# INOGENI



# INOGENI U-CAM

## User guide

Version 1.4

October 25, 2022

## VERSION HISTORY

| Version | Date | Description |
|---------|------|-------------|
| 1.0 | April 19, 2022 | First release. |
| 1.1 | May 27, 2022 | Second release.<br>Adding RS232 API.<br>Adding REST API. |
| 1.2 | July 12, 2022 | Update image to support new hardware revision.<br>Change REST API functions to have same name as RS232 API.<br>Add http to https redirection for webpage. |
| 1.3 | September 30, 2022 | Added audio input switching. |
| 1.4 | October 25, 2022 | Modified "disableSerialInterface" function description.<br>Adding new functions over LAN interface. |

## CONTENTS

Here is a typical connection diagram used for the U-CAM device in a videoconferencing setup.
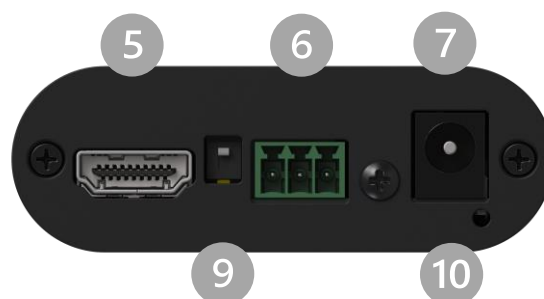


INOGENI U-CAM
## USB Camera + USB Micro to HDMI 4K

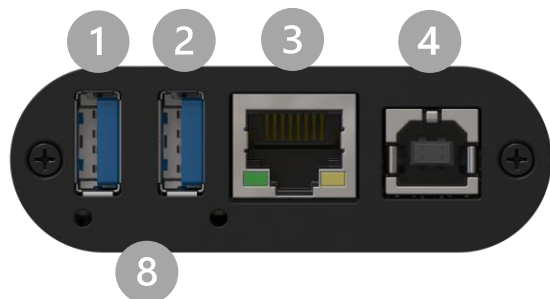Here are the devices interfaces.



1. USB input #1
2. USB input #2
3. LAN port
4. USB 2.0 output
5. HDMI output
6. RS232 port
7. +12V power input
8. USB device detection leds
9. Factory upgrade switch
10. System activity led

## LEDS BEHAVIOR

Here are the leds behavior:

| USB input | |
| --- | --- |
| **OFF** | No USB camera/device connected. |
| **SOLID** | USB camera/device connected. |
| **System activity led** | |
| **OFF** | No power present on board. |
| **BLINK** | System firmware running correctly. |

## SPECIFICATIONS

Here is the complete specification.

| Physical details | |
| --- | --- |
| **Dimensions (W x L x H)** | 70 x 83 x 23 mm |
| **Power supply** | 12V |
| **Power consumption** | Up to 1.2A |
| **Weight** | 136 g |
| **Package content** | 1 x USB 2.0 Type-B to Type-A cable. <br> 1 x terminal block connection. <br> 1 x 12V power supply. |
| **Operating temperature** | 0° to 45° C (32° to 113° F) |
| **Storage temperature** | -40° to 105° C (-40° to 221° F) |
| **Origin** | Canada |
| **Warranty** | 2 years |

| USB inputs | |
| --- | --- |
| **2x USB 3.0 inputs** | Capture video and audio from USB 3.0 and 2.0 cameras <br> Capture is done from a single camera at a time |
| **Video capabilities** | MJPEG: Up to 1080p30 using USB 2.0/3.0 <br> YUYV: Up to 1080p60 using USB 3.0, 1080p10 using USB 2.0 |
| **Audio capabilities** | Device will capture embedded audio from USB camera or external USB device and will output through HDMI. |

| HDMI output | |
| --- | --- |
| **Resolution** | 3840x2160p23.98/24/25/29.97/30 fps, 1080p50/60 fps, 720p50/60 fps |
| **Connector** | HDMI |

| USB output | |
| --- | --- |
| **USB-B connector** | Device will expose a UVC interface over USB 2.0 up to 720p30 MJPEG |

| Compatibility | |
| --- | --- |
| **Operating system** | NO driver installation necessary <br> Windows 7 and above (32/64-bit) <br> macOS 10.10 and above, |
| **Cameras Supported** | Cameras (or video Source) with an USB output |
| **Software Compatibility** | UVC-compliant. Runs with all software compatible to DirectShow/MediaFoundation, V4L2, QuickTime and AVFoundation. |

Here is the complete list of commands provided through the serial connection. Pinout is indicated on the enclosure.

Typically, commands will return ACK in case of success and NACK in case of failure.

