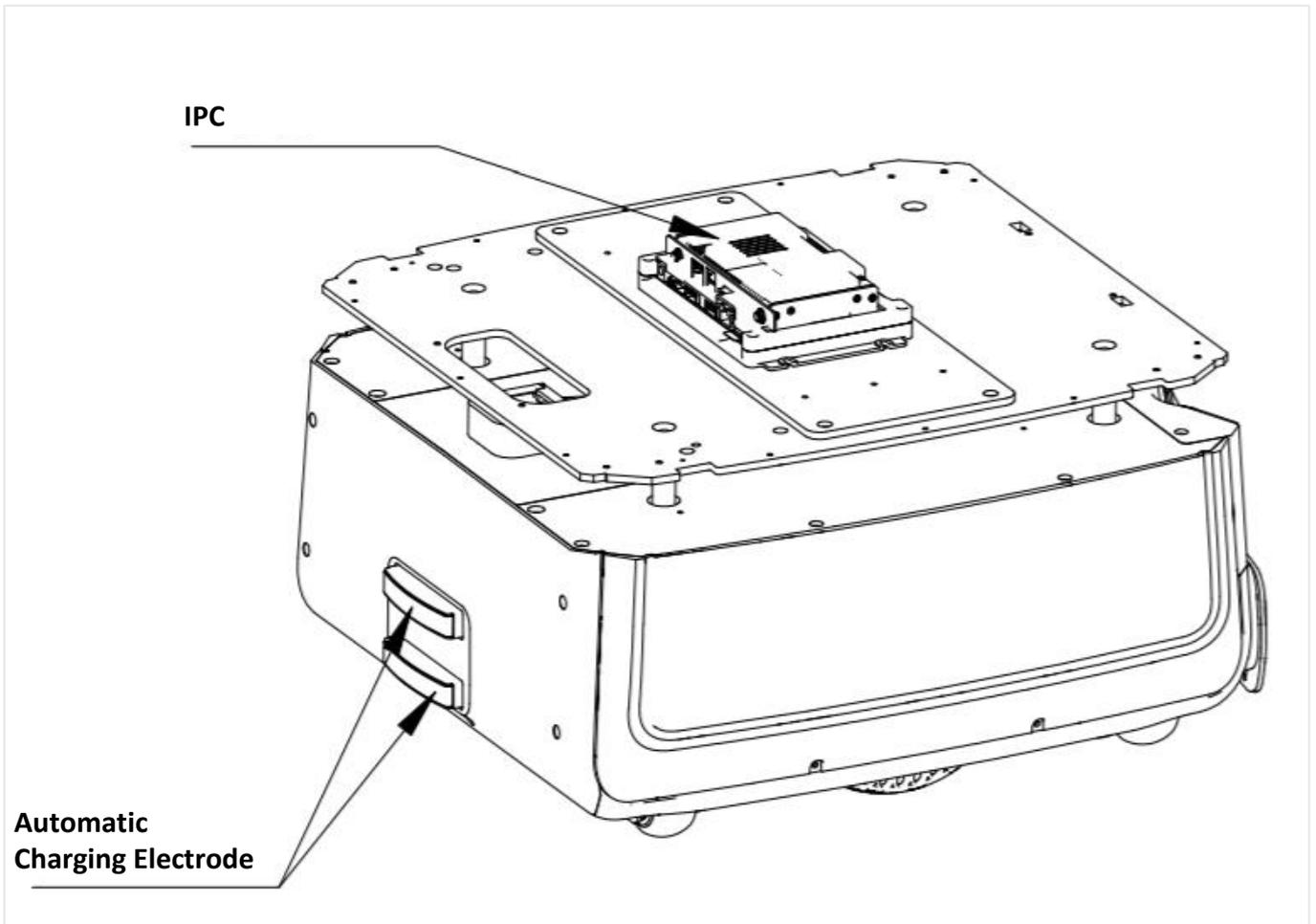
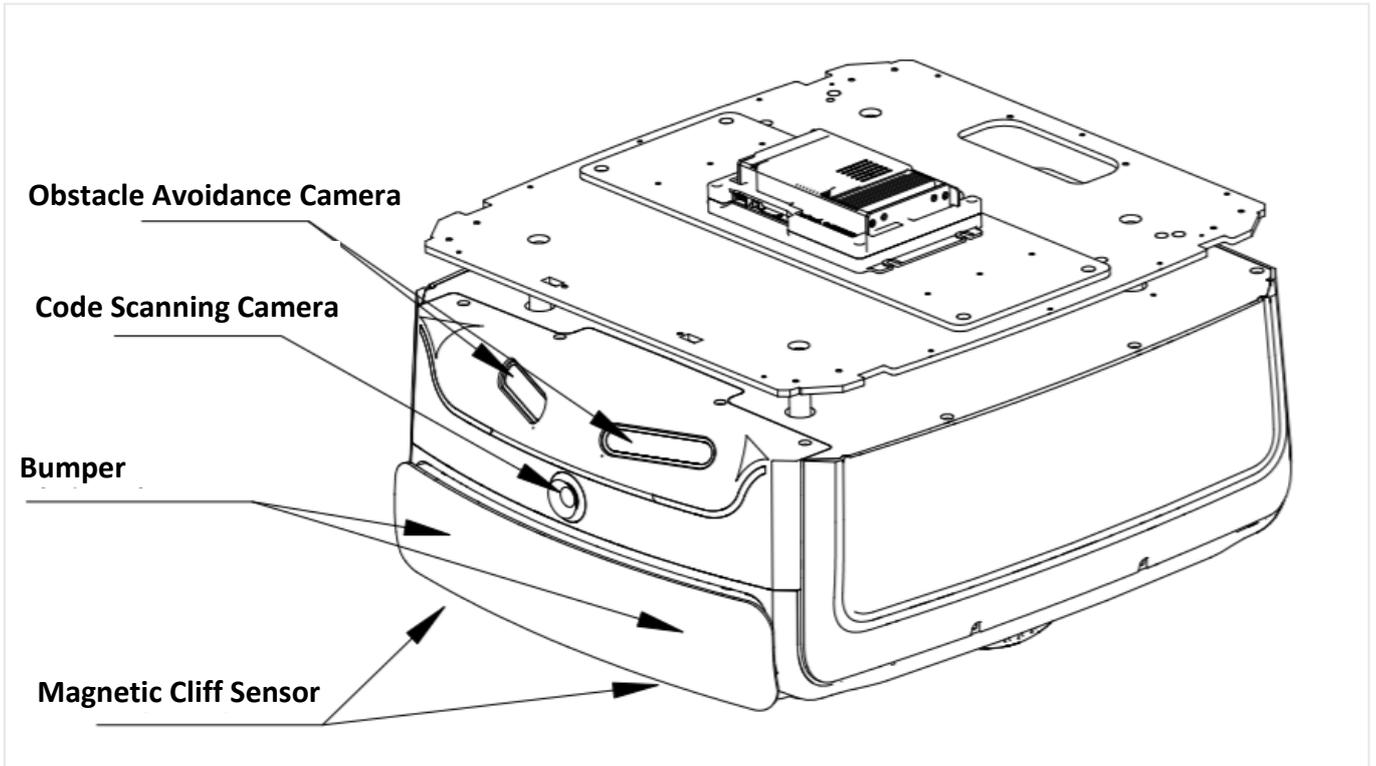
2.5 Comprehensive Safety

Hermes uses multi-sensor fusion methods such as dual-depth magnetic sensors, bumper sensors, and lidar to accurately identify and avoid both moving and static obstacles. The robot also supports fall avoidance, collision avoidance, and emergency braking functionality to ensure comprehensive safety during the move process.

