ATLONA
a PANDUIT company

OMEGA™

HDBaseT Transmitter with USB-C and HDMI Inputs plus USB Hub

Application Programming Interface

AT-OME-SW21-TX | Atlona APIs **Presentation**

Version Information

Version	Release Date	Notes
1	May 2025	Factory release

Information

Telnet port: 23
SSH port: 22
WS: 80
WSS: 443

Websocket address = ws://<IP>/ws
Secure Websocket address = wss://<IP>/ws

RS-232 settings: 115200, N, 8, 1 (Default)

This unit also supports TCP Proxy for external device control to the local RS-232 port with the following port assignments:

TCP Port	Port Description
9001	HDBaseT OUT
9002	RS-232 port

NOTE: This product does not support TCP Proxy port 9000.

Commands

The following is a list of JSON commands for the AT-OME-SW21-TX. Commands are case-sensitive. Do not change spacing or lettering. Each command is terminated with a carriage return. If the command fails or is entered incorrectly, then the feedback is “Unknown command”.

IMPORTANT: Each command must be in JSON RPC 2.0 structure: {"jsonrpc":"2.0","id":}. Id is optional but is recommended, as it will allow the difference between commands to be seen because it gives the ID in the response.

AV Commands

Request Object	Description
VideoSwitch.Get	Displays the current I/O settings.
VideoAutoSw.Set	Sets auto-switching to on/off.
VideoSwitch.Set	Switches input to output.
VideoInputStatus.Get	Gets the status of the input.
AudioInputFormat.Get	Gets input audio format information
VideoOutputStatus.Get	Gets the status of the output
AudioOutFormat.Get	Gets the audio format information of output
VideoHDMIOut5V.Set	Enables/disables +5V when no output signal is present
VideoHDMIOut5V.Get	Gets the status of HDMI output +5V when no output signal is present
VideoHDBTStatus.Get	Gets the HDBaseT status
AudioOutMute.Set	Sets the audio out's mute status.
AudioOutMute.Get	Gets the audio out's mute status.
VideoScaler.Set	Enables/disables the scaler on the HDMI output.
VideoScaler.Get	Gets the status of the scaler on the HDMI output.

HDCP Commands

Request Object	Description
HdcpCompliant.Set	Sets HDCP compliance for the inputs.
HdcpCompliant.Get	Gets the HDCP status of inputs.

EDID Commands

Request Object	Description
EdidInput.Set	Sets the input EDID.
EDIDInput.Get	Gets the inputs' EDID status.

USB Commands

Request Object	Description
UsbSwitch.Get	Displays the current USB switching settings.
UsbSwitchMode.Set	Sets USB Host switching mode.
UsbSwitchHost.Set	Sets which USB Host is switched to USB device on manual mode.
UsbVbus.Set	To "bounce" Vbus on USB device hub when host is switched.

Display Commands

Request Object	Description
DisplayCtrl.Get	Gets the display control mode.
DisplayCtrlMode.Set	Sets the control method of the display to CEC, RS-232 or IP.
DisplayCtrlOperation.Set	Controls the display.
DisplayPowerOnAuto.Set	Enable to set the display to power off when the power settings are met. The display will automatically turn on as soon as a signal is received and all timers have expired.
DisplayCtrlDelay.Set	Sets the delay for the unit to control the display power on/off.
DisplayDisplayWarm.Set	Sets the time between when the projector lamp has been turned on to when it can receive new commands.
DisplayCtrlLampCoolDown.Set	Sets the time between when projector lamp has been turned off to when it can receive new commands.
DisplayCtrlIP.Set	Sets the IP parameters for the unit to send commands to the display.
DisplayCtrlRs232.Set	Sets the RS-232 parameters for the unit to send commands to the display.
DisplayCtrlCmd.Set	Sets the command string to send to the display.

Occupany Sensor Commands

Request Object	Description
OccSensor.Get	Gets the status of the Occupancy Sensor.
OccSensorAdd	Adds the OCS-900 as a sensor. Up to 2 AT-OCS-900Ns can be added.
OccSensorDelete	Removes the AT-OCS-900N as a sensor.
OccSensorIdentify	The LEDs of the AT-OCS-900N will blink to help identify a specific unit when there are more than one.
OccSensorSettings	Enables/Disables the AT-OCS-900N will send auto on/off commands.

