Shenzhen HEISHA Technology Co., LTD

Connection Guidance of HEISHA DOCK On-Cloud API

Version: V2.6-20240104

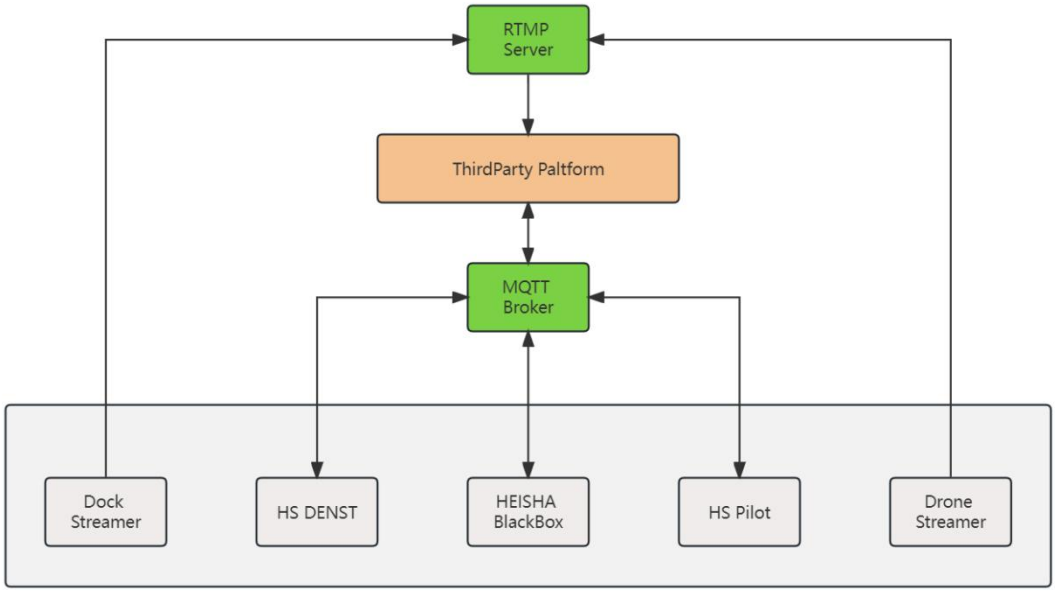
Version	Edit date	Edit content	Note
V2.1	20230906	1. HS Pilot Property report add takeoff point location, which includes altitude, UAV altitude can be obtained based on the takeoff location altitude superimposed on the relative height; 2. Add camera type to the HS Pilot Property report; 3. Modify the route sending to link form; 4. Add camera and video source direct selection command;	
V2.2	20231205	1. Add HS dnest event Post; 2. Add BlackBox takeoff interface;	
V2.3	20231226	1. Add results upload request interface to HS Pilot	
V 2.4	20231229	1. Add results upload interface in BlackBox	
V 2.5	20240103	1. Adding a Log Upload Interface to BlackBox	
V2.6	20240104	1. HS DNEST adds Property Post for weather station data	

Shenzhen HEISHA Technology Co., LTD

Catalog

I. Brief

HEISHA DOCK opens up the cloud API based on the MQTT protocol, and the communication links with third-party cloud platforms are shown in the figure below:



The HEISHA DOCK on Cloud API consists of 5 main components, which are:

Dock Streamer-used to push RTMP video streams from airport surveillance cameras

Drone Streamer-Used to push RTMP video streams from drone cameras

HS DNEST-Provides dock-related operational interfaces

HS Pilot-Provides the interface to the drone.

HEISHA BlackBox-Packages HS dnest and HS Pilot interfaces, providing some combined operation interfaces.

To interface with HEISHA DOCK on the third cloud platform, you need to prepare an MQTT Broker that supports MQTT version 3.1.1 and an RTMP server.HEISHA BlackBox provides a configuration interface for configuring parameters such as MQTT Broker address, port, username, password and RTMP push stream address.After successful configuration, HEISHA DOCK publishes status messages to the MQTT Broker.Third-party cloud platforms can connect to the MQTT Broker and subscribe to the Reply threads to obtain the real-time status of the drone docks and drones, and they can also send commands to HEISHA DOCK through the service requests of the published Replys.

II. Interface List

Each of the three components HS DNEST, HEISHA BlackBox, and HS Pilot opens a

set of stylized API interfaces with the interface subject format:

heisha/{component name}/{device unique code}/sender/interface name

The names of the three components are: dnest, bl a ck-box and pilot.

For the same device, all three components have the same device unique code.

The HEISHA DOCK on cloud API uses client and server to distinguish different senders, where HS dnest, HEISHA BlackBox, and HS Pilot all belong to client, and the third-party cloud platform is server.

Two interfaces are developed for each component: post - for status Post and service - for service delivery.

Each interface has a corresponding reply interface, post_reply for post and service_reply for service.

