Socket API

Basic Description of the Central Control Environment (instructions)

1. Purpose

This document provides a detailed explanation of the functionality and syntax of the control API commands for the central control module. It enables readers to quickly understand the relevant features of the central control module and guide developers in their subsequent coding work. This guide facilitates the completion of the development of the control host for Yealink VCS products' network mode.

2. Introduction

2.1 Network Mode Configuration Description

- The device supports network mode to control the host device, using TCP protocol to establish a connection. The client needs to know the ip address and port number of the host to connect to the host side.
- The following parameters need to be configured for network mode:

1> The control port of the network mode of the center control is configured as: 6024
2> IP address can be viewed according to the actual situation
3>If the central control system's network mode has enabled password authentication, when establishing a TCP connection using the TCP protocol, you will need to input a password and authenticate it successfully before being able to execute central control commands normally. The password format for input is "XXXX\r\n", where XXXX represents the correct password.
For example, after establishing a TCP connection, if you receive the message "Password:\r\n", it indicates that you need to input the authentication password.
If an incorrect password is entered, the connection will return "Invalid Password!\r\nPassword:\r\n", prompting you to re-enter the password for authentication.
Once you enter the correct password, the connection will respond with "welcome yealink central control". This indicates that the authentication was successful and you can proceed with controlling the central system.

2.2 Serial Port Mode Configuration Description

Some devices support serial port mode for controlling host devices, such as M500 and M900. The client needs to connect to the host device using a serial cable, and the host device will recognize the serial port connection and establish communication using the serial protocol.
The baud rate, data bits, parity bit, and stop bit of the serial port can be configured through the device's web interface.

3. API Directives

3.1 API Basic Syntax Description

This introduction to the device center control API command description file. In order to facilitate the description of the API command syntax, the format of the API writing the following statements and conventions:

- Instruction words are all lowercase

- Command basic structure: command + parameter
- Parameter composition: parameter name + parameter value ------ separated by colon, the command parameter overall with the httpapi format, in the form of json string transmission
- Note: The 'app' parameter is unique and separate from other parameters. It operates independently outside the JSON format during transmission. This parameter indicates that the instruction being called is specific to the app. If this parameter is not included, the default behavior is to execute instructions at the system level.

3.2 Description of the Basic Packet Format

1. Description of sending format Use \r\n (carriage return line feed) as the end identifier.

All API commands must be terminated with the delimiter \r\n.

2. Description of return format: Adoption of \r\n as end identifier

All return packets will have the end identifier of \r\n

3. Command basic format:

- Command + \r\n (no parameter type)
- Command + space + json string + \r\n (with parameter type)
- Command + space + app:value + space + json string + \r\n (with parameters and app parameter type)

3.2 Detailed Description of API Commands

The currently supported API command list is described below. Please refer to the command documentation. For debugging, you can use a tcp debugging tool to set up and test API commands, according to the environment setup instructions provided above. The commands and parameters supported may vary depending on the version and device model. Parameters enclosed in "[]" are optional.

System Layer

Preface: In the following command instructions, the request parameters in "[]" are all optional. Input parameters for commands must be provided in json string format. Fields involving Chinese characters, such as names, must use UTF-8 encoding.

System Syntax

1. Get System Version Information

- **Return Parameters**

Name	Typology	Parameter Description
model	string	device models
firmware	string	firmware version
hardware	string	hardware version
serialnumber	string	device sn number
macaddress	string	device mac address
cc-version	string	center control version

- **Description**

System command	system get version

Supported Models	Meetingboard 65/86, Meetingboard 65/75/86 Pro, MeetingDisplay, MeetingEye 500, MeetingBar A10/A40/A50, UVC40
Send Format	system get version\r\n
Return Format	system get version {"model":"MeetingEye 500","firmware":"280.320.254.125","hardware":"280.0.0.0.0.0.0","macaddress":"00:15:65:00:00:00","serialnumber":"506607D117000009","cc-version":"1.0.0.2"}\r\n
Example	Send: system get version \r\n Return: system get version {"model":"MeetingEye 500","firmware":"280.320.254.125","hardware":"280.0.0.0.0.0.0","macaddress":"00:15:65:00:00:00","serialnumber":"506607D117000009","cc-version":"1.0.0.2"} \r\n

2. Obtaining System Status

- **Return Parameters**

Name	Typology	Range of Values	Parameter Description
status	string	[sleeping,wake-up]	System Status sleeping: sleeping wake-up: wake up

- **Description**

System command	system get status
Supported Models	Meetingboard 65/86, Meetingboard 65/75/86 Pro, MeetingDisplay, MeetingEye 500, MeetingBar A10/A40/A50
Send Format	system get status\r\n
Return Format	system get status {"status":"sleeping"}\r\n
Example	Send: system get status \r\n Return: system get status {"status":"sleeping"} \r\n

3. Setting the System Status

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
status	string	[sleeping,wake-up,reboot,reset]	Yes	System Status sleeping: sleep wake-up: wake up reboot: reboot reset: factory restore

sn	string	NULL	No	Device unique identifier SN

- **Description**

System command	system set status {"status":"<value>",["sn":"<value>"]}
Supported Models	Meetingboard 65/86, Meetingboard 65/75/86 Pro, MeetingDisplay, MeetingEye 500, MeetingBar A10/A40/A50, Yealink RoomConnect, UVC40, SmartVision 40 AP08, AVBridge, CM20, CM50, CS10, CS10-D **Note:** 1. Yealink RoomConnect/UVC40 only support reboot, restore to factory settings 2. Only Yealink RoomConnect supports passing the sn parameter 3. SmartVision 40 supports reboot 4. AP08, AVBridge, CM20, CM50, CS10, CS10-D support both reboot and factory reset
Send Format	system set status {"status":"wake-up",["sn":"506607D117000009"]} \r\n
Return Format	system set status {"status":"wake-up",["sn":"506607D117000009"]} \r\n
Example	**With SN scenario:** Send: system set status {"status":"wake-up","sn":"506607D117000009"} \r\n Return: system set status {"status":"wake-up","sn":"506607D117000009"} \r\n **In the case without an SN:** Send: system set status {"status":"wake-up"} \r\n Return: system set status {"status":"wake-up"} \r\n

4. Setting Log Sever Address

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
enable	int	[0,1]	Yes	Log server function switch 1: Enable 0: Disable

facility	int	[0-23]	Yes	Log storage device 0: kernel messages 1: User message 2: Mail system 3: System daemon 4: security/authorization messages 5: internally generated system information 6: printing subsystem 7: network messaging subsystem 8: UUCP subsystem The clock daemon 10: security/authorization messages 11: FTP daemon 12: NTP subsystem 13: Log audit 14: log warning15: Clock daemon 16: For local use 0 17: For local use 1 18: For local use 2 19: For local use 3 20: For local use 4 21: For local use 5 22: For local use 6 23: For local use 7
level	int	[1-7]	Yes	Log level
transport-type	int	[0,1,2]	Yes	transport type 0: UDP 1: TCP 2: TLS
port	int	NULL	Yes	ports
server	string	NULL	Yes	The server address for log synchronization.

- **Description**

System command	system set log-server {"enable":<value>,"facility":<value>,"level":<value>,"transport-type":<value>,"port":<value>,"server":"<value>"}
Supported Models	Meetingboard 65/86, Meetingboard 65/75/86 Pro, MeetingDisplay, MeetingEye 500, MeetingBar A10/A40/A50, UVC40 **Note:** 1. When deactivating the log server, you can skip additional parameters; otherwise, parameters are required. 2. Recommended parameters: facility (16), level (6), transport-type (0)
Send Format	system set log-server {"enable": 1,"server":"syslog.test.yealink.com", "port":514, "facility":1, "level":6,"transport-type":2} \r\n

Return Format	system set log-server {"enable": 1,"server":"syslog.test.yealink.com", "port":514, "facility":1, "level":6,"transport-type":2} \r\n
Example	Send: system set log-server {"enable": 1,"server":"syslog.test.yealink.com", "port":514,"facility":1, "level":6,"transport-type":2}\r\n Return: system set log-server {"enable": 1,"server":"syslog.test.yealink.com", "port":514, "facility":1, "level":6,"transport-type":2}\r\n

5. Get device list

- **Return Parameters**

Name	Typology	Range of Values	Note
device-list	device_info []	NULL	List of device information

device_info

Name	Typology	Note
model	string	Equipment type
firmware	string	Firmware version
hardware	string	hardware version
serialnumber	string	device sn number
macaddress	string	device mac address
id	int	Device identification (can be used to set or get display parameters)

- Description

System command	system get devices
Supported Models	Yealink RoomConnect, SmartVision 40, MeetingEye 500
Send Format	system get devices \r\n
Return Format	system get devices {"device-list": [{ "model":"UVC86", "firmware":"128.423.253.104", "hardware":"263.0.19.0.3.0.36", "macaddress":"00:15:65:00:00:00", "serialnumber":"506607D117000009", }, { "model":"UVC84", "firmware":"130.303.253.44", "hardware":"261.0.5.10.43.0.58", "macaddress":"00:24:13:00:00:00", "serialnumber":"803032E070000031", }] } \r\n

Example	Send: system get devices \r\n Return: system get devices {"device-list": [{"model":"UVC86", "firmware":"128.423.253.104", "hardware":"263.0.19.0.3.0.36", "macaddress":"00:15:65:00:00:00", "serialnumber":"506607D117000009",}, {"model":"UVC84", "firmware":"130.303.253.44", "hardware":"261.0.5.10.43.0.58", "macaddress":"00:24:13:00:00:00", "serialnumber":"803032E070000031", "id": 0}]} \r\n

6. Retrieve Working Hours

- **Return Parameters**

Name	Typology	Range of Values	Parameter Description
value	int	>0	Operating time (in minutes)

- **Description**

System command	system get uptime
Supported Models	Meetingboard 65/86, Meetingboard 65/75/86 Pro, MeetingDisplay, MeetingEye 500, MeetingBar A10/A40/A50, SmartVision 40
Send Format	system get uptime\r\n
Return Format	system get uptime {"value":840} \r\n
Example	Send: system get uptime\r\n Return: system get uptime {"value":840} \r\n

7. Retrieve CPU Information

- **Return Parameters**

Name	Typology	Range of Values	Parameter Description
cpu-usage	int	[1~100]	cpu usage rate (in percent, integer)
cpu-temp	int	[-10,100]	cpu temperature (unit: °C)

- **Description**

System command	system get cpu-info
Supported Models	Meetingboard 65/86, Meetingboard 65/75/86 Pro, MeetingDisplay, MeetingEye 500, MeetingBar A10/A40/A50, SmartVision 40 Note: 1. SmartVision 40 supports cpu-temp
Send Format	system get cpu-info\r\n

Return Format	system get cpu-info {"cpu-usage":25} \r\n
Example	Send: system get cpu-info\r\n Return: system get cpu-info {"cpu-usage":25} \r\n **SmartVision 40:** Send: `system get cpu-info\r\n` Return:``system get cpu-info {"cpu-usage":11,"cpu-temp":69} \r\n

8. Retrieve system hardware information

- **Return Parameters**

Name	Typology	Parameter Description
model	string	device models
firmware	string	firmware version
hardware	string	hardware version
serialnumber	string	device sn number
macaddress	string	device mac address
cc-version	string	center control version
vendor	string	Device Manufacturer Name

- **Description**

System command	system get sys-info
Supported Models	SmartVision 40
Send Format	system get sys-info\r\n
Return Format	system get sys-info {"model":"SmartVision40","firmware":"286.0.253.1068","hardware":"286.0.16.0.0.0.0","serialnumber":"506656F110000056","macaddress":"24:9a:d8:db:a2:3b","cc-version":"1.0.0.16","vendor":"Yealink"}
Example	` ` Send: ` ` system get sys-info ` ` Return: system get sys-info {"model":"SmartVision40","firmware":"286.0.253.1068","hardware":"286.0.16.0.0.0.0","serialnumber":"506656F110000056","macaddress":"24:9a:d8:db:a2:3b","cc-version":"1.0.0.16","vendor":"Yealink"} \r\n

9. Read usb connection status

- **Return Parameters**

Name	Typology	Range of Values	Description

status	string	[on,off]	Connection status: on: connected off: not connected

- **Description**

System command	system get pc-connect-status
Supported Models	SmartVision 40
Send Format	system get pc-connect-status\r\n
Return Format	system get pc-connect-status {"status":"on"} \r\n
Example	Send： `system get pc-connect-status\r\n` Return:``system get pc-connect-status {"status":"on"} \r\n

10. Read memory information

- **Return Parameters**

Name	Typology	Range of Values	Description
memory-usage	int	[0,100]	Memory utilization
total-memory	int	>0	Total memory (MB)
free-memory	int	>=0	Unused memory (MB)
used-memory	int	>0	Used memory (in MB)

- **Description**

sytem directive	system get memory-info
Supported Models	Smartvision 40
Send Format	system get memory-info\r\n
Return Format	system get memory-info {"memory-usage":20,"total-memory":953,"free-memory":758,"used-memory":195} \r\n
Example	Send: ` ` system get memory-info\r\n `Return:` system get memory-info {"memory-usage":20,"total-memory":953,"free-memory":758,"used-memory":195} \r\n

11. Read device status

- **Return Parameters**

Name	Typology	Range of Values	Description

normal	string	[true,false]	Device health status: normal =!(memorytoohigh || cpuusagetoohigh)
memorytoohigh	string	[true,false]	Memory usage too high — returns true if over 80% (by default)
cpuusagetoohigh	string	[true,false]	cpu usage too high — returns true if over 80% (by default)

- **Description**

System command	system get alert-status
Supported Models	SmartVision 40
Send Format	system get alert-status\r\n
Return Format	system get alert-status {"normal":"true","memorytoohigh":"false","cpuusagetoohigh":"false"} \r\n
Example	Send: `system get alert-status\r\n` Return:``system get alert-status {"normal":"true","memorytoohigh":"false","cpuusagetoohigh":"false"} \r\n

12. Configure alert notification rules

- **Request Parameters**

Parameter	Typology	Range of Values	Description
cpu	int	[0,100]	Set high cpu threshold, unit: %, default is 80. If the requested value is out of range, it will revert to the default value.
memory	int	[0,100]	Set high memory threshold, unit: %, default is 80. If the requested value is out of range, it will revert to the default value.