Network Commands

Request Object	Description
Network.Get	Gets the current network settings.
Network.Set	Sets the network settings for the switcher.
NetworkHostname.Set	Sets the unit's host name.
NetworkHostname.Get	Gets the unit's host name.
NetworkCtlProtocolsEnable.Set	Sets the network control protocol was enabled or disabled.
NetworkCtlProtocols.Get	Displays the network control protocol.

System Commands

Request Object	Description
Platform.FactoryReset	Factory resets the unit.
Platform.Reboot	Reboots the unit.
System.Get	Displays the current system's information.
SystemFrontPanel.Set	Enables/disables the front panel buttons to help identify the device.
SystemHdbt.Get	Get the System HDBaseT Info.
SystemHdbtStart.Set	Set the System HDBaseT information to start or stop.

VideoSwitch.Get

Displays the current I/O settings.

Structure
`{"jsonrpc": "2.0","id": "X","method": "VideoSwitch.Get"}`

Identifier	Request Object	Parameters
X	id	**Default:** VideoSwitchGetResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands

Example

```
{
"jsonrpc": "2.0",
"id": "VideoSwitchGetResults",
"method": "VideoSwitch.Get"
}
```

Feedback

```
{
"id": "VideoSwitchGetResults",
"jsonrpc": "2.0",
"result": {
"autoswitch": true,
"switch": {
"in": "in1",
"out": "out1"
}
},
"switchdefault": "in1"
}
```

VideoAutoSw.Set

Sets auto-switching to on/off.

Structure
`{"jsonrpc": "2.0","id": "X","method": "VideoAutoSw.Set","params": Y}`

Identifier	Request Object	Parameters
X	id	**Default:** VideoSwitchGetResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands
Y	params	true, false true: Auto-switching enabled. false: Auto-switching disabled.

Example

```
{
"jsonrpc": "2.0",
"id": "VideoAutoSwSetResults",
"method": "VideoAutoSw.Set",
"params": true
}
```

Feedback

```
{
"id": "VideoAutoSwSetResults",
  "jsonrpc": "2.0",
"result": {
"autosw": true
}
}
```

VideoSwitch.Set

Switches input to output.

Structure
{"jsonrpc": "2.0","id": "X","method": "VideoSwitch.Set","params":{"in": "Y","out": "Z"}}

Identifier	Request Object	Parameters
X	id	**Default:** VideoSwitchSetResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands
Y	in	in1, in2 in1: USB-C IN in2: HDMI IN
Z	out	out1, out2 out1: HDMI OUT out2: HDBaseT OUT

Example


```
{
"jsonrpc": "2.0",
"id": "VideoSwitchSetResults",
"method": "VideoSwitch.Set",
"params": {
"in": "in2",
"out": "out1"
}
}
```


Feedback


```
{
"id": "VideoSwitchSetResults",
"jsonrpc": "2.0",
"result": {
"in": "in2",
"out": "out1"
}
}
```

VideoInputStatus.Get

Gets the status of the input.

Structure
{"jsonrpc": "2.0","id": "X","method": "VideoInputStatus.Get","params": "Y"}

Identifier	Request Object	Parameters
X	id	**Default:** VideoInputStatusGetResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands
Y	params	in1, in2 in1: USB-C IN in2: HDMI IN

Example

```
{
  "jsonrpc": "2.0",
  "id": "VideoInputStatusGetResults",
  "method": "VideoInputStatus.Get",
  "params": "in2"
}
```

Feedback

```
{
    "id": "VideoInputStatusGetResults",
    "result": {
    "source": in2,
    "cableconnection": "Conneted",
    "signal":"Valid",
    "videoformat":"3840x2160,60;none hdr;rgb;8bit;16:9",
    "hdcp":"hdcp2.2",
    "signaltype":"hdmi"
 }
 }
a
}
```

AudioInputFormat.Get

Gets input audio format information.

