



# INOGENI TOGGLE ROOMS User Guide

User guide

Version 1.1

March 25, 2024

## VERSION HISTORY

Version	Date	Description
0.1	January 17, 2024	Preliminary user guide for device launch.
0.2	January 24, 2024	- Added new options to set built-in EDIDs - Updated RESTAPI and serial commands for EDID and EDIDUSR.
0.3	March 15, 2024	- Updated the connectivity diagram
1.0	March 20, 2024	- Updated serial and REST APIs. - Updated certification page.
1.1	March 25, 2024	- Adding precisions to priority functions.

## CONTENTS

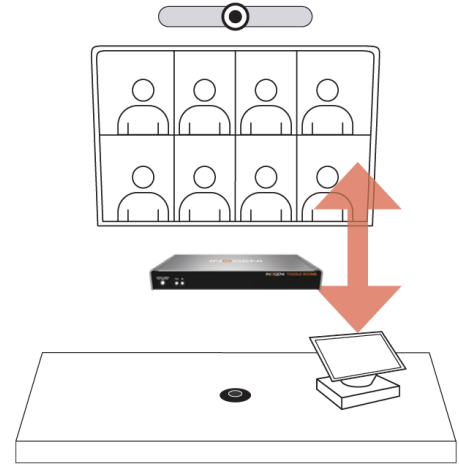
Version history .....	1
Typical applications.....	2
Room PC mode with BYOD content sharing .....	2
BYOM mode.....	2
Block diagram .....	3
Connectivity diagram .....	4
Device interfaces .....	5
LEDs behavior .....	6
Operating modes .....	6
Automatic.....	6
Manual.....	6
Manual with fallback .....	6
Specifications .....	7
Serial communication protocol.....	9
REST API.....	15
Telnet .....	23
INOGENI Maestro application.....	24
Mechanical specification .....	24
DIP switches .....	27
Troubleshooting section.....	27
Support.....	28
Certifications .....	28

## TYPICAL APPLICATIONS

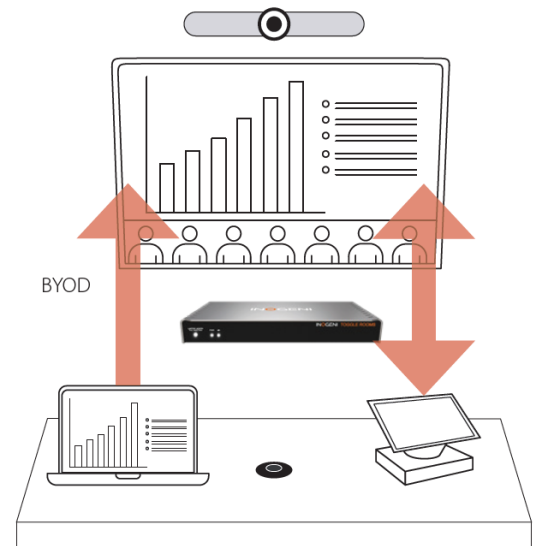
Here is a typical connection diagram used for the TOGGLE ROOMS device in a videoconferencing setup.

### ROOM PC MODE WITH BYOD CONTENT SHARING

In this mode, only the Room PC USB and HDMI connections are routed to the main USB and HDMI peripherals.

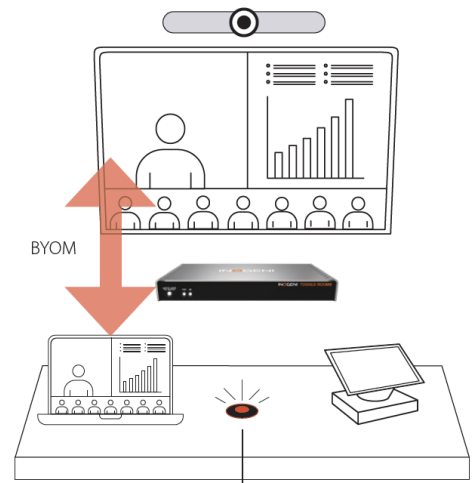


The Room PC is the system that is currently selected to the main USB and HDMI peripherals. However, if the user would like to send HDMI content from the laptop's USB-C or HDMI connection to the Room PC, it is possible to do so with the HDMI SHARE output connection.



### BYOM MODE

In this mode, the laptop is the system that is currently selected to the main USB and HDMI peripherals. The Room PC is completely disconnected from the setup.



# BLOCK DIAGRAM

Here is a simple block diagram to better understand the usage of the product.

## TOGGLE ROOMS BYOD/BYOM switcher

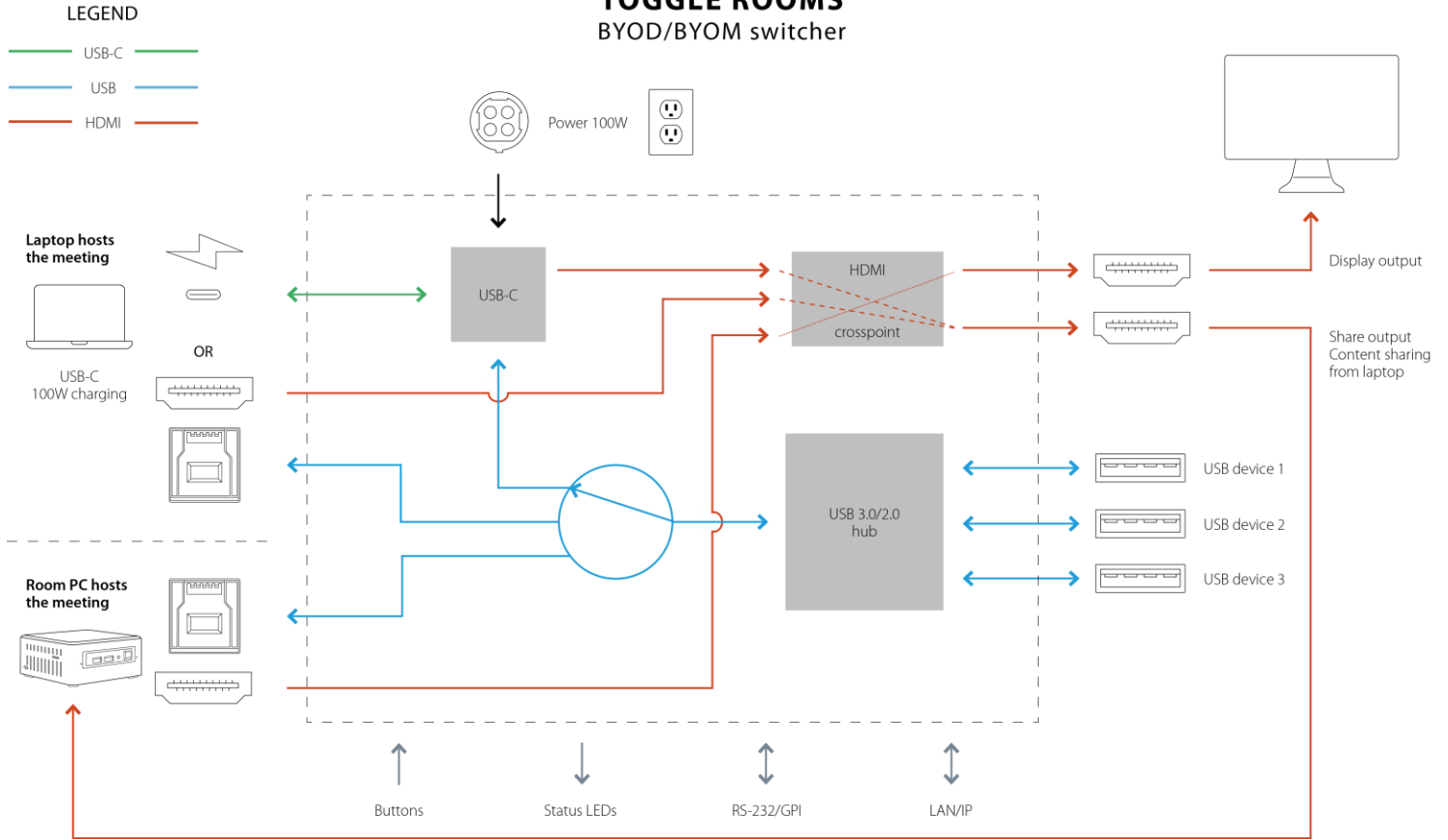
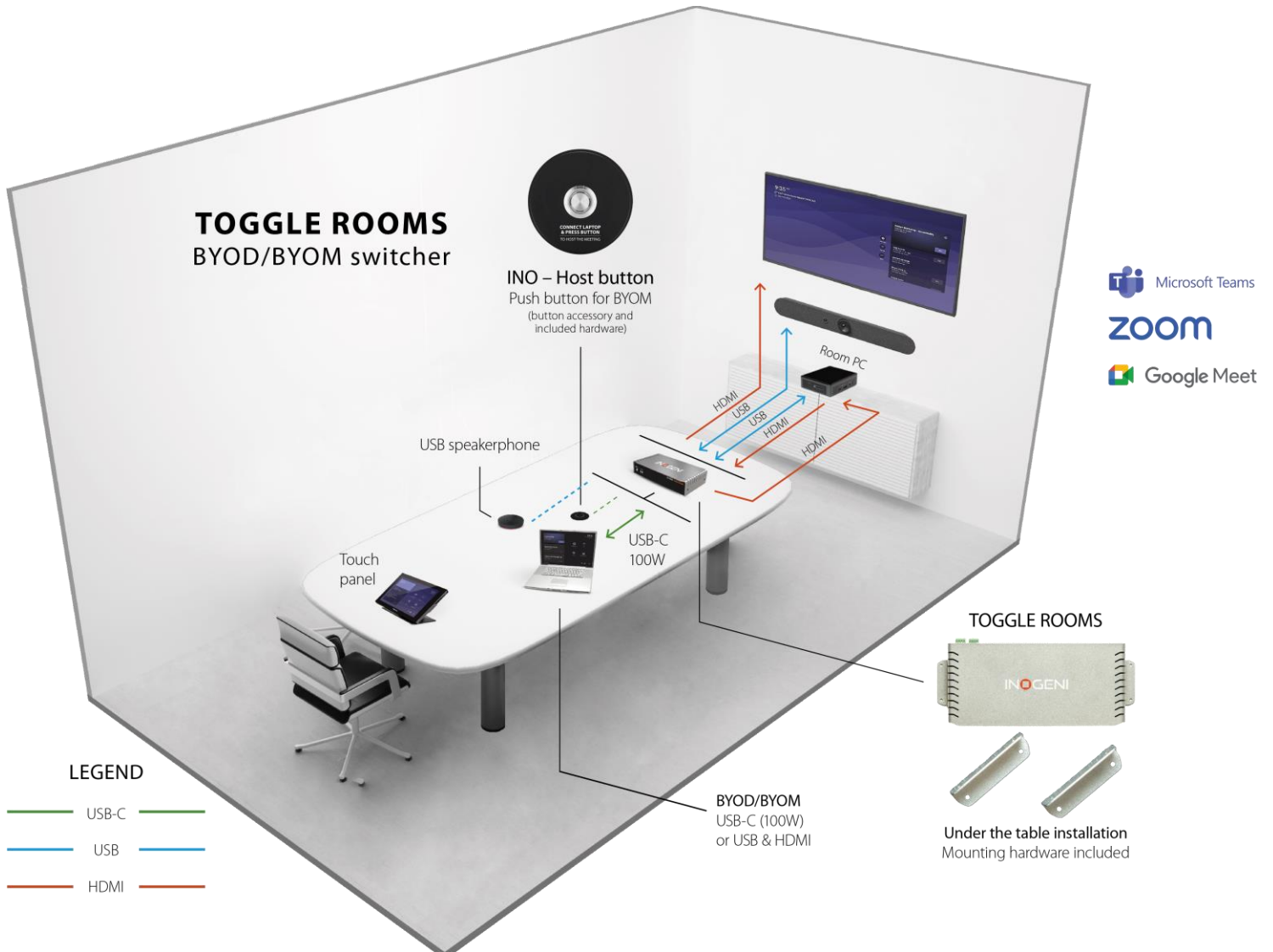