- **Description**

System command	system set alert-rules {"cpu":90,"memory":70}
Supported Models	SmartVision 40
Send Format	system set alert-rules {"cpu":90,"memory":70} \r\n
Return Format	system set alert-rules {"cpu":90,"memory":70} \r\n
Example	Send： `system set alert-rules {"cpu":90,"memory":70}\r\n` Return:``system set alert-rules {"cpu":90,"memory":70} \r\n

13. Retrieve the current time

- **Return Parameters**

Parameter	Typology	Range of Values	Description
value	int	NA	linux timestamp

- **Description**

System command	system get current-time
Supported Models	SmartVision 40
Send Format	system get current-time\r\n
Return Format	system get current-time {"value":1725459796} \r\n
Example	Send: `system get current-time\r\n` Return:``:system get current-time {"value":1725459796} \r\n

14. Retrieve Hardware Information

- **Return Parameters**

Name	Typology	Range of Values	Parameter Description
» hardware_list	body	object	Yes
»» type	body	string	Yes
»» index	body	string	No

- **Description**

System command	system get hardware
Supported Models	AP08
Send Format	system get hardware {"hal_list": [{"type": "<value>","mode": "<value>","index": [<value>]}]} \r\n
Return Format	system get hardware {"hal_list": [{"type":"<value>","mode":"<value>","status": [{"index": <value>,"status":"<value>"}]}]}\r\n
Example	Send: system get hardware {"hal_list": [{"type":"gpio","mode":"input","index": [1,2]}]} \r\n Return: system get hardware {"hal_list": [{"type":"gpio","mode":"input","status": [{"index":1,"status":"high"}, {"index":2,"status":"low"}]}]}\r\n

15. Get BYOD-EXTENDER Status

- **Return Parameters**

Name	Typology	Range of Values	Description

status	string	[available,in_use, disconnected, unknown]	USB Connection Status: available: MVC-BYOD-Extender is connected to host, but not in use in_use: MVC-BYOD-Extender is currently in use disconnected: MVC-BYOD-Extender not connected unknown: Unknown status

- **Description**

System command	byod get extender-status
Supported Models	Yealink RoomConnect
Send Format	byod get extender-status {"sn":"806009F040000357"} \r\n
Return Format	byod get extender-status {"status":"unknown"} \r\n
Example	Send: byod get extender-status {"sn":"806009F040000357"} \r\n Return: byod get extender-status {"status":"unknown"} \r\n

16. Configure device location switch

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
status	string	[on,off]	Yes	on: Enable device location off: Disable device location

- **Description**

System command	system set locating-status
Supported Models	AP08, AVBridge, CM20, CM50, CS10, CS10-D
Send Format	system set locating-status {"status": "on"} \r\n
Return Format	system set locating-status {"status": "on"} \r\n
Example	Send: system set locating-status {"status": "on"} \r\n Return: system set locating-status {"status": "on"} \r\n

17. Configure LED On/Off Switch

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description

enable	string	{on, off}	Yes	on: Turn on LED light off: Turn off LED light Note: Turning on/off only controls the LED for the relevant event

- **Description**

System command	system set led-preset-status
Supported Models	CM20, CM50, CS10, CS10-D
Send Format	system set led-preset-status {"enable": "on"} \r\n
Return Format	system set led-preset-status {"enable": "on"} \r\n
Example	Send: system set led-preset-status {"enable": "on"} \r\n Return: system set led-preset-status {"enable": "on"} \r\n

18. Get LED switch status

- **Return Parameters**

Name	Typology	Range of Values	Parameter Description
enable	string	[on,off]	on: Enabled off: Disabled

- **Description**

System command	system get led-preset enable
Supported Models	CM20, CM50, CS10, CS10-D
Send Format	system get led-preset enable\r\n
Return Format	system get led-preset enable {"enbale":"off"} \r\n
Example	Send: system get led-preset enable\r\n Return: system get led-preset enable {"enbale":"off"} \r\n

19. Set the LED event color

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
event	string	{mute,unmute}	Yes	Light event
color	string	{red, green, orange, #xxxxxx}	Yes	The indicator light color for the corresponding event

- **Description**

system	system set led-preset config
Supported Models	CM20, CM50, CS10, CS10-D
Send Format	system set led-preset config {"event":"mute", "color":"red"} \r\n
Return Format	system set led-preset config {"event":"mute", "color":"red"} \r\n
Example	Send: system set led-preset config {"event":"mute", "color":"red"} \r\n Return: system set led-preset config {"event":"mute", "color":"red"} \r\n

20. Retrieve the color of the LED event

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
event	string	{mute, unmute}	Yes	Event to retrieve the light color

- Return Parameters

Name	Typology	Range of Values	Parameter Description
event	string	{mute, unmute}	-
color	string	{red, green, orange,#xxxxxx}	Light color

- Description

System Syntax	system get led-preset config
Supported Models	CM20, CM50, CS10, CS10-D
Send Format	system get led-preset config { “event":"unmute"} \r\n
Return Format	system get led-preset config { “event":"unmute","color":"#856974"} \r\n
Example	Send: system get led-preset config { “event":"unmute"} \r\n Return: system get led-preset config { “event":"unmute","color":"#856974"} \r\n

Button Syntax

1. Remote Control Button Operation

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
key	string	[power,F1,F2,F3,volume+,volume-,zoom+,zoom-,up,down,right,left,select,mute,back,call,delete,hangup,0~9*#]	Yes	Remote control button name

- **Description**

Button Command	button {"key": " <value>"}
Supported Models	MeetingEye 500, UVC40, SmartVision 40 **Note:** **1. UVC40 models only support volume+ and volume-** **2. SmartVision 40 only supports volume+ and volume-**
Send Format	button {"key":"power"}\r\n
Return Format	button {"key":"power"}\r\n
Example	Send: button {"key":"power"} \r\n Return: button {"key":"power"} \r\n

Camera Syntax

1. Control Camera Movement

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
direction	string	[up,down,left,right,stop]	Yes	Camera movement direction up: control camera direction up down: control camera direction down left: control camera direction left right: control camera direction right stop: control camera stop
sn	string	NULL	No	Camera Unique Identifier **Note: If the command is requested with the sn parameter, it means that the control is for the device with the specified sn, otherwise the control is for the first device in the camera list**

- **Description**

Camera Command.	camera move {"direction":"<value>",["sn":"<value>"]}

Supported Models	Meetingboard 65/86, Meetingboard 65/75/86 Pro, MeetingEye 500, MeetingBar A10/A40/A50, Yealink RoomConnect, UVC40, UVC85, UVC86, SmartVision 40, AVHub, SmartVision 80 **Note:** **1. For Yealink RoomConnect and AVHub, the 'sn' parameter is mandatory when using this command** **2. MeetingBar A10/A40/A50, the 'sn' parameter is optional** **3. For MeetingBoard 65/86/75-Pro, MeetingEye 500, if there is only one camera, the 'sn' parameter is optional. If there are multiple cameras, use the 'sn' to specify the correct camera for operation** **4. UVC and SmartVision devices do not require the 'sn' parameter**
Send Format	camera move {"direction":"up",["sn":"506607D117000009"]} \r\n
Return Format	camera move {"direction":"up",["sn":"506607D117000009"]} \r\n
Example	**With sn parameter case:** Send: camera move {"direction":"up","sn":"506607D117000009"} \r\n Return: camera move {"direction":"up",["sn":"506607D117000009"} \r\n **Without sn parameter case:** Send: camera move {"direction":"up"} \r\n Return: camera move {"direction":"up"} \r\n

2. Control Camera Focus

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
direction	string	[in,out,stop]	Yes	Camera focus direction in: zoom in out: zoom out stop: control camera to stop changing
sn	string	NULL	No	Camera Unique Identifier **Note: If the command is requested with the sn parameter, it means that the control is for the device with the specified sn, otherwise the control is for the first device in the camera list**

- **Description**

Camera Command.	**camera zoom {"direction":"<value>",["sn":"<value>"]}**
Functional Description	Control camera focus

Supported Models	Meetingboard 65/86, Meetingboard 65/75/86 Pro, MeetingEye 500, MeetingBar A10/A40/A50, Yealink RoomConnect, UVC40, UVC85, UVC86, SmartVision 40, AVHub, SmartVision 80 **Note:** **1. For Yealink RoomConnect and AVHub, the 'sn' parameter is mandatory when using this command** **2. MeetingBar A10/A40/A50, the 'sn' parameter is optional** **3. For MeetingBoard 65/86/75-Pro, MeetingEye 500, if there is only one camera, the 'sn' parameter is optional. If there are multiple cameras, use the 'sn' to specify the correct camera for operation** **4. UVC and SmartVision devices do not require the 'sn' parameter**
Send Format	camera zoom {"direction":"in",["sn":"506607D117000009"]} \r\n
Return Format	camera zoom {"direction":"in",["sn":"506607D117000009"]} \r\n
Example	**With sn parameter case:** Send: camera zoom {"direction":"in","sn":"506607D117000009"} \r\n Return: camera zoom {"direction":"in",["sn":"506607D117000009"} \r\n **Without sn parameter case:** Send: camera zoom {"direction":"in"} Return: camera zoom {"direction":"in"} \r\n

3. Obtaining the Camera Position

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
sn	string	NULL	No	Camera Unique Identifier **Note: If the command is requested with the sn parameter, it means that the device with the specified sn is acquired, otherwise the first device in the camera list is acquired.**

- **Return Parameters**

Name	Typology	Range of Values	Parameter Description
x	double	Reference range: [0~3360] There are model differences, subject to actual acquisition	x-axis coordinates **Note: limited by focal length value**
y	double	Reference range: [0~1890] There are model differences, subject to actual acquisition	y-axis coordinates **Note: limited by focal length value**

z	double	Reference range: [-1890~0] There are model differences, subject to actual acquisition	focal length

- **Description**

Camera Command.	**camera get position {["sn":"<value>"]}**
Supported Models	Meetingboard 65/86, Meetingboard 65/75/86 Pro, MeetingEye 500, MeetingBar A10/A40/A50, Yealink RoomConnect, UVC40, UVC85, UVC86, SmartVision 40, AVHub, SmartVision 80 **Note:** **1. For Yealink RoomConnect and AVHub, the 'sn' parameter is required for this command** **2. For MeetingBar A10/A40/A50, the 'sn' parameter is optional** **3. For MeetingBoard 65/86/75-Pro and MeetingEye 500, if there is only one camera, the 'sn' parameter is optional. If multiple cameras are present, you must use the 'sn' parameter to specify the target camera for correct operation** **4. UVC devices and SmartVision devices do not require the 'sn' parameter**
Send Format	camera get position {["sn":"506607D117000009"]} r\n
Return Format	camera get position {"x":1910.0,"y":1060.0,"z":50.5} \r\n
Example	**With sn parameter case:** Send: camera get position {"sn":"506607D117000009"} \r\n Return: camera get position {"x":1910.0,"y":1060.0,"z":50.5} \r\n **Without sn parameter case:** Send: camera get position \r\n Return: camera get position {"x":1910.0,"y":1060.0,"z":50.5} \r\n

4. Setting the Camera Position

- **Request Parameters**

Name	**Typology**	**Range of Values**	**Compulsory**	**Parameter Description**
sn	string	NULL	No	Camera Unique Identifier **Note: If the command is requested with the sn parameter, it means that the device with the specified sn is acquired, otherwise the first device in the camera list is acquired**.

x	double	Reference range: [0~3360] There are model differences, subject to actual acquisition	Yes	x-axis coordinates **Note: limited by the focal length value, the input parameter is of double type, even the integer value needs to have a decimal point, otherwise it can not be successfully executed**
y	double	Reference range: [0~1890] There are model differences, subject to actual acquisition	Yes	y-axis coordinates **Note: limited by the focal length value, the input parameter is of double type, even the integer value needs to have a decimal point, otherwise it can't be successfully executed**
z	double	Reference range: [-1890~0] There are model differences, subject to actual acquisition	Yes	Focus **Note: the input parameter is of type double, even integer values need to have a decimal point, otherwise it will not be executed successfully**

- **Description**

Camera Command.	**camera set position {"x":<value>,"y":<value>,"z":<value>,["sn":"<value>"]}**
Supported Models	Meetingboard 65/86, Meetingboard 65/75/86 Pro, MeetingEye 500, MeetingBar A10/A40/A50, Yealink RoomConnect, UVC40, UVC85, UVC86, SmartVision 40, AVHub, SmartVision 80 **Note:** **1. For Yealink RoomConnect and AVHub, the 'sn' parameter is required for this command** **2. For MeetingBar A10/A40/A50, the 'sn' parameter is optional** **3. For MeetingBoard 65/86/75-Pro and MeetingEye 500, if there is only one camera, the 'sn' parameter is optional. If there are multiple cameras, you must use the 'sn' parameter to specify the target camera for proper operation** **4. UVC devices and SmartVision devices do not require the 'sn' parameter**
Send Format	camera set position {"x":1910.0,"y":1060.0,"z":50.5,["sn":"506607D117000009"]} r\n
Return Format	camera set position {"x":1910.0,"y":1060.0,"z":50.5,["sn":"506607D117000009"]} \r\n

Example	**With sn parameter case:** Send: camera set position {"x":1910.0,"y":1060.0,"z":50.5,"sn":"506607D117000009"} \r\n Return: camera set position {"x":1910.0,"y":1060.0,"z":50.5, sn":"506607D117000009"} \r\n **Without sn parameter case:** Send: camera set position {"x":1910.0,"y":1060.0,"z":50.5} \r\n Return: camera set position {"x":1910.0,"y":1060.0,"z":50.5} \r\n

5. Getting a List of Cameras

- **Return Parameters**

Name	Typology	Parameter Description
sn-list	string	Camera sn list

- **Description**

Camera Command.	camera get list
Supported Models	MeetingBoard 65/86/65 Pro/75 Pro/86 Pro, MeetingEye 500, MeetingBar A10/A40/A50, Yealink RoomConnect, AVHub
Send Format	camera get list\r\n
Return Format	camera get list {"sn-list":["803032E070000031"]} \r\n
Example	Send: camera get list \r\n Return: camera get list {"sn-list":["803032E070000031"]} \r\n

6. Getting Camera Details

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
sn	string	NULL	Yes	Camera unique identification

- **Return Parameters**

Name	Typology	Parameter Description
ip	string	Camera ip
mac	string	Camera mac address
name	string	Camera name
firmware	string	Camera software version
hardware	string	Camera hardware version

spec	string	Camera specifications
model	string	Camera models
sn	string	Camera sn

- **Description**

Camera Command.	**camera get detail {"sn":"<value>"}**
Supported Models	Meetingboard 65/86, Meetingboard 65/75/86 Pro, MeetingEye 500, MeetingBar A10/A40/A50, Yealink RoomConnect
Send Format	camera get detail {"sn":"506607D117000009"} r\n
Return Format	camera get detial {"ip":"169.254.1.150","mac":"80:5E:C0:60:00:62","name":"Yealink UVC84 - 1","firmware":"262.302.5.5","hardware":"262.0.96.0.0.0.0","spec":"PTZ 12x Optical Zoom","model":"UVC84","sn":"8703018090000132"}\r\n
Example	Send: camera get detail {"sn":"506607D117000009"} \r\n Return: camera get detial {"ip":"169.254.1.150","mac":"80:5E:C0:60:00:62","name":"Yealink UVC84 - 1","firmware":"262.302.5.5","hardware":"262.0.96.0.0.0.0","spec":"PTZ 12x Optical Zoom","model":"UVC84","sn":"8703018090000132"} \r\n

7. Setting the Camera AI Mode

- **Request Parameters**

Name	**Typology**	**Range of Values**	**Compulsory**	**Parameter Description**

type	string	[ptz,auto-frame, speaker-tracking, view-cropping, multi-screen,smart-gallery,pip,multi-pip,presenter-tracking,intellifocus,virtual-background,multi-stream-intelliframe]	Yes	Mode Types ptz: Manual control auto-frame: auto framing speaker-tracking: speaker tracking view-cropping: view cropping multi-screen: multi-screen/trs id="5"/> smart-gallery: smart gallery/ multi-stream mode pip: Picture-in-picture multi-pip: multi-screen + picture-in-picture presenter-tracking: presenter tracking mode intellifocus: multi-speaker tracking virtual-background: virtual background multi-stream-Intelliframe: multi-stream Intelliframe
sn	string	NULL	No	Camera Unique Identifier **Note: If the command is requested with the sn parameter, it means that the device with the specified sn is acquired, otherwise the first device in the camera list is acquired**.