Structure
`{ "jsonrpc": "2.0","id": "X","method": "AudioInputFormat.Get","params": "Y"}`

Identifier	Request Object	Parameters
X	id	**Default:** AudioInputFormatGetResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands
Y	params	in1, in2 in1: USB-C IN in2: HDMI IN

Example

```
{
  "jsonrpc": "2.0",
  "id": "AudioInputFormatGetResults",
  "method": "AudioInputFormat.Get",
  "params": "in2"
}
```

Feedback

```
{
    "id": "AudioInputFormatGetResults",
    "result": {
    "source": "in2",
    "audioformat": "PCM;48kHZ",
   }
"jsonrpc": "2.0"
}
```

VideoOutputStatus.Get

Gets the status of the output.

Structure
{"jsonrpc": "2.0","id": "X","method": "VideoOutputStatus.Get","params": "Y"}

Identifier	Request Object	Parameters
X	id	**Default:** VideoOutputStatusGetResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands
Y	params	out1, out2 out1: HDMI OUT out2: HDBaseT OUT

Example

```
{
  "jsonrpc": "2.0",
  "id": "VideoOutputStatusGetResults",
  "method": "VideoOutputStatus.Get",
  "params": "out1"
}
```

Feedback

```
{
    "id": "VideoOutputStatusGetResults",
    "result": {
    "sink": "out1",
    "activeinput":"in1",
    "cableconnection": "Conneted",
    "signal":"Valid",
    "videoformat":"3840x2160,60;none hdr;rgb;8bit;16:9",
    "hdcp":"hdcp2.2",
    "signaltype":"hdmi"
  }
    "jsonrpc": "2.0"
}
```

AudioOutFormat.Get

Gets the audio format information of output.

Structure
{"jsonrpc": "2.0","id": "X","method": "AudioOutFormat.Get","params": "Y"}

Identifier	Request Object	Parameters
X	id	**Default:** VideoOutputStatusGetResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands
Y	params	out1, out2 out1: HDMI OUT out2: HDBaseT OUT

Example

```
{
  "jsonrpc": "2.0",
  "id": "AudioOutFormatGetResults",
  "method": "AudioOutFormat.Get",
  "params": "out1"
}
```

Feedback

```
{
  "id": "AudioOutFormatGetResults",
  "jsonrpc": "2.0",
  "result": {
    "audioformat": "48kHZ",
    "sink": "out1"
  }
}
```

VideoHDMIOut5V.Set

Enables/disables +5V when no output signal is present.

Structure
`{"jsonrpc": "2.0","id": "X","method": "VideoHDMIOut5V.Set","params":Y}`

Identifier	Request Object	Parameters
X	id	**Default:** VideoHDMIOut5VSetResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands
Y	params	true, false

Example

```
{
  "jsonrpc": "2.0",
  "id": "VideoHDMIOut5VSetResults",
  "method": "VideoHDMIOut5V.Set",
  "params": false
}
```

Feedback

```
{
    "id": "VideoHDMIOut5VSetResults",
    "result": {
    "HDMIOut5V": false
    }
    "jsonrpc": "2.0"
}
```

VideoHDMIOut5V.Get

Gets the status of HDMI output +5V when no output signal is present.

Structure
{"jsonrpc": "2.0","id": "X","method": "VideoHDMIOut5V.Get"}

Identifier	Request Object	Parameters
X	id	**Default:** VideoHDMIOut5VSetResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands

Example

```
{
  "jsonrpc": "2.0",
  "id": "VideoHDMIOut5VGetResults",
  "method": "VideoHDMIOut5V.Get"
}
```

Feedback

```
{
    "id": "VideoHDMIOut5VGetResults",
    "result": {
    "HDMIOut5V": true
    }
"jsonrpc": "2.0"
}
```

VideoHDBTStatus.Get

Gets the HDBT status.

Structure
{"jsonrpc": "2.0","id": "X","method": "VideoHDBTStatus.Get"}

Identifier	Request Object	Parameters
X	id	**Default:** VideoHDMIOut5VSetResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands

Example

```
{
  "jsonrpc": "2.0",
  "id": "VideoHDBTStatusGetResults",
  "method": "VideoHDBTStatus.Get"
}
}
```

Feedback

```
{
    "id": "X",
    "jsonrpc": "2.0",
    "result": {
        "link": "up",
        "remote ip": "",
        "remote model": "AT-OME-EX-RX"
    }
}
```

AudioOutMute.Set

Sets the audio out's mute status.