Specifically, the list of HEISHA DOCK on cloud APIs is as follows:

No.	Interface topic	Interface name
1	heisha/black -box/${unique device code }/client/post	BlackBox Property Post
2	heisha/black-box/${unique device code }/server/post_reply	BlackBox Property Post Reply
3	h eisha /black -box/${unique device code }/server/service	BlackBox Service delivery
4	heisha/black -box/${unique device code }/client/service_reply	BlackBox Service delivery Reply
5	heisha/dnest /${unique device code }/client/post	HS dnest Property Post
6	heisha/dnest /${unique device code }/server/post _reply	HS dnest Property Post Reply
7	heisha/dnest/${unique device code}/server/service	HS dnest Service delivery
8	heisha/dnest/${unique device code}/client/service_reply	HS dnest Service delivery Reply
9	heisha/dnest/${unique device code}/client/thing	HS dnest Incident Post
10	heisha/dnest/${unique device code}/server/thing_reply	HS dnest Incident Post Reply
11	heisha/pilot/${unique device code}/client/post	HS Pilot Property Post
12	heisha/pilot/${unique device code}/server/post_reply	HS Pilot Property Post Reply
13	heisha/pilot/${unique device code}/server/service	HS Pilot Service delivery
14	heisha/pilot/${unique device code}/client/service_reply	HS Pilot Service delivery Reply

III. Interface Details

3.1 BlackBox Property Post

Topic: heisha/black_box/${unique device code }/client/post
Payload:


```
{
    "direction": "up",
    "type": "post",
"group": 64,
    "data": {
        "dockModel": 1,
        "droneModel": 6,
        "workingMode": 0,
        "taskStep": 0,
        "stepStatus": 0,
        "taskStatus": 0,
        "photoTotal": 0,
        "upNum": 0,
        "taskId": "id-123asd",
        "pointNum": 0,
        "pointIndex": 0,
    }
}
```


Data parsing:

Parameters	Instruction
direction	Data direction (string): "up"-BlackBox-->third-party cloud platforms
type	Package type (string): "post" Property uploading
group	Property grouping (integer) (reserved)
data	Property Data
data :dockModel	Dock model (integer): 0-Unknown 1-D135 2-D80 3-D50 4-DCap 5-DCap Pro 6-R80s
data :droneModel	Drone model (integer):

	0-Unknown 1-Mavic Mini 3 2-Mavic Mini 3Pro 3-Mavic 3M 4-Mavic 3 Enterprise 5-Matrix 30 6-Matrix 300 RTK 7-Matrix 350 RTK
data :workingMode	Working mode (integer): 0-Free 1-Flight preparation completed 2-Pending takeoff 3-Manual flight mode 4-Point-to-point Mode 5-Waypoint Mode
data:taskStep	Task step (integer): 1-Open the dock 2-Release the centering bar 3-The drone is turned on 4-Ready to go 5-Rising 6-Flying 7-Returning 8-Landing Success/Failure 9-Results data uploading in progress 10-Closing the dock
data :stepStatus	Step status (integer): 1-Success 2-Failure
data:taskStatus	Task status (integer): 1-Preparation 2-Implementation 3-Success 4-Failure 5-Canceled
data :photoTotal	Total number of photos taken (integer)
data :upNum	Number of uploaded images (integer)
data :taskId	UUID of the currently executing task (string)
data :pointNum	Total number of waypoints (integer)
data :pointIndex	Index of the waypoint being flown to (Integer, starting at 0)

3.2 BlackBox Property Post Reply

Topic：heisha/black-box/${unique device code }/server/post_reply
Payload：

```
{
    "direction": "down",
    "type": "post"
}
```

Data parsing：

Parameter	Instructions
direction	Data direction (string): "down" - third-party cloud platform --> BlackBox
type	Package type (string): "post"-property uploaded

3.3 BlackBox Service Delivery

Topic：heisha/black-box/${unique device code }/server/service
Payload：

```
{
    "direction": "down",
    "type": "service",
    "code": 0,
    "params": {}  // optional
}
```

Parameter parsing :

Parameter	Instructions
direction	Data direction (character string): "Down" - -a third-party cloud platform- - -> BlackBox
type	Package type (character string): “service ” -Service Delivery
code	Service Code (integer): 0-Undefined 1-Point-to-point flight 2-One key preflight 3-One-click charging 4-Route distribution 5-Start the task 6-Pause the task 7-Restore tasks 8-Return to the dock 9-Return home to a safe landing site

	10-Local landing 11-Virtual joystick 12-Take-off to the specified altitude 13-Upload the outcome data 14-Upload the logs
params	Parameter (Json), configure different Parameter according to different services

Service details:

（1） Point-to-point flight

Function Instructions: After receiving the pointing flight mission, HEISHA BlackBox will control the UAV to fly from the current position to the specified point, and the flight altitude and flight speed can be configurable (see HS Pilot related interface).

Payload：

```
{
    "direction": "down",
    "type": "service",
    "code": 1,
    "params": {
        "uid": "taskID",
        "latitude": 22.5635654,
        "longitude": 118.7654357,
        "altitude " : 300.0
    }
}
```

Parameter parsing:

Parameter	Instructions
direction	Data direction (character string): "Down" - -a third-party cloud platform- - -> BlackBox
type	Package type (character string): “service ” -Service Delivery
code	Service Code (integer): 1: Point-to-point flight
params	Parameter (Json), configure different Parameter according to different services
params:uid	The unique code (string) is the UUID
params:latitude	Target latitude (floating point)
params:longitude	Target longitude (floating point)
params:altitude	Target altitude (floating point)

（2） One key preflight

Function Instructions: After receiving the one-key preflight command, HEISHA BlackBox will start the standby process. The standby process includes tightening the return lever and return the UAV to the center of the takeoff and landing platform, opening the UAV,

opening the remote control of the UAV, releasing the charging bar, and opening the dock cover.