- **Description**

Camera Command.	**camera set ai-mode {"type":"<value>",["sn":"<value>"]}**

Supported Models	**MeetingBoard 65/86/65 Pro/75 Pro/86 Pro, MeetingEye 500, MeetingBar A10/A40/A50, Yealink RoomConnect, UVC40, UVC85, UVC86, SmartVision 40, AVHub, SmartVision 80** **Note:** **1. For Yealink RoomConnect, the ‘sn’ parameter is required for this command** **2. For MeetingBar A10/A40/A50, the ‘sn’ parameter is optional; A10 only supports modes: “ptz, auto-frame, speaker-tracking, view-cropping, multi-screen, pip”** **3. For MeetingBoard 65/86/75-Pro and MeetingEye 500, when there's only one camera, you may provide the ‘sn’ parameter optionally. However, if multiple cameras are present, you must use ‘sn’ to specify the target camera for correct operation** **4. UVC40 does not require the ‘sn’ parameter; UVC40 only supports “ptz, auto-frame, speaker-tracking” modes** **5. After MeetingEye 500 connects with UVC86, the supported modes are "ptz, auto-frame, speaker-tracking, smart-gallery, presenter-tracking"; with UVC84, supported modes are "ptz, auto-frame"** **6. MeetingBoard 65/86 supports "ptz, auto-frame, speaker-tracking, multi-screen, pip, smart-gallery, presenter-tracking" modes** **7. UVC86 does not require ‘sn’, and only supports "ptz, auto-frame, speaker-tracking, multi-screen, presenter-tracking" modes** **8. SmartVision 40 does not require ‘sn’; supported modes are "ptz, auto-frame, speaker-tracking, multi-screen, intellifocus, virtual-background, multi-stream-intelliframe"** **9. AVHub requires ‘sn’ and only supports "ptz, auto-frame, speaker-tracking, multi-screen, presenter-tracking" modes** **10. UVC85 does not require ‘sn’; supported modes are "ptz, auto-frame, speaker-tracking, multi-stream-intelliframe"** **11. SmartVision 80 does not require ‘sn’; supported modes are "ptz, auto-frame, speaker-tracking, multi-screen, presenter-tracking, intellifocus"**
Send Format	camera set ai-mode {"type":"ptz",["sn":"506607D117000009"]} r\n
Return Format	camera set ai-mode {"type":"ptz",["sn":"506607D117000009"]} \r\n
Example	**With sn parameter case:** Send: camera set ai-mode {"type":"ptz","sn":"506607D117000009"} \r\n Return: camera set ai-mode {"type":"ptz",["sn":"506607D117000009"} \r\n **Without sn parameter case:** Send: camera set ai-mode {"type":"ptz"} \r\n Return: camera set ai-mode {"type":"ptz"} \r\n

8. Apply Camera Preset Position

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
id	int	[1~99]	Yes	Preset Position ID Supported preset position values for AVHub and UVC86 devices are in the range [1~9]
sn	string	NULL	No	camera unique identification

- **Description**

Camera Command.	camera recall preset {"id":<value>,["sn":"<value>"]}
Supported Models	Meetingboard 65/86, Meetingboard 65/75/86 Pro, MeetingEye 500, MeetingBar A10/A40/A50, Yealink RoomConnect, AVHub, UVC85, UVC86, SmartVision 80 **Note:** **1. For Yealink RoomConnect and AVHub, the 'sn' parameter is required for this command** **2. For MeetingBar A10/A40/A50, the 'sn' parameter is optional** **3. For MeetingBoard 65/86/75-Pro and MeetingEye 500, when there's only one camera, you may provide the 'sn' parameter optionally; if multiple cameras are present, you must use 'sn' to specify the target camera for correct operation** **4. UVC and SmartVision devices do not require the 'sn' parameter**
Send Format	camera recall preset {"id":2,["sn":"8703018090000132"]} \r\n
Return Format	camera recall preset {"id":2,["sn":"8703018090000132"]} \r\n
Example	**With sn parameter case:** Send: camera recall preset {"id":2,"sn":"8703018090000132"} \r\n Return: camera recall preset {"id":2, sn":"8703018090000132"} \r\n **Without sn parameter case:** Send: camera recall preset {"id":2} \r\n Return: camera recall preset {"id":2} \r\n

9. Setting Camera Preset Position

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
id	int	[1~99]	Yes	Preset Position ID Supported preset position values for AVHub, UVC85, and UVC86 devices are in the range [1~9]
sn	string	NULL	No	camera unique identification

- **Description**

Camera Command.	camera set preset {"id":<value>,["sn":"<value>"]}
Supported Models	Meetingboard 65/86, Meetingboard 65/75/86 Pro, MeetingEye 500, MeetingBar A10/A40/A50, Yealink RoomConnect, AVHub, UVC85, UVC86, SmartVision 80 **Note:** **1. For Yealink RoomConnect and AVHub, the "sn" parameter is required for this command.** **2. For MeetingBar A10/A40/A50, the "sn" parameter is optional.** **3. For MeetingBoard 65/86/75-Pro and MeetingEye 500, if only one camera is connected, the "sn" parameter is optional; if multiple cameras are connected, use "sn" to specify the targeted camera for accurate operations.** **4. For UVC devices and SmartVision devices, the "sn" parameter is not needed.**
Send Format	camera set preset {"id":2,["sn":"8703018090000132"]} \r\n
Return Format	camera set preset {"id":2,["sn":"8703018090000132"]} \r\n
Example	**With sn parameter case:** Send: camera set preset {"id":2,"sn":"8703018090000132"} \r\n Return: camera set preset {"id":2, sn":"8703018090000132"} \r\n **Without sn parameter case:** Send: camera set preset {"id":2} \r\n Return: camera set preset {"id":2} \r\n

10) Non-sleep mode switch

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
status	string	[on,off]	Yes	Non-sleep mode status on: enable off: disable
sn	string	NULL	No	camera unique identification

- **Description**

Camera Command.	camera keep-alive set switch {"status":<value>,"sn":<value>}
Supported Models	Yealink RoomConnect **Note: YRC requires the parameter sn to be included in the command.**
Send Format	camera keep-alive set switch {"status":"off","sn":"506011D110000054"} \r\n
Return Format	camera keep-alive set switch {"status":"off","sn":"506011D110000054"} \r\n

Example	Send: camera keep-alive set switch {"status":"off","sn":"506011D110000054"} \r\n Return: camera keep-alive set switch {"status":"off","sn":"506011D110000054"} \r\n

11) Video Fence switch

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
status	string	[on,off]	Yes	Video Fence status on: open off: closed
sn	string	NULL	No	camera unique identification

- **Description**

Camera Command.	camera video-fence set switch {"status":,"sn":}
Supported Models	Yealink RoomConnect, SmartVision 40 Note: For YRC, the sn parameter is required when using this command; SmartVision 40 does not require the sn parameter.
Send Format	camera video-fence set switch {"status":"on","sn":"806009E070000512"} \r\n
Return Format	camera video-fence set switch {"status":"on","sn":"806009E070000512"} \r\n
Example	Send: camera video-fence configuration switch {"status":"on","sn":"806009E070000512"} Return: camera video-fence set switch {"status":"on","sn":"806009E070000512"} \r\n

12) Set Multi-camera Tracking

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description

type	string	[close,multi-camera-intellifocus, mic-camera-linkage,multi-stream-intelliframe,multi-camera-speaker-tracking]	Yes	Mode Types close: Off multi-camera-intellifocus: multi-speaker tracking mic-camera-linkage: mic-camera linkage multi-stream-IntelliFrame: IntelliFrame multi-camera-speaker-tracking: Single speaker tracking

- **Description**

Camera Command.	camera set multi-camera-tracking-function {"type":<value>}
Supported Models	AVHub,Yealink RoomConnect
Send Format	camera set multi-camera-tracking-function {"type":"multi-camera-intellifocus"}\r\n
Return Format	camera set multi-camera-tracking-function {"type":"multi-camera-intellifocus"}\r\n
Example	Send: camera set multi-camera-tracking-function {"type":"multi-camera-intellifocus"}\r\n Return: camera set multi-camera-tracking-function {"type":"multi-camera-intellifocus"}\r\n

13. Retrieve camera status

- **Return Parameters**

Parameter	Typology	Range of Values	Description
status	string	[available,in_use]	Camera usage status: available: not in use in_use: currently in use

- **Description**

Camera Command.	camera get status
Supported Models	SmartVison 40
Send Format	camera get status\r\n
Return Format	camera get status {"status":"in_use"} \r\n
Example	Send: camera get status\r\n Return: camera get status {"status":"in_use"} \r\n

14. Get the AI people count

- **Return Parameters**

Name	Typology	Range of Values	Description
quantity	int	[-1, non-negative integer]	-1: Headcount mode disabled Non-negative integer: Number of people returned when headcount is enabled

- **Description**

Camera Command.	camera get people-count
Supported Models	SmartVision 40
Send Format	camera get people-count\r\n
Return Format	camera get people-count {"quantity":1} \r\n
Example	Send: `camera get people-count\r\n` Reply: ``camera get people-count {"quantity":1} \r\n

15. Read video parameters

- **Return Parameters**

Name	Typology	Range of Values	Description
Brightness	int	value [0,100] or -1 (nosupport)	SmartVision 40 supports returning values
Staturation	int	value [0,100] or [0,10]	SmartVision 40: [0,100]
Contrast	int	value [0,100] or -1 (nosupport)	SmartVision 40 supports returning values
WhiteBalance	int	value [2800,6800] or -1 (nosupport)	Only in manual white balance mode will a value be returned.
WhiteBalanceMode	string	[AutoWhiteBalance,ManualWhiteBalance,OnePush, ` ` Incandescent,Fluorescent,Daylight,CloudyDaylight,Shade]	SmartVision 40 supports Auto White Balance, Manual White Balance, and One-Push.
sharp	int	[0-31]	SmartVision 40 range: [0-31]

- **Description**

Camera Command.	camera get video-parameter
Supported Models	**SmartVision 40**
Send Format	camera get video-paremeter\r\n

Return Format	camera get video-parameter {"whiteBalance":-1,"whiteBalanceMode":"AutoWhiteBalance","contrast":50,"brightness":50,"saturation":50,"sharpness":16} \r\n
Example	Send: camera get video-parameter\r\n Return: camera get video-parameter {"whiteBalance":-1,"whiteBalanceMode":"AutoWhiteBalance","contrast":50,"brightness":50,"saturation":50,"sharpness":16} \r\n

16. Retrieve Video Wall Status

- **Return Parameters**

Name	Typology	Range of Values	Description
status	string	[on,off]	Video wall status: on: enabled off: disabled

- **Description**

Camera Command.	camera get people-count
Supported Models	SmartVision 40
Send Format	camera get people-count\r\n
Return Format	camera get people-count {"quantity":1} \r\n
Example	Send: camera get people-count\r\n Reply: ``camera get people-count {"quantity":1} \r\n

17. Set Active Camera

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
sn	string	NULL	No	camera unique identification

- **Description**

Camera Command.	camera set active {"sn":" <value>"}
Supported Models	MeetingBoard 65/86/75-Pro, MeetingEye 500, Avhub
Send Format	Activate camera {"sn":"8703018090000132"}
Return Format	Activate camera {"sn":"8703018090000132"}
Example	**With sn parameter case:** Send: Activate camera {"sn":"8703018090000132"} \r\n Return: Activate camera {"sn":"8703018090000132"}

Screen Syntax

1. Getting Screen Brightness

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
id	int	NULL	No	Screen ID (value taken from the device identifier obtained in the device list information)

- **Return Parameters**

Name	Typology	Range of Values	Parameter Description
value	int	[1~100]	screen brightness value

- **Description**

Screen Command	screen get brightness
Supported Models	Meetingboard 65/86, Meetingboard 65/75/86 Pro, MeetingDisplay
Send Format	screen get brightness {"id":0} \r\n
Return Format	screen get brightness {"value":50}\r\n
Example	Send: screen get brightness {"id":0} \r\n Return: screen get brightness {"value":50} \r\n

2. Setting Screen Brightness

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
id	int	NULL	No	Screen ID (value taken from the device identifier obtained in the device list information)
value	int	[1~100]	Yes	screen brightness value

- **Description**

Screen Command	screen set brightness {"value":<value>}
Supported Models	Meetingboard 65/86, Meetingboard 65/75/86 Pro, MeetingDisplay
Send Format	screen set brightness {"id":0, "value":80} \r\n
Return Format	screen set brightness {"id":0, "value":80} \r\n
Example	Send: screen set brightness {"id":0,"value":80} \r\n Return: screen set brightness {"id":0, "value":80} \r\n

3. Retrieve display parameters

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
id	int	NULL	No	Screen ID (obtain from device list, default is 0)

- **Return Parameters**

Name	Typology	Range of Values	Parameter Description
contrast	int	[1~100]	Screen contrast ratio
saturation	int	[1~100]	Screen Saturation Value
color-temperature	int	[0~3]	Color Temperature 0: Warm 1: Default 2: Cool 3: Custom

- **Description**

Screen Command	screen get display-parameter
Supported Models	Meetingboard 65/86, Meetingboard 65/75/86 Pro, MeetingDisplay
Send Format	screen get display-parameter {"id": 0} \r\n
Return Format	screen get display-parameter {"contrast": 60,"saturation": 60,"color-temperature": 0} \r\n
Example	Send: screen get display-parameter {"id": 0} \r\n Return: screen get display-parameter {"contrast": 60,"saturation": 60,"color-temperature": 0} \r\n

4. Retrieve color temperature information list

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
id	int	NULL	No	Screen ID (obtain from device list, default is 0)