Structure
{"jsonrpc": "2.0","id": "X","method": "AudioOutMute.Set","params":Y}

Identifier	Request Object	Parameters
X	id	**Default:** AudioOutMuteSetResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands
Y	params	true, false

Example

```
{
  "jsonrpc": "2.0",
  "id": "AudioOutMuteSetResults",
  "method": "AudioOutMute.Set",
  "params": false
}
```

Feedback

```
{
    "id": "AudioOutMuteSetResults",
    "result": {
    "AudioOutMute": false
    }
    "jsonrpc": "2.0"
}
```

AudioOutMute.Get

Gets the audio out's mute status.

Structure
{"jsonrpc": "2.0","id": "X","method": "AudioOutMute.Get"}

Identifier	Request Object	Parameters
X	id	**Default:** AudioOutMuteGetResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands

Example

```
{
  "jsonrpc": "2.0",
  "id": "AudioOutMuteGetResults",
  "method": "AudioOutMute.Get"
}
```

Feedback

```
{
    "id": "AudioOutMuteGetResults",
    "result": {
    "AudioOutMute": false
    }
"jsonrpc": "2.0"
}
```

VideoScaler.Set

Enables/disables the scaler on the HDMI output.

Structure
{"jsonrpc": "2.0","id": "X","method": "VideoScaler.Set","params":Y}

Identifier	Request Object	Parameters
X	id	**Default:** VideoScalerSetResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands
Y	params	true, false

Example

```
{
  "jsonrpc": "2.0",
  "id": "VideoScalerSetResults",
  "method": "VideoScaler.Set",
  "params": true
}
```

Feedback

```
{
    "id": "VideoScalerSetResults",
    "result": {
    "Scaler": true
    }
    "jsonrpc": "2.0"
}
```

VideoScaler.Get

Gets the status of the scaler on the HDMI output.

Structure
{"jsonrpc": "2.0","id": "X","method": "VideoScaler.Get"}

Identifier	Request Object	Parameters
X	id	**Default:** VideoScalerGetResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands

Example

```
{
  "jsonrpc": "2.0",
  "id": "VideoScalerGetResults",
  "method": "VideoScaler.Get"
}
```

Feedback

```
{
    "id": "VideoScalerGetResults",
    "result": {
    "Scaler": true
    }
"jsonrpc": "2.0"
}
```

HdcpCompliant.Set

Sets HDCP compliance for the inputs.

Structure
`{"jsonrpc": "2.0","id": "X","method": "HdcpCompliant.Set","params": {"source": "Y","hdcpCompliant": S}}`

Identifier	Request Object	Parameters
X	id	**Default:** HdcpCompliantSetResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands
Y	source	in1, in2 in1: USB-C IN in2: HDMI IN
S	hdcpCompliant	true, false

Example

```
{
  "jsonrpc": "2.0",
  "id": "HdcpCompliantSetResults",
  "method": "HdcpCompliant.Set",
  "params": {
    "source": "in1",
    "hdcpCompliant": true
  }
}
```

Feedback

```
{
    "id": "HdcpCompliantSetResults",
    "result": {
        "source": "in1",
        "hdcpCompliant": true
        },
    "jsonrpc": "2.0"
}
```

HdcpCompliant.Get

Gets the HDCP status of the inputs.

Structure
{"jsonrpc": "2.0","id": "X","method": "HdcpCompliant.Get"}

Identifier	Request Object	Parameters
X	id	**Default:** HdcpCompliantGetResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands

Example

```
{
  "jsonrpc": "2.0",
  "id": "HdcpCompliantGetResults",
  "method": "HdcpCompliant.Get"
}
```

Feedback

```
{
 "id": "User1",
 "result": {
  "hdcpCompliants": [
  {
   "source": "in1",
   "hdcpCompliant": true
  },
 {
   "source": "in2",
   "hdcpCompliant": true
  }
  ]
 },
 "jsonrpc": "2.0"
}
```

EdidInput.Set

Sets input EDID.