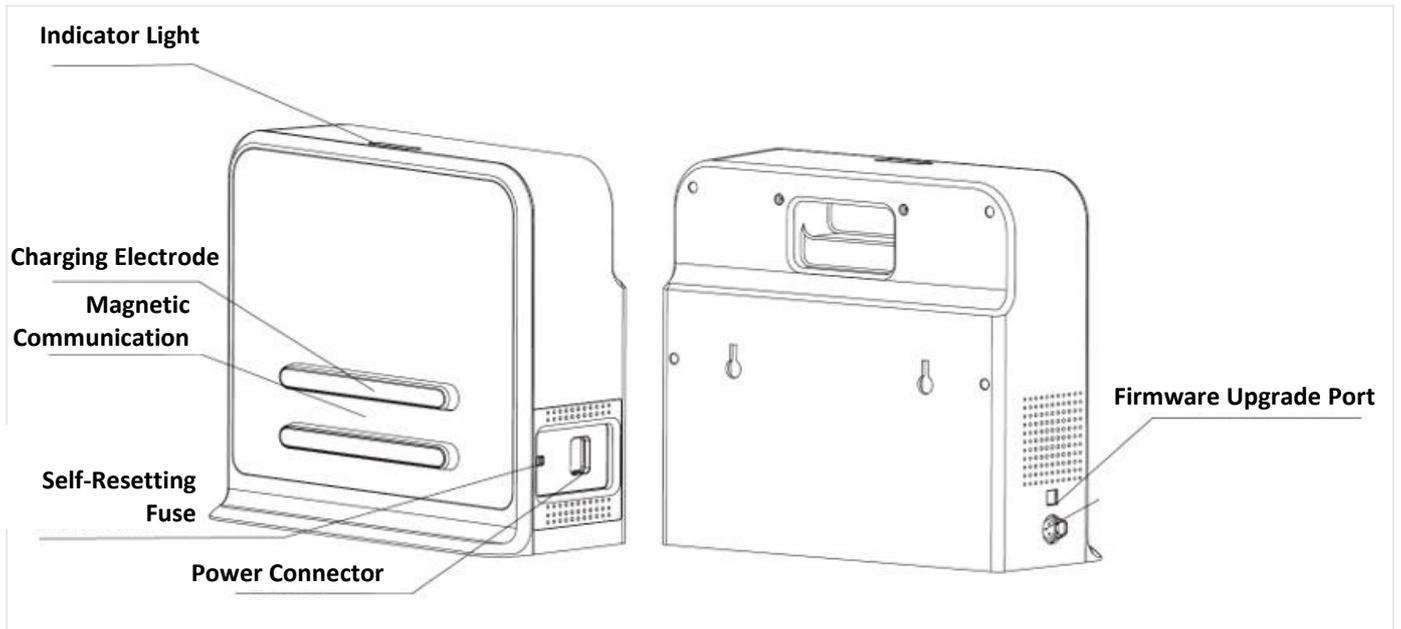
2.6 Autonomous Recharging

Sufficient power is ensured for Hermes to complete assigned tasks smoothly. Hermes automatically returns to the charging station when its battery charge falls below a configured threshold or when its tasks are completed.

3. Hermes Schematic Diagram



4. Hermes Charging Stand



5. Parameters

Name		Hermes	
Core Features		SLAM Localization and Navigation	
Item		Index	Value
Mass and Volume		Length x Width	545 x 465 mm
		Height	272 mm (excludes controller)
		Weight	40 kg (excludes controller)
		Max Load	60 kg
Sensor Performance Parameters	Lidar Sensor	Max Scan Radius (90% Surface Reflective Rate)	25 m, 360° (standard) 30 m, 360° (Optional)
		Depth Camera Sensor	Quantity
	Detection Range		0.2m-10m
	Field of View (FOV)		Single H: 87° V: 58° Double H: 168° V: 58°
	Boundary Sensor	Quantity	2 units
		Max Detection Range	3 cm
	Bumper Sensor	Quantity	2 units
		Trigger Method	Physical Collision
Mapping Performance		Map Resolution	5 cm
		Max Mapping Area	500m x 500m
Motion Parameters		Max Move Speed	1.2m/s (Can be customized 1.5 m/s)
		Max Cross Slope	16°
		Passing Obstacle Height	2 cm
		Passing Obstacle Width	4 cm
User Port	Hardware Port	Ethernet	1x RJ45 Gigabit Ethernet Port
		Power Connector	DC 12 V

		Wi-Fi	2.4 GHz/5 GHz
		4G	1x 4G
		SIM card slot	1 unit
		HDMI	1x HDMI
		Audio	1x 3.5mm Headset Socket 1x LINE_MIC Audio Jack (Co-lay with headset socket) 1x Dual-Channel 5w/8Ω Amplifier Jack
	Type-C	Standard USB 3.0 Type-C Port	
	Software API	SLAMWARE™	
Battery and Capacity	Capacity Specifications	16 Ah (Standard) 25 Ah (Optional)	
	No-Load Operating Time	9.5h (16Ah) 15h (25Ah)	
	Charging Time	3h-4h	
	Power Dissipation in Stand-By Time	24W	
	Battery Life	800 charges	
Power Consumption	Standby power consumption	32W (16Ah, No-load)	
	Motion power consumption	39W (No-load)	
	Full load power consumption	31W (Standby) 62W (Motion)	
	Maximum access power consumption	126W	
Wheelset	Wheelset Parameters	6.5-inch In-Wheel Motor	
Operating Environment	Charging Station	DC25.5V 6A	
	Operating Temperature	-10°C to +50°C	
	Operating Humidity	5 to 90%rh (no condensation)	