- Return Parameters

Name	Typology	Range of Values	Note
color-temperature-list	color_temperature_list []	NULL	Color temperature information list

color_temperature_list

Name	Typology	Note
value	int	Color Temperature Mode
name	string	Color Temperature Name

- **Description**

Screen Command	screen get color-temperature-info
Supported Models	Meetingboard 65/86, Meetingboard 65/75/86 Pro, MeetingDisplay
Send Format	screen get color-temperature-info {"id": 0} \r\n
Return Format	screen get color-temperature-info {"color-temperature-list": [{"value":0, "name":"warm"}, {"value":1, "name":"default"}, {"value":2, "name":"cold"}, {"value":3, "name":"custom"}]}\r\n
Example	Send: screen get color-temperature-info {"id": 0} \r\n Return: screen get color-temperature-info {"color-temperature-list": [{"value":0, "name":"warm"}, {"value":1, "name":"default"}, {"value":2, "name":"cold"}, {"value":3, "name":"custom"}]}\r\n

5. Configure display settings

- **Request Parameters**

Name	Typology	Range of Values	Parameter Description
id	int	NULL	Screen ID (obtain from device list, default is 0)
contrast	int	[1~100]	Screen contrast ratio
saturation	int	[1~100]	Screen Saturation Value
color-temperature	int	[0~3]	Color Temperature 0: Warm 1: Default 2: Cool 3: Custom

- **Description**

Screen Command	screen set display-parameter {"value":<value>}
Supported Models	Meetingboard 65/86, Meetingboard 65/75/86 Pro, MeetingDisplay
Send Format	screen set display-parameter {"id":0, "contrast": 60,"saturation": 60,"color-temperature": 0} \r\n
Return Format	screen set display-parameter {"id":0, "contrast": 60,"saturation": 60,"color-temperature": 0} \r\n

Example	Send: screen set display-parameter {"id":0, "contrast": 60,"saturation": 60,"color-temperature": 0} \r\n Return: screen set display-parameter {"id":0, "contrast": 60,"saturation": 60,"color-temperature": 0} \r\n

6. Reset display settings

- **Request Parameters**

Name	Typology	Range of Values	Parameter Description
id	int	NULL	Screen ID (obtain from device list, default is 0)

- **Description**

Screen Command	screen reset display-parameter {"value":<value>}
Supported Models	Meetingboard 65/86, Meetingboard 65/75/86 Pro, MeetingDisplay
Send Format	screen reset display-parameter {"id":0} \r\n
Return Format	screen reset display-parameter {"id":0} \r\n
Example	Send: screen reset display-parameter {"id":0} \r\n Return: screen reset display-parameter {"id":0} \r\n

7. Set Custom Color Temperature

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
id	int	NULL	No	Screen ID (obtain from device list, default is 0)
red	int	[0~255]	Yes	Red temperature value
green	int	[0~255]	Yes	Green temperature value
blue	int	[0~255]	Yes	Blue temperature value

- **Description**

Screen Command	screen set custom-color-temperature {"red": <value>,"green": <value>,"blue": <value>}
Supported Models	Meetingboard 65/86, Meetingboard 65/75/86 Pro, MeetingDisplay
Send Format	screen set custom-color-temperature {"id":0, "red": 128,"green": 116,"blue": 131} \r\n
Return Format	screen set custom-color-temperature {"id":0, "red": 128,"green": 116,"blue": 131} \r\n

Example	Send: screen set custom-color-temperature {"id":0, "red": 128,"green": 116,"blue": 131} \r\n Return: screen set custom-color-temperature {"id":0, "red": 128,"green": 116,"blue": 131} \r\n

8. Get list of signal sources

- **Return Parameters**

Name	Typology	Range of Values	Note
input-source-list	source_list []	NULL	Input source list

source_list

Name	Typology	Note
type	string	Screen input source types Default: Default Input android: android OS windows: windows OS HdmiIn: hdmi input typec: Type-C input HdmiIn 1: hdmi 1 input HdmiIn 2: hdmi 2 input HdmiIn 3: hdmi 3 input

- **Description**

Screen Command	input-source get list
Supported Models	MeetingBoard 65/86, MeetingDisplay
Send Format	input-source get list \r\n
Return Format	input-source get list {"input-source-list": [{"type":"Android"}, {"type":"HdmiIn 1"}, {"type":"HdmiIn 2"}, {"type":"HdmiIn 3"}]} \r\n
Example	Send: input-source get list \r\n Return: input-source get list {"input-source-list": [{"type":"Android"}, {"type":"HdmiIn 1"}, {"type":"HdmiIn 2"}, {"type":"HdmiIn 3"}]} \r\n

9. Get current signal source

- **Return Parameters**

Name	Typology	Note

type	string	Screen input source types Default: Default Input android: android OS windows: windows OS HdmiIn: hdmi input typec: Type-C input HdmiIn 1: hdmi 1 input HdmiIn 2: hdmi 2 input HdmiIn 3: hdmi 3 input

- **Description**

Screen Command	input-source get current
Supported Models	MeetingBoard 65/86, MeetingDisplay
Send Format	input-source get current \r\n
Return Format	input-source get current {"type":"HdmiIn 1"} \r\n
Example	Send: input-source get current \r\n Return: input-source get current {"type":"HdmiIn 1"} \r\n

10. Set current input source

- **Request Parameters**

Name	Typology	Note
type	string	Screen Input Source Type Default: Default Input android: android OS windows: windows OS HdmiIn: hdmi Input typec: Type-C Input HdmiIn 1: hdmi 1 Input HdmiIn 2: hdmi 2 Input HdmiIn 3: hdmi 3 Input

- **Description**

Screen Command	input-source set current {"type":<value>}
Supported Models	MeetingBoard 65/86, MeetingDisplay
Send Format	input-source set current {"type":"Android"} \r\n
Return Format	input-source set current {"type":"Android"} \r\n

Example	Send: input-source switched to {"type":"Android"} Return: input-source set current {"type":"Android"}

11. Get the default boot input source

- **Return Parameters**

Name	Typology	Note
type	string	Screen Input Source Type Default: Default Input android: android OS windows: windows OS HdmiIn: hdmi Input typec: Type-C Input HdmiIn 1: hdmi 1 Input HdmiIn 2: hdmi 2 Input HdmiIn 3: hdmi 3 Input

- **Description**

Screen Command	input-source get default
Supported Models	MeetingBoard 65/86, MeetingDisplay
Send Format	input-source get default \r\n
Return Format	input-source get default {"type":"HdmiIn 3"} \r\n
Example	Send： input-source get default \r\n Return: input-source get default {"type":"HdmiIn 3"} \r\n

12. Set default input source on startup

- **Return Parameters**

Name	Typology	Note
type	string	Screen Input Source Type Default: Default Input android: android OS windows: windows OS HdmiIn: hdmi Input typec: Type-C Input HdmiIn 1: hdmi 1 Input HdmiIn 2: hdmi 2 Input HdmiIn 3: hdmi 3 Input

- **Description**

Screen Command	input-source set default {"type": <value>}

Supported Models	MeetingBoard 65/86, MeetingDisplay
Send Format	input-source set default {"type":"HdmiIn 1"} \r\n
Return Format	input-source set default {"type":"HdmiIn 1"} \r\n
Example	Send: set input source to default: {"type":"HdmiIn 1"} \r\n Return: input-source set default {"type":"HdmiIn 1"} \r\n

app syntax

1. Retrieve the foreground app

- **Return Parameters**

Name	Typology	Note
id	string	App package name
name	string	App name
version	string	App version number

- Description

System command	app get foreground
Supported Models	Meetingboard 65/86, Meetingboard 65/75/86 Pro, MeetingDisplay, MeetingEye 500, MeetingBar A10/A40/A50
Send Format	app get foreground\r\n
Return Format	app get foreground {"id":"com.yealink.BYOD","name":"BYOD","version":"1.0"} \r\n
Example	Send: Request app foreground status\r\n Return: app get foreground {"id":"com.yealink.BYOD","name":"BYOD","version":"1.0"}

2. Retrieve app information list.

- **Return Parameters**

Name	Typology	Range of Values	Note
app-list	app_list []	NULL	List of device information

app_list

Name	Typology	Note
id	string	App package name
name	string	App name
version	string	App version number

- Description

System command	app info
Supported Models	Meetingboard 65/86, Meetingboard 65/75/86 Pro, MeetingDisplay, MeetingEye 500, MeetingBar A10/A40/A50
Send Format	app info\r\n
Return Format	app info {"app-list": [{"id":"com.yealink.BYOD","name":"BYOD","version":"2.0"}, {"id":"us.Zoom.zoompresence","name":"Zoom Rooms","version":"6.0.6.4892"}]}\r\n
Example	Send: app Information\r\n Return: app info {"app-list": [{"id":"com.yealink.BYOD","name":"BYOD","version":"2.0"}, {"id":"us.Zoom.zoompresence","name":"Zoom Rooms","version":"6.0.6.4892"}]}\r\n

3) Bring the App to the foretend.

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
id	string	Retrieve the ID value from the app information list by referring to the basic information document interface.	Yes	App package name

- Description

System command	app start {"id":<value>}
Supported Models	Meetingboard 65/86, Meetingboard 65/75/86 Pro, MeetingDisplay, MeetingEye 500, MeetingBar A10/A40/A50
Send Format	app start {"id":"com.yealink.byod"} \r\n
Return Format	app start {"id":"com.yealink.byod"} \r\n
Example	Send： app launch {"id":"com.yealink.byod"} \r\n Response: app start {"id":"com.yealink.byod"} \r\n

Audio Syntax

1. Retrieve audio source list

- **Return Parameters**

Name	Typology	Parameter Description

input-source	string	Input source list
output-source	string	Output source list

- **Description**

Audio Command	audio get source-info
Supported Models	Meetingboard 65/86, Meetingboard 65/75/86 Pro, MeetingDisplay, MeetingEye 500, MeetingBar A10/A40/A50
Send Format	audio get source-info \r\n
Return Format	audio get source-info {"input-source": [],"output-source": ["AUTO","HDMI"]}\r\n
Example	Send: audio get source-info \r\n Return: audio get source-info {"input-source": [],"output-source": ["AUTO","HDMI"]} \r\n

2. Setting the Audio Input Source

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description

type	string	Only supported for retrieving the list of audio sources to obtain input source information.	Yes	Audio input source identification AUTO: Automatic VCP: CP96X device LINE: LINE input USB_LINE: USB to LINE input BUILT_IN: Built-in mic HANDSET: Controller BT_HANDSET: Bluetooth Controller WIRED_MIC: Wired Microphone WIRELESS_MIC: Wireless Microphone XLR: Three-prong power input

- **Description**

Audio Command	audio set input-source {"type":"<value>"}
Supported Models	Meetingboard 65/86, Meetingboard 65/75/86 Pro, MeetingEye 500, MeetingBar A10/A40/A50
Send Format	audio set input-source {"type":"VCP"} \r\n
Return Format	audio set input-source {"type":"VCP"} \r\n
Example	Send: audio set input-source {"type":"VCP"} \r\n Return: audio set input-source {"type":"VCP"} \r\n

3. Setting the Audio Output Source

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
type	string	Only supports retrieving the list of audio sources to obtain output source details.	Yes	Audio Output Source ID

AUTO: AUTO

VCP: CP96X device

hdmi: hdmi

LINE: LINE Out

USB_LINE: USB to LINE Output

BUILT_IN: Built-in Speaker

HEADSET: Headset

BT_HEADSET: Bluetooth Headset

WIRED_SPEAKER: Wired Speaker

- **Description**

Audio Command	audio set output-source {"type":"<value>"}
Supported Models	Meetingboard 65/86, Meetingboard 65/75/86 Pro, MeetingDisplay, MeetingEye 500, MeetingBar A10/A40/A50
Send Format	audio set output-source {"type":"LINE"} \r\n
Return Format	audio set output-source {"type":"LINE"} \r\n
Example	Send: audio set output-source {"type":"LINE"} \r\n Return: audio set output-source {"type":"LINE"} \r\n

4. Retrieve Audio mute Status (Deprecated, use mic-mute)

- **Return Parameters**

Name	Typology	Range of Values	Parameter Description
status	string	[on,off]	Mute status on: on off: off

- **Description**

Audio Command	audio get mute
Supported Models	Meetingboard 65/86, Meetingboard 65/75/86 Pro, MeetingEye 500, MeetingBar A10/A40/A50, Yealink RoomConnect, UVC40, SmartVision 40
Send Format	audio get mute\r\n
Return Format	audio get mute {"status":"on"}\r\n
Example	Send: audio get mute \r\n Return: audio get mute {"status":"on"} \r\n

5. Setting the Audio Mute State（MVC）

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
status	string	[on,off]	Yes	Mute status on: on off: off

- **Description**

Audio Command	audio set mute {"status":"<value>"}
Supported Models	UVC40, Yealink RoomConnect, SmartVision 40, AP08, AVBridge, CM20, CM50, CS10, CS10-D

Note:

1. AP08, AVBridge, CM20, CM50, CS10, CS10-D set mute status for all device channels, and are not synchronized with the status fetched/set from the single-channel mute API.
   | Send Format| audio set mute {"status":"off"}\r\n
   | Return Format| audio set mute {"status":"off"}\r\n
   | Example| Send:
   audio set mute {"status":"off"} \r\n
   Return:
   audio set mute {"status":"off"} \r\n

6. Setting the Audio Mute State（VCS）

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
key	string	[mute]	Yes	key name

- **Description**

Button Command	button {"key": "mute"}
Supported Models	Meetingboard 65/86, Meetingboard 65/75/86 Pro, MeetingDisplay, MeetingEye 500, MeetingBar A10/A40/A50
Send Format	button {"key": "mute"}\r\n
Return Format	button {"key": "mute"}\r\n
Example	Send: button {"key": "mute"} \r\n Return: button {"key": "mute"} \r\n

7. Getting the Volume

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
type	string	[idle,talk]	No	Volume type idle: ringer volume talk: call volume

- **Return Parameters**

Name	Typology	Range of Values	Parameter Description
value	int	[0~15]	loudness value

- **Description**

Audio Command	audio get volume {["type":"<value>"]}
Supported Models	Meetingboard 65/86, Meetingboard 65/75/86 Pro, MeetingDisplay, MeetingEye 500, MeetingBar A10/A40/A50, UVC40, Yealink RoomConnect **Note:

1. When the UVC40 device calls this API, no parameters are required.

2. For MeetingBoard 65/86/MeetingEye 500, the volume type is optional, default is "idle" (in Single Volume Mode, idle and talk volumes are synchronized).**
   | Send Format| audio get volume {["type":"idle"]} \r\n
   | Return Format| audio get volume {"value":5}\r\n
   | Example| **With type situation:**
   Send:
   audio get volume {"type":"idle"} \r\n
   Return:
   audio get volume {"value":5} \r\n
   不带type情况:
   Send:
   audio get volume \r\n
   Return:
   audio get volume {"value":5} \r\n

8. Setting the Volume

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
value	int	[0~15]	Yes	loudness value
type	string	[idle,talk]	No	Volume type idle: ringer volume talk: call volume

- **Description**

Audio Command	audio set volume {"value":<value>,["type":"<value>"]}
Supported Models	Meetingboard 65/86, Meetingboard 65/75/86 Pro, MeetingDisplay, MeetingEye 500, MeetingBar A10/A40/A50, Yealink RoomConnect **Note:

1. For MeetingBoard 65/86/MeetingEye 500, the volume type is optional, default is "idle" (in Single Volume Mode, idle and talk volumes are synchronized).**
   | Send Format| audio set volume {"value":5,["type":"idle"]} \r\n
   | Return Format| audio set volume {"value":5,["type":"idle"]} \r\n
   | Example| **With type situation:**
   Send:
   audio set volume {"value":5,"type":"idle"} \r\n
   Return:
   audio get volume {"value":5, type":""} \r\n
   Without type situation:
   Send:
   audio set volume {"value":5}\r\n
   Return:
   audio set volume {"value":5} \r\n

9) Set input audio noise reduction.