Note that if serial interface was disabled using REST API, commands will not be parsed, and no response will be provided.

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

| Command | Argument(s) | Description |
|---|---|---|
| **HELP** | None | Return command list with description |
| **RSTR** | None | Restore default settings |
| **IP** | None | Return IP address |
| **VERSION** | None | Return firmware version |
| **QUIT** | None | Restart application |
| **STATUS** | None | Return device, video/audio inputs and HDMI output status |
| **PAN** | 1 argument (integer)<br>The sign specifies the direction.<br>We multiply the argument by the camera smallest step, and if the speed is too fast, we go as fast as the camera allow. We recommend using values between -10 and 10. | Relative pan |
| **TILT** | 1 argument (integer)<br>The sign specifies the direction.<br>We multiply the argument by the camera smallest step, and if the speed is too fast, we go as fast as the camera allow. We recommend using values between -10 and 10. | Relative tilt |
| **ZOOM** | 1 argument (integer)<br>The sign specifies the direction.<br>We multiply the argument by the camera smallest step, and if the speed is too fast, we go as fast as the camera allow. We recommend using values between -10 and 10. | Relative zoom |
| **SETHDMI** | 1 argument (integer)<br>0 => 1080P60<br>1 => 1080P50<br>2 => 720P60<br>3 => 720P50<br>4 => 4K24<br>5 => 4K25<br>6 => 4K30 | Set HDMI output mode |
| **SETVIDEOFORMAT** | 1 argument (integer)<br>0 => 1st preferred format<br>1 => 2nd preferred format<br>2 => 3rd preferred format<br>3 => 4th preferred format | Set USB video input format |
| **SETAUDIOINPUT** | 1 argument (integer)<br>The argument specifies the index of the audio input. To see available inputs, use STATUS command. | Set audio input |

## LAN COMMUNICATION PROTOCOL

You can access the device settings through its LAN interface. The LAN interface use DHCP. You can obtain the IP from the Inogeni Control App or from the serial port IP command.

## CDC-NCM COMMUNICATION PROTOCOL

The device can also be controlled through CDC-NCM interface exposed on the USB2.0 device port.

This interface has the same functions as the LAN interface, except the requests are done through USB in order to ease configuration.

CDC-NCM IP address: 169.254.10.10

## WEBPAGE

Here is the webpage that can be used to configure and upgrade the device.

The username is "**admin**" and the default password is "**admin**".
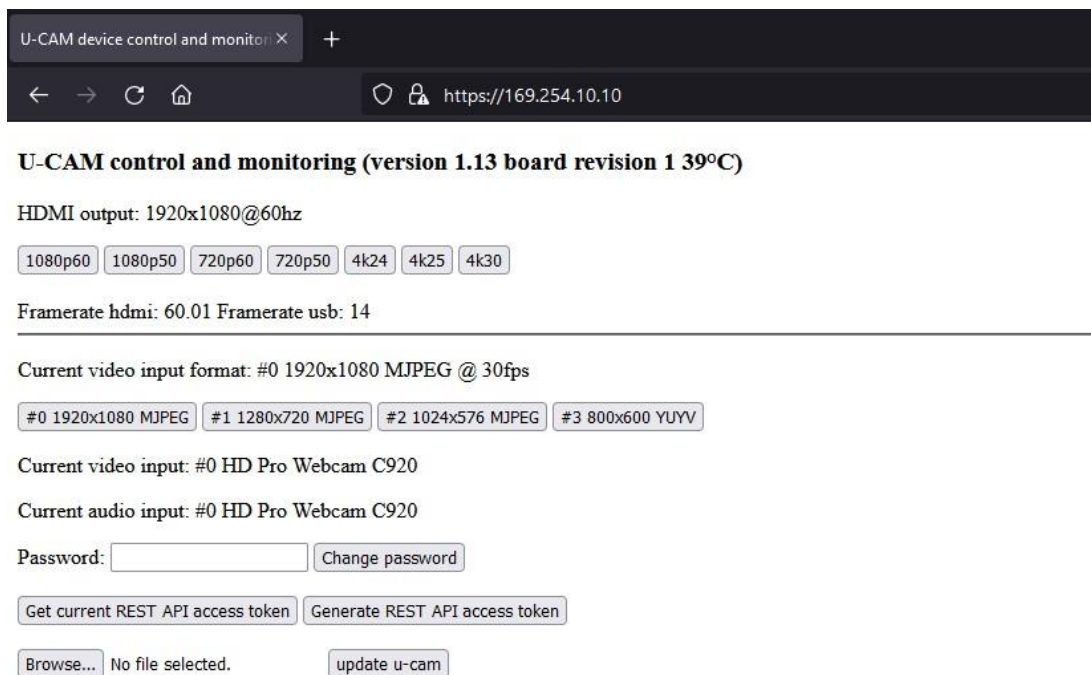


The first time you access the webpage, your web browser is likely to complain that the connection is insecure. The reason for this is because we are using self-signed HTTPS certificate, because certificate providers will not provide certificates for address that are not globally accessible.