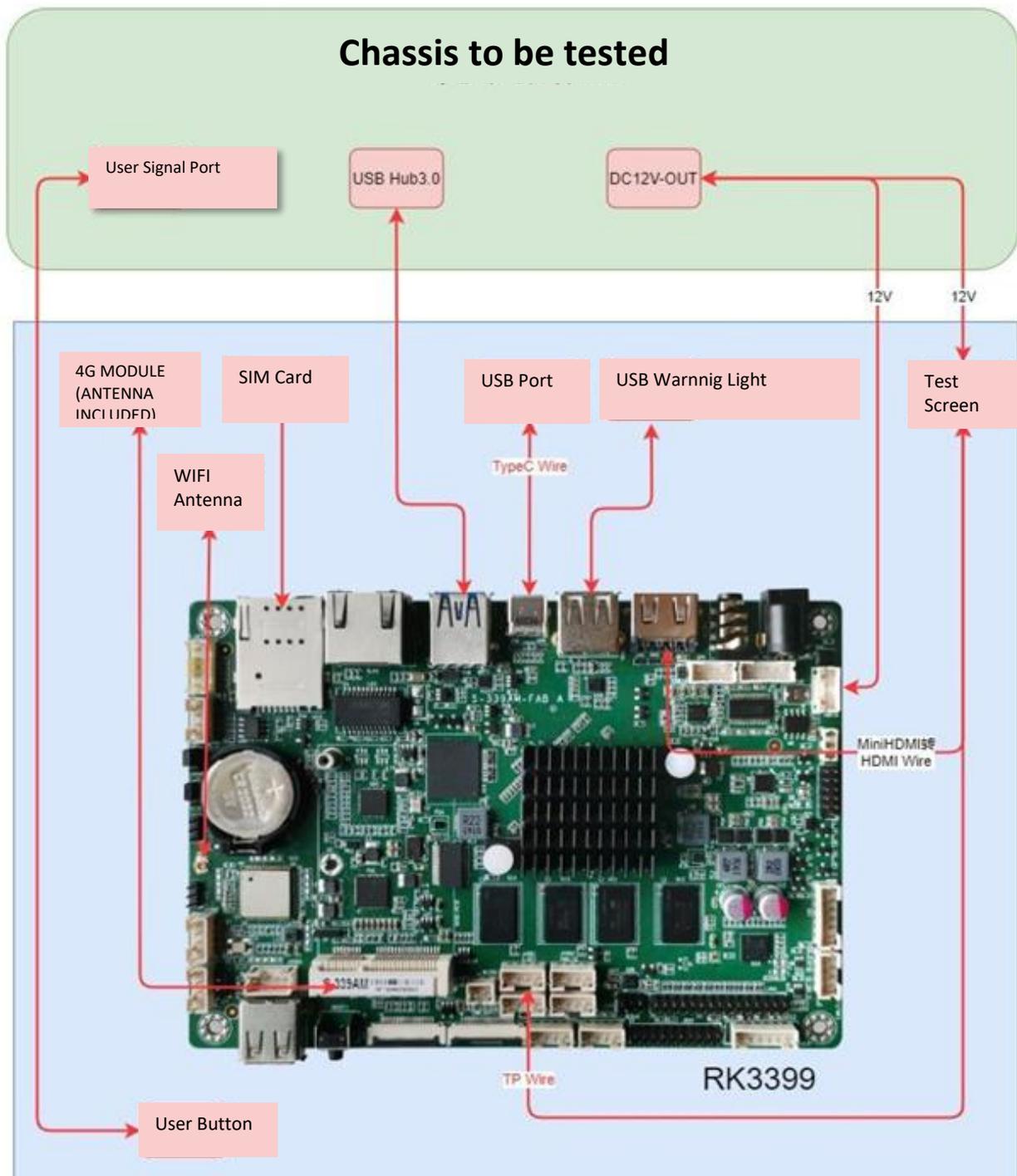
6. Instructions Of Hardwares

6.1 Wire Instructions

Interfaces	Picture	Index	Function	Note
Power ON/OFF		--	Power switch of the entire system. Short press until the indicator light turns on power-on, long press until the indicator light turns off power-off.	--
Shut Down		--	Effectively control Hermes in an emergency. After pressing, Hermes will immediately shut down.	It is a switch that must be designed according to rigid regulations. If the switch is not designed, Hermes will refuse to run.
Brake		--	When robot is energized, press this switch to release the brake and you can push the robot freely; press it once again to brake, and the robot will not be able to push it freely.	Invalid if the robot is not turned on
24V Power		Rated Voltage 24V Maximum current 20A	Maximum output power is 480W, can be used to supply power to external devices	No overload
12V RK3399 Power Supply Wire		Rated Voltage 12V Maximum Current 20A	Cube2 supplies power to the RK3399 development board.	

Depth Camera Interface		USB3.0	Connected to the RK3399 development via USB3.0 to obtain information.	Need to pay attention to the left and right sequence.
Docking Camera		USB2.0	Connected to the RK3399 development via USB3.0 to obtain information of code scanning camera.	Assist precise docking.
12V -DISP		Rated Voltage 12V	12V working power of Cube2 to provide the external display screen of human-computer interaction.	
USB-Type-C		USB3.0	Used for communication between 3399 and Cube2.	Connect to Hub3.0

6.2 Important Interfaces of Developing Board on RK3399

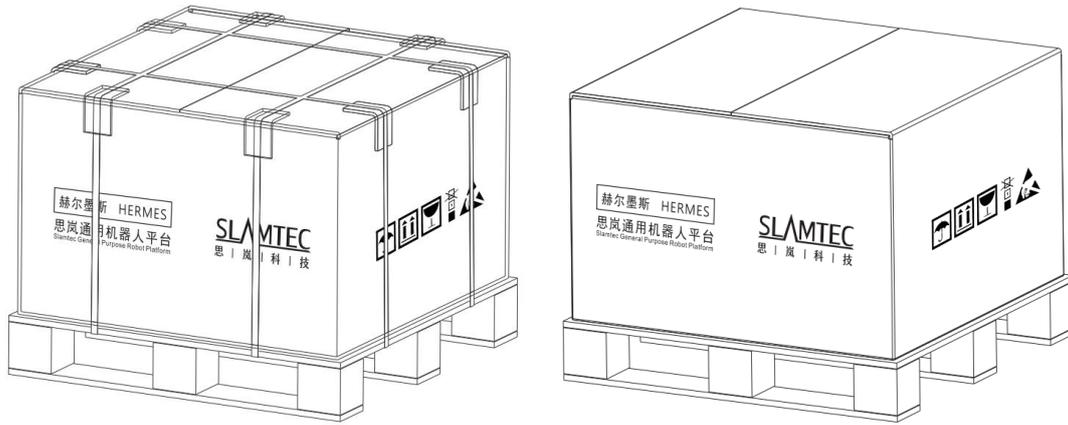


7.

Setup Instructions

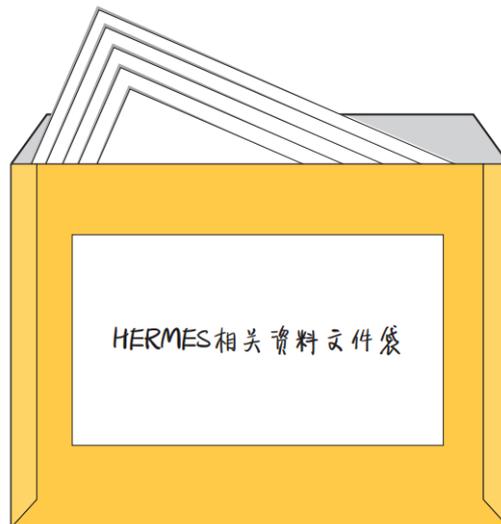
7.1 Opening

1. After receiving the Hermes machine, please check whether the packing box is intact as shown in the left picture, and whether the box on the right side of the Hermes sample of the outer box is checked. After confirming that the packing is complete and the box contains Hermes, use packing pliers to cut the packing tape, remove all packing straps and paper corners, it should be shown as on the right picture.



7-1 Hermes unpacking

2. Remove the document bag on the top of the box. The document bag contains Hermes-related paper materials, such as parameter descriptions and quick start guides, please keep them properly.