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
ai-mode	string	[on,off]	No	AI intelligent noise reduction mode on: enable off: disable
level	int	[0,1,2.3]	No	Inhibition Level 0: Off 1: Weak 2: Normal 3: Strong
sn	string	NULL	No	Device unique identifier

- **Description**

Audio Command	audio input set noise-reduction {"sn":"<value>","ai-mode":"<value>","level":"<value>"}
Supported Models	Yealink RoomConnect

Note:

1. Supported models for ai-mode: avhub, uvc84, uvc86
2. Supported models for level: avhub, uvc84, uvc86, vcm34, vcm35, vcm38, cm20
   | Send Format| audio input set noise-reduction {"sn":"506607D117000009","ai-mode":"on","level":2} \r\n
   | Return Format| audio input set noise-reduction {"sn":"506607D117000009","ai-mode":"on","level":2} \r\n
   | Example| Send:

   audio input set noise-reduction {"sn":"506607D117000009","ai-mode":"on","level":2} \r\n

   Return:

   audio input set noise-reduction {"sn":"506607D117000009","ai-mode":"on","level":2} \r\n

10) Set input audio gain.

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
status	string	[on,off]	Yes	Gain control status: on - enabled off - disabled
gain-value	int	[-30~30]	No	Gain adjustment value
rca-value	int	[-12~40]	No	RCA input adjustment value
line-value	int	[-12~40]	No	Line input adjustment value
attenuation-value	int	[-30~30]	No	Attenuation Adjustment Value
sn	string	NULL	No	Device unique identifier

- **Description**

Audio Command	audio input set gain {"status":"<value>","sn":"<value>","gain-value":"<value>","rca-value":"<value>","line-value":"<value>"}
Supported Models	Yealink RoomConnect

Note:

1. rca-value and line-value parameters only affect the adjusted device if it’ s avhub
   | Send Format| audio input set gain {"status":"on","sn":"506011D110000054","gain-value":20,"rca-value":-6,"attenuation-value":20,"line-value":12} \r\n
   | Return Format| audio input set gain {"status":"on","sn":"506011D110000054","gain-value":20,"rca-value":-6,"attenuation-value":20,"line-value":12} \r\n
   | Example| Send:

   audio input set gain {"status":"on","sn":"506011D110000054","gain-value":20,"rca-value":-6,"attenuation-value":20,"line-value":12} \r\n

   Return:

   audio input set gain {"status":"on","sn":"506011D110000054","gain-value":20,"rca-value":-6,"attenuation-value":20,"line-value":12} \r\n

11) Set up input audio echo cancellation.

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
status	string	[on,off]	Yes	Echo cancellation status on: enable off: disable
suppress-level	int	[0,1,2]	No	Echo cancellation level 0: Low 1: Medium 2: High
reverb-level	int	[0,1,2]	No	Room reverberation level 0: Low 1: Medium 2: High
manual-dalay	string	[on,off]	No	Manual delay on: Turn on off: Turn off
delay-value	int	[-100~500]	No	Audio AEC delay count (unit: ms)
sn	string	NULL	No	Device unique identifier

- **Description**

Audio Command	audio input set echo-cancellation {"status":"<value>","sn":"<value>","suppress-level":"<value>","reverb-level":"<value>","manual-dalay":"<value>","delay-value":"<value>"}
Supported Models	Yealink RoomConnect
Send Format	audio input set echo-cancellation {"status":"on","sn":"506607D117000009","suppress-level":1,"reverb-level":0,"manual-dalay":"on","delay-value":200} \r\n

Return Format	audio input set echo-cancellation {"status":"on","sn":"506607D117000009","suppress-level":1,"reverb-level":0,"manual-delay":"on","delay-value":200} \r\n
Example	Send: audio input: echo cancellation enabled {"status":"on","sn":"506607D117000009","suppress-level":1,"reverb-level":0,"manual-delay":"on","delay-value":200} \r\n Return: audio input set echo-cancellation {"status":"on","sn":"506607D117000009","suppress-level":1,"reverb-level":0,"manual-dalay":"on","delay-value":200} \r\n

12) Set input audio equalizer

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
status	string	[on,off]	Yes	Equalizer status on: enable off: disable
mode	string	[custom,bass-boost, treble-boost,vocal-boost]	Yes	Equalizer mode custom Custom bass-boost: Bass boost treble-boost: Treble boost vocal-boost: Vocal boost
frequency-range-list	int	[-12~12]	No	Custom mode frequency band list, the data in the list corresponds to: ``20Hz, 63Hz, 125Hz, 250Hz, 500Hz, 1000Hz, 2000Hz, 4000Hz, 8000Hz, 20000Hz``
sn	string	NULL	No	Device unique identifier

- **Description**

Audio Command	audio input set equalizer {"status":"<value>","sn":"<value>","mode":"<value>","frequency-range-list":"<value>"}
Supported Models	Yealink RoomConnect Note: 1. The rca-value and line-value parameters are only applicable to avhub devices for adjustment.
Send Format	audio input set equalizer {"status":"on","sn":"506011D110000054","mode":"custom","frequency-range-list": [12,5,-12,3,6,10,-5,2,0,3,9]} \r\n
Return Format	audio input set equalizer {"status":"on","sn":"506011D110000054","mode":"custom","frequency-range-list": [12,5,-12,3,6,10,-5,2,0,3,9]} \r\n

Example	Send: audio input: set equalizer {"status":"on","sn":"506011D110000054","mode":"custom","frequency-range-list": [12,5,-12,3,6,10,-5,2,0,3,9]} \r\n Return: audio input set equalizer {"status":"on","sn":"506011D110000054","mode":"custom","frequency-range-list": [12,5,-12,3,6,10,-5,2,0,3,9]} \r\n

13) Set output audio gain.

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
status	string	[on,off]	Yes	Gain control status: on - enabled off - disabled
gain-value	int	[-30~30]	No	Gain adjustment value
attenuation-value	int	[-30~30]	No	Attenuation Adjustment Value
line-value	int	[-50-30]	No	Line output adjustment value
sn	string	NULL	No	Device unique identifier

- **Description**

| Audio Command| xml
<command>audio output set gain</command> {"status":" <value>","sn":" <value>","gain-value":" <value>"}
|----------|----------
| Supported Models| Yealink RoomConnect
Note:
1. The rca-value and line-value parameters are only applicable to avhub devices for adjustment.
| Send Format| audio output set gain {"status":"on","sn":"506011D110000054","gain-value":20,"attenuation-value":18,"line-value":10} \r\n
| Return Format| audio output set gain {"status":"on","sn":"506011D110000054","gain-value":20,"attenuation-value":18,"line-value":10} \r\n
| Example| Send:
audio output: set gain {"status":"on","sn":"506011D110000054","gain-value":20,"attenuation-value":18,"line-value":10} \r\n
Return:
audio output set gain {"status":"on","sn":"506011D110000054","gain-value":20,"attenuation-value":18,"line-value":10} \r\n

14) Set up output audio equalizer.

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description

status	string	[on,off]	Yes	Equalizer status on: enable off: disable
mode	string	[custom,bass-boost, treble-boost,vocal-boost]	Yes	Equalizer mode custom Custom bass-boost: Bass boost treble-boost: Treble boost vocal-boost: Vocal boost
frequency-range-list	int	[-12~12]	No	Custom mode frequency band list, the data in the list corresponds to: ``20Hz, 63Hz, 125Hz, 250Hz, 500Hz, 1000Hz, 2000Hz, 4000Hz, 8000Hz, 20000Hz``
sn	string	NULL	No	Device unique identifier

- **Description**

| Audio Command| xml
<status></status>
<sn></sn>
<mode></mode>
<frequency-range-list></frequency-range-list>
|----------|----------
| Supported Models| Yealink RoomConnect
Note:
1. The rca-value and line-value parameters are only applicable to avhub devices for adjustment.
| Send Format| audio output set equalizer {"status":"on","sn":"506011D110000054","mode":"custom","frequency-range-list": [12,5,-12,3,6,10,-5,2,0,3,9]} \r\n
| Return Format| audio output set equalizer {"status":"on","sn":"506011D110000054","mode":"custom","frequency-range-list": [12,5,-12,3,6,10,-5,2,0,3,9]}\r\n
| Example| Send：
audio output: set equalizer {"status":"on","sn":"506011D110000054","mode":"custom","frequency-range-list": [12,5,-12,3,6,10,-5,2,0,3,9]} \r\n
Return:
audio output set equalizer {"status":"on","sn":"506011D110000054","mode":"custom","frequency-range-list": [12,5,-12,3,6,10,-5,2,0,3,9]} \r\n

15. Get mic mute status

- **Return Parameters**

Name	Typology	Range of Values	Parameter Description
status	string	[on,off]	Mute status on: on off: off

- **Description**

Audio Command	audio get mic-mute

Supported Models	Meetingboard 65/86, Meetingboard 65/75/86 Pro, MeetingEye 500, MeetingBar A10/A40/A50, Yealink RoomConnect, UVC40, SmartVision 40
Send Format	audio get mic-mute\r\n
Return Format	audio get mic-mute {"status":"on"} \r\n
Example	Send: audio get mic-mute \r\n Return: audio get mic-mute {"status":"on"} \r\n

16. Retrieve Speaker Mute Status

- **Return Parameters**

Name	Typology	Range of Values	Parameter Description
status	string	[on,off]	Mute status: on: enabled off: disabled

- **Description**

aduio command	audio get speaker-mute
Supported Models	SmartVision 40
Send Format	audio get speaker-mute\r\n
Return Format	audio get speaker-mute {"status":"off"} \r\n
Example	Send: `audio get speaker-mute` Return:``audio get speaker-mute {"status":"off"}

17. Retrieve Audio Fence (audio wall) switch status

- **Return Parameters**

Name	Typology	Range of Values	Description
status	string	[on,off]	Audio wall switch status: on: enabled off: disabled

- **Description**

Audio Command	audio get fence-enable
Supported Models	SmartVision 40
Send Format	audio get fence-enable\r\n
Return Format	audio get fence-enable {"status":"off"} \r\n

Example	Send: audio get fence-enable\r\n Return: audio get fence-enable {"status":"off"} \r\n

18. Get device audio channel information (deprecated; use 19 instead)

- **Return Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
max_channel_count	int	-	Yes	Maximum number of channels
channel_enable	bool []	-	Yes	Channel Enable Status

- **Description**

Audio Command	audio get device-sound-channel-info
Supported Models	CM50
Send Format	audio get device-sound-channel-info \r\n
Return Format	audio get device-sound-channel-info {"max_channel_count":8,"channel_enable":[true,true,true,true,true,true,true,true]} \r\n
Example	Send: audio get device-sound-channel-info \r\n Return: audio get device-sound-channel-info {"max_channel_count":8,"channel_enable":[true,true,true,true,true,true,true,true]} \r\n

19. Retrieve device channel audio source information

- **Return Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
max_channel_count	int	-	Yes	Maximum number of channels
channel_enable	bool []	-	Yes	Channel Enable Status

- **Description**

Audio Command	audio get lobe-enable
Supported Models	CM50
Send Format	audio get lobe-enable\r\n
Return Format	audio get lobe-enable {"max_channel_count":8,"channel_enable":[true,true,true,true,true,true,true,true]}\r\n

Example	Send: audio get lobe-enable\r\n Return: audio get lobe-enable {"max_channel_count":8,"channel_enable": [true,true,true,true,true,true,true,true]}\r\n

20. Get device audio channel activation status (deprecated; use 21 instead)

- **Return Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
active_status	bool []	NULL	Yes	Channel Activation Status

- **Description**

Audio Command	audio get device-sound-active-status
Supported Models	CM50
Send Format	audio get device-sound-active-status\r\n
Return Format	audio get device-sound-active-status {"active_status": [true,true,true,true,true,true,true,true]} \r\n
Example	Send: audio get device-sound-active-status \r\n Return: audio get device-sound-active-status {"active_status": [true,true,true,true,true,true,true,true]} \r\n

21. Retrieve device audio channel activation status

- **Return Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
active_status	bool []	NULL	Yes	Channel Activation Status

- **Description**

Audio Command	audio get lobe-activation
Supported Models	CM50
Send Format	audio get lobe-activation\r\n
Return Format	audio get lobe-activation {"active_status": [true,true,true,true,true,true,true,true]}\r\n
Example	Send: audio get lobe-activation\r\n Return: audio get lobe-activation {"active_status": [true,true,true,true,true,true,true,true]}\r\n

22. Get speaker position coordinates (deprecated; use 23 instead)

- **Return Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
doa	doa []	NULL	Yes	Device doa array information

doa

Name	Typology	Range of Values	Compulsory	Parameter Description
dev_type	string	NULL	Yes	Equipment type
dev_sn	string	NULL	Yes	Device Serial Number
snd_src_num	int	NULL	Yes	Number of sound sources
snd_src	snd_src []	NULL	Yes	Audio source data

snd_src

Name	Typology	Range of Values	Compulsory	Parameter Description
x	int	NULL	Yes	X-axis values
y	int	NULL	Yes	Y-axis value
z	int	NULL	Yes	Z-axis value

- **Description**

Audio Command	audio get device-sound-talker-position
Supported Models	CM50
Send Format	audio get device-sound-talker-position
Return Format	audio get device-sound-talker-position {"doa": [{"dev_type":"CM50","dev_sn":"sfsfsfsff","snd_src_num":1,"snd_src": [{"x":0,"y":0,"z":0}]}]}\r\n
Example	Send: audio get device-sound-talker-position\r\n Return: audio get device-sound-talker-position {"doa": [{"dev_type":"CM50","dev_sn":"sfsfsfsff","snd_src_num":1,"snd_src": [{"x":0,"y":0,"z":0}]}]}\r\n

23. Retrieve speaker location coordinates

- **Return Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
doa	doa []	NULL	Yes	Device doa array information

doa

Name	Typology	Range of Values	Compulsory	Parameter Description
dev_type	string	NULL	Yes	Equipment type
dev_sn	string	NULL	Yes	Device Serial Number
snd_src_num	int	NULL	Yes	Number of sound sources
snd_src	snd_src []	NULL	Yes	Audio source data

snd_src

Name	Typology	Range of Values	Compulsory	Parameter Description
x	int	NULL	Yes	X-axis values
y	int	NULL	Yes	Y-axis value
z	int	NULL	Yes	Z-axis value

- **Description**

Audio Command	audio get sound-locate
Supported Models	CM50
Send Format	audio get sound-locate
Return Format	audio get sound-locate {"doa": [{"dev_type":"CM50","dev_sn":"sfsfsfsff","snd_src_num":1,"snd_src": [{"x":0,"y":0,"z":0}]}]}\r\n
Example	Send: audio get sound-locate\r\n Return: audio get sound-locate {"doa": [{"dev_type":"CM50","dev_sn":"sfsfsfsff","snd_src_num":1,"snd_src": [{"x":0,"y":0,"z":0}]}]}\r\n

24. Get audio parameter presets (deprecated, use 25 instead)

- **Request Parameters**

Name	Typology	Required	Constraint	Description
» preset_info	object	true	none	none
»» preset_type	string	true	none	none
»» preset_id	string	true	none	Preset Unique ID

»» preset_name	string	true	none	When preset_type is set to official preset, the name and id are identical.