Figure 1: Basic block diagram when RoomPC and content sharing mode is activated.

# CONNECTIVITY DIAGRAM

Here is a simple connectivity diagram showing



## DEVICE INTERFACES

Here are the devices interfaces.



Figure 2: Front side connections




Figure 3: Back side connections

Items	
1	LAPTOP HOST THE MEETING button. This button will connect HDMI and USB peripherals to the laptop connection for BYOM.
2	PWR and charging status leds.
3	24VDC power input.
4	USB-C laptop connection.
5	USB-B laptop connection.
6	HDMI laptop connection.
7	USB-B Room PC connection.
8	HDMI Room PC connection.
9	USB devices.
10	HDMI share output from laptop.
11	HDMI display output.
12	LAN interface.
13	RS232 and remote interface.
14	GPI/button interface.

## LEDS BEHAVIOR

Here are the LEDs behavior:

LAPTOP HOSTS THE MEETING	
<b>OFF</b>	Laptop not selected.
<b>SOLID</b>	Laptop selected.
<b>BLINK</b>	Error condition. 1. When the user tries to switch to laptop if this one is not present or if USB or HDMI connections are missing. 2. When the user tries to switch host if button is locked through our API.
PWR	
<b>OFF</b>	Device not powered.
<b>SOLID</b>	Device powered.
Charging 	
<b>OFF</b>	Laptop is not charging.
<b>SOLID</b>	Laptop is charging.

## OPERATING MODES

There are the operating modes supported by the device. They will be explained here.

### AUTOMATIC

**This is the default mode.** This mode will switch automatically to the last source (USB or HDMI) connected if the operation mode is set to BYOM. If the current source is disconnected, the device will switch back to the other source if it is detected. Push-button action and remote control are also supported.

### MANUAL

The manual mode will enable you to force a specific source selection. Push-button action and remote control are also supported.

### MANUAL WITH FALLBACK

The manual mode with fallback supports the same features as the manual mode. It will only add the possibility to switch to the other detected source connection automatically if the selected source is disconnected.

## SPECIFICATIONS

Here is the complete specification.

Physical details	
<b>Dimensions (W x L x H)</b>	25.11 cm x 10.97 cm x 3.26 cm 9.89" x 4.32" x 1.28"
<b>Weight</b>	770g
<b>Power supply</b>	160W (85-264VAC 50/60Hz to 24V/6.67A DC)
<b>Power supply dimensions (W x L x H)</b>	175 mm x 72 mm x 35 mm 6.89" x 2.83" x 1.38"
<b>Package contents</b>	<ul style="list-style-type: none"> <li>1 x Toggle Rooms</li> <li>1 x USB-C to USB-C cable – 6ft</li> <li>1 x USB3.0 cable (USB-A to USB-B) – 3ft</li> <li>2 x terminal block 4-pos</li> <li>2 x mounting brackets</li> <li>4 x M2.5 mounting screws for brackets on product</li> <li>4 x screws for Toggle Rooms table/wall mount</li> <li>1 x 24V/160W PSU</li> <li>1 x AC power cord</li> <li>1 x country-specific power plug (USA/CA or EU/UK/AU/BIS)</li> <li>1 x PSU mounting brackets</li> <li>4 x screws for PSU table/wall mount</li> <li>1 x quickstart guide</li> <li>4 x rubber feet</li> </ul>
<b>Operating temperature</b>	0° to 45° C (32° to 113° F)
<b>Storage temperature</b>	-40° to 105° C (-40° to 221° F)
<b>Relative humidity</b>	0% to 90% non-condensing
<b>Mounting options</b>	Ability to mount under the table or on a wall.
<b>UPC code</b>	051497418694
<b>Origin</b>	Canada
<b>Warranty</b>	2 years

HOST - LAPTOP	
<b>1x USB-C connector</b>	Supports USB-C DisplayPort Alternate Mode <ul style="list-style-type: none"> <li>- DisplayPort up to 3840x2160p60 / 4096x2160p60</li> <li>- USB3.0 (USB 3.1 Gen 1 / 5 Gbps)</li> <li>- USB2.0 (480 Mbps)</li> <li>- Charging up to 100W</li> <li>- USB-C cable locking option</li> </ul>
<b>1x USB connector</b>	USB 3.0 Type-B
<b>1x HDMI connector</b>	HDMI 2.0 – Up to 3840x2160p60 / 4096x2160p60 – 18Gbps Cable locking option.

HOST - ROOMPC	
<b>1x USB connector</b>	USB 3.0 Type-B
<b>1x HDMI connector</b>	HDMI 2.0 – Up to 3840x2160p60 / 4096x2160p60 – 18Gbps Cable locking option.

HDMI DISPLAY output	
<b>Resolution</b>	HDMI 2.0 – Up to 3840x2160p60 / 4096x2160p60 – 18Gbps
<b>Connector</b>	HDMI with cable locking option.

HDMI SHARE output	
<b>Resolution</b>	Up to 3840x2160p60 / 4096x2160p60 – 18Gbps
<b>Connector</b>	HDMI with cable locking option.



USB devices	
<b>Connectors</b>	3 x USB 3.0 Type-A ports.
<b>Power</b>	1.8A shared between downstream ports.

Control	
<b>Control options</b>	Front button – for laptop selection RS232 GPI LAN USB
<b>IP interface</b>	10/100Mbps Supports DHCP or static addressing. IP control available through RESTAPI and telnet connections.
<b>RS232 interface</b>	4-pos terminal block connector Baud rates: 9600 [default], 19200, 38400 and 115200 Data bits: 8 Stop bits: 1 Parity: None Flow control: None
<b>GPI interface</b>	4-pos terminal block connector 2x Contact-closure control. GPI: <ul style="list-style-type: none"> <li>- Controlled by open-drain IO (short to ground) or driven IO.</li> <li>- Supported voltage range: 0 to 12V max.</li> <li>- Voltage threshold is 2.3V.</li> </ul> VOUT: <ul style="list-style-type: none"> <li>- Able to power up the led on the button of our INO-Button accessory.</li> <li>- Logic-low level: 0 @ 0.5V</li> <li>- Logic-high level: 4.5 @ 5V</li> </ul>

HDMI video	
<b>HDCP compliance</b>	Compliant with HDCP2.3, HDCP2.2 and HDCP1.4
<b>HDMI compliance</b>	Compliant with HDMI2.0b, HDMI1.4 and DVI1.0
<b>Sampling frequency</b>	600MHz
<b>Video scaling</b>	Crosspoint switch supports video downscaling from 4K to 1080p.
<b>Chroma subsampling</b>	YUV/RGB 4:4:4, 4:2:2
<b>CEC</b>	Ability to send CEC commands to connected HDMI display sink.

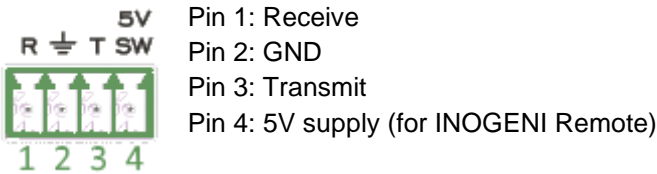
HDMI audio	
<b>Audio</b>	Audio passthrough from input to output
<b>Formats</b>	LPCM, Dolby Digital, DTS up to 192kHz

Certifications	
<b>Device</b>	FCC, CE, UKCA, RoHS, IEC62368, RCM, SoV
<b>Power supply</b>	FCC, CE, UKCA, RoHS, IEC62368, RCM, CCC, CB, EAC, VI, UL
<b>TAA-compliance</b>	Yes

Compatibility	
<b>Operating system</b>	NO driver installation necessary Windows 7 and above (32/64-bit) macOS 10.10 and above Linux (kernel v2.6.38 and above)

## SERIAL COMMUNICATION PROTOCOL

Here is the complete list of commands provided through the serial connection. As written on the back of the device, here is the pinout of the terminal block.