7-2 Hermes document bag

7.2 Placing Charging Base

Hermes can be charged by returning to the charging base automatically, so the position of the charging base will affect the automatic return charging function of Hermes. When Hermes returns to the charging station, it will generate propelling force. Therefore, it is best to place the charging stand with its back against the wall. The wall must meet the following conditions:

- The charging base must be attached to the wall, without no obstacle in the middle, try to avoid the wall with skirting, etc.
- The wall material cannot be high-permeability materials, such as mirror or glass
- The wall width needs to be at least three times wider than the width of the charging stand
- The wall must be a straight wall, not a curved wall

The charging base needs to be connected to a 220V power supply. The length of the external power cord of the charging base is 1.5m. Therefore, it is necessary to ensure that there is a 220 V interface within

1.5m of the wall against which the charging base rests. The ground wiring harness is messy causing unnecessary trouble).

The ground in front of the charging base must meet the following conditions:

- a. Open, with no obstacles within a radius of 2m in front of the charging stand as the center of the circle
- b. Plane, no slope
- c. The ground cannot be covered with soft carpet that causes Hermes to sink more than 2cm

To ensure that Hermes automatically returns to the charging function, it is necessary to ensure that Hermes always starts from the charging base.

Note: If you want to change the position of the charging stand, you need to set the corresponding setting according to the position you need when loading the map. For detailed operations, please see the SDK corresponding interface operation document.

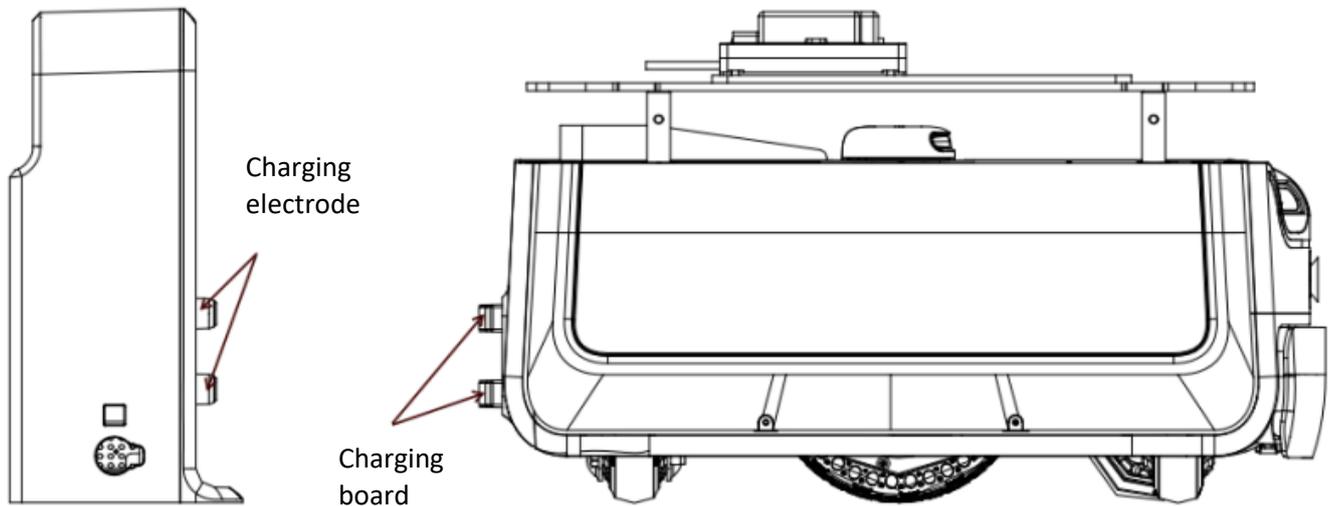
7.3 Power ON/OFF

Hold the power switch until the power switch indicator lights up and release it. The machine system enters the startup process, the front light strip lights up, and the wheel hub is in the brake state, indicating that the startup is complete.

Hold the power switch until the power switch indicator light goes out, then release it, the machine enters the shutdown process, the light strip on the front of the machine goes out, and the machine can be pushed arbitrarily, indicating that the shutdown is complete.

7.4 Charging

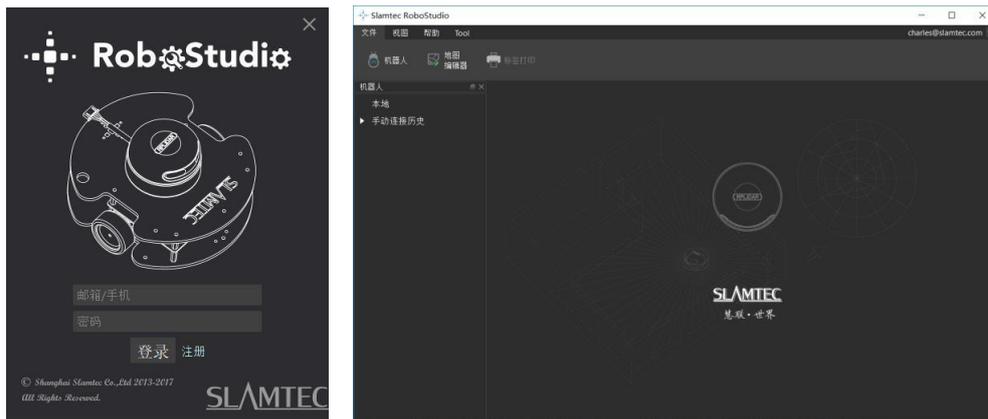
As shown in the figure below, align the charging piece of the machine with the charging electrode of the charging pile. After waiting for 10 seconds, the front light strip of the robot lights up, the wheel hub enters the braking state, and the breathing light of the charging pile flashes, indicating that charging has started.



7-3 Hermes charging

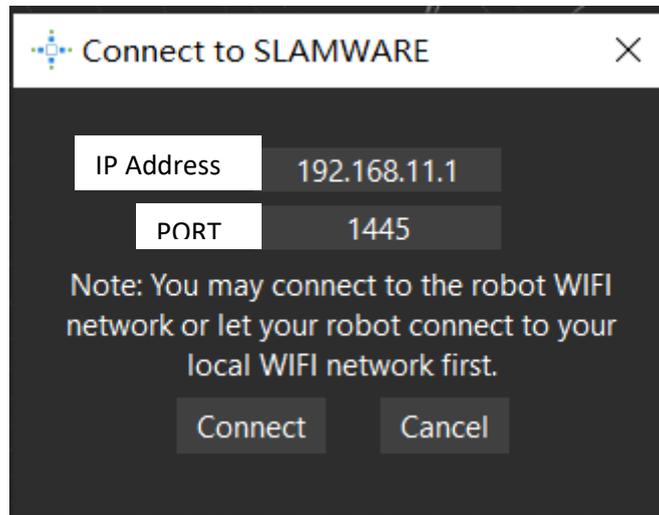
7.5 Map Building and Upload

Download and install the robot graphical control management tool Robot Studio from our official website <http://www.slamtec.com/cn/RoboStudio> to adjust and use Hermes, sign up account and log in. In the menu-toolbar area, click "File-Robot" in turn, and a docking page named Robot will appear on the left side of the workspace, as shown in the figure below. The user connects/disconnects the robot through this page.