- **Description**

Audio Command	audio get param-preset
Supported Models	AP08, CM50, CM20
Send Format	audio get param-preset \r\n
Return Format	audio get param-preset {"preset_info": [{"preset_type":" <value>","preset_name":" <value>","preset_id":" <value>"}]}
Example	Send: audio parameter preset applied. Return: audio set param-preset {"preset_info": [{"preset_type":"custom_preset","preset_name":"Custom 1","preset_id":"OFQjdE"}, {"preset_type":"official_preset","preset_name":"General","preset_id":"general_preset"}]} \r\n

25. Retrieve audio parameter presets

- **Request Parameters**

Name	Typology	Required	Constraint	Description
» preset_info	object	true	none	none
»» preset_type	string	true	none	none
»» preset_id	string	true	none	Preset Unique ID
»» preset_name	string	true	none	When preset_type is set to official preset, the name and id are identical.

- **Description**

Audio Command	audio get preset
Supported Models	AP08, AVBridge, CM50, CM20
Send Format	audio get preset\r\n

Return Format	audio get preset {"preset_info": [{"preset_type":"<value>","preset_name":"<value>","preset_id":"<value>"}]}\r\n
Example	Send: audio set preset Return: audio set preset {"preset_info": [{"preset_type":"custom_preset","preset_name":"Custom 1","preset_id":"OFQjdE"}, {"preset_type":"official_preset","preset_name":"General preset","preset_id":"general_preset"}]}

26. Configure audio parameter presets (deprecated, use 27 instead)

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
preset_name	string	/	Yes	The unique preset name should use UTF-8 encoding for Chinese.

- **Description**

Audio Command	audio set param-preset {"preset_name":"<value>"}
Supported Models	AP08, CM50, CM20
Send Format	audio set param-preset {"preset_name":"<value>"} \r\n
Return Format	audio set param-preset {"preset_name":"<value>"} \r\n
Example	Send: audio set param-preset {"preset_name":"Custom 1"} \r\n Return: audio set param-preset {"preset_name":"Custom 1"}

27. Configure audio parameter presets

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
preset_name	string	/	Yes	The unique preset name should use UTF-8 encoding for Chinese.

- **Description**

Audio Command	audio set preset {"preset_name":"<value>"}
Supported Models	AP08, AVBridge, CM50, CM20
Send Format	audio set preset {"preset_name":" "} \r\n
Return Format	audio set preset {"preset_name":" "} \r\n

Example	Send: audio set preset {"preset_name":"Custom 1"} \r\n Return: audio set -preset {"preset_name":"Custom1"} \r\n

28. Set Gain (Deprecated, recommended to use 29)

- **Request Parameters**

Name	Typology	Required	Constraint	Description
» arr_gain_info	object	true	none	none
»» name	string	true	none	Channel Name Displayed on the Designer UI
»» value	integer	true	[-60, 20]	none

- **Description**

Audio Command	audio set gain {"arr_gain_info": [{"name":"<value>","value":<value>}]}
Supported Models	AP08, CM50, CM20
Send Format	audio set gain {"arr_gain_info": [{"name":"","value":}]} \r\n
Return Format	audio set gain {"arr_gain_info": [{"name":"","value":}]} \r\n
Example	Send: audio: set gain parameters. {"arr_gain_info": [{"name":"USB-Input-1","value":10}, {"name":"USB-Input-2","value":10}]} \r\n Return: audio set gain {"arr_gain_info": [{"name":"USB-Input-1","value":10}, {"name":"USB-Input-2","value":10}]} \r\n

29. Set channel gain

- **Request Parameters**

Name	Typology	Required	Constraint	Description
» arr_gain_info	object	true	none	none
»» name	string	true	none	Channel Name Displayed on the Designer UI
»» value	integer	true	[-60, 20]	none

- **Description**

Audio Command	audio set channel gain {"arr_gain_info": [{"name":"<value>","value":<value>}]}
Supported Models	AP08, AVBridge, CM50, CM20
Send Format	audio set channel gain {"arr_gain_info": [{"name":"","value":}]}

Return Format	audio set channel gain {"arr_gain_info": [{"name":"","value":}]}
Example	Send: audio configure channel gain {"arr_gain_info": [{"name":"USB-Input-1","value":10}, {"name":"USB-Input-2","value":10}]} \r\n Return: audio set channel gain {"arr_gain_info": [{"name":"USB-Input-1","value":10}, {"name":"USB-Input-2","value":10}]} \r\n

30. Get Current Gain (Deprecated, use 31 instead)

Request Parameters:

Name	Typology	Required	Constraint	Description
» arr_gain_info	object	true	none	none

Return Parameters

Name	Typology	Required	Constraint	Description
» arr_gain_info	object	true	none	none
»» name	string	true	none	Channel Name Displayed on the Designer UI
»» value	integer	true	[-60, 20]	none

- **Description**

Audio Command	audio get gain {"arr_gain_info": ["<value>"]}
Supported Models	AP08, CM50, CM20
Send Format	audio get gain {"arr_gain_info": [" <value>"]}
Return Format	audio get gain {"arr_gain_info": [{"name":" <value>","value":<value>}]}
Example	Send: audio get gain {"arr_gain_info": ["USB_IN-1","USB_IN-0"]} \r\n Return: audio get gain {"arr_gain_info": [{"name":"USB-Input-1","value":10}, {"name":"USB-Input-2","value":10}]} \r\n

31. Get current gain

Request Parameters:

Name	Typology	Required	Constraint	Description
arr_gain_info	string	true	none	none

Return Parameters

Name	Typology	Required	Constraint	Description

» arr_gain_info	object	true	none	none
»» name	string	true	none	Channel Name Displayed on the Designer UI
»» value	integer	true	[-60, 20]	none

- **Description**

Audio Command	audio get channel gain {"arr_gain_info": ["<value>"]}
Supported Models	AP08, AVBridge, CM50, CM20
Send Format	audio get channel gain {"arr_gain_info": [" <value>"]}
Return Format	audio get channel gain {"arr_gain_info": [{"name":" <value>","value":<value>}]}
Example	Send: Retrieve audio channel gain: {"arr_gain_info": ["USB_IN-1","USB_IN-0"]} \r\n Return: audio get channel gain {"arr_gain_info": [{"name":"USB-Input-1","value":10}, {"name":"USB-Input-2","value":10}]} \r\n Special Note: If no parameters are specified, the gain values for all channels will be returned.

32. Configure external audio input noise reduction

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
level	int	[0,1]	Yes	Noise cancellation level 1: Enabled 0: Disabled
sn	string	NULL	No	Device unique identifier

- **Description**

| Audio Command| xml
<value></value>
<value></value>
```audio external-input set noise-reduction {"level":" <value></value> ","sn":" <value></value> "}
|----------|----------
| Supported Models| Yealink RoomConnect
| Send Format| audio external-input set noise-reduction {"level":0,"sn":"506011D110000054"} \r\n
| Return Format| audio external-input set noise-reduction {"level":0,"sn":"506011D110000054"} \r\n
| Example| Send：
audio external input noise reduction setting {"level":0,"sn":"506011D110000054"} \r\n
Return:
audio external-input set noise-reduction {"level":0,"sn":"506011D110000054"} \r\n

**33. Set channel mute status**
```

- **Request Parameters**

Request Parameters:

Body:

Name	Typology	Required	Constraint	Description
» Mute channel	object	true	none	none
»» name	string	true	none	Channel Name Displayed on the Designer UI
»» status	string	true	[off, on]	none

- **Description**

Audio Command	audio set channel mute {"channel_mute": [{"name":"<value>","status":<value>}]}
Supported Models	AP08, AVBridge, CM50, CM20
Send Format	audio set channel mute {"channel": [{"name":" <value>","value":<value>}]}
Return Format	audio set channel mute {} \r\n
Example	Send: audio - set channel mute status {"channel_mute": [{"name":"Pickup-Area-1","status":"on"}, {"name":"Amplifier-Area-1","status":"off"}]} \r\n Return: audio set channel mute {"channel_mute": [{"name":"Pickup-Area-1","status":"on"}, {"name":"Amplifier-Area-1","status":"off"}]} \r\n

34. Get channel mute status

Request Parameters:

Body:

Name	Typology	Required	Constraint	Description
» Mute channel	object	true	none	none

Return Parameters

Body:

Name	Typology	Required	Constraint	Description
» Mute channel	object	true	none	none
»» name	string	true	none	Channel Name Displayed on the Designer UI
»» status	string	true	[off, on]	none

- **Description**

Audio Command	audio get channel mute {"channel_mute": ["<value>"]}

Supported Models	AP08, AVBridge, CM50, CM20
Send Format	audio get channel mute {"channel_mute": [" <value>"]} \r\n
Return Format	xml

<value></value>
```
```json
{"channel_mute": [{"name":" <value>","status":<value>}]} ```
```json
{"channel_mute": [{"name":" ","status":}]}
| Example| Send：
audio get channel mute {"channel_mute": ["Pickup-Area-1","Amplifier-Area-1"]} \r\n
Return:
audio get channel mute {"channel_mute": [{"name":"Pickup-Area-1","status":"on"}, {"name":"Amplifier-Area-1","status":"off"}]} \r\n
```

35. Get channel volume

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
arr_gain_info	string	-	Yes	Channel name array; in essence, gain is a physical quantity. This API requires specifying the channel name; if the channel name is not provided, it will return the gain value.

- **Return Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
arr_gain_info	arr_gain_info []	-	Yes	-

arr_gain_info:

Name	Typology	Range of Values	Parameter Description
name	string	-	Channel Name
value	int	[0-100]	Channel volume may have discrepancies. If you set the value to 50, you might retrieve 51.

- **Description**

Audio Command	audio get channel volume
Supported Models	AP08, AVBridge, CM50, CM20
Send Format	audio get channel volume {"arr_gain_info": ["Dante-Input-1","Dante-Input-2"]}\r\n

Return Format	audio get channel volume {"arr_gain_info": [{"name":"Dante-Input-1","value":51}, {"name":"Dante-Input-2","value":60}]}
	Send: Retrieve audio channel volume Return: Retrieve audio channel volume {"arr_gain_info": [{"name":"Dante-Input-1","value":51}, {"name":"Dante-Input-2","value":60}]}

36. Set channel volume

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
arr_gain_info	arr_gain_info []	-	Yes	-

arr_gain_info:

Name	Typology	Range of Values	Parameter Description
name	string	-	Channel Name
value	int	[0-100]	Channel volume may have discrepancies. If you set the value to 50, you might retrieve 51.

- **Description**

Audio Command	audio set channel volume
Supported Models	AP08, AVBridge, CM50, CM20
Send Format	audio set channel volume {"arr_gain_info": [{"name":"Dante-Input-1","value":50}, {"name":"Dante-Input-2","value":60}]}\r\n
Return Format	audio set channel volume {"arr_gain_info": [{"name":"Dante-Input-1","value":50}, {"name":"Dante-Input-2","value":60}]}\r\n
	Send: set channel volume for audio {"arr_gain_info": [{"name":"Dante-Input-1","value":50}, {"name":"Dante-Input-2","value":60}]}\r\n Return: set channel volume for audio {"arr_gain_info": [{"name":"Dante-Input-1","value":50}, {"name":"Dante-Input-2","value":60}]}\r\n

meeting-platform Syntax

1. Access to the List of Meeting Platforms

- **Return Parameters**

Name	Typology	Range of Values	Parameter Description

mode-list	string	Reference range: [ume,yms,zoom,general ,tencent,feishu,byod]	List of meeting platforms for devices:

ume: UME meeting

yms: YMS meeting

zoom: Zoom meeting

general: Yealink meeting

tencent: Tencent meeting

feishu: Feishu meeting

byod: BYOD mode

- **Description**

meeting-platform command	meeting-platform get mode-list
Supported Models	Meetingboard 65/86, Meetingboard 65/75/86 Pro, MeetingEye 500, MeetingBar A10/A40/A50 **Note:** 1. The meeting platforms listed are based on those installed or available on the device.
Send Format	meeting-platform get mode-list\r\n
Return Format	meeting-platform get mode-list {"mode-list":["ume","yms","zoom"]} \r\n
Example	Send: meeting-platform get mode-list \r\n Return: meeting-platform get mode-list {"mode-list":["ume","yms","zoom"]}\r\n

2. Setting the Current Conference Platform

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
type	string	Reference range: [ume,yms,zoom, general,tencent, feishu,byod]	Yes	Device installed conference platform

ume: ume meeting

yms: yms meeting

zoom: zoom meeting

general: Yealink meeting

tencent: Tencent meeting

feishu: Feishu meeting

byod: BYOD mode

- **Description**

meeting-platform command	**meeting-platform set {"type":"<value>"}**
Supported Models	Meetingboard 65/86, Meetingboard 65/75/86 Pro, MeetingEye 500, MeetingBar A10/A40/A50 **Note:** 1. The available meeting platforms should be determined based on those currently present, as retrieved via the get API.
Send Format	meeting-platform set {"type":"ume"}\r\n
Return Format	meeting-platform set {"type":"ume"}\r\n
Example	Send: meeting-platform set {"type":"ume"} \r\n Return: meeting-platform set {"type":"ume"} \r\n

3. Get current meeting platform

- **Return Parameters**

Name	**Typology**	**Range of Values**	**Parameter Description**
name	string	NULL	Current platform name

- **Description**

meeting-platform command	**meeting-platform get current-mode**
Supported Models	Meetingboard 65/86, Meetingboard 65/75/86 Pro, MeetingEye 500, MeetingBar A10/A40/A50
Send Format	meeting-platform get current-mode
Return Format	meeting-platform get current-mode {"name":"ume"}

Example	Send: meeting-platform get current-mode \r\n Return: meeting-platform get current-mode {"name":"ume"} \r\n

Bluetooth Syntax

1. Configure Bluetooth Switch

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
status	string	[on,off]	Yes	Bluetooth status off: off on: on

- **Description**

bluetooth command	bluetooth set {"status":"<value>"}
Supported Models	Meetingboard 65/86, Meetingboard 65/75/86 Pro, MeetingEye 500, MeetingBar A10/A40/A50, Yealink RoomConnect
Send Format	bluetooth set {"status":"on"}\r\n
Return Format	bluetooth set {"status":"on"}\r\n
Example	Send: bluetooth set {"status":"on"} \r\n Return: bluetooth set {"status":"on"} \r\n

Network Syntax

1. Access to network information

- **Return Parameters**

Name	Typology	Range of Values	Note
network-list	network_info []	NULL	network information list

network_info

Name	Typology	Range of Values	Note
type	int	[0,1]	Acquire network method 0: Dynamic acquisition 1: Static setting

port-type	int	[0,1,2]	Network Port Type 0: LAN Port 1 1: LAN Port 2 2: Wireless Port 3: AP Port
mode	int	[0,1,2]	IP type 0: IPv4 1: IPv6 2: IPv4 and IPv6
ip	string	NULL	IP address
mask	string	NULL	subnet mask
gateway	string	NULL	Gateway
primary-dns	string	NULL	Preferred DNS server name
second-dns	string	NULL	Secondary DNS server name.