Payload:


```
{
    "direction": "down",
    "type": "service",
    "code": 2
}
```


（3） One key charging

Function Instructions: After receiving the one-key preflight command, HEISHA BlackBox will start the charging process, including tightening the centering bar, close the dock canopy, close the UAV, close the remote control of the UAV, and start charging.

Payload:


```
{
    "direction": "down",
    "type": "service",
    "code": 3
}
```


（4） Flight route issued

Function Instructions: The third-party platform sends route data to HEISHA BlackBox.

PayLoad：


```
{
    "direction": "down"
    "type": "service",
    "code": 4,
    "params": {
        "waylineId": "7c9c2825-1e65-4b1b-a6c0-2254905c9298",
        "waylineFile":
"https://airservice-private.oss-cn-shanghai.aliyuncs.com/kmz/7c9c2825-1e65-4b1b-a6c0-2254905c9298.kmz?Expires=1693918040&OSSAccessKeyId=LTAI5tMKVbEnyT2gVaY1SCCr&Signature=e8kPDx48X%2FKIgsVoAIt7C7FfRbw%3D"
    }
}
```


Parameter parsing:

Parameter	Instructions
direction	Data direction (character string): "Down" - -a third-party cloud platform- - -> BlackBox
type	Package type (character string): “service ” -Service Delivery
code	Service Code (integer): 4: Route distribution
params	waypoint array, and each array element represents a waypoint
params:waylineId	Route ID, the string

params:waylineFile	The Route download link to the route file in JSON format

（5） Start the task

Function Instructions: It is used to start a route flight mission. After receiving the start mission command, HEISHA BlackBox will automatically take off the UAV and fly according to the route.

Payload：

```
{
    "direction": "down",
    "type": "service",
    "code": 5
}
```

（6）Suspend task

Function Instructions: It is used to suspend a flight mission or a pointing mission. After receiving the pause mission command, the HEISHA BlackBox will hover in the current position.

Payload：

```
{
    "direction": "down",
    "type": "service",
    "code": 6
}
```

（7） Resume the task

Function Instructions: It is used to resume a suspended flight mission or pointing mission. After receiving the recovery mission command, HEISHA BlackBox will control the UAV to continue to fly to the next target point.

Payload：

```
{
    "direction": "down",
    "type": "service",
    "code": 7
}
```

（8） Return to the dock

Function Instructions: It is used to trigger the return of the UAV and make the UAV return to the dock position. If there is an accurate landing function, the landing control UAV will automatically land into the dock.

Payload：

```
{
    "direction": "down",
    "type": "service",
    "code": 8
}
```

（9） Return home to a safe landing point

Function Instructions: used to trigger the UAV to return and enable the UAV to return and land to the safe landing point.

Payload:

```
{
    "direction": "down",
    "type": "service",
    "code": 9
}
```

（10） On-site landing

Function Instructions: To trigger the drone landing.

Payload:

```
{
    "direction": "down",
    "type": "service",
    "code": 10
}
```

（11） Virtual joystick

Function Instructions: When the UAV is in hovering state, the UAV can be flown manually through virtual joystick command.

Payload:

```
{
    "direction": "down",
    "type": "service",
    "code": 11,
    "params": {
        "PITCH": 200,
        "ROLL": 200,
        "THR": 200,
        "YAW": 200
    }
}
```

Parameter parsing:

Parameter	Instructions
direction	Data direction (character string): "Down" - -a third-party cloud platform- - -> BlackBox
type	Package type (character string): “service ” -Service Delivery
code	Service Code (integer): 11: Virtual joystick
params	Parameter (Json), configure different Parameter according to different services
params:PITCH	Pitch rod amount (whole type): 100~300, the median is 200, less than 200 forward fly, more than 200 backward
params:ROLL	Roll bar quantity (whole type): 100~300, the median is

	200, less than 200 to fly left, more than 200 to fly right
params:THR	Throttle rod quantity (whole type): 100~300, the median is 200, less than 200 down, more than 200 rise
params:YAW	Yaw rod amount (integer): 100~300, median 200, less than 200 counterclockwise, more than 200 clockwise

（12） Take-off to the specified altitude

Function Instructions: When the UAV is in disarm state, the takeoff command can hover the aircraft arm and climb to the specified altitude.

Payload：

```
{
    "direction": "down",
    "type": "service",
    "code": 12,
    "params": {
        "height ": 100.0
    }
}
```

Parameter parsing:

Parameter	Instructions
direction	Data direction (character string): "Down" - -a third-party cloud platform- - -> BlackBox
type	Package type (character string): “service ” -Service Delivery
code	Service Code (integer): 12: Take off
params	Parameter (Json), configure different Parameter according to different services
params:height	Relative height (floating point type): in meters

（13）Upload results data

PayLoad：

```
{
    "direction": "down",
    "type": "service",
    "code": 13,
    "params": {
        "bucket ": "bucket _name",
        "path": "path_name"
    }
}
```

Parameter Instructions ：

Parameter	Instructions
direction	Data direction (character string):