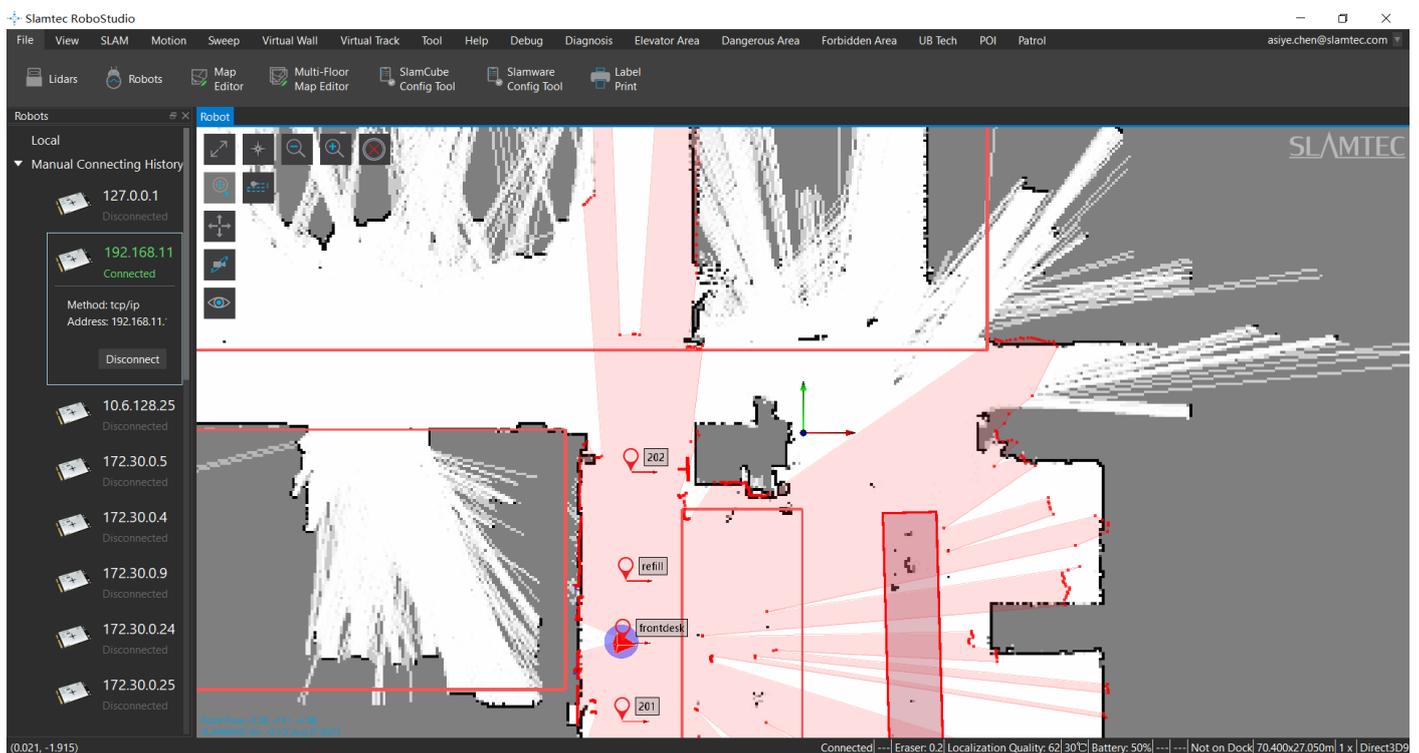
7-4 RoboStudio Robot page

Right-click the blank space of the robot list, click "Manually connect robots" in the pop-up menu, and the connect robot dialog box will pop up (as shown below). Enter the IP address and port number of the Hermes robot above and click the connect button to connect (This connection method is suitable for machines that have been assigned an IP address through the Web portal management backstage). When the computer has been connected to the SSID of the aforementioned Hermes robot through a network adapter (SSID can be viewed on the label of the machine), enter the IP address 192.168.11.1 by default (note that your wireless network adapter IP address should be set to automatically obtain it using DHCP).



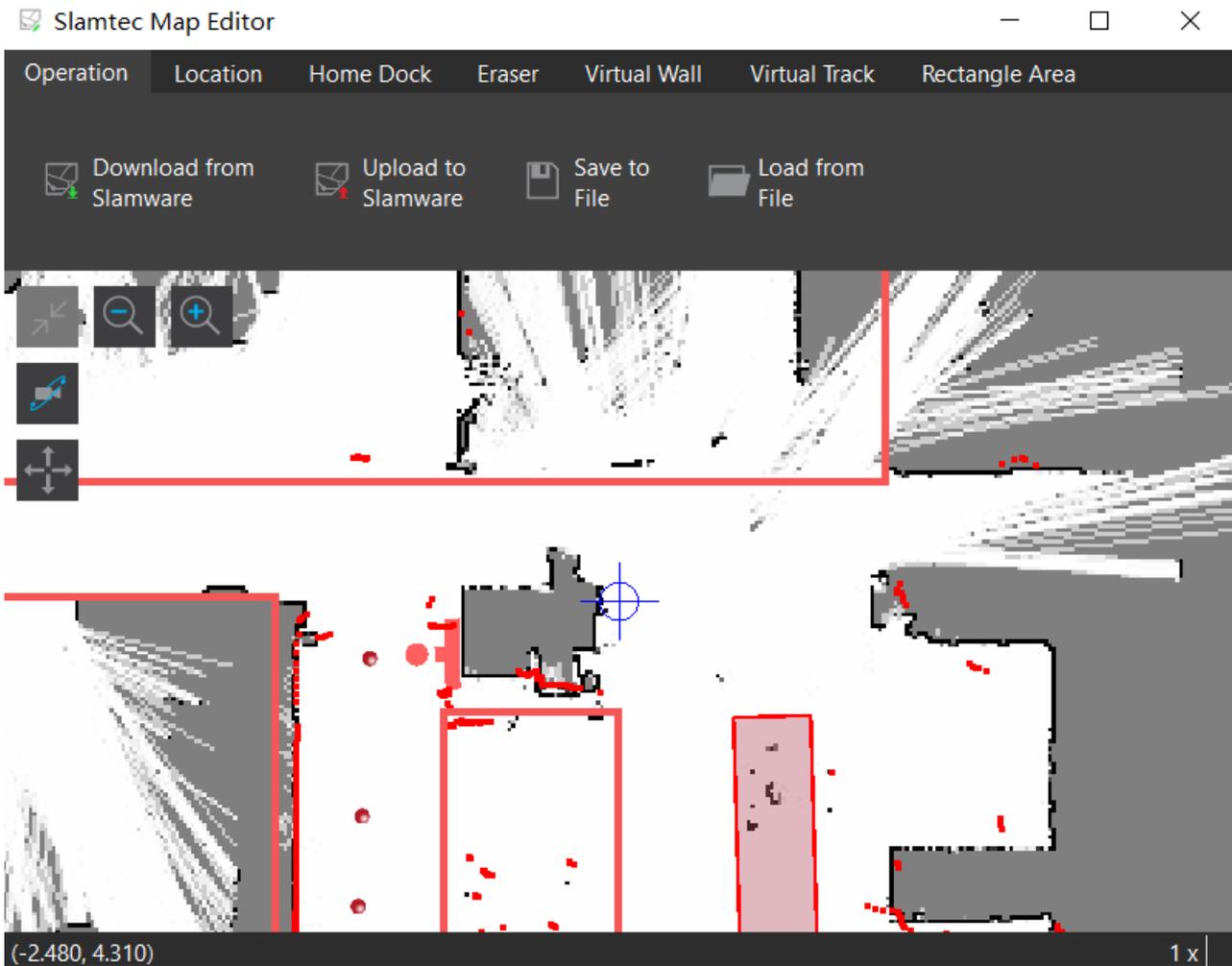
7-5 RoboStudio Robot connecting page

When finishing connecting, the work area will display the robot, map information, and status information. In the robot page, you will find that the name of the connected device turns green, and the status is "Connected", as shown in the figure below.