**NOTE:** The user needs to put a **space character** between the command name and argument.

<b>ARG</b>	Lists all the available options for the arguments to be used with the command.
<b>RX</b>	When command does not have any argument or only first argument is provided, it will return information from the device.
<b>TX</b>	When command have all arguments, it will apply the configuration to the device.

You need to add a carriage return **<CR>** character and a line feed **<LF>** character at the end of the command string.

Typically, commands will return "ACK**<CR><LF>**" in case of success and "NACK**<CR><LF>**" in case of failure.

**Baud rate:** 9600 [default] // **Data bits:** 8 // **Stop bits:** 1 // **Parity:** None // **Flow control:** None


Command	REQ/ ARG	Arguments	Return
<b>HELP</b> Return commands list with description.	RX	N/A	List of all the supported commands.
<b>RSTR</b> Restore default settings (including password and REST API token).	RX	N/A	ACK <b>&lt;CR&gt;&lt;LF&gt;</b>
<b>REBOOT</b> Reboot the device.	RX	N/A	ACK <b>&lt;CR&gt;&lt;LF&gt;</b>
<b>VERSION</b> Return firmware version.	RX	N/A	MAJOR=<Integer> <b>&lt;CR&gt;&lt;LF&gt;</b> MINOR=<Integer> <b>&lt;CR&gt;&lt;LF&gt;</b> ACK <b>&lt;CR&gt;&lt;LF&gt;</b>
<b>STATUS</b> Return laptop and RoomPC information, display and share output timings.	RX	N/A	List of all the status of the device.
<b>USBHOST</b> Get/Set USB host to use.	ARG	<b>&lt;host&gt; options:</b> 0 => RoomPC 1 => Laptop USB-C 2 => Laptop USB-B 3 => OFF	
		TX	<host>
	RX	N/A	USBHOST=<host> <b>&lt;CR&gt;&lt;LF&gt;</b> ACK <b>&lt;CR&gt;&lt;LF&gt;</b>
<b>DISPLAYSRC</b> Get/Set which HDMI source to be routed to display output.	ARG	<b>&lt;src&gt; options:</b> 0 => RoomPC 1 => Laptop USB-C 2 => Laptop HDMI 3 => OFF	
		TX	<src>

Command	REQ/ ARG	Arguments	Return
	RX	N/A	DISPLAYSRC=<src><CR><LF> ACK<CR><LF>
<b>SHARESRC</b>  Get/Set which HDMI source to be routed to share output.		<b>&lt;src&gt; options:</b> 0 => RoomPC [Not supported in automatic mode] 1 => Laptop USB-C 2 => Laptop USB-B 3 => OFF	
	TX	<src>	ACK<CR><LF>
	RX	N/A	SHARESRC=<src><CR><LF> ACK<CR><LF>
<b>OPMODE</b>  Get/Set operation mode.  By default, the device will operate in <b>RoomPC / BYOD mode</b> – RoomPC USB and HDMI peripherals selected, and laptop sends video content only to SHARE output. The user will need to trigger our API or use the GPI interface to enter BYOM mode.  When <b>BYOM mode</b> is set, the device will automatically switch all HDMI and USB peripherals to the laptop as soon as it is detected.  When <b>Custom mode</b> is set, the user can set the USB, display and share source switching modes independently.		<b>&lt;opMode&gt; options:</b> 0 => RoomPC with BYOD/content sharing [default] 1 => BYOM 2 => Custom	
	TX	<opMode>	ACK<CR><LF>
	RX	N/A	OPMODE=<opMode><CR><LF> ACK<CR><LF>
<b>USBHOSTSWMODE</b>  Get/Set USB host switching mode. The operation mode must be set to “Custom” to use this.		<b>&lt;swMode&gt; options:</b> 0 => Automatic mode [default] 1 => Manual mode 2 => Manual mode with fallback	
	TX	<swMode>	ACK<CR><LF>
	RX	N/A	USBHOSTSWMODE=<swMode><CR><LF> ACK<CR><LF>
<b>DISPLAYSWMODE</b>  Get/Set HDMI display source switching mode. The operation mode must be set to “Custom” to use this.		<b>&lt;swMode&gt; options:</b> 0 => Automatic mode [default] 1 => Manual mode 2 => Manual mode with fallback	
	TX	<swMode>	ACK<CR><LF>
	RX	N/A	DISPLAYSWMODE=<swMode><CR><LF> ACK<CR><LF>
<b>SHARESWMODE</b>  Get/Set HDMI share source switching mode. The operation mode must be set to “Custom” to use this.		<b>&lt;swMode&gt; options:</b> 0 => Automatic mode [default] 1 => Manual mode 2 => Manual mode with fallback	
	TX	<swMode>	ACK<CR><LF>
	RX	N/A	SHARESWMODE=<swMode><CR><LF> ACK<CR><LF>
<b>PRIORUSBHOST</b>  Get/Set USB priority. Only applicable when USB host switching mode is automatic.		<b>&lt;host&gt; options:</b> 0 => RoomPC 1 => Laptop USB-C 2 => Laptop USB-B 3 => Last detected host [default]	
	TX	<host>	ACK<CR><LF>
	RX	N/A	PRIORUSBHOST=<host><CR><LF> ACK<CR><LF>

Command	REQ/ ARG	Arguments	Return	
<b>PRIORDISPLAYSRC</b>  Get/Set display source priority. Only applicable when display source switching mode is automatic.	<b>ARG</b>	<b>&lt;src&gt; options:</b> 0 => RoomPC 1 => Laptop USB-C 2 => Laptop HDMI 3 => Last detected source [default]		
		<b>TX</b>	<b>&lt;src&gt;</b>	ACK<CR><LF>
		<b>RX</b>	N/A	PRIORDISPLAYSRC=<src><CR><LF> ACK<CR><LF>
<b>PRIORSHARESRC</b>  Get/Set share source priority. Only applicable when share source switching mode is automatic.	<b>ARG</b>	<b>&lt;src&gt; options:</b> 0 => RoomPC 1 => Laptop USB-C 2 => Laptop HDMI 3 => Last detected source [default]		
		<b>TX</b>	<b>&lt;src&gt;</b>	ACK<CR><LF>
		<b>RX</b>	N/A	PRIORSHARESRC=<src><CR><LF> ACK<CR><LF>
<b>NETWORK</b>  Get/Set network settings.	<b>ARG</b>	<b>&lt;mode&gt; options:</b> static => addressing is static dhcp => use DHCP addressing  If mode is static, ip and netmask are required while gateway is optional.		
		<b>TX</b>	<b>&lt;mode&gt; &lt;ip&gt; &lt;netmask&gt; &lt;gateway&gt;</b>	ACK<CR><LF>
		<b>RX</b>	N/A	MODE=<mode><CR><LF> IP=<ip><CR><LF> NETMASK=<netmask><CR><LF> GATEWAY=<gateway><CR><LF> ACK<CR><LF>
<b>HOSTNAME</b>  Get/Set the hostname of the device. This command will change the device name when probed over the network and the name of the USB HID interface.	<b>ARG</b>	<b>&lt;hostname&gt; option:</b> String defined hostname to be shown on the network and USB HID interface. This string must not have space characters.		
		<b>TX</b>	<b>&lt;hostname&gt;</b>	ACK<CR><LF>
		<b>RX</b>	N/A	HOSTNAME=<hostname><CR><LF> ACK<CR><LF>
<b>CECPASSTHROUGHEN</b>  Get/Set CEC passthrough setting from source to display. This setting allows CEC commands to be sent or not from the video source to the connected display. Manual CEC commands will continue to work regardless of this setting.	<b>ARG</b>	<b>&lt;enable&gt; options:</b> 0 => OFF 1 => ON		
		<b>TX</b>	<b>&lt;enable&gt;</b>	ACK<CR><LF>
		<b>RX</b>	N/A	ENABLE=<enable><CR><LF> ACK<CR><LF>
<b>CECPOWER</b>  Power ON/OFF the display.	<b>ARG</b>	<b>&lt;ctrl&gt; options:</b> 0 => power off 1 => power on		
		<b>TX</b>	<b>&lt;ctrl&gt;</b>	ACK<CR><LF>
<b>CECTOGGLEMUTE</b>  Toggle mute control.	<b>TX</b>		ACK<CR><LF>	