- **Description**

network command	**network get info**
Supported Models	UVC40, MeetingBoard 65/86, MeetingEye 500, MeetingBar A10/A40, Yealink RoomConnect **Note:

1. UVC40 only provides wireless LAN information
2. MeetingBoard 65/86 supports both Wired LAN 1 and wireless LAN
3. SmartVision 40 only supports a wired LAN connection**
   | Send Format| network get info\r\n
   | Return Format| network get info {"network-list":[{"type":0,"port-type":2,"mode":0,"ip":"","mask":"","gateway":"","primary-dns":"","second-dns":""},{"type":0,"port-type":0,"mode":0,"ip":"10.50.67.59","mask":"255.255.255.0","gateway":"10.50.67.254","primary-dns":"10.100.1.10","second-dns":"192.168.1.22"}]}\r\n
   | Example| Send：
   network get info \r\n
   Return:
   network get info {"network-list":[{"type":0,"port-type":2,"mode":0,"ip":"","mask":"","gateway":"","primary-dns":"","second-dns":""},{"type":0,"port-type":0,"mode":0,"ip":"10.50.67.59","mask":"255.255.255.0","gateway":"10.50.67.254","primary-dns":"10.100.1.10","second-dns":"192.168.1.22"}]} \r\n

camera-layout syntax

1. Setting the camera layout state

- **Request Parameters**

Name	**Typology**	**Range of Values**	**Compulsory**	**Parameter Description**

status	string	[on,off]	Yes	Camera layout status on: enabled off: disabled

- **Description**

camera-layout command	camera-layout set switch {"status":"<value>"}
Supported Models	Yealink RoomConnect
Send Format	camera-layout set switch {"status":"on"}\r\n
Return Format	camera-layout set switch {"status":"on"}\r\n
Example	Send: camera-layout set switch {"status":"on"} \r\n Return: camera-layout set switch {"status":"on"} \r\n

2. Setting the camera layout type

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
type	string	[fullscreen, div2$_{div9,1x1}$1x8,1xN,divide]	Yes	Layout type fullscreen: single full-screen layout div2~div9: two-split~nine-split layout divide: equal split layout 1xN：1+N layout 1x1~1x8: 1+1~1+8 layout
focus-camera	string	NULL	No	The unique identifier of the camera in the layout for the large image display.

- **Description**

camera-layout command	camera-layout set type {"type":"<value>",["focus-camera":"<value>"]}

Supported Models	Yealink RoomConnect，MeetingEye 500 Note： 1. The type parameter for the MeetingEye 500 can have values of fullscreen, divide, and 1xN. 2. Yealink RoomConnect supports the type parameter with values of fullscreen, div2 to div9, 1x1 to 1x8 When setting the fullscreen mode, you need to use the Set Camera Layout/Camera Position feature to specify the camera with position 0 to apply the fullscreen mode. 3.Only Yealink RoomConnect includes the small-screen-position, small-screen-size, main-screen-type, and second-screen-type parameters, and these four parameters only apply to the Picture-in-Picture layout. 4.Only Yealink RoomConnect supports the sn parameter. 5.Only the MeetingEye 500 model supports the focus-camera parameter.
Send Format	**Yealink RoomConnect：** camera-layout set type {"type":"fullscreen","sn":"8703018090000132", "pip-param": { "small-screen-position":"top-left","small-screen-size":"one-fourth","main-screen-type":"panorama","second-screen-type":"auto-frame"}}\r\n **M500:** camera-layout set type {"type":"fullscreen","focus-camera": "8703018090000132"}\r\n
Return Format	**Yealink RoomConnect：** camera-layout set type {"type":"fullscreen","sn":"8703018090000132", "pip-param": { "small-screen-position":"top-left","small-screen-size":"one-fourth","main-screen-type":"panorama","second-screen-type":"auto-frame"}}\r\n **M500:** camera-layout set type {"type":"fullscreen","focus-camera": "8703018090000132"}\r\n
Example	**Yealink RoomConnect：** Send： camera-layout set type {"type":"fullscreen","sn":"8703018090000132", "pip-param": { "small-screen-position":"top-left","small-screen-size":"one-fourth","main-screen-type":"panorama","second-screen-type":"auto-frame"}}\r\n Return: camera-layout set type {"type":"fullscreen","sn":"8703018090000132", "pip-param": { "small-screen-position":"top-left","small-screen-size":"one-fourth","main-screen-type":"panorama","second-screen-type":"auto-frame"}}\r\n **M500:** Send： camera-layout set type {"type":"fullscreen","focus-camera": "8703018090000132"} \r\n Return: camera-layout set type {"type":"fullscreen","focus-camera": "8703018090000132"} \r\n

3) Set up camera layout and camera position.(MVC)

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description

position-list	object	NULL	Yes（AVhub）	Getting a list of cameras
>>	sn		Yes（AVhub）	Camera unique identification
>>	left	[0~1920]	Yes（AVhub）	Left coordinate of the camera layout screen position.
>>	top	[0~1080]	Yes（AVhub）	Camera layout screen display position left coordinate
>>	width	[0~1920]	Yes（AVhub）	Width of the camera layout screen display.
>>	height	[0~1080]	Yes（AVhub）	Height of the camera layout screen display.

- **Description**

camera-layout command	**camera-layout set position {"position-list":[{"sn":"<value>","left":<value>,"top":<value>,"width":<value>,"height":<value>},{"sn":"<value>","left":<value>,"top":<value>,"width":<value>,"height":<value>}]}**
Supported Models	AVHub **Note:** The number of camera position parameters in the position-list can be optionally set based on the actual number of cameras.
Send Format	camera-layout set position {"position-list": [{"sn":"806009E070000512","left":0,"top":134,"width":1440,"height":810}, {"sn":"806007D120000442","left":1440,"top":134,"width":480,"height":270}]}\r\n
Return Format	camera-layout set position {"position-list": [{"sn":"806009E070000512","left":0,"top":134,"width":1440,"height":810}, {"sn":"806007D120000442","left":1440,"top":134,"width":480,"height":270}]}\r\n
Example	Send: camera-layout set position {"position-list": [{"sn":"806009E070000512","left":0,"top":134,"width":1440,"height":810}, {"sn":"806007D120000442","left":1440,"top":134,"width":480,"height":270}]}\r\n Return: camera-layout set position {"position-list": [{"sn":"806009E070000512","left":0,"top":134,"width":1440,"height":810}, {"sn":"806007D120000442","left":1440,"top":134,"width":480,"height":270}]}\r\n

4.Setting up camera layout and camera position（YRC）

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
sn	string	NULL	Yes	Camera unique identification

position	int	[0~8]	Yes	Location identifier

- **Description**

camera-layout command	camera-layout set position {"sn":"<value>","position":<value>}
Supported Models	Yealink RoomConnect

Note:

When configuring fullscreen mode, you need to set the camera layout and specify camera position 0 in order to enable fullscreen mode
| Send Format| camera-layout set position {"sn":"8703018090000132","position":2}\r\n
| Return Format| camera-layout set position {"sn":"8703018090000132","position":2}\r\n
| Example| Send：
camera-layout set position {"sn":"8703018090000132","position":2} \r\n
Return:
camera-layout set position {"sn":"8703018090000132","position":2} \r\n
Layout positions follow left-to-right order, with equal divisions based on the maximum quantity of 1+N as a reference:
Equal parts:
1+N：

5) Obtain camera layout type

- **Return Parameters**

Name	Typology	Range of Values	Parameter Description
type	string	[fullscreen,1xN,divide]	Layout type fullscreen: single full-screen layout divide: equal split layout 1xN: 1+ layout
focus-camera	string	NULL	The unique identifier of the camera in the layout for the large image display.

- **Description**

camera-layout command	camera-layout get type
Supported Models	MeetingEye 500 Note: 1.The value range for the **type parameter of the AVHub and MeetingEye 500 model is fullscreen, divide, 1xN.**
Send Format	camera-layout get type \r\n
Return Format	camera-layout get type {"type":"fullscreen","focus-camera":"8703018090000132"} \r\n

Example	Send: camera-layout get type \r\n Return: camera-layout get type {"type":"fullscreen","focus-camera":"8703018090000132"} \r\n

Upgrade Syntax

1. Device upgrade

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
sn	string	NULL	No	Device serial number for upgrade.
device	string	NULL	No	Upgraded device type
url	string	NULL	Yes	Firmware upgrade address (URL address or local path)
time	int	-1~23	No	Upgrade time -1：Upgrade immediately 0~23: Upgrade time

- **Description**

| upgrade Command | upgrade firmware start {["sn":" ",], ["device":" "],path:" ",time:} | | ---------- | ---------- | | Supported Models | MeetingBoard 65/86/65 Pro/75 Pro/86 Pro, MeetingDisplay, MeetingEye 500, MeetingBar A10/A40/A50, Yealink RoomConnect

注意：

1、仅Yealink RoomConnect支持传sn和device参数， 至少填写sn和device其中一个参数

2、sn与device理论上只填一种，同时填写，只生效sn参数 | | Command Format | upgrade firmware start {"sn":"801203F04C600791","url":"https://packet-nexus.yealink.com/repository/repo-packet-develop/AllRom/MeetingBarA40-rom/289.320.254.315/MeetingBarA40-289.320.254.315-mix.rom","time":-1} \r\n ||

返回格式 | upgrade firmware start {"sn":"801203F04C600791","url":"https://packet-nexus.yealink.com/repository/repo-packet-develop/AllRom/MeetingBarA40-rom/289.320.254.315/MeetingBarA40-289.320.254.315-mix.rom","time":-1} \r\n ||

Example | **携带sn参数情况**：

Send：upgrade firmware start {"sn":"803032E070000031","url":"https://packet-nexus.yealink.com/service/rest/repository/browse/repo-packet-release/AllRom/UVC86-rom/151.0.249.13/UVC86-151.0.249.13.rom","time":-1} \r\n

Return: upgrade firmware start {"sn":"803032E070000031","url":"https://packet-nexus.yealink.com/service/rest/repository/browse/repo-packet-release/AllRom/UVC86-rom/151.0.249.13/UVC86-151.0.249.13.rom","time":-1} \r\n

With device parameter：

Send： upgrade firmware start {"device":"uvc86","url":"https://packet-nexus.yealink.com/service/rest/repository/browse/repo-packet-release/AllRom/UVC86-rom/151.0.249.13/UVC86-151.0.249.13.rom","time":-1} \r\n

Return: upgrade firmware start {"device":"uvc86","url":"https://packet-nexus.yealink.com/service/rest/repository/browse/repo-packet-release/AllRom/UVC86-rom/151.0.249.13/UVC86-151.0.249.13.rom","time":-1} \r\n

2. Device upgrade cancellation

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
sn	string	NULL	No	Device serial number that stopped upgrading
device	string	NULL	No	Upgraded device type

- **Description**

Upgrade Commands	upgrade firmware cancel {["sn":"<value>"],["device":"<value>"]}
Supported Models	Yealink RoomConnect **Note:** **1. Fill in either SN or device theoretically. If both are filled in at the same time, only the SN parameter will take effect.** **2. If neither SN nor device is filled in, all upgrades will be canceled.**
Send Format	upgrade firmware cancel {"sn":"803032E070000031","device":"uvc86"} \r\n
Return Format	upgrade firmware cancel {"sn":"803032E070000031","device":"uvc86"} \r\n
Example	**With sn:** Send: upgrade firmware cancel {"sn":"803032E070000031"} \r\n Return: upgrade firmware cancel{"sn":"803032E070000031"}\r\n **With device:** Send: upgrade firmware cancel {"device":"uvc86"} \r\n Return: upgrade firmware cancel{"device":"uvc86"} \r\n

3. Obtain upgrade status

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
sn	string	NULL	Yes	Get the serial number of the device in upgrade status.

- **Return Parameters**

Name	Typology	Range of Values	Parameter Description
status	string	[downloading,upgrading,success,fail]	Download status: downloading: downloading upgrading: upgrading success: success fail: failed
progress	int	0~100	Upgrade progress, return a normal value if upgrading, otherwise return -1.