Structure
`{"jsonrpc": "2.0","id": "X","method": "EdidInput.Set","params": {"source": "Y","edidmode": Z}}`

Identifier	Request Object	Parameters
X	id	**Default:** EdidInputSetResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands
Y	source	in1, in2 in1: USB-C IN in2: HDMI IN
Z	edidmode	1-10, 98, 99 1: Copy from the hdmi output connected display (Default) 2: Fixed 3840x2160@60 Hz 2.0CH PCM Audio with HDR; 3: Fixed 3840x2160@30 Hz 2.0CH PCM Audio with SDR; 4: Fixed 1920x1080P@60Hz 2.0CH PCM Audio with SDR; 5: Fixed 1920x1080P@30Hz 2.0CH PCM Audio with SDR; 6: Fixed 1280x720P@60Hz 2.0CH PCM Audio with SDR; 7: Fixed 1280x800 2.0CH PCM Audio with SDR; 8: Fixed 1920x1200 2.0CH PCM Audio with SDR; 9: Fixed 3440x1440@50 2.0CH PCM Audio with SDR 10: Fixed 2560x1080@60 2.0CH PCM Audio with SDR 99: mem1 98: mem2

Example

```
{
  "jsonrpc": "2.0",
  "id": "EdidInputSetResults",
  "method": "EdidInput.Set",
  "params": {
    "source": "in1",
    "edidmode": 2
  }
}
```

Feedback

```
{
    "id": "EdidInputSetResults",
    "result":  {
    "source": "in1",
    "edidmode": 2
    },
    "jsonrpc": "2.0"
}
```

EDIDInput.Get

Gets the inputs' EDID status.

Structure
{"jsonrpc": "2.0","id": "X","method": "EDIDInput.Get"}

Identifier	Request Object	Parameters
X	id	**Default:** EDIDInputGetResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands

Example

```
{
  "jsonrpc": "2.0",
  "id": "EDIDInputGetResults",
  "method": "EDIDInput.Get"
}
```

Feedback

```
{
    "id": "EDIDInputGetResults",
    "result": {
    "mem1":true,
"mem1EDID ": "xxxx",
"mem2":true,
    "mem2EDID ": "xxxx",
    "edidsettings":[
        {
            "source": "in1",
            "edidmode": 2
        },
        {
            "source": "in2",
            "edidmode": 2
        }
        ]
   },
    "jsonrpc": "2.0"
}
```

UsbSwitch.Get

Displays the current USB switching settings.

Structure
{"jsonrpc": "2.0","id": "X","method": "UsbSwitch.Get"}

Identifier	Request Object	Parameters
X	id	**Default:** UsbSwitchGetResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands

Example

```
{
  "jsonrpc": "2.0",
  "id": "UsbSwitchGetResults",
  "method": "UsbSwitch.Get"
}
```

Feedback

```
{
  "id": "UsbSwitchGetResults",
  "jsonrpc": "2.0",
  "result": {
    "followvideousbmap": [
      {
        "maptohost": "host1",
        "source": "in1"
      },
      {
        "maptohost": "host2",
        "source": "in2"
      }
    ],
    "usbSwitchHost": "host1",
    "usbVbus": "true",
    "usbswitchmode": "follow video"
  }
}
```

UsbSwitchMode.Set

Sets USB Host switching mode. Default is follow video.

Structure
`{"jsonrpc": "2.0","id": "X","method": "UsbSwitchMode.Set","params": "Y"}`

Identifier	Request Object	Parameters
X	id	**Default:** UsbSwitchModeSetResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands
Y	params	follow video, manual

Example

```
{
  "jsonrpc": "2.0",
  "id": "UsbSwitchModeSetResults",
  "method": "UsbSwitchMode.Set",
  "params": "follow video"
}
```

Feedback

```
{
    "id": "UsbSwitchModeSetResults",
    "result": {
     "usbswitchmode": "follow video"
},
    "jsonrpc": "2.0"
}
```

UsbSwitchHost.Set

Sets which USB Host is switched to USB device on manual mode.