	"Down" - -a third-party cloud platform- -> HS Pilot
type	Package type (character string): “service ” -Service Delivery
code	Service Code (integer): 13: Request to upload the results
params	Parameter (Json), configure different Parameter according to different services
params:bucket	MinIO Bucket name (character string)
params:path	Storage path (string), automatic recognition.jpg and. The mp4 format file and deposited in path_name / photo and path_name / video, respectively

（14）Upload the log

PayLoad：

```
{
    "direction": "down",
    "type": "service",
    "code": 14,
    "params": {
        "end point": "URL ",
        "accessKey": "access_key",
        "secretKey": "secret_key",
        "bucket ": "bucket _name",
        "path": "path_name"
    }
}
```

Parameter Instructions ：

Parameter	Instructions
direction	Data direction (character string): "Down" - -a third-party cloud platform- -> HS Pilot
type	Package type (character string): “service ” -Service Delivery
code	Service Code (integer): 14: Request to upload the log
params	Parameter (Json), configure different Parameter according to different services
params :endpoint	MinIO, URL of the server, can include port number with ':' interval (string)
params :accessKey	access key (aka user ID) (String)
params:secretKey	secret key (aka password) (String)
params:bucket	MinIO Bucket name (character string)
params:path	Storage path (string), log data within 30 days

3.4 BlackBox Service Delivery in reply

Topic: heisha / black-box / $ {device unique code} / client / service _ report
Payload：

```
{
    "direction": "up ",
    "type": "service ",
    "code": 0,
    "result": 0
}
```

Instructions: Service Delivery reply is used to confirm whether Service Delivery is responded normally. After the third-party platform issues the service, if no reply is received, it can be resent and other relevant processing.
Parameter parsing:

Parameter	Instructions
direction	Data direction (character string): "Up" -BlackBox- -> Third-party cloud platform
type	Package type (character string): "service" -Service
code	Service code (integer): Reference to the previous section

result	Service call feedback code:
	SERVICE _RESULT_ACCEPTED = 0
	
	// The Service has responded
	SERVICE _RESULT_REJECTED = 1
	// Services are available (reserved)
	SERVICE _RESULT_SUCCESS = 2
	// The Service call was successful
	SERVICE _RESULT_FAIL = 3
	// Service call failed (reserved)

3.5 HS DNEST Property report

Topic: heishs / dnest / $ {device unique code} / client / post
Payload：

```
{
    "direction": "up",
    "type": "post",
    "group": 0,
    "data": {}
}
```

Data parsing ：

Parameter	Instructions
direction	Data direction (character string): "Up" -HS DNEST- -> Third-party cloud platform
type	Package type (character string): 'Post " -Property report
group	Data grouping (integer): 0-Canopy Property report 1-Charging bar Property report 2-Charger Propertys report 5-Remote control base Property report
data	Property data, parsed according to the different groups

（1）Canopy Property

Payload：

```
{
    "direction": "up",
    "type": "post",
    "group": 0,
    "data": {
        "status": 1,
        "canopystatus": 1,
        "tem": [1287, 0],
        "hum": [81, 0]
    }
}
```

Data parsing ：

Parameter	Instructions
direction	Data direction (character string): "Up" -HS DNEST- -> Third-party cloud platform
type	Package type (character string): 'Post " -Property report
group	Data grouping (integer): 0-Canopy properties
data	Property data, parsed according to the different groups
data:status	Online status of canopy assembly: 0-Offline 1-Online
data:canopystatus	canopy status (whole): 0-Unknown 1-Close 2-Open 4-Opening 5-Closing 6-Fault
data:tem	Interior of the dock temperature:

	Array, only one valid, integer, the actual temperature is (integer-1000) * 0.1 degrees Celsius, if 1287 represents a temperature of 28.7 degrees.
data:hum	Humidity inside the machine dock: Array, only one valid, integer, representing the humidity percentage.

（2）Charging bar Property

Payload：

```
{
    "direction": "up",
    "type": "post",
    "group": 1,
    "data": {
        "status": 1,
        "barstatus": 2
    }
}
```

Data parsing ：

Parameter	Instructions
direction	Data direction (character string): "Up" -HS DNEST- -> Third-party cloud platform
type	Package type (character string): 'Post " -Property report
group	Data grouping (integer): 1-Charging bar Property
data	Property data, parsed according to the different groups
data:status	On-line status of the return rod assembly: 0-Offline 1-Online
data:barstatus	Charging bar status (integer): 0-Unknown 1-Release 2-Tightening 3-In the first tightening group 5-In the second tightening group 6-Fault

（3）Charger Property

Payload：

```
{
    "direction": "up",
    "type": "post",
    "group": 2,
```

```
        "data": {
            "status": 1,
            "cdstatus": [4, 1, 2],
            "cdvoltage": 0,
            "cdcurrent": 0
        }
    }
```

Data parsing ：

Parameter	Instructions
direction	Data direction (character string): "Up" -HS DNEST- -> Third-party cloud platform
type	Package type (character string): 'Post " -Property report
group	Data grouping (integer): 2-Charger properties
data	Property data, parsed according to the different groups
data:status	Online status of the charger: 0-Offline 1-Online
data:cdstatus	Charger Status (array): First-Charge state (whole type): 0-No charge 1-Charging initialization 2-Shutoff and charging 3-Starting up and charging in process 4-Charging completed 5-Cooling 8-Fault Second-Battery detection result (whole): 0-Unknown 1-Battery is detected 2-No battery detected Third-UAV switch status (integer): 0-Unknown 1-Boot 2-Shutdown
data:cdvoltage	Charging voltage (integer): The actual voltage is (integer value * 0.1V), such as 524 indicates the charging voltage 52.4V
data:cdcurrent	Charging current (whole): The actual current is (integer value * 0.1A), such as 145 indicates the charging current 14.5A