7-6 RoboStudio Robot connecting page

Left click the spot inside the map-building area, let Hermes build the map. When finishing, please use the virtual wall function to isolate the places that need to be isolated. Then click "File-Map Editor" in turn, choose to save the file or upload to the firmware to save the map.



7-7 RoboStudio Robot map-editing page

7.6 Host Computer Completes Startup and Loading The Map

Below is a reference example of how to use the SDK to complete the startup and loading of a specific map on the host computer.

```

1.  /*
2.  * upload map file
to Hermes 3. */
4.  #include <iostream>
5.  #include <rpos\robot_platforms\slamware_core_platform.h>
6.  #include <rpos\robot_platforms\objects\composite_map_reader.h>
7.  #include <rpos\core\pose.h>
8.
9.  using namespace std;
10.
11. int main()
12. {
13.     try
14.     {
15.         string map_path = ".\\map.stcm"; //the path of map
16.         string Hermes_ip = "192.168.11.1"; //the ip of Hermes
17.         int Hermes_port = 1445; //the port of Hermes ,default is 1445
18.
19.         rpos::robot_platforms::SlamwareCorePlatform Hermes =
20.             rpos::robot_platforms::SlamwareCorePlatform::connect(Hermes_ip, Hermes_port);
21.         //connect to the Hermes
22.         rpos::robot_platforms::objects::CompositeMapReader cmapreader;

```

```

23. //map reader
24. rpos::core::Pose Hermes_pose = rpos::core::Pose(rpos::core::Location(0, 0, 0));
25. //the Hermes pose in map(Hermes_pose should be the Hermes's real pose in new map)
26. //using Hermes.getpose() to get the old Hermes pose
27. auto map = cmapreader.loadFile(map_path);
28. //load map
29. Hermes.setCompositeMap(*map, Hermes_pose);
30. //set compositemap
31. rpos::core::Pose home_pose = rpos::core::Pose(rpos::core::Location(0, 0, 0));
32. //the home pose in map(home_pose should be the home's real pose in new map)
33. //using Hermes.gethomepose() to get the old home pose
34. Hermes.setHomePose(home_pose);
35. //set home pose
36. }
37. catch (rpos::robot_platforms::ConnectionFailException &e)
38. {
39.     cout << "connect failed on " << e.what() << endl;
40. }
41. catch (rpos::system::detail::ExceptionBase &e)
42. {
43.     cout << "failed on " << e.what() << endl;
44. }
45.
46. return 0;
47. }

```

Further motion deployment details, please see the SDK instruction document

8. Connection to Computer

8.1 Wired Connection

Based on the actual development and application needs, the RJ45 universal network port is designed on the expansion interface layer of Hermes, which can be directly connected to the PC through the network cable to facilitate users to expand or develop robots.

8.2 Wireless Connection

In order to facilitate users to debug and meet the needs of actual scene applications in the later period, Hermes also supports wireless connections. Users can directly use their computer to connect to Hermes' SSID and then use the default IP address 192.168.11.1 to connect to Hermes (IP address should be set to be automatically obtained using DHCP), and it also supports the use of Portal management tools to assign a specific IP address to Hermes then use that IP address to connect to Hermes.

9. Adjusting Tools

9.1 RoboStudio

RoboStudio graphical tools are used to adjust and use Hermes, please download and install it from our official website: <http://www.slamtec.com/cn/RoboStudio>, and coming with manual to introduce those features.

9.2 Web Management Backstage

In the process of developing, adjusting and using SLAMWARE equipment, various operations can be performed on the equipment through the web management backstage, such as viewing basic information, updating the version, configuring WiFi, etc. (Default username: admin, default password: admin111)

Currently, web management backstage supports following functions:

1. Check basic information of the device
2. Restart the module
3. Update the firmware

Slamtec will regularly provide Hermes firmware updates and upgrades. You can easily upgrade the firmware for Hermes through the management backstage. Please obtain the latest firmware from sales or technical support personnel. The update process will last 5-10 minutes. The buzzer of Hermes will continue to sound during the update, and Hermes will automatically restart after the update. Before that, please make sure that Hermes has sufficient power.

4. Configuring WIFI
5. Start the SLAMWARE Core diagnostic mode
6. Modify the administrator password

More details on usages please refer to :

<https://wiki.slamtec.com/display/SD/SQ001+SLAMWARE+Web+Portal+Function+Overview>