Command	REQ/ ARG	Arguments	Return
<b>CECVOLUP</b> Increase display volume.	<b>TX</b>		ACK<CR><LF>
<b>CECVOLDOWN</b> Decrease display volume.	<b>TX</b>		ACK<CR><LF>
<b>EDID</b> Set specific EDID modes to be reported to video source.		<b>&lt;src&gt; options:</b> 0 => RoomPC 1 => Laptop USB-C 2 => Laptop HDMI  <b>&lt;edid&gt; options:</b> 0 => Passthrough 1 => User EDID 2 => 3840x2160p60 3 => 3840x2160p50 4 => 3840x2160p30 5 => 3840x2160p25 6 => 1920x1080p60 7 => 1920x1080p50 8 => 1280x720p60 9 => 1280x720p50 10 => 5120x2160p30 11 => 5120x2160p25	
	<b>TX</b>	<src> <edid>	ACK<CR><LF>
	<b>RX</b>	<src>	EDID=<edid><CR><LF> ACK<CR><LF>
<b>EDIDUSR</b> Set user EDID to be sent to specified source. Must have set the according video source EDID in user EDID mode.	<b>ARG</b>	<b>&lt;src&gt; options</b> 0 => RoomPC 1 => Laptop USB-C 2 => Laptop HDMI  <b>&lt;edidUsr&gt; =&gt; formatted 256 bytes array</b>	
	<b>TX</b>	<src> <256 bytes array>	ACK<CR><LF>
	<b>RX</b>	<src>	EDIDUSR=<edidUsr><CR><LF> ACK<CR><LF>
<b>USBC4K60EN</b> Get/Set the USB-C working mode.  <b>NOTE:</b> Enabling DisplayPort signal to support 4K60 will disable USB3.0 connectivity on USB-C port. USB2.0 will remain active.  Disabling this option will allow user to support USB3.0 and 4K30 video.	<b>ARG</b>	<b>&lt;mode&gt; options:</b> 0 => Disable 4K60 [default] 1 => Enable 4K60	
	<b>TX</b>	<mode>	ACK<CR><LF>
	<b>RX</b>	N/A	USBC4K60EN=<mode><CR><LF> ACK<CR><LF>
<b>HDCPCTL</b> Get/Set the HDCP setting.	<b>ARG</b>	<b>&lt;src&gt; options:</b> 0 => RoomPC 1 => Laptop USB-C 2 => Laptop HDMI  <b>&lt;hdcp&gt; options:</b> 0 => Disabled 1 => HDCP v1.4 2 => HDCP v2.2 3 => Auto	
	<b>TX</b>	<src> <hdcp>	ACK<CR><LF>
	<b>RX</b>	<src>	HDCP=<hdcp><CR><LF> ACK<CR><LF>

Command	REQ/ ARG	Arguments	Return
<b>GPICFG</b>  Get/Set the GPI configuration.  <b>NOTE:</b> In <b>pulse mode</b> , a short to GND on this pin will trigger the function. The function will be executed on GPI falling edge. GPI rising edge has no effect.  In <b>level mode</b> , the function will be executed on short to GND and open states.		<b>&lt;gpi&gt; options:</b> 1 => GPI1 2 => GPI2  <b>&lt;mode&gt; options:</b> 0 => Pulse mode [default] 1 => Level mode  <b>&lt;function&gt; options:</b> 0 => Disabled. 1 => BYOM mode control [default GPI1] SHORT = BYOM OPEN = ROOMPC 2 => USB host control [default GPI2] SHORT = LAPTOP OPEN = ROOMPC 3 => Display video source control SHORT = LAPTOP USB-C/HDMI OPEN = ROOMPC 4 => Laptop video source control SHORT = LAPTOP USB-C OPEN = LAPTOP HDMI	
	<b>TX</b>	<gpi> <mode> <function>	ACK<CR><LF>
	<b>RX</b>	<gpi>	MODE=<mode><CR><LF> FUNCTION=<function><CR><LF> ACK<CR><LF>
<b>VOUT</b>  Get/Set the VOUT level.  <b>NOTE:</b> By default, the firmware will drive this output to power up a led when enabling the BYOM mode. If the user wants to override this behavior, it is possible by changing the <vout> option.	<b>ARG</b>	<b>&lt;vout&gt; options:</b> 0 => Controlled by firmware. 1 => Logic-low. 2 => Logic-high.	
	<b>TX</b>	<vout>	ACK<CR><LF>
	<b>RX</b>	N/A	VOUT=<vout><CR><LF> ACK<CR><LF>
<b>BAUDRATE</b>  Set RS232 baud rate.	<b>ARG</b>	<b>&lt;baudrate&gt; options</b> 0 => 9600 1 => 19200 2 => 38400 3 => 115200	
	<b>TX</b>	<baudrate>	ACK<CR><LF>
	<b>RX</b>	N/A	BAUDRATE=<baudrate><CR><LF> ACK<CR><LF>
<b>BTNLOCK</b>  Get/Set the button lock status.	<b>ARG</b>	<b>&lt;lockState&gt; options:</b> 0 => Not locked 1 => Locked	
	<b>TX</b>	<lockState>	ACK<CR><LF>
	<b>RX</b>	N/A	BTNLOCK=<lockState><CR><LF> ACK<CR><LF>
<b>SCALER</b>  Get/Set the scaler options over the HDMI video outputs.	<b>ARG</b>	<b>&lt;output&gt; options:</b> 0 => Display output 1 => Share output  <b>&lt;enable&gt; options:</b> 0 => OFF 1 => ON	
	<b>TX</b>	<output> <enable>	ACK<CR><LF>
	<b>RX</b>	<output>	ENABLE=<enable><CR><LF> ACK<CR><LF>

Command	REQ/ ARG	Arguments	Return																																				
<b>USBDEVEN</b>  Get/Set the power on USB devices ports according to specific hosts.		<b>&lt;host&gt; options:</b> 0 => RoomPC 1 => Laptop USB-C 2 => Laptop HDMI 3 => When no host detected  <b>&lt;devices&gt; options:</b> Bitmask to enabled ports.																																					
	ARG																																						
		<table border="1"> <thead> <tr> <th>&lt;devices&gt;</th> <th>USB #1</th> <th>USB #2</th> <th>USB #3</th> </tr> </thead> <tbody> <tr><td>0</td><td>OFF</td><td>OFF</td><td>OFF</td></tr> <tr><td>1</td><td>ON</td><td>OFF</td><td>OFF</td></tr> <tr><td>2</td><td>OFF</td><td>ON</td><td>OFF</td></tr> <tr><td>3</td><td>ON</td><td>ON</td><td>OFF</td></tr> <tr><td>4</td><td>OFF</td><td>OFF</td><td>ON</td></tr> <tr><td>5</td><td>ON</td><td>OFF</td><td>ON</td></tr> <tr><td>6</td><td>OFF</td><td>ON</td><td>ON</td></tr> <tr><td>7</td><td>ON</td><td>ON</td><td>ON</td></tr> </tbody> </table>	<devices>	USB #1	USB #2	USB #3	0	OFF	OFF	OFF	1	ON	OFF	OFF	2	OFF	ON	OFF	3	ON	ON	OFF	4	OFF	OFF	ON	5	ON	OFF	ON	6	OFF	ON	ON	7	ON	ON	ON	
	<devices>	USB #1	USB #2	USB #3																																			
	0	OFF	OFF	OFF																																			
1	ON	OFF	OFF																																				
2	OFF	ON	OFF																																				
3	ON	ON	OFF																																				
4	OFF	OFF	ON																																				
5	ON	OFF	ON																																				
6	OFF	ON	ON																																				
7	ON	ON	ON																																				
<b>TX</b>	<host> <devices>		ACK<CR><LF>																																				
<b>RX</b>	<host>		DEVICES=<devices><CR><LF> ACK<CR><LF>																																				
<b>AUTOHDMICEPWR</b>  Get/Set the automatic CEC power control of the connected display. When enabled, the device will turn on/off the display depending on the actual state of the HDMI source routed to the display.		<b>&lt;enable&gt; options:</b> 0 => OFF 1 => ON																																					
	<b>TX</b>	<enable>		ACK<CR><LF>																																			
	<b>RX</b>	N/A		ENABLE=<enable><CR><LF> ACK<CR><LF>																																			
<b>HTTPEN</b>  Get/Set HTTP control setting.		<b>&lt;enable&gt; options:</b> 0 => OFF 1 => ON																																					
	<b>TX</b>	<enable>		ACK<CR><LF>																																			
	<b>RX</b>	N/A		ENABLE=<enable><CR><LF> ACK<CR><LF>																																			

You can enable a bearer authentication in the HTTP header (Authorization: Bearer <token>) through our configuration page to increase security on the API.

There will be a return code to each call with the following commands:

200 => success

400 => error

401 => authorization error

<b>ARG</b>	Lists all the available options for the arguments to be used with the command.
<b>RX</b>	When command does not have any body arguments or only first argument is provided, it will return information from the device.
<b>TX</b>	When command have all body arguments, it will apply the configuration to the device.

The return body will usually be JSON formatted with a "message" field containing a JSON string explaining the cause of the error or "success" in case of success. Note that we are using self-signed certificates.

Here is the complete list of commands supported through the REST API (excluding password change, firmware update, bearer token get/set):

Command URL / Description	REQ/ ARG	Body arguments	Return body
<b>HTTP GET</b> <b>https://&lt;IP&gt;/api/v1/help</b>  Return commands list with description.	RX	N/A	JSON object with multiple fields
<b>HTTP GET/POST</b> <b>https://&lt;IP&gt;/api/v1/rstr</b>  Restore default settings (including password and REST API token).	RX	N/A	{ "message": <String> }
<b>HTTP GET/POST</b> <b>https://&lt;IP&gt;/api/v1/reboot</b>  Reboot the device.	RX	N/A	{ "message": <String> }
<b>HTTP GET</b> <b>https://&lt;IP&gt;/api/v1/version</b>  Return firmware version.	RX	N/A	{ "major": <Integer>, "minor": <Integer> }
<b>HTTP GET</b> <b>https://&lt;IP&gt;/api/v1/status</b>  Return laptop and RoomPC information, display and share output timings.	RX	N/A	JSON object with multiple fields
<b>HTTP GET/POST</b> <b>https://&lt;IP&gt;/api/v1/usbHost</b>  Get/Set USB host to use.		<b>&lt;usbHost&gt; options:</b> 0 => RoomPC 1 => Laptop USB-C 2 => Laptop USB-B 3 => OFF	
	<b>TX</b>	usbHost=<host>	{ "message": <String> }
	<b>RX</b>	N/A	{ "usbHost": <host>, "message": <String> }