（4）Remote control base Property

Payload：


```
{
    "direction": "up",
    "type": "post",
    "group": 5,
    "data": {
        "status": 1,
        "tem": 1335,
        "hum": 625,
        "remote_state": 0
    }
}
```


Data parsing ：

Parameter	Instructions
direction	Data direction (character string): "Up" -HS DNEST- -> Third-party cloud platform
type	Package type (character string): 'Post " -Property report
group	Data grouping (integer): 5-Remote control base Property
data	Property data, parsed according to the different groups
data:status	Online status of the remote control base: 0-Offline 1-Online
data:tem	Air temperature around the remote control (whole type): The actual temperature is (integer value-1000) * 0.1 ° C, for example, 1335 indicates that the temperature around the remote control is 33.5 ° C.
data:hum	Humidity around the remote control (whole type): percentage
data:remote_state	Remote control switch status (whole type): 0-Shutdown 1-Boot

（5）The Edge computes module properties

Payload：


```
{
    "direction": "up",
    "type": "post",
    "group": 3,
    "data": {
        "rs485_yl_mode": {
            "yl_online": 0,
            "yl_tem": 0,
```

```
                "yl_hum": 0,
                "yl_rainfall": 0,
                "yl_windspeed": 0,
                "yl_winddirection": 0
            }
        }
    }
```


Data parsing ：

Parameter	Instructions
direction	Data direction (character string): "Up" -HS DNEST- -> Third-party cloud platform
type	Package type (character string): 'Post " -Property report
group	Data grouping (integer): 3-Edge computing module properties
data	Property data, parsed according to the different groups
data: rs485_yl_mode	Weather Station Properties: (json)
rs485_yl_mode: yl_online	Online status of each weather station (byte): Binary bitx: 0-offline, 1-online bit0 hygrometer bit 1 rain gauge bit 2 anemometer bit 3 wind ometer
rs485_yl_mode: yl_tem	Ambient temperature (complete type): degrees * 10 + 1000
rs485_yl_mode: yl_hum	Ambient humidity (integer): RH% * 10
rs485_yl_mode: yl_rainfall	Rainfall (integer), mm / h
rs485_yl_mode: yl_windspeed	Wind speed (whole type), m / s * 100
rs485_yl_mode: yl_winddirection	Wind direction (whole), 0 due north, 90 due east, 180 due south, 270 due west

3.6 HS DNEST Property Post Reply

Topic：heisha/dnest/${unique device code }/server/post_reply

Payload：


```
{
    "direction": "down",
    "type": "post"
}
```

3.7 HS DNEST Service Delivery

Topic: heisha / dnest / $ {device unique code} / server/service
Payload：

```
{
    "direction": "down",
    "type": "service",
    "code": 0
}
```

Parameter parsing:

Parameter	Instructions
direction	Data direction (character string): "Down" - a third-party cloud platform- -> HS DNEST
type	Package type (character string): “service ” -Service Delivery
code	Service Code (integer): 0-System reset 1-Open the drone remote control 2-Turn off the UAV remote control 40-Canopy reset 41-Open dock cover 42-Close dock cover 80-Charging bar reset 81-Tighten the charging bar 82-Release the charging bar 121-Start charging 123-Stop the charging 124-UAV boot up 125-Drone to power off

3.8 HS DNEST Service Delivery in reply

Topic: heisha / dnest / $ {device unique code} / client / service _ report
Payload：

```
{
    "direction": "up ",
    "type": "service ",
    "code": 0,
    "result": 0
}
```

Parameter Parse

Parameter	Instructions

direction	Data direction (character string): "Up" -HS DNEST- -> Third-party cloud platform
type	Package type (character string): "service" -Service
code	Service code (integer): Reference to the previous section

result	Service call feedback code:
	SERVICE _RESULT_ACCEPTED = 0
	
	// The Service has responded
	SERVICE _RESULT_REJECTED = 1
	// Services are available (reserved)
	SERVICE _RESULT_SUCCESS = 2
	// The Service call was successful
	SERVICE _RESULT_FAIL = 3
	// Service call failed (reserved)

3.9 HS DNEST Event Post

Topic: heisha / dnest / $ {device unique code} / client / thing
Payload:

```
{
    "direction": "up ",
    "type": "thing",
    "level": 0,
    "code": 0,
    "params": {
    }
}
```

【Parameter Instructions 】

Parameter	Instructions
direction	Data direction
type	Package type
level	Event Level: THING _LEVEL_INFORMATION = 0 // message THING _LEVEL_WARNING = 1 // report an emergency THING _LEVEL_FAULT = 2 // hitch
code	Event code, refer to thing code Instructions table, as below
params	With with Parameter, resolved according to different codes, refer to thing code Instructions table

[Interface Instructions]