Structure
{"jsonrpc": "2.0","id": "X","method": "UsbSwitchMode.Set","params": "Y"}

Identifier	Request Object	Parameters
X	id	**Default:** UsbSwitchHostSetResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands
Y	params	host1, host2, host3

Example

```
{
  "jsonrpc": "2.0",
  "id": "UsbSwitchHostSetResults",
  "method": "UsbSwitchHost.Set",
  "params": "host1"
}
```

Feedback

```
{
    "id": "UsbSwitchHostSetResults",
    "result": {
     "toHost": "host1"
    },
    "jsonrpc": "2.0"
}
```

UsbVbus.Set

To "bounce" Vbus on USB device hub when host is switched. Default is set to true.

Structure
`{"jsonrpc": "2.0","id": "X","method": "UsbVbus.Set","params": "Y"}`

Identifier	Request Object	Parameters
X	id	**Default:** UsbVbusSetResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands
Y	params	true, false

Example

```
{
  "jsonrpc": "2.0",
  "id": "UsbVbusSetResults",
  "method": "UsbVbus.Set",
  "params": "true"
}
```

Feedback

```
{
    "id": "UsbVbusSetResults",
    "result":{
         "usbVbus": "true"
    },
    "jsonrpc": "2.0"
}
```

DisplayCtrl.Get

Gets the display control mode.

Structure
{"jsonrpc": "2.0","id": "X","method": "DisplayCtrl.Get"}

Identifier	Request Object	Parameters
X	id	**Default:** DisplayCtrlGetResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands

Example

```
{
   "jsonrpc": "2.0",
   "id": "X",
   "method": "DisplayCtrl.Get"
}
```

Feedback

```
{
    "id": "X",
    "jsonrpc": "2.0",
    "result": {
        "DisplayControl/AutocontrolDelay":
120,
        "DisplayControl/
AutocontrolEnable": "VideoPresence",
        "DisplayControl/DisplayWarmTime":
"10",
        "DisplayControl/LampCoolDownTime":
"10",
        "DisplayControl/RS232": [
            {
                "Baudrate": "115200",
                "Databits": "8",
                "Parity": "N",
                "Stopbits": "1",
                "port": "uart1"
            },
            {
                "Baudrate": "115200",
                "Databits": "8",
                "Parity": "N",
                "Stopbits": "1",
                "port": "uart2"
            }
        ],
        "DisplayControl/command/
RepeatStatus": false,
        "DisplayControl/command/
RepeatTimes": "2",
        "DisplayControl/command/mute": "\\
x50\\x44\\x43",
        "DisplayControl/command/off": "\\
x40\\x36",
        "DisplayControl/command/on": "\\
x40\\x04",
        "DisplayControl/command/unmute":
"\\x50\\x44\\x43",
        "DisplayControl/command/
volumedown": "\\x50\\x44\\x42",
        "DisplayControl/command/volumeup":
"\\x50\\x44\\x41",
        "DisplayControl/controlmethod":
"CecAPI",
        "DisplayControl/ip/address": "",
        "DisplayControl/ip/mode": "Non-
Login",
        "DisplayControl/ip/password": "",
        "DisplayControl/ip/port": "",
        "DisplayControl/ip/username": "",
        "DisplayControl/manufacturer": "",
        "DisplayControl/model": "",
        "DisplayControl/products": ""
    }
}
```

DisplayCtrlMode.Set

Sets the control method of the display to CEC, RS-232 or IP.

Structure
{"jsonrpc": "2.0","id": "X","method": "DisplayCtrlMode.Set","params":"Y"}

Identifier	Request Object	Parameters
X	id	**Default:** DisplayCtrlModeSetResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands
Y	params	IP, RS232, CecAPI

Example

```
{
  "jsonrpc": "2.0",
  "id": "DisplayCtrlModeSetResults",
  "method": "DisplayCtrlMode.Set",
  "params": "RS232"
}
```

Feedback

```
{
    "id": "DisplayCtrlModeSetResults",
    "result": {
    "displaycontrolmethod": "RS232"
},
    "jsonrpc": "2.0"
}
```

DisplayCtrlOperation.Set

Controls the Display.