Command URL / Description	REQ/ ARG	Body arguments	Return body
<b>HTTP GET/POST</b> <b>https://&lt;IP&gt;/api/v1/ displaySrc</b>	ARG	<b>&lt;displaySrc&gt; options:</b> 0 => RoomPC 1 => Laptop USB-C 2 => Laptop HDMI 3 => OFF	
	TX	displaySrc=<src>	{ "message": <String> }
	RX	N/A	{ "displaySrc": <src>, "message": <String> }
<b>HTTP GET/POST</b> <b>https://&lt;IP&gt;/api/v1/ shareSrc</b>	ARG	<b>&lt;shareSrc&gt; options:</b> 0 => RoomPC [Not supported in automatic mode] 1 => Laptop USB-C 2 => Laptop HDMI 3 => OFF	
	TX	shareSrc=<src>	{ "message": <String> }
	RX	N/A	{ "shareSrc": <src>, "message": <String> }
<b>HTTP GET/POST</b> <b>https://&lt;IP&gt;/api/v1/ opMode</b>	ARG	<b>&lt;opMode&gt; options:</b> 0 => RoomPC with BYOD/content sharing [default] 1 => BYOM 2 => Custom	
	TX	opMode=<opMode>	{ "message": <String> }
	RX	N/A	{ "opMode": <opMode>, "message": <String> }
<b>HTTP GET/POST</b> <b>https://&lt;IP&gt;/api/v1/ usbHostSwMode</b>	ARG	<b>&lt;usbHostSwMode&gt; options:</b> 0 => Automatic mode [default] 1 => Manual mode 2 => Manual mode with fallback	
	TX	usbHostSwMode=<swMode>	{ "message": <String> }
	RX	N/A	{ "usbHostSwMode": <swMode>, "message": <String> }
<b>HTTP GET/POST</b> <b>https://&lt;IP&gt;/api/v1/ displaySwMode</b>	ARG	<b>&lt;displaySwMode&gt; options:</b> 0 => Automatic mode [default] 1 => Manual mode 2 => Manual mode with fallback	

Command URL / Description	REQ/ ARG	Body arguments	Return body
Get/Set HDMI display source switching mode. The operation mode must be set to "Custom" to use this.	<b>TX</b>	displaySwMode=<swMode>	{ "message": <String> }
	<b>RX</b>	N/A	{ "displaySwMode": <swMode>, "message": <String> }
<b>HTTP GET/POST</b> <b>https://&lt;IP&gt;/api/v1/ shareSwMode</b>	<b>ARG</b>	<b>&lt;shareSwMode&gt; options:</b> 0 => Automatic mode [default] 1 => Manual mode 2 => Manual mode with fallback	
		<b>TX</b>	shareSwMode=<swMode>
	<b>RX</b>	N/A	{ "shareSwMode": <swMode>, "message": <String> }
	Get/Set HDMI share source switching mode. The operation mode must be set to "Custom" to use this.		
<b>HTTP GET/POST</b> <b>https://&lt;IP&gt;/api/v1/ priorUsbHost</b>	<b>ARG</b>	<b>&lt;host&gt; options:</b> 0 => RoomPC 1 => Laptop USB-C 2 => Laptop USB-B 3 => Last detected host	
		<b>TX</b>	<host>
	<b>RX</b>	N/A	{ "priorUsbHost": <host>, "message": <String> }
	Get/Set USB priority. Only applicable when USB host switching mode is automatic.		
<b>HTTP GET/POST</b> <b>https://&lt;IP&gt;/api/v1/ priorDisplaySrc</b>	<b>ARG</b>	<b>&lt;src&gt; options:</b> 0 => RoomPC 1 => Laptop USB-C 2 => Laptop HDMI 3 => Last detected source [default]	
		<b>TX</b>	<src>
	<b>RX</b>	N/A	{ "priorDisplaySrc": <src>, "message": <String> }
	Get/Set display source priority. Only applicable when display source switching mode is automatic.		
<b>HTTP GET/POST</b> <b>https://&lt;IP&gt;/api/v1/ priorShareSrc</b>	<b>ARG</b>	<b>&lt;src&gt; options:</b> 0 => RoomPC 1 => Laptop USB-C 2 => Laptop HDMI 3 => Last detected source [default]	
		<b>TX</b>	<src>
	<b>RX</b>	N/A	{ "priorShareSrc": <src>, "message": <String> }
	Get/Set share source priority. Only applicable when share source switching mode is automatic.		

Command URL / Description	REQ/ ARG	Body arguments	Return body
<b>HTTP GET/POST</b> <b>https://&lt;IP&gt;/api/v1/network</b>  Get/Set network settings.		<b>&lt;mode&gt; options:</b> static => addressing is static dhcp => use DHCP addressing  If mode is static, ip and netmask are required while gateway is optional.	
	ARG	<b>&lt;ip&gt; option:</b> String defined IP address. Example: 192.168.0.20  <b>&lt;netmask&gt; option:</b> String defined netmask address. Example: 255.255.0.0  <b>&lt;gateway&gt; option:</b> String defined gateway address. Example: 192.168.0.1	
	TX	mode=<mode> ip=<ip> netmask=<netmask> gateway=<gateway>	<pre>{   "message": &lt;String&gt; }</pre>
	RX	N/A	<pre>{   "mode": &lt;static,dhcp&gt;,   "ip": &lt;ip&gt;,   "netmask": &lt;netmask&gt;,   "gateway": &lt;gateway&gt;,   "message": &lt;String&gt; }</pre>
<b>HTTP GET/POST</b> <b>https://&lt;IP&gt;/api/v1/hostname</b>  Get/Set the hostname of the device. This command will change the device name when probed over the network and the name of the USB HID interface.	ARG	<b>&lt;hostname&gt; option:</b> String defined hostname to be shown on the network and USB HID interface. This string must not have space characters.	
	TX	hostname=<hostname>	<pre>{   "message": &lt;String&gt; }</pre>
	RX	N/A	<pre>{   "hostname": &lt;hostname&gt;,   "message": &lt;String&gt; }</pre>
<b>HTTP GET/POST</b> <b>https://&lt;IP&gt;/api/v1/cecPassthroughEn</b>  Get/Set CEC passthrough setting from source to display. This setting allows CEC commands to be sent or not from the video source to the connected display. Manual CEC commands will continue to work regardless of this setting.	ARG	<b>&lt;enable&gt; options:</b> 0 => OFF 1 => ON	
	TX	enable=<enable>	<pre>{   "message": &lt;String&gt; }</pre>
	RX	N/A	<pre>{   "enable": &lt;enable&gt;,   "message": &lt;String&gt; }</pre>
<b>HTTP GET/POST</b> <b>https://&lt;IP&gt;/api/v1/cecPower</b>  Power ON/OFF the display.	ARG	<b>&lt;ctrl&gt; options:</b> 0 => power off 1 => power on	
	TX	ctrl=<ctrl>	<pre>{   "message": &lt;String&gt; }</pre>
<b>HTTP GET/POST</b> <b>https://&lt;IP&gt;/api/v1/cecToggleMute</b>  Toggle mute control.	TX	N/A	<pre>{   "message": &lt;String&gt; }</pre>
<b>HTTP GET/POST</b> <b>https://&lt;IP&gt;/api/v1/cecVolUp</b>  Increase display volume.	TX	N/A	<pre>{   "message": &lt;String&gt; }</pre>

Command URL / Description	REQ/ ARG	Body arguments	Return body
<b>HTTP GET/POST</b> <b>https://&lt;IP&gt;/api/v1/cecVolDown</b>  Decrease display volume.	TX	N/A	{ "message": <String> }
<b>HTTP GET/POST</b> <b>https://&lt;IP&gt;/api/v1/edid</b>  Set specific EDID modes to be reported to video source.	ARG	<b>&lt;src&gt; options:</b> 0 => RoomPC 1 => Laptop USB-C 2 => Laptop HDMI  <b>&lt;edid&gt; options:</b> 0 => Passthrough 1 => User EDID 2 => 3840x2160p60 3 => 3840x2160p50 4 => 3840x2160p30 5 => 3840x2160p25 6 => 1920x1080p60 7 => 1920x1080p50 8 => 1280x720p60 9 => 1280x720p50 10 => 5120x2160p30 11 => 5120x2160p25	
	TX	src=<src> edid=<edid>	{ "message": <String> }
	RX	src=<src>	{ "edid": <edid>, "message": <String> }
<b>HTTP GET/POST</b> <b>https://&lt;IP&gt;/api/v1/edidUsr</b>  Set user EDID to be sent to specified source. Must have set the according video source EDID in user EDID mode.	ARG	<b>&lt;src&gt; options</b> 0 => RoomPC 1 => Laptop USB-C 2 => Laptop HDMI  <b>&lt;edidUsr&gt; =&gt; Filetype formatted 256 bytes array</b>	
	TX	src=<src> edidUsr=<256 bytes array>	{ "message": <String> }
	RX	src=<src>	{ "edidUsr": <edidUsr>, "message": <String> }
<b>HTTP GET/POST</b> <b>https://&lt;IP&gt;/api/v1/usbc4K60En</b>  Get/Set the USB-C working mode.  <b>NOTE:</b> Enabling DisplayPort signal to support 4K60 will disable USB3.0 connectivity on USB-C port. USB2.0 will remain active.  Disabling this option will allow user to support USB3.0 and 4K30 video.	ARG	<b>&lt;mode&gt; options:</b> 0 => Disable 4K60 [default] 1 => Enable 4K60	
	TX	usbc4K60En=<mode>	{ "message": <String> }
	RX	N/A	{ "usbc4K60En": <mode>, "message": <String> }