It is used for the client to report specific events to the server in real time. Events are

divided into three levels: message, alarm and fault according to the emergency degree. Event Post requires timely reply from the server. If the client does not receive the server reply, it will continuously send 60 times per second. The code field is divided into three levels according to level before offset, with 0~49 for message, 50~99 for alarm, and 100~149 for fault. K100_GT module offset is 0 * 256, K100 _ GZ module offset 1 * 256, K100_CD module offset 2 * 256, K100 _ KT module offset 3 * 256, and K100 _ TX module offset 4 * 256. That is, the final code value = the value + offset according to the level classification.

thing code Instructions Table

code	Instructions	level	params
K 100_GT (Offset 0 * 256)			
1	433 Remote control A key short press	0	-
2	433 Remote control A key long press	0	-
3	433 Remote control B key short press	0	-
4	433 Remote control B key press	0	-
5	433 Remote control C key short press	0	-
6	433 Remote control key press	0	-
7	433 Remote control D key short press	0	-
8	433 Remote control D key press	0	-
9	Dock cover state reset	0	-
57	The dock cover opened abnormally	1	-
58	The MCU soft-reset event	1	-
101	Dock cover condition failure	2	-
102	Unresponsive command to Canopy motor drive (communication failure)	2	-
103	Reply command but not executed to canopy motor drive (motor abnormal)	2	-
K100 _ GZ (offset of 1 * 256)			
357	Fail to tighten the charging bar	2	-
358	Fail to release the charging bar	2	-
K100_CD (Offset 2 * 256)			
562	The battery temperature too high	1	-
563	The battery is seriously aging	1	-
564	The charging current too high	1	-
565	Charging time too long	1	-
612	Battery type not match	2	-
613	The charging voltage too high	2	-

614	The charging voltage too low	2	-
615	The power meter no data	2	-
616	Temperature communication error	2	-
K100 _ KT (offset 3 * 256)			
869	Heavy rainfall	2	-
870	Too much smoke	2	-
871	No temperature probe 1	2	-
872	No temperature probe 2	2	-
K100 _ TX (offset 4 * 256)			
1025	One-key preflight successful	0	-
1026	One-click charging successful	0	-
1031	The T100 firmware update successful	0	“file_name”： Firmware file name
1032	The K100 firmware updated successful	0	“file_name”： Firmware file name
1125	One-key preflight failed	2	“alarm”： 1: The landing board, charging board and remote control offline 3: The charging bar not tightened 4: Charging bar error 5: No battery was detected 6: The UAV is not turned on, and the remote control is not turned on 7: Charging bar error 8: Charging bar not open 9: Canopy board offline 10: Canopy board error 11: Canopy open timeout
1126	One-click charging failed	2	“alarm”： 2: The landing board, charging board and remote control offline 12: The charging bar tightening timeout 13: Charging bar error 14: Canopy board offline 15: Canopy board error 16: Drum close timeout 17: No battery detected 18: The drone shutdown failed 19: The remote control

			shutdown failed
1131	The T100 firmware update failed	2	“file_name”： Firmware file name
1132	The K100 firmware update failed	2	“file_name”： Firmware file name

3.10 HS DNEST Event Post and Reply

Topic: heisha / dnest / $ {device unique code} / client / thing

Payload：

```
{
    "direction": "down",
    "type": "thing",
    "level": 0,
    "code": 0
}
```

【Parameter Instructions 】

Parameter	Instructions
direction	Data direction
type	Package type
level	Event Level: THING _LEVEL_INFORMATION = 0 // message THING _LEVEL_WARNING = 1 // report an emergency THING _LEVEL_FAULT = 2 // hitch
code	Event code, refer to the thing code Instructions table in the previous section

[Interface Instructions]

Use to confirm the server's Reply to event Post.

3.11 HS Pilot Property Post

Topic: heisha / pilot / $ {device unique code} / client / post

Payload：

```
{
    "direction": "up",
    "type": "post",
    "group ": 32,
    "data": {
```

```
"product": {
        "connection": false,
        "type": 6
},
"remoteController": {
        "batteryLevel": 100,
        "connection": false,
        "signalQuality": 100
},
"battery": [{
        "connection": false,
        "percent": 100,
        "temperature": 27.4,
        "voltage": 52.212
}, {
        "connection": false,
        "percent": 99,
        "temperature": 27.1,
        "voltage": 52.221
}],
"flightController": {
        "attitude": {
                "pitch": 0.8,
                "roll": 1.4,
                "yaw": -131.8
        },
        "connection": false,
        "distanceToHome": 0,
        "gpsSatellite": 0,
        "gpsSignal": 0,
        "horizontalSpeed": 0,
        "isFlying": false,
        "location": {
                "height": 0,
                "latitude": 0,
                "longitude": 0
        },
        "takeoffLocation": {
                "altitude": 0,
                "latitude": 0,
                "longitude": 0
        },
        "verticalSpeed": -0
},
```

```
            "RTK": {
                "accurateLanding": true,
                "connected": true,
                "enable": false,
                "healthy": false,
                "source": 1,
                "supported": true
            },
            "payloadLeftMain": {
                "camera": {
                    "connection": false,
                    "cameraType": 6,
                    "isRecording": false,
                    "mode": 0,
                    "recordingTime": 0,
                    "videoSource": 1,
                    "zoomRatio": 1
                },
                "gimbal": {
                    "connection": false,
                    "pitch": 28.8,
                    "yaw": 0
                }
            },
            "dockLocation": {
                "latitude": 22.6761094,
                "longitude": 114.0623609
            },
            "achievementManager": {
    "fileTotal": 16,
    "upNum": 2,
    "mnt Status": 0
    }
        }
    }
```