The webpage can set HDMI resolution, USB video input format, webpage password, or the REST API access token. Please note that in the case of the REST API token, we can only ask for the device to generate a new randomly generated token. It can also be used to upgrade the device firmware.

The REST API need to have Autorization: Bearer <REST API access token> in the HTTP header. The response will be JSON formatted with a "message" field containing a JSON string explaining the cause of the error if any. Note that we are using self-signed certificate.

Here is the complete list of commands supported through the REST API

| Command URL | Body arguments | Return code | Return body |
|---|---|---|---|
| **GET**<br>**https://<IP>/api/v1/rstr** | | 200 => success<br>401 => authorization error | JSON object with multiple fields |
| **GET**<br>**https://<IP>/api/v1/version** | | 200 => success<br>401 => authorization error | JSON object with multiple fields |
| **GET**<br>**https://<IP>/api/v1/status** | | 200 => success<br>401 => authorization error | JSON object with multiple fields |
| **POST**<br>**https://<IP>/api/v1/pan** | <Integer><br>The sign specifies the direction.<br>We multiply the argument by the camera smallest step, and if the speed is too fast, we go as fast as the camera allow. We recommend using values between -10 and 10. | 200 => success<br>400 => error<br>401 => authorization error | JSON object with message field explaining error if any |
| **POST**<br>**https://<IP>/api/v1/tilt** | <Integer><br>The sign specifies the direction.<br>We multiply the argument by the camera smallest step, and if the speed is too fast, we go as fast as the camera allow. We recommend using values between -10 and 10. | 200 => success<br>400 => error<br>401 => authorization error | JSON object with message field explaining error if any |
| **POST**<br>**https://<IP>/api/v1/zoom** | <Integer><br>The sign specifies the direction.<br>We multiply the argument by the camera smallest step, and if the speed is too fast, we go as fast as the camera allow. We recommend using values between -10 and 10. | 200 => success<br>400 => error<br>401 => authorization error | JSON object with message field explaining error if any |
| **POST**<br>**https://<IP>/api/v1/setHdmi** | mode=<hdmiModeIndex><br>0 => 1080P60<br>1 => 1080P50<br>2 => 720P60<br>3 => 720P50<br>4 => 4K24<br>5 => 4K25<br>6 => 4K30 | 200 => success<br>400 => error<br>401 => authorization error | JSON object with message field explaining error if any |
| **POST**<br>**https://<IP>/api/v1/setVideoFormat** | format=<formatIndex><br>0 => 1st preferred format<br>1 => 2nd preferred format<br>2 => 3rd preferred format | 200 => success<br>400 => error<br>401 => authorization error | JSON object with message field |

| | 3 => 4th preferred format | | explaining error if any |
|---|---|---|---|
| **POST**<br>**https://\<IP>/api/v1/setAudioInput** | input=\<Integer><br>The argument specifies the index of the audio input. To see available inputs, use "status" command. | 200 => success<br>400=> error | JSON object with message field explaining error if any |
| **POST**<br>**https://\<IP>/api/v1/**<br>**disableSerialInterface** | \<Integer><br>If integer is 1, will disable serial interface, otherwise will enable it.<br>This function must be used when using the "serialRead" and "serialWrite" commands. | 200 => success<br>400 => error<br>401 => authorization error | JSON object with message field explaining error if any |
| **GET**<br>**https://\<IP>/api/v1/serialRead** | | 200 => success<br>400 => error<br>401 => authorization error | JSON object with message field containing characters read from serial port |
| **POST**<br>**https://\<IP>/api/v1/serialWrite** | \<Content to write> | 200 => success<br>400 => error<br>401 => authorization error | JSON object with message field explaining error if any |

## INOGENI CONTROL APP

You can use our Control App to monitor firmware information, upgrade and configure your unit.

> ⚠️ **NOTE:** You need to use the USB-B to USB-A cable provided with the box for the Control App to detect the unit.

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 for immediate help or if you have any technical question about our products.
- Extensive **Knowledge Base** to learn from other customers experiences.

INOGENI, Inc.
979 de Bourgogne avenue, suite 530
Québec
G1W 2L4 (QC) Canada