Command URL / Description	REQ/ ARG	Body arguments	Return body	
<b>HTTP GET/POST</b> <b>https://&lt;IP&gt;/api/v1/hdcpCtl</b>  Get/Set the HDCP setting.		<b>&lt;src&gt; options:</b> 0 => RoomPC 1 => Laptop USB-C 2 => Laptop HDMI		
	<b>ARG</b>	<b>&lt;hdcp&gt; options:</b> 0 => Disabled 1 => HDCP v1.4 2 => HDCP v2.2 3 => Auto		
	<b>TX</b>	src=<src> hdcp=<hdcp>	<pre>{   "message": &lt;String&gt; }</pre>	
	<b>RX</b>	src=<src>	<pre>{   "hdcp": &lt;hdcp&gt;,   "message": &lt;String&gt; }</pre>	
<b>HTTP GET/POST</b> <b>https://&lt;IP&gt;/api/v1/gpiCfg</b>  Get/Set the GPI configuration.  <b>NOTE:</b> In <b>pulse mode</b> , a short to GND on this pin will trigger the function. The function will be executed on GPI falling edge. GPI rising edge has no effect.  In <b>level mode</b> , the function will be executed on short to GND and open states.	<b>ARG</b>	<b>&lt;gpi&gt; options:</b> 1 => GPI1 2 => GPI2		
		<b>&lt;mode&gt; options:</b> 0 => Pulse mode [default] 1 => Level mode		
		<b>&lt;function&gt; options:</b> 0 => Disabled. 1 => BYOM mode control [default GPI1] SHORT = BYOM OPEN = ROOMPC 2 => USB host control [default GPI2] SHORT = LAPTOP OPEN = ROOMPC 3 => Display video source control SHORT = LAPTOP USB-C/HDMI OPEN = ROOMPC 4 => Laptop video source control SHORT = LAPTOP USB-C OPEN = LAPTOP HDMI		
	<b>TX</b>	gpi=<gpi> mode=<mode> function=<function>	<pre>{   "message": &lt;String&gt; }</pre>	
	<b>RX</b>	gpi=<gpi>	<pre>{   "mode": &lt;mode&gt;,   "function": &lt;function&gt;,   "message": &lt;String&gt; }</pre>	
<b>HTTP GET/POST</b> <b>https://&lt;IP&gt;/api/v1/vout</b>  Get/Set the VOUT level.  <b>NOTE:</b> By default, the firmware will drive this output to power up a led when enabling the BYOM mode. If the user wants to override this behavior, it is possible by changing the <vout> option.	<b>ARG</b>	<b>&lt;vout&gt; options:</b> 0 => Controlled by firmware. 1 => Logic-low. 2 => Logic-high.		
		<b>TX</b>	vout=<vout>	<pre>{   "message": &lt;String&gt; }</pre>
		<b>RX</b>	N/A	<pre>{   "vout": &lt;vout&gt;,   "message": &lt;String&gt; }</pre>

Command URL / Description	REQ/ ARG	Body arguments	Return body																																		
<b>HTTP GET/POST</b> <b>https://&lt;IP&gt;/api/v1/  baudRate</b>  Set RS232 baud rate.	<b>ARG</b>   	<b>&lt;baudrate&gt; options</b> 0 => 9600 1 => 19200 2 => 38400 3 => 115200																																			
		<b>TX</b> baudrate=<baudrate>	{ "message": <String> }																																		
		<b>RX</b> N/A	{ "baudrate": <baudrate>, "message": <String> }																																		
<b>HTTP GET/POST</b> <b>https://&lt;IP&gt;/api/v1/  btnLock</b>  Get/Set the button lock status.	<b>ARG</b>   	<b>&lt;lockState&gt; options:</b> 0 => Not locked 1 => Locked																																			
		<b>TX</b> btnLock=<lockState>	{ "message": <String> }																																		
		<b>RX</b> N/A	{ "btnLock": <lockState>, "message": <String> }																																		
<b>HTTP GET/POST</b> <b>https://&lt;IP&gt;/api/v1/  scaler</b>  Get/Set the scaler options over the HDMI video outputs.	<b>ARG</b>   	<b>&lt;output&gt; options:</b> 0 => Display output 1 => Share output  <b>&lt;enable&gt; options:</b> 0 => OFF 1 => ON																																			
		<b>TX</b> output=<output> enable=<enable>	{ "message": <String> }																																		
		<b>RX</b> output=<output>	{ "enable": <enable>, "message": <String> }																																		
<b>HTTP GET/POST</b> <b>https://&lt;IP&gt;/api/v1/  usbDevEn</b>  Get/Set the power on USB devices ports according to specific hosts.	<b>ARG</b>   	<b>&lt;host&gt; options:</b> 0 => RoomPC 1 => Laptop USB-C 2 => Laptop HDMI 3 => When no host detected.  <b>&lt;devices&gt; options:</b> <b>Bitmask to enabled ports.</b>																																			
																																					
		<table border="1"> <thead> <tr> <th>&lt;devices&gt;</th> <th>USB #1</th> <th>USB #2</th> <th>USB #3</th> </tr> </thead> <tbody> <tr><td>0</td><td>OFF</td><td>OFF</td><td>OFF</td></tr> <tr><td>1</td><td>ON</td><td>OFF</td><td>OFF</td></tr> <tr><td>2</td><td>OFF</td><td>ON</td><td>OFF</td></tr> <tr><td>3</td><td>ON</td><td>ON</td><td>OFF</td></tr> <tr><td>4</td><td>OFF</td><td>OFF</td><td>ON</td></tr> <tr><td>5</td><td>ON</td><td>OFF</td><td>ON</td></tr> <tr><td>6</td><td>OFF</td><td>ON</td><td>ON</td></tr> <tr><td>7</td><td>ON</td><td>ON</td><td>ON</td></tr> </tbody> </table>	<devices>	USB #1	USB #2	USB #3	0	OFF	OFF	OFF	1	ON	OFF	OFF	2	OFF	ON	OFF	3	ON	ON	OFF	4	OFF	OFF	ON	5	ON	OFF	ON	6	OFF	ON	ON	7	ON	ON
<devices>	USB #1	USB #2	USB #3																																		
0	OFF	OFF	OFF																																		
1	ON	OFF	OFF																																		
2	OFF	ON	OFF																																		
3	ON	ON	OFF																																		
4	OFF	OFF	ON																																		
5	ON	OFF	ON																																		
6	OFF	ON	ON																																		
7	ON	ON	ON																																		
<b>TX</b> host=<host> devices=<devices>	{ "message": <String> }																																				

Command URL / Description	REQ/ ARG	Body arguments	Return body
	RX	host=<host>	{ "devices": <devices>, "message": <String> }
<b>HTTP GET/POST</b> <b>https://&lt;IP&gt;/api/v1/ autoHdmiCecPwr</b>	ARG	<b>&lt;enable&gt; options:</b> 0 => OFF 1 => ON	
Get/Set the automatic CEC power control of the connected display. When enabled, the device will turn on/off the display depending on the actual state of the HDMI source routed to the display.	TX	enable=<enable>	{ "message": <String> }
	RX	N/A	{ "enable": <enable>, "message": <String> }
<b>HTTP GET/POST</b> <b>https://&lt;IP&gt;/api/v1/ httpEn</b>	ARG	<b>&lt;enable&gt; options:</b> 0 => OFF 1 => ON	
Get/Set HTTP control setting.	TX	enable=<enable>	{ "message": <String> }
	RX	N/A	{ "enable": <enable>, "message": <String> }

It is also possible to embed arguments to an API call inside the URL to ease configuration with some control systems with the following topology:

**GET** https://<IP>/api/v1/<COMMAND>?<ARG1>=value&<ARG2>=value

where <COMMAND>, <ARG1> and <ARG2> are command and associated arguments.

For example, using the **usbHost** command, you can issue the following request:

**GET** https://<IP>/api/v1/**usbHost?host=1**

This request will set the USB host to laptop USB-C port.

The following commands allow to perform password management and bearer token management. The authentication used is basic auth, and we use the same user and password as the webpage (default user=admin and password=[SERIAL\_NUM] where [SERIAL\_NUM] is the serial number of the device located under the unit).

Command URL / Description	Body arguments	Return body
<b>HTTP POST</b> <b>https://&lt;IP&gt;/api/v1/ changeUsername? username=&lt;newUsername&gt;</b>		{ "message": <String> }
Change the username to <newUsername>.		
<b>HTTP POST</b> <b>https://&lt;IP&gt;/api/v1/ changePassword? password=&lt;newPassword&gt;</b>		{ "message": <String> }
Change the password to <newPassword>.		
<b>HTTP GET</b> <b>https://&lt;IP&gt;/api/v1/ getAccessToken</b>		{ "token": <String> }

Return the bearer token.	If no bearer token is set, the "token" field will be null.
<b>HTTP POST</b> <b>https://&lt;IP&gt;/api/v1/generateAccessToken</b>	<pre>{   "message": &lt;String&gt; }</pre>
Generate random access token.	

The bearer token is generated using a random process. The format of the bearer token only supports the following:

- Alphanumeric (A to Z) upper and lowercase characters.
- Numbers 0-9.

## TELNET

You can use any telnet application to communicate with the device using TCP. Make sure to use the right IP address and **port 23**.

Use the serial communication protocol to configure the device.

The `quit` command can be used to ask server for disconnection.



# INOGENI MAESTRO APPLICATION

You can use our INOGENI Maestro application to monitor firmware information and upgrade your unit.