Data parsing ：

Parameter	Instructions
direction	Data direction (character string): "Up" -HS Pilot- -> Third-party cloud platform
type	Package type (character string): 'Post " -Property report
group	Property grouping (integer) (reserved)

data	Property data, parsed according to the different groups
data:product	DJI SDK
data:product:connection	DJI SDK Connection status: false- not connected true- -Connected
data:product:type	Drone model: 0-Unknown 1-Mavic Mini 3 2-Mavic Mini 3Pro 3-Mavic 3M 4-Mavic 3 Enterprise 5-Matrix 30 6-Matrix 300 RTK 7-Matrix 350 RTK
data:remoteController:connection	Remote control connection status: false- not connected true- -Connected
data:remoteController:batteryLevel	Remote control power (percentage)
data:remoteController:signalQuality	Remote control signal strength (percentage)
data:battery	UAV battery status (array): Array length is equal to the number of cells
data:battery:connection	Battery connection status false- not connected true- -Connected
data:battery:percent	Battery level (%)
data:battery:voltage	Battery voltage (floating point)
data:battery:temperature	Battery temperature (floating point)
data:flightController	Flight control state
data:flightController:connection	Flight control connection status: false- not connected true- -Connected
data:flightController:attitude	UAV posture
data:flightController:attitude:pitch	Pitch Angle (floating point)
data:flightController:attitude:roll	Roll Angle (floating point)
data:flightController:attitude:yaw	Yaw angle (floating point)
data:flightController:distanceToHome	Distance from the Home Point (in meters)
data:flightController:gpsSatellite	GPS star number
data:flightController:gpsSignal	GPS signal intensity
data:flightController:horizontalSpeed	Horizontal flight speed (in m / s)

data:flightController:verticalSpeed	Vertical flight speed (in m / s)
data:flightController:isFlying	Flight sign: false-not flying true-flight
data:flightController:location	UAV location
data:flightController:location:height	Relight height relative altitude (floating point)
data:flightController:location:latitude	Current latitude of the UAV (floating point)
data:flightController:location:longitude	Current UAV longitude (floating point)
data:flightController:takeoffLocation	The takeoff position of the drone
data:flightController:takeoffLocation:altitude	UAV altitude altitude altitude (floating point)
data:flightController:takeoffLocation:latitude	UAV takeoff position latitude (floating point)
data:flightController:takeoffLocation:longitude	UAV takeoff position longitude (floating point)
data:flightController:RTK	RTK state
data:flightController:RTK:supported	Whether the RTK is supported: False- not support True-support
data:flightController:RTK:source	RTK signal source: 0-No 1-Chihiro 2-Base station 4-Custom-defined network 5-NTRIP
data:flightController:RTK:enable	Enable mark: false — not enabled true-enable
data:flightController:RTK:connected	Connection flag: false-linkage true- -Connected
data:flightContorller:RTK:healthy	Positioning status: false-not located true- located
data:flightController:RTK:accurateLand ing	The RTK precision landing enables: false-close true-open
data:payloadLeftMain	Main payload status
data:payloadLeftMain:camera	Main camera status
data:payloadLeftMain:camera:connection	Camera connection status: False-Unconnected

	True- -Connected
data:payloadLeftMain:camera:cameraType	Payload model (whole type): 0-Unknown 1-M3E 2-M3T 3-M3M 4-M30 5-M30T 6-ZENMUSE_H20 7-ZENMUSE_H20T 8-ZENMUSE_P1 9-ZENMUSE_L1
data:payloadLeftMain:camera:mode	Camera mode: 0-Take photos 1-Camera
data:payloadLeftMain:camera:videoSource	Video source: 0-Default 1-Wide Angle 2-Zoom 3-Infrared
data:payloadLeftMain:camera:zoomRatio	Zoom ratio (integer)
data:payloadLeftMain:camera:isRecording	Video status: false — no video taken true- recording
data:payloadLeftMain:camera:recordingTime	Camera Time (in seconds)
data:payloadLeftMain:gimbal	The state of the cloud
data:payloadLeftMain:gimbal:connection	Gimbal connection status: false- not connected true- Connected
data:payloadLeftMain:gimbal:pitch	Gimbal pitch Angle (unit: degree)
data:payloadLeftMain:gimbal:yaw	Tangle relative to fuselage yaw (unit: degree)
data:dockLocation	Dock location positioned by the UAV
data:dockLocation:latitude	Dock location latitude (floating point)
data:dockLocation:longitude	Dock location and longitude (floating point)
data : achievementManager	Results Management Module
data : achievementManager : file Total	Total amount of media files (integer)
data : achievementManager : u pNum	Number of uploaded files (integer)
data : achievementManager : m ntStatus	Results processing status: 0-Initializing 1-Free 2-Start synchronizing the data

	3-Synchronizing the data 4-Data synchronization completed 5-Start pulling up a file 6-Pulling up a file 7-One file pull complete 8-Pull all the files complete 9-Start uploading a file 10-Uploading a file 11-A file upload complete 12-All files upload complete