Structure
{"jsonrpc":"2.0","id":"DisplayCtrlOperationSetResults","method":"DisplayCtrlOperation.Set","params":{"command":"Y","controlmode":"M"}}

Identifier	Request Object	Parameters
X	id	**Default:** DisplayCtrlOperationSetResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands
Y	source	poweron, poweroff, volumeup, volumedown, mute, unmute
Z	controlmode	IP, RS232, CecAPI

Example

```
{
  "jsonrpc": "2.0",
  "id": "DisplayCtrlOperationSetResults",
  "method": "DisplayCtrlOperation.Set",
  "params": {
    "command": "poweron",
    "controlmode": "CecAPI"
  }
}
```

Feedback

```
{
    "id": "DisplayCtrlOperationSetResults",
    "result": {
    "DisplayCtrlOperation":true
},
    "jsonrpc": "2.0"
}
```

DisplayPowerOnAuto.Set

Enable to set the display to power off when the power settings are met. The display will automatically turn on as soon as a signal is received and all timers have expired.

Modes	Description
Active Video Presence	Device will send the power off command to the display if no active source is detected on the input, and power on command when an active source is detected. Power timers will be followed.
Active Video Presence w/Occupancy Sensor*	When the occupancy sensor (AT-OCS-900N) is triggered and source signal is active or inactive, it will send the on or off command based on physical and signal presence.
Occupancy Sensor only	Power on and off commands will be sent based on the OCS-900N sensor status. The sensor must be connected to the same network as the OME-SW21-TX and connected on the I/O page.
Disabled	No display control

Structure
`{"jsonrpc": "2.0","id": "X","method": "DisplayPowerOnAuto.Set","params":"Y" }`

Identifier	Request Object	Parameters
X	id	**Default:** DisplayCtrlModeSetResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands
Y	params	VideoPresence, Video/OccSensor, OccSensor, Disabled

Example

```
{
  "jsonrpc": "2.0",
  "id": "DisplayPowerOnAutoSetResults",
  "method": "DisplayPowerOnAuto.Set",
  "params": "Video/OccSensor"
}
```

Feedback

```
{
  "id": "DisplayPowerOnAutoSetResults",
  "jsonrpc": "2.0",
  "result": {
    "displaypowonauto": "Video/OccSensor"
  }
}
```

DisplayCtrlDelay.Set

Sets the delay for the unit to control the display power on/off. Parameter is set in seconds.

NOTE: The parameter will not be triggered unless DisplayPowerOnAuto.Set is set to true.

Structure
`{"jsonrpc": "2.0","id": "X","method": "DisplayCtrlDelay.Set","params": Y}`

Identifier	Request Object	Parameters
X	id	**Default:** DisplayCtrlDelaySetResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands
Y	params	15-900

Example

```
{
  "jsonrpc": "2.0",
  "id": "DisplayCtrlDelaySetResults",
  "method": "DisplayCtrlDelay.Set",
  "params": 120
}
```

Feedback

```
{
  "id": "DisplayCtrlDelaySetResults",
  "jsonrpc": "2.0",
  "result": {
    "displayctrldelay": 120
  }
}
```

DisplayDisplayWarm.Set

Sets the time between when the projector lamp has been turned on to when it can receive new commands. Parameter is set in seconds.

Structure
`{"jsonrpc": "2.0","id": "X","method": "DisplayDisplayWarm.Set","params": Y}`

Identifier	Request Object	Parameters
X	id	**Default:** DisplayDisplayWarmSetResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands
Y	params	10-300

Example

```
{
    "jsonrpc": "2.0",
    "id": "DisplayDisplayWarmSetResults",
    "method": "DisplayDisplayWarm.Set",
    "params":"15"
}
```

Feedback

```
{
  "id": "DisplayDisplayWarmSetResults",
  "jsonrpc": "2.0",
  "result": {
    "DisplayWarmTime": "15"
  }
}
```

DisplayCtrlLampCoolDown.Set

Sets the time between when projector lamp has been turned off to when it can receive new commands. Parameter is set in seconds.