**NOTE:** You need to use the USB-B to USB-A cable provided with the box for the Maestro application to detect the unit.

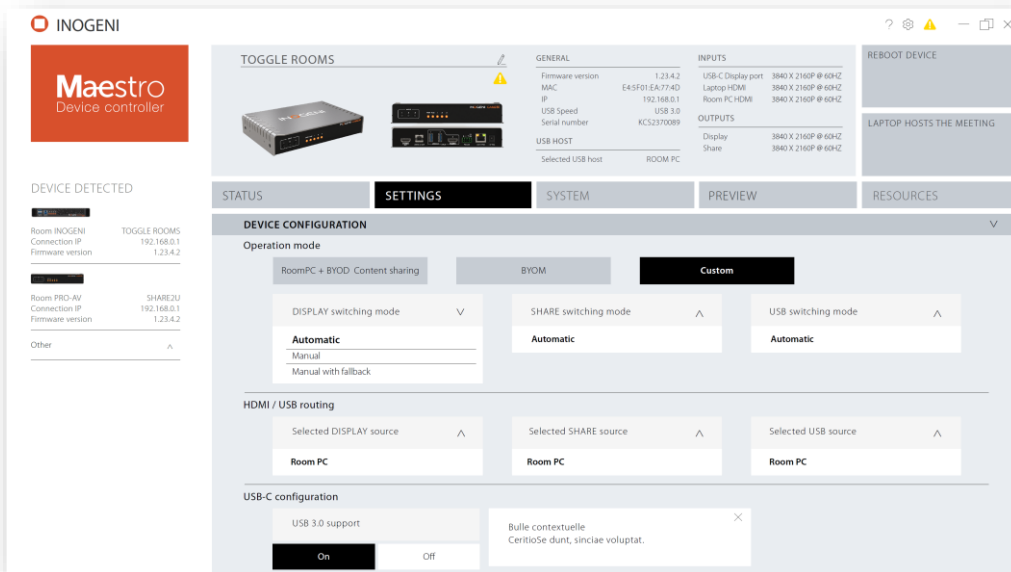
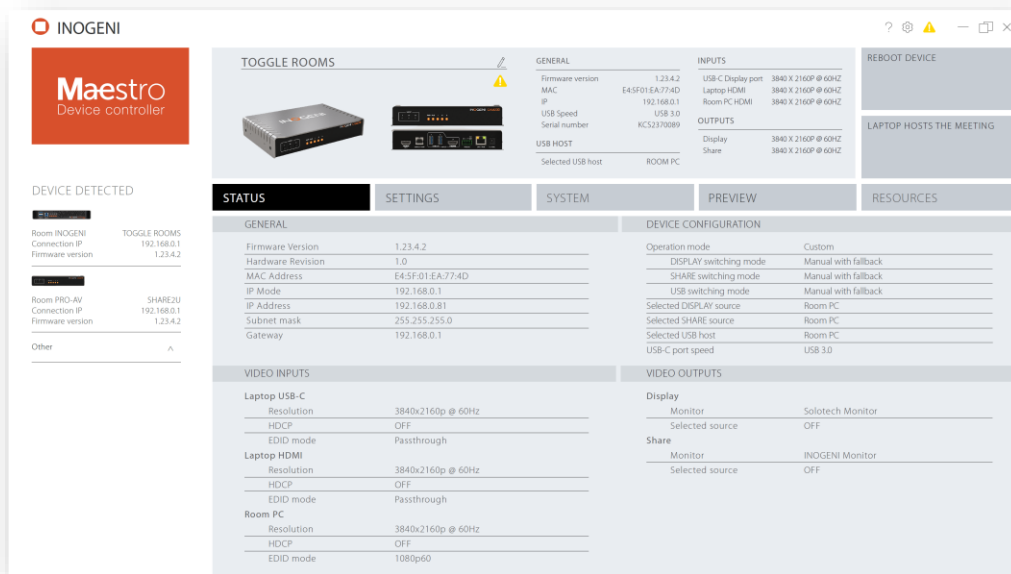


Figure 4: INOGENI Maestro application preview

## MECHANICAL SPECIFICATION

You can find the mechanical specification of the device. All dimensions are in **mm [in]**.