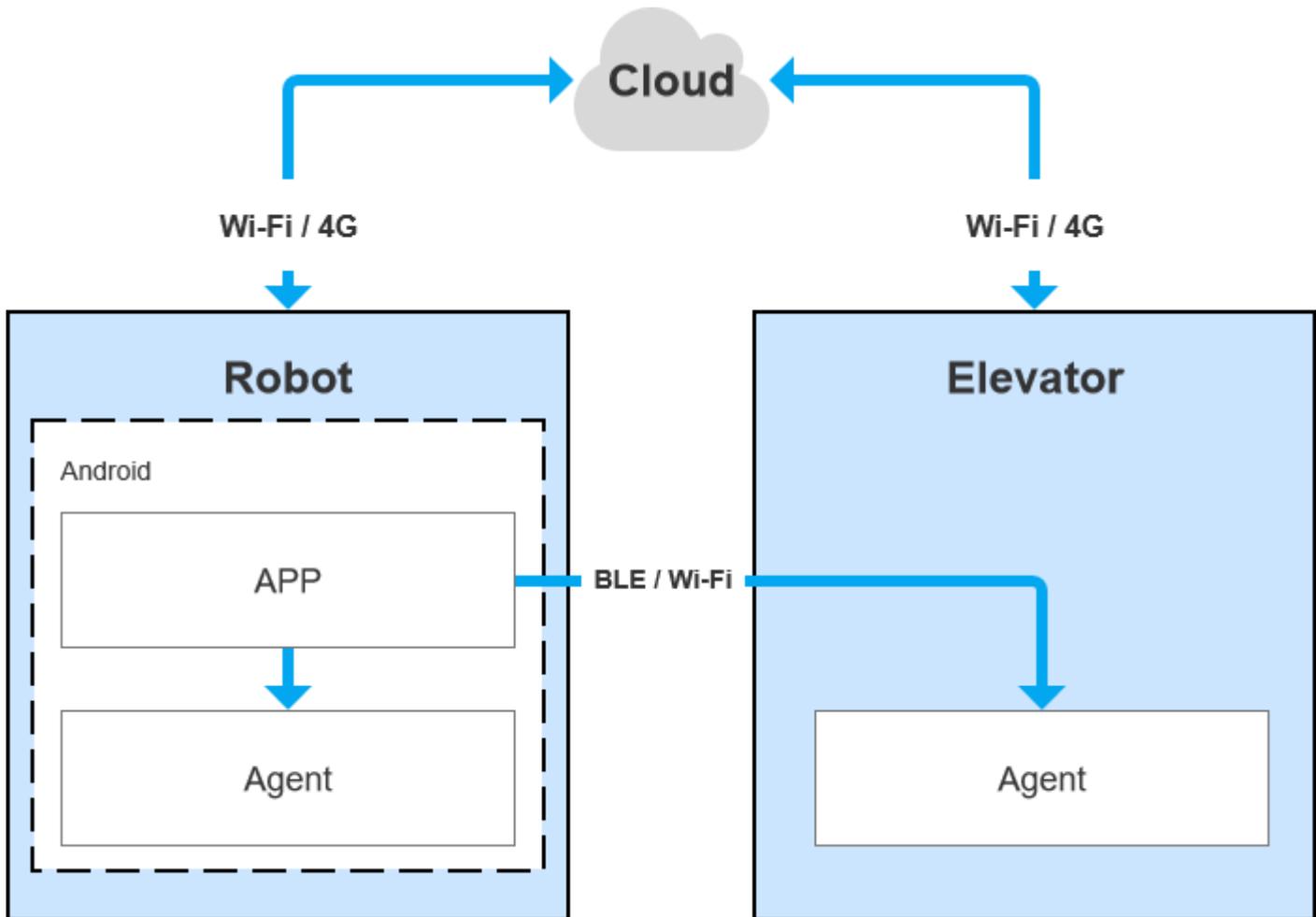
10. Developer Tools

10.1 Start Using

The Agent SDK of the Hermes chassis is developed based on the C++ language to reduce user access costs and improve the robustness of the SDK. At the same time, it has strong compatibility and supports multiple languages such as Java, C++, C, and Kotlin. The following is a detailed introduction to the relevant examples and usage guidelines of the Agent SDK call based on the Hermes chassis.

10.2 Inter-system Adjusting Framework

Robot App controls the robot's positioning, movement, and return to piles through communication; at the same time, Robot App sends instructions to the robot according to various business scenarios, and Robot Agent will provide data interfaces, task operation interfaces, and business services to Robot App.



10-1 COMMUNICATION BETWEEN EACH SYSTEM OF HERMES

10.3 Instructions of Each System

10.3.1 Robot Agent

Robot Agent is a service program running on the Hermes chassis, through which the cloud and the elevator control terminal communicate with the elevator control equipment. Inside the Robot system, the Robot Agent communicates with the Robot App, receives instructions from the Robot App to control the Robot, and sends the Robot status at the same time.

The core function of Robot Agent is:

- ✓ Communicate with the Robot Cloud, Robot APP, send the Robot status and receive instructions

10.3.2 Elevator Agent

Elevator Agent is a service program running on the Linux system of the elevator control main control box. The cloud and the robot end communicate with the elevator control equipment through it. Inside the elevator control system, the Elevator Agent communicates with the Elevator Controller through UDP, sending elevator control instructions and obtaining elevator status.

The core function of Elevator Agent is:

- ✓ Communicate with the Robot Cloud, Robot APP, send the elevator status and receive instructions of elevator control
- ✓ Communicate with the Elevator Controller, receive the elevator status and send instructions of elevator control

10.3.3 Robot Cloud

Robot Cloud is a set of services provided to realize remote management, scheduling, and control of robots. It runs in the cloud and communicates with the robot at one end and the elevator control equipment at the other end. Therefore, elevator control is part of the function. Robot Cloud communicates with the Elevator Agent running on the main control box of the elevator control through the MQTT protocol.

Its core functions include:

- ✓ Send instructions, control the elevator to the designated floor
- ✓ Send instructions, control the elevator to open the door
- ✓ Send instructions, control the elevator to close the door
- ✓ Receive the up/down status of elevator
- ✓ Receive the current floor of the elevator

10.3.4 Robot App

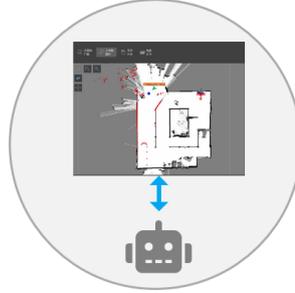
10.3.4.1 Open Source Application-Hotel Delivery APP

The hotel delivery app is a service program running on the robot. It draws a map and loads it locally on the robot through RoboStudio graphical tools. At the same time, combined with the intelligent elevator control developed by SLAMTEC, the robot can move up and down with the elevator autonomously and finally realize multi-floor movement.

DEPLOYMENT



1. Personnel installing SLAMTEC elevator control device



2. Personnel sketch multi-floor map and load it to the Robot.

DELIVERY PROCESS



3. Waiter receive the takeout of the customer, and send it to their room using Robot.



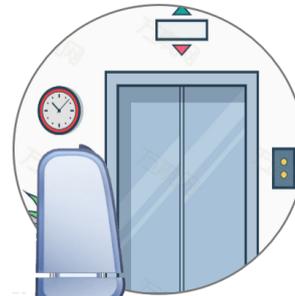
4. Waiter entering the process code



5. Click "delivery to room", Robot open the gate, waiter placing the takeout.



6. Waiter the room number, click "leave right now", Robot perform delivery task.



7. Robot come to the elevator, communicate with the elevator control device, Robot start calling elevator process, transfer to designated floor



8. Robot exit the elevator to come to the room of customer, customer click "confirm", Robot open the gate, customer take the takeout, Robot close the gate, return to the front desk, complete the delivery mission.

10-2 HOTEL DELIVERY BUSINESS FLOWCHART

Its core functions include:

- ✓ Get configuration information of the machine
- ✓ Check mission info
- ✓ Create mission
- ✓ Cancel all mission
- ✓ Cancel some mission
- ✓ Get status of current mission
- ✓ Pause/ continue the mission
- ✓ Start picking up items
- ✓ End picking up items

Its basic function is similar to Robot Cloud's elevator control function :

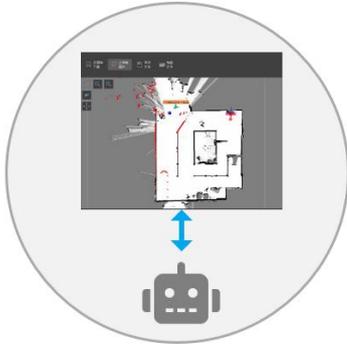
- ✓ Send instruction to control the elevator to designated floor
- ✓ Send instruction to control the elevator to open the door
- ✓ Send instruction to control the elevator to close the door
- ✓ Get up/ down status of elevator
- ✓ Get the current floor of elevator

NOTE: The hotel's multi-floor delivery solution needs to cooperate with the elevator control system, also needs to customize the cabin model. Please contact the marketing department for delivery business.

10.3.4.2 Open Source application – Restaurant Delivery App

The restaurant delivery app is a service program running on the robot. The application scene is a restaurant. It draws a map and loads it to the robot's local area through RoboStudio graphical tools. It is a universal application for human-computer interaction to achieve multi-point task delivery.