3.12 HS pilot Property Post Reply

Topic：heisha/pilot/${unique device code }/server/post_reply
Payload：

```
{
    "direction": "down ",
    "type": "post "
}
```

3.13 HS Pilot Service Delivery

Topic：heisha/pilot/${unique device code }/server/service
Payload：

```
{
    "direction": "down",
    "type": "service",
    "code": 0,
    "params": {}
}
```

Parameter parsing:

Parameter	Instructions
direction	Data direction (character string): "Down" - -a third-party cloud platform- -> HS Pilot
type	Package type (character string): “service ” -Service Delivery
code	Service Code (integer): 0-Undefined 8-Switch to the photo mode 9-Switch to the camera mode 10-Take a picture 11-Start the video recording

	12-Stop the video recording 13-Zoom in 14-Zoom out 15-Switch between the camera video source in the cycle mode 16-Specify the camera and the video source 17-Set the cradle head angle 44-Upload the outcome data
params	Parameter (Json), configure different Parameter according to different services

Service details:

（1）Switch to the photo mode

PayLoad：

```
{
    "direction": "down",
    "type": "service",
    "code": 8
}
```

（2）Switch to the camera mode

PayLoad：

```
{
    "direction": "down",
    "type": "service",
    "code": 9
}
```

（3）Take a picture

PayLoad：

```
{
    "direction": "down",
    "type": "service",
    "code": 10
}
```

Instructions: Effective when the camera is in photo mode

（4）Start the video recording

PayLoad：

```
{
    "direction": "down",
    "type": "service",
    "code": 11
}
```

Instructions: Effective when the camera is in camera mode

（5）Stop the video recording

PayLoad：

```
{
```

```
        "direction": "down",
        "type": "service",
        "code": 12
    }
```

Instructions: Effective when the camera is in camera mode

（6）Zoom in

PayLoad：

```
    {
        "direction": "down",
        "type": "service",
        "code": 13
    }
```

Instructions: Effective when the video source is a zoom camera

（7）Zoom out

PayLoad：

```
    {
        "direction": "down",
        "type": "service",
        "code": 14
    }
```

Instructions: Effective when the video source is a zoom camera

（8）Switch over the video source in the cycle mode

PayLoad：

```
    {
        "direction": "down",
        "type": "service",
        "code": 15
    }
```

Instructions: For example, the camera supports three video sources, wide Angle, zoom and infrared, so it switches between the three video sources in sequence

（9）Specify the camera and the video source

PayLoad：

```
    {
        "direction": "down",
        "type": "service",
        "code": 17,
        "params": {
            "index": 0,
            "source": 1
        }
    }
```

Parameter Instructions ：

Parameter	Instructions
direction	Data direction (character string):

	"Down" - -a third-party cloud platform- -> HS Pilot
type	Package type (character string): “service ” -Service Delivery
code	Service Code (integer): 17: Set the Gimbal Angle
params	Parameter (Json), configure different Parameter according to different services
params:index	Specify the payload position: 0-Default 1-Main payload (bottom left) 2-Lower right 3-Above 4-FPV
params:source	Video source: 0-Default 1-Wide Angle 2-Zoom 3-Infrared

（10）Set the Gimbal Angle

PayLoad：

```
{
    "direction": "down",
    "type": "service",
    "code": 17,
    "params": {
        "PITCH": 0,
        "YAW": 0
    }
}
```

Parameter Instructions ：

Parameter	Instructions
direction	Data direction (character string): "Down" - a third-party cloud platform- -> HS Pilot
type	Package type (character string): “service ” -Service Delivery
code	Service Code (integer): 17: Set the gimbal Angle
params	Parameter (Json), configure different Parameter according to different services
params:PITCH	Gimbal pitch Angle (unit: degree)
params:YAW	The yaw Angle of the fuselage first (unit: degrees)

（11）Upload results data

PayLoad：

```
{
    "direction": "down",
    "type": "service",
    "code": 44,
    "params": {
        "bucket ": "bucket _name",
        "path": "path_name"
    }
}
```

Parameter Instructions ：

Parameter	Instructions
direction	Data direction (character string): "Down" - a third-party cloud platform- -> HS Pilot
type	Package type (character string): “service ” -Service Delivery
code	Service Code (integer): 44: Request to upload the results
params	Parameter (Json), configure different Parameter according to different services
params:bucket	MinIO Bucket name (character string)
params:path	Storage path (string), automatic recognition.jpg and. The mp4 format file and deposited in path_name / photo and path_name / video, respectively

3.14 HS Pilot Service Delivery in reply

Topic：heisha/pilot/${unique device code }/client/service_reply

Payload：

```
{
    "direction": "up ",
    "type": "service ",
    "code": 0,
    "result": 0
}
```

Parameter Parsing:

Parameter	Instructions
direction	Data direction (character string): "Up" -HS Pilot- -> Third-party cloud platform
type	Package type (character string): "service" -Service

code	Service code (integer): Reference to the previous section

result	Service call feedback code:
	SERVICE _RESULT_ACCEPTED = 0
	
	// The Service has responded
	SERVICE _RESULT_REJECTED = 1
	// Services are available (reserved)
	SERVICE _RESULT_SUCCESS = 2
	// The Service call was successful
	SERVICE _RESULT_FAIL = 3
	// Service call failed (reserved)