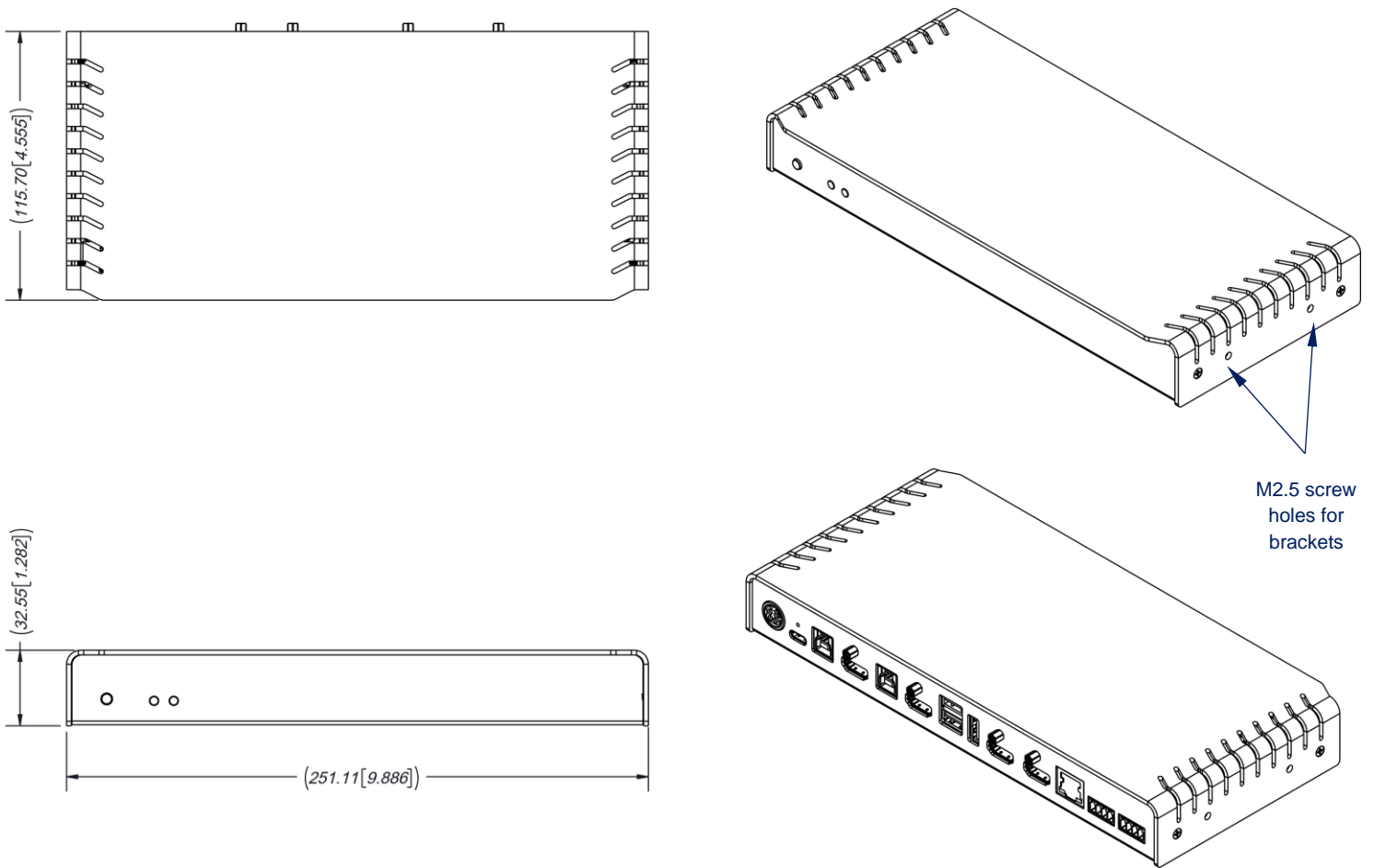


Figure 5: Top plate dimensions

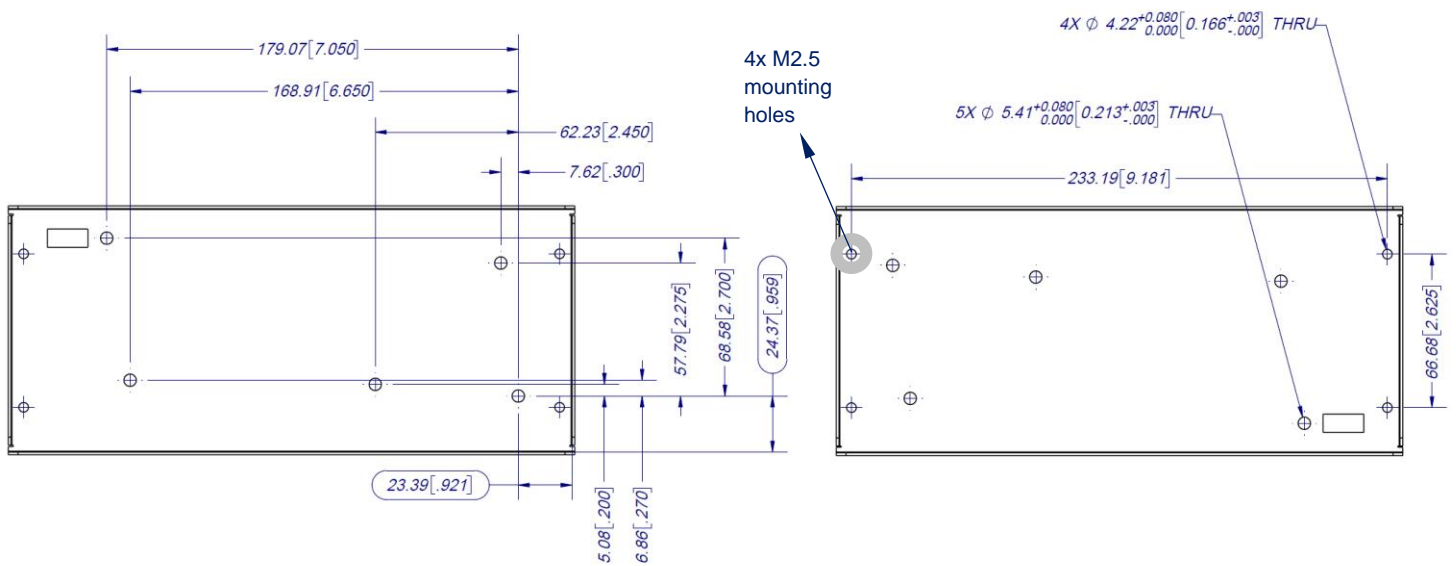


Figure 6: Bottom plate dimensions and holes positions

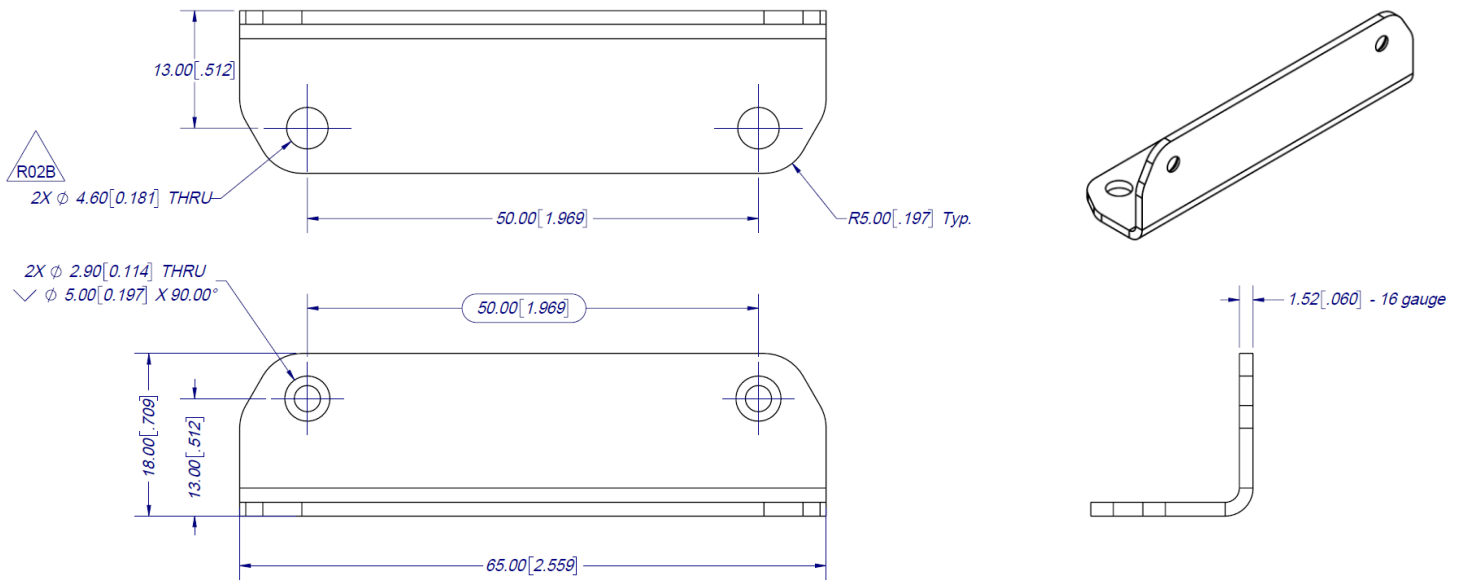


Figure 7: Bracket dimensions

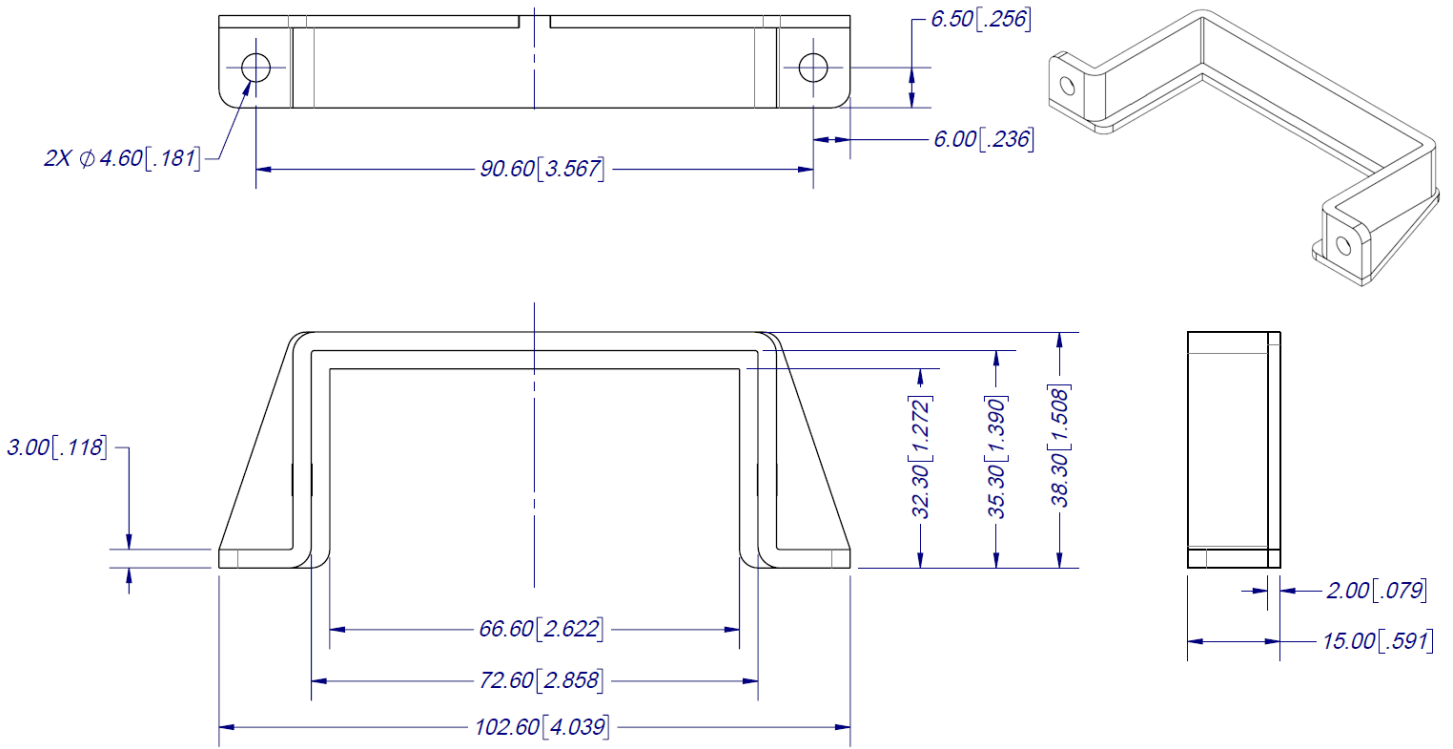


Figure 8: Power supply bracket dimensions

## DIP SWITCHES

Here you can find the behavior of the DIP switches located at the back of the unit.

Switch	Position	Description
<b>SW1</b>	OFF	For future use.
	ON	
<b>SW2</b>	OFF	For future use.
	ON	
<b>SW3</b>	OFF	For future use.
	ON	
<b>SW4</b>	OFF	For future use.
	ON	
<b>SW5</b>	OFF	Reserved.
	ON	
<b>SW6</b>	OFF	Disable 5V on terminal block
	ON	Enable 5V on terminal block. This switch must be set to power up the connected remote.

## TROUBLESHOOTING SECTION

Here is the troubleshooting section for the device.

Problem	Resolution
<b>My laptop is not charging using my USB-C cable.</b>	Check if the cable is rated to support USB-C power delivery. Also check if the cable used is among the ones that we already support. Visit <a href="https://inogeni.com/product/toggle-rooms/">https://inogeni.com/product/toggle-rooms/</a> for the complete list.
<b>The device does not automatically switch USB host and HDMI source.</b>	By default, the device is operating in “RoomPC with BYOD / content sharing” mode to avoid disruption of a current video meeting. See “Operation mode” API to properly set the operation you need.

Engineered by video professionals, for video professionals, it is your most compatible USB 3.0 device. INOGENI expertise at your fingertips:


- **Expert Technical Support team** at [support@inogeni.com](mailto:support@inogeni.com) for immediate help or if you have any technical question about our products.
- Extensive **Knowledge Base** to learn from other customers' experiences.

© Copyright 2024 by INOGENI INC. All Rights Reserved.


INOGENI name and logo are trademarks or registered trademarks of INOGENI. Use of this product is subject to the terms and conditions of the license and limited warranty in effect at the time of purchase. Product specifications can change without notice.


INOGENI, Inc.  
1045 Avenue Wilfrid-Pelletier  
Suite 101  
Québec, QC, Canada, G1W0C6  
(418) 651-3383


## CERTIFICATIONS

 **FCC Radio Frequency Interference Statement Warning**  
This device complies with Part 15 of the FCC rules. Operation is subject to the following two conditions:  
(1) this device may not cause harmful interference, and  
(2) this device must accept any interference received including interference that may cause undesired operation.


**IC Statement**  
This Class A digital apparatus complies with Canadian CAN ICES-3(A)/NMB-3(A).

 **CE Statement**  
We, INOGENI Inc., declare under our sole responsibility that the Toggle Rooms, to which this declaration relates, is in conformity with European Standards EN 55032, EN 55035, and RoHS Directive 2011/65/EU + 2015/863/EU.

 **UKCA Statement**  
This device is compliant with the Electromagnetic Compatibility Regulations 2016 No. 1091 as part of the requirements leading to the UKCA marking.

 **WEEE Statement**  
The European Union has established regulations for the collection and recycling of all waste electrical and electronic equipment (WEEE). Implementation of WEEE regulations may vary slightly by individual EU member states. Please check with your local and state government guidelines for safe disposal and recycling or contact your national WEEE recycling agency for more information.

 **RCM Statement**  
This device is compliant with Regulator Compliance Mark (RCM) certification.

 **NOM Statement**  
This device is compliant with the NOM-019 standard.