- **Description**

Upgrade Commands	upgrade firmware get status {"sn":"<value>"}
Supported Models	Yealink RoomConnect
Send Format	upgrade firmware get status {"sn":"803032E070000031"}\r\n
Return Format	upgrade firmware get status {"status":"upgrading", "progress": 48} \r\n
Example	Send: upgrade firmware get status {"sn":"803032E070000031"}\r\n Return: upgrade firmware get status {"status":"upgrading", "progress": 48} \r\n

Diagnostics Syntax

1. Access system logs

- **Return Parameters**

Name	Typology	Range of Values	Parameter Description
File path	string	NULL	File path (log file can only be obtained by manually downloading it)

- **Description**

Diagnostics Commands	diagnostics get log
Supported Models	Yealink RoomConnect, MeetingDisplay, MeetingBoard 65/86/65 Pro/75 Pro/86 Pro, MeetingEye 500, MeetingBar A10/A40/A50
Send Format	diagnostics get log \r\n
Return Format	diagnostics get log {file} \r\n
Example	Send: diagnostics get log \r\n Return: diagnostics get log {file}\r\n

2. Network diagnosis

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
action	string	ping traceroute	Yes	ping command: ping -c num ip traceroute command: traceroute -m num -I ip
num	int	1~30	Yes	Number of ping attempts; or maximum number of hops for traceroute tracking
ip	string	NULL	Yes	Target IP/domain

- **Return Parameters**

Name	Typology	Range of Values	Parameter Description

result	string	NULL	Executing statement returns content

- **Description**

Diagnostics Commands	**diagnostics network {"action":"<value>","num":"<value>","ip":"<value>"}**
Supported Models	Meetingboard 65/86, Meetingboard 65/75/86 Pro, MeetingDisplay, MeetingEye 500, MeetingBar A10/A40/A50
Send Format	diagnostics network {"action": "ping","num": 5,"ip": "10.50.150.1"} \r\n
Return Format	diagnostics network "PING 10.56.23.9 (10.56.23.9) 56 (84) bytes of data. 64 bytes from 10.56.23.9: icmp_seq=1 ttl=64 time=0.111 ms 64 bytes from 10.56.23.9: icmp_seq=2 ttl=64 time=0.049 ms 64 bytes from 10.56.23.9: icmp_seq=3 ttl=64 time=0.050 ms 64 bytes from 10.56.23.9: icmp_seq=4 ttl=64 time=0.047 ms 64 bytes from 10.56.23.9: icmp_seq=5 ttl=64 time=0.072 ms – 10.56.23.9 ping statistics —5 packets transmitted, 5 received, 0% packet loss, time 4099ms rtt min/avg/max/mdev = 0.047/0.065/0.111/0.026 ms "\r\n
Example	Send: diagnostics network {"action": "ping","num": 5,"ip": "10.50.150.1"} \r\n Return: diagnostics network "PING 10.56.23.9 (10.56.23.9) 56 (84) bytes of data. 64 bytes from 10.56.23.9: icmp_seq=1 ttl=64 time=0.111 ms 64 bytes from 10.56.23.9: icmp_seq=1 ttl=64 time=0.049 ms 64 bytes received from 10.56.23.9: icmp_seq=1 ttl=64 time=0.050 ms 64 bytes from 10.56.23.9: icmp_seq=1 ttl=64 time=0.047 ms 64 bytes from 10.56.23.9: icmp_seq=1 ttl=64 time=0.072 ms – 10.56.23.9 ping statistics —5 packets transmitted, 5 received, 0% packet loss, time 4099ms rtt min/avg/max/mdev = 0.047/0.065/0.111/0.026 ms "\r\n

3. Start or stop packet capture files

- **Request Parameters**

Name	**Typology**	**Range of Values**	**Compulsory**	**Parameter Description**
filter	string	Filtering by IP, port, protocol, etc. is possible.	No	Filter condition
interface	string	wan: currently active network interface	Yes (start capturing packets)	Interface name
filter-type	int	0: Custom		

1: SIP or H245 or H225

2: RTP

3: Not RTP| Yes (start capturing packets)| Filter type; the filter takes effect if the value is 0.
| operation| string| start: start capturing packets

stop: stop capturing packets

get: retrieve captured packet files| Yes| Packet capturing action; when stopping or getting, there is no need to carry the above three parameters.
Note:
1. It is recommended to send the get command every 5 minutes after starting the packet capturing to obtain the packet capture file. The packet capture files need to be concatenated by yourself (you may get empty files because the device side has not yet generated the file at this time); if more than 10 minutes have passed without getting, it is considered to abandon the most recent packet capture file and stop capturing.
2. When stopping, the device side responds that the most recent file has not been uploaded.

- **Return Parameters**

Name	Typology	Range of Values	Parameter Description
Packet capture filename	string	NULL	Respond to get/stop operations; upload the packet capture file itself without a file type extension.

Name	Typology	Range of Values	Parameter Description
File stream	file	NULL	File stream data (requires initiating a download to obtain the file).

- **Description**

Diagnostics Commands	diagnostics packetcapture {"operation":"<value>", "interface":"<value>","filter":"<value>","filter-type": "<value>"}
Supported Models	MeetingBoard 65/86/65 Pro/75 Pro/86 Pro, MeetingDisplay, MeetingEye 500, MeetingBar A10/A40/A50 **Note:** 1. Packet capture data can only be retrieved using the get API once the cache reaches 5MB. After retrieval, the data will be cleared; you must wait until the cache hits 5MB again before more data becomes available. 2. If real-time data retrieval is not required, it is recommended to stop capturing the current collected packet data directly. 3. Capturing 5MB of packets is necessary because if the cache is too small, asynchronous packet capturing may result in incomplete packet data.

Send Format	Packet capture started: diagnostics packetcapture {"operation":"start","interface":"wan","filter":"tcp","filter-type":"0"} \r\n Retrieve packet capture: diagnostics packetcapture {"operation":"get"} stop packet capture: diagnostics packetcapture {"operation":"stop"}
Return Format	Packet capture initiated: diagnostics packetcapture {"operation":"start","interface":"wan","filter":"tcp","filter-type":"0"} \r\n Retrieve packet capture: diagnostics packetcapture/tmp/cc_pcap/cc_merge.pcap Stop packet capture: diagnostics packetcapture/tmp/cc_pcap/cc_merge.pcap
Example	Packet capture started: Send: Starting diagnostics packet capture on the wan interface with tcp filtering (filter type: 0).

Return: diagnostics - Packet Capture {"operation":"start","interface":"wan","filter":"tcp","filter-type":"0"} \r\n
start packet capture:
Send: diagnostics packetcapture {"operation":"get"} \r\n
Return: diagnostics packetcapture/tmp/cc_pcap/cc_merge.pcap
Stop packet capture:
Send: diagnostics packetcapture {"operation":"stop"}
Return: diagnostics packetcapture/tmp/cc_pcap/cc_merge.pcap

4. Alarm

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
severity	string	critical: severe		

major: primary

minor: minor

all: all| No| If severity is not specified, it defaults to critical.
| from-time| long| Unit: seconds;
the earliest time point is 7 days ago.| No| 1. Respond with a list of alarm events between the from-time and the time the command is sent.
2. If from-time is not specified, default to responding with a list of alarm events from the last ten minutes.
3. The timestamp refers to the number of seconds elapsed between a certain moment and "00:00:00" UTC on January 1, 1970.

- **Return Parameters**

Name	Typology	Range of Values	Note
alert-list	alert_list []	NULL	List of device information

alert_list

Name	Typology	Range of Values	Parameter Description
name	string	Disk slave disconnect: Accessory disconnected	

Update Configuration failure: Configuration file update failed

Update Firmware failure: Firmware upgrade failed

Wireless microphone low battery: Low battery for wireless microphone| Event Name
| severity| string| critical: severe

major: primary

minor: minor| NULL
| action-time| long| Unit: seconds| Alarm timestamp
| mac| string| NULL| device mac address
| ip| string| NULL| Device IP address

- **Description**

Diagnostics Commands	diagnostics get alert {["severity":"<value>"], ["from-time":"<value>"]}
Supported Models	Meetingboard 65/86, Meetingboard 65/75/86 Pro, MeetingDisplay, MeetingEye 500, MeetingBar A10/A40/A50
Send Format	diagnostics get alert {"severity": "all", "from-time": 1720430000} \r\n
Return Format	diagnostics get alert {"alert-list": [{"action-time":1720431643,"ip":"10.56.23.9","mac":"24:9A:D8:5A:4C:95","name":"Dsk slave disconnect","severity":"Minor"}, {"action-time":1720431749,"ip":"10.56.23.9","mac":"24:9A:D8:5A:4C:95","name":"Dsk slave disconnect","severity":"Minor"}, {"action-time":1720508139,"ip":"","mac":"24:9A:D8:5A:4C:95","name":"Dsk slave disconnect","severity":"Minor"}]} \r\n

Example	Send: diagnostics get alert {"severity": "all", "from-time": 1720430000} \r\n Return: diagnostics get alert {"alert-list": [{"action-time":1720431643,"ip":"10.56.23.9","mac":"24:9A:D8:5A:4C:95","name":"Dsk slave disconnect","severity":"Minor"}, {"action-time":1720431749,"ip":"10.56.23.9","mac":"24:9A:D8:5A:4C:95","name":"Dsk slave disconnect","severity":"Minor"}, {"action-time":1720508139,"ip":"","mac":"24:9A:D8:5A:4C:95","name":"Dsk slave disconnect","severity":"Minor"}]} \r\n

call syntax

1. Get call status

- **Return Parameters**

Name	Typology	Range of Values	Parameter Description
call-state	string	[incoming, incall, idle]	incoming: incoming call incall: In call idle: idle Both "incoming" and "incall" states are considered "In a Call"
app-info	app_info	NULL	If an app is currently on a call, its call info will be displayed. Otherwise, all related info will show as ""

app_info

Name	Typology	Note
id	string	App package name
name	string	App name
version	string	App version number

- **Description**

Diagnostics Commands	call get state
Supported Models	Yealink RoomConnect, MeetingBoard 65/86/65 Pro/75 Pro/86 Pro, MeetingDisplay, MeetingEye 500, MeetingBar A10/A40/A50, SmartVision 40 Note: SmartVision 40 supports both "incall" and "idle" states
Send Format	call get state \r\n
Return Format	call get state {"call-status":"idle","app-info": {"id":"","name":"","version":""}}\r\n
Example	Send: call get state\r\n Return: call get state {"call-status":"idle","app-info": {"id":"","name":"","version":""}}\r\n

splitroom syntax

1. Setting dividable meeting room status

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
sn	string	NULL	Yes	Requestor identification
value	int	[0~3]	Yes	Dividable Meeting Room Status 0: Disable Dividable Meeting Room function 1: Standalone Mode under Dividable Meeting Room 2: Merged Room under Dividable Meeting Room 3: Master Room under Dividable Meeting Room

- **Description**

Upgrade Commands	splitroom set status {"sn":" <value>","value":" <value>"}
Supported Models	Yealink RoomConnect
Send Format	splitroom set status {"sn":"506011D110000054","value":"1"} \r\n
Return Format	splitroom set status {"sn":"506011D110000054","value":"1"} \r\n
Example	Send: splitroom set status {"sn":"506011D110000054","value":"1"} \r\n Return: splitroom set status {"sn":"506011D110000054","value":"1"} \r\n

division syntax

1. Set the current partition status of the device (deprecated, use 2 instead)

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
value	int	[0, 1]	Yes	0: Merged state 1: Split state

- **Description**

division instruction	division set status {"value":}
Supported Models	AP08
Send Format	division set status {"value":1} \r\n
Return Format	division set status {"value":1} \r\n

Example	Send: set division status {"value":1} \r\n Return: division set status {"value":1} \r\n

2. Set the current partition status of the device

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
room_info	room_info []	-	Yes	Room Information

room_info:

Name	Typology	Range of Values	Compulsory	Parameter Description
room_id	int []	[1,3]	Yes	The ID of the smallest partitioned room contained in each room after segmentation

- **Description**

division-room command	division-room set config
Supported Models	AP08
Send Format	division set config {"room_info": [{"room_id": [1,2]}, {"room_id": [3]}]} \r\n
Return Format	division set config {"room_info": [{"room_id": [1,2]}, {"room_id": [3]}]} \r\n
Example	Send: division set config {"room_info": [{"room_id": [1,2]}, {"room_id": [3]}]} \r\n Return: division set config {"room_info": [{"room_id": [1,2]}, {"room_id": [3]}]} \r\n

3. Retrieve current device split status (deprecated, use method 4 instead)

- **Return Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
value	int	[0, 1]	Yes	0: Merged state 1: Split state

- **Description**

division instruction	division get status {"value":<value>}
Supported Models	AP08
Send Format	division get status {} \r\n
Return Format	division get status {"value":1} \r\n
Example	Send: Query the current status of the division {} \r\n Return: division get status {"value":1} \r\n

4. Retrieve current device partition status

- **Return Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
room_info	room_info []	-	Yes	Room Information

room_info:

Name	Typology	Range of Values	Compulsory	Parameter Description
room_id	int []	[1,3]	Yes	The ID of the smallest partitioned room contained in each room after segmentation

- **Description**

division-room	division-room get config
Supported Models	AP08
Send Format	division-room get config\r\n
Return Format	division-room get config {"room_info": [{"room_id": [1]}, {"room_id": [2]}, {"room_id": [3]}]} \r\n
Example	Send: division-room get configuration\r\n Response: division-room get configuration {"room_info": [{"room_id": [1]}, {"room_id": [2]}, {"room_id": [3]}]} \r\n

physical-interface syntax

1. Retrieve physical interface list

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
type	string	NULL	Yes	Interface Type ALL: ALL Types AUDIO: AUDIO Interface VIDEO: VIDEO Adapter Interface OTHERS: Other Interfaces

- **Description**

division instruction	phiysical-interface get list {"type":<value>}
Supported Models	Meetingboard 65/86, Meetingboard 65/75/86 Pro, MeetingDisplay, MeetingEye 500, MeetingBar A10/A40/A50

Send Format	phiysical-interface get list {"type":"ALL"} \r\n
Return Format	phiysical-interface get list {"interface-list": [{"id":0,"name":"HDMI_IN","type":"VIDEO","connected":false,"status":"on"}, {"id":1,"name":"HDMI_IN","type":"VIDEO","connected":false,"status":"on"}]} \r\n
Example	Send: get the list of physical interfaces {"type":"ALL"} \r\n Return: phiysical-interface get list {"interface-list": [{"id":0,"name":"HDMI_IN","type":"VIDEO","connected":false,"status":"on"}, {"id":1,"name":"HDMI_IN","type":"VIDEO","connected":false,"status":"on"}]} \r\n

2. Configure Physical Interface Switch

- **Request Parameters**

Name	Typology	Range of Values	Compulsory	Parameter Description
name	string	NULL	Yes	Interface name
id	int	NULL	Yes	Actually obtain interface ID
status	string	[on,off]	Yes	Interface status on: enabled off: disabled

- **Description**

division instruction	phiysical-interface set enable {"name":<value>, "id":<value>, "status":<value>}
Supported Models	Meetingboard 65/86, Meetingboard 65/75/86 Pro, MeetingDisplay, MeetingEye 500, MeetingBar A10/A40/A50
Send Format	physical-interface set enable {"name": "HDMI_IN", "id":2, "status": "off"}
Return Format	physical-interface set enable {"name": "HDMI_IN", "id":2, "status": "off"}
Example	Send: physical-interface set enable {"name": "HDMI_IN", "id":2, "status": "off"} Return: physical-interface set enable {"name": "HDMI_IN", "id":2, "status": "off"}