DEPLOYMENT



1. Personnel sketching map and load to the Robot

DELIVERY PROCESS



2. Restaurant waiter sets menu into the Robot



3. Waiter enters designated table number



4. Robot send items to the destination



5. Robot informing customer "Here's your order!" when reaching destination



6. Robot returning to picking up point and wait for another order

10-3 RESTAURANT DELIVERY BUSINESS FLOWCHART

Its core functions include:

- ✓ Get device battery percentage
- ✓ Get health status info of the device
- ✓ Get POI info
- ✓ Get password of the action
- ✓ Create new action
- ✓ Get current action
- ✓ Stop current action
- ✓ Shut down of restart Robot

10.4 Demonstrations of Code

1. The following is an example of the Robot APP calling the Robot Agent interface to check the battery status

GET <http://127.0.0.1:1448/api/core/system/v1/power/status>

The format of the returned data is application/json

```
interface AgentApi {
    /**
     * get方式调用 value: 接口地址
     * PowerStatus: 接口返回json格式对应的bean格式的data class的值
     */
    @GET("/core/system/v1/power/status")
    fun queryPowerStatus(): Call<PowerStatus>
}

/**
 * batteryPercentage : 90电池电量百分比, 0 ~ 100
 * dockingStatus : 对桩状态
 * isCharging : 是否正在充电...
 */
data class PowerStatus(
    val batteryPercentage: Int, val dockingStatus: String, val isCharging: Boolean,
    val isDCConnected: Boolean, val powerStage: String, val sleepMode: String
)

/**
 * retrofit接口代理类
 */
object AgentServiceCreator {
    fun <T> create(serviceClass: Class<T>, timeout: Long): T =
        Retrofit.Builder().baseUrl("http://127.0.0.1:1448/api/")
            .addConverterFactory(GsonConverterFactory.create()).client(
                OkHttpClient.Builder().retryOnConnectionFailure(true)
                    .connectTimeout(timeout, TimeUnit.SECONDS)
                    .addInterceptor { chain ->
                        val originalRequest = chain.request()
                        val requestBuilder =
                            originalRequest.newBuilder().addHeader("Connection", "close")
                        chain.proceed(requestBuilder.build())
                    }.build()
            ).build().create(serviceClass)
}

// 查询电量, 返回值为powerstatus
val powerStatus = AgentServiceCreator.create(AgentApi::class.java, 1L).queryPowerStatus().await()
    Result.success(powerStatus)
```

2. The following is the Robot App calling the Robot Agent interface to make the robot move across floors

POST <http://127.0.0.1:1448/api/core/motion/v1/actions>

The request message format is application/json

```

{
  "action_name": "slamtec.agent.actions.MultiFloorMoveAction",
  "options": {
    "target": {
      "poi_name": "201" //表示前往名称为 201的poi
    }
  }
}
} // JavaScript Document

```

10.5 Details Of Robot API

FUNCTION OF MODULES	API LIST
Locate, map building-related functions	Get the Robot location
	Set the Robot location
	Get quality of location
	Whether support locating
	Start/Close locating
	Whether start map building
	Start/ pause map building
	Get location of power station
	Set location of power station
	Get current map
	Delete current map
Mark map elements manually	Get all virtual line segment
	Add virtual line segment
	Adjust virtual line segment
	Delete virtual line segment
	Delete virtual line segment
	Get all POI in current map
	Add POI
	Empty POI
	Find POI according to ID
	Modify POI
	Delete POI
Motion control of the Robot	Get all supported actions
	Get current action
	Stop current action
	Create new action
	Check action status
Firmware Upgrade	Get firmware upgrade process
Android application management	Get all custom installed apps
	Install APP

	Uninstall an APP	
Multi-floor map and POI management, taking elevator, etc.	Move across floors	
	Go back across floors	
	Get all floor info	
	Get floor info of floor of the robot	
	Setup info of floor of the robot	
	Get POI info	
	Upload map to the robot	
	Long-lasting save the map	
	Reload the map	
	Delivery service-related interfaces	Get password of the action
		Set password of the action
Get configuration info of the device		
Get setup info related to the delivery		
Check task info		
Create task		
Cancel all task		
Cancel some task		
Get current mission status		
Stop/ continue current mission		
Start picking up items		
End picking up items		
Get info of objects		
NOTE: delivery business please inform marketing department		
Restaurant delivery service-related interfaces		Get the current battery percentage
	Get the health info of the device	
	Get POI info	
	Get password of the action	
	Make up new actions	
	Get current action	
	Terminate current action	
	Shut down or restart the robot	

MORE DETAILS OF API: <https://github.com/slamtec/HermesSampleApp>

11. Introduction And Use Of Elevator Control

11.1 Introduction



11-1 INTELLIGENT ELEVATOR CONTROL DEVELOPED BY SLAMTEC

Intelligent elevator control, independently developed by Shanghai SLAMTEC CO., Ltd, can provide accurate detection of elevator status, program control call elevators, control elevators, combined with Mercury II hotel robots, helping the robots to take and exit the elevators autonomously. Providing reliable solutions to multi-floor operation scenarios. In addition to working with hotel robots, the product can be used as an independent set of solutions. Through the API interface we provide, this product can also interact with other smart devices or products to meet the different customization needs of customers. At the same time, the product can also be seamlessly connected to the Hermes chassis, to provide technical service support for more specific needs in other scenarios.

11.2 Features

11.2.1 Intelligence Elevator Control

The core function of this product is to help robots and other intelligent devices to control the elevator, such as calling the elevator, going to the designated floor, and controlling the elevator to open and close the door. This function greatly improves the business scenario of robots and breaks the previous limitation that robots can only work on the same floor.

11.2.2 Intelligence Status Detection

Through the built-in sensor and adaptive algorithm, this product can accurately detect the current floor, up and down status of the elevator, real-time to the millisecond level, with an accuracy of more than 96%, providing a very critical foundation so that the robot can enter and exit the elevator correctly and complete the work.

11.2.3 Elevator Diversity Adaption

The main control box has developed a variety of optimized filtering algorithms, so that this product is suitable for many different types of elevators, such as single-door elevators, double-door elevators, front and rear door elevators, etc. As it covers as many scenes as possible, this product can be used in most elevators at present. Users only need to simply calibrate and adjust through the APP provided after installing the equipment.

11.2.4 Multi-Level Network Adaption

Considering the limitations of the elevator network, how to ensure data interconnection and program interoperability is a very critical factor, and relatively it is also a difficult point. After continuous optimization iterations, we finally found a way to solve this problem. This product provides 4G, WIFI, BLE and other multi-level network communication methods to interact with robots and other smart devices to ensure the reliability of the communication link.

11.2.5 OTA Remote Upgrade

Support OTA remote upgrade of software and firmware, and batch deployment. At the same time, it provides support for VPN, which can remotely log in to the Linux system of this product for upgrades and other operations. Through the OTA function, real-time updates, upgrades and optimization can be done.

11.2.6 Interface Support

The intelligent elevator control provides a consistent API interface to the outside world, supports Bluetooth, MQTT, and HTTP communication protocols. Customers can control and interact with the elevator by writing their own program code. At the same time, it supports secondary development to meet the different customization needs of customers.

11.2.7 Swipe To Bypass

For some elevators that require a swipe card to ride, this product can also be easily supported. Just install the equipment we specify when installing the device, and then turn on the swipe card to bypass it in the APP. In this way, even if there is a card reader, it can be easily bypassed and freely enter and exit the elevator.

11.2.8 Unified Management

The cloud platform provided by SLAMTEC can perform real-time monitoring of the status of all deployed products, batch upgrades, and further data mining through online management, information statistics, equipment monitoring, etc., to provide exclusive accurate data models for hotels or commercial buildings, to guide them to improve their overall operational capabilities and service quality.

To know more about the detailed solution of elevator control, obtain it from the marketing department. [Elevator control user manual](#)->>