Structure
{"jsonrpc":"2.0","id":"X","method":"DisplayCtrlLampCoolDown.Set","params":"Y"}

Identifier	Request Object	Parameters
X	id	**Default:** DisplayCtrlLampCoolDownSetResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands
Y	params	10-300

Example

```
{
  "jsonrpc": "2.0",
  "id": "DisplayCtrlLampCoolDownSetResults",
  "method": "DisplayCtrlLampCoolDown.Set",
  "params": "15"
}
```

Feedback

```
{
  "id": "DisplayCtrlLampCoolDownSetResults",
  "jsonrpc": "2.0",
  "result": {
    "LampCoolDownTime": "15"
  }
}
```

DisplayCtrlIP.Set

Sets the IP parameters for the unit to send commands to the display.

Structure
`{"jsonrpc": "2.0","id": "X","method": "DisplayCtrlIP.Set","params": {"mode":"M","ipaddress":"Y","port":Z, "username":"N","password":"N"}}`

Identifier	Request Object	Parameters
X	id	**Default:** DisplayCtrlIPSetResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands
Y	ipaddress	xxx.xxx.xxx.xxx
Z	port	xxx (up to a 5 digit port)
M	mode	Login, Non-Login
N	username /password	xx (username and password can be up to 64 characters)

Example

```
{
  "jsonrpc": "2.0",
  "id": "DisplayCtrlIPSetResults",
  "method": "DisplayCtrlIP.Set",
  "params": {
    "ipaddress": "192.168.23.189",
    "port": 23,
    "mode": "Login",
    "username": "admin",
    "password": "admin"
  }
}
```

Feedback

```
{
  "id": "DisplayCtrlIPSetResults",
  "jsonrpc": "2.0",
  "result": {
    "ipaddress": "192.168.23.189",
    "mode": "Login",
    "password": "admin",
    "port": 23,
    "username": "admin"
  }
}
```

DisplayCtrlRs232.Set

Sets the RS-232 parameters for the unit to send commands to the display.

Structure
`{"jsonrpc": "2.0","id": "X","method": "DisplayCtrlRs232.Set","params": {"port": "W","baudrate": "Y","parity": "Z", "dataBit": "M","stopBit": "N"}}`

Identifier	Request Object	Parameters
X	id	**Default:** DisplayCtrlRs232SetResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands
W	port	uart1, uart2
Y	baudrate	9600, 19200, 38400, 57600, 115200
Z	parity	N (none), O (odd), E (even)
M	dataBit	7, 8
N	stopBit	1, 2

Example

```
{
  "jsonrpc": "2.0",
  "id": "DisplayCtrlRs232SetResults",
  "method": "DisplayCtrlRs232.Set",
  "params": {
    "port": "uart1",
    "baudrate": "115200",
    "parity": "N",
    "dataBit": "8",
    "stopBit": "1"
  }
}
```

Feedback

```
{
    "id": "DisplayCtrlRs232SetResults",
    "result": true,
    "jsonrpc": "2.0"
}
```

DisplayCtrlCmd.Set

Sets the command string to send to the display.

Structure
`{"jsonrpc": "2.0","id": "X","method": "DisplayCtrlCmd.Set","params": {"cmd": "Y","data": "W"}}`

Identifier	Request Object	Parameters
X	id	**Default:** DisplayCtrlCmdSetResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands
Y	cmd	poweron, poweroff, volumeup, volumedown, mute, unmute
Z	data	xxxxxxxxxxx Command for display control

Example

```
{
  "jsonrpc": "2.0",
  "id": "DisplayCtrlCmdSetResults",
  "method": "DisplayCtrlCmd.Set",
  "params": {
    "data": "02 50 4f 4e 03",
    "cmd": "poweron"
  }
}
```

Feedback

```
{
    "id": "DisplayCtrlCmdSetResults",
    "result": true,
    "jsonrpc": "2.0"
}
```

OccSensor.Get

Gets the status of the Occupancy Sensor.

Structure
{"jsonrpc": "2.0","id": "X","method": "OccSensor.Get"}

Identifier	Request Object	Parameters
X	id	**Default:** OccSensorGetResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands

Example

```
{
  "jsonrpc": "2.0",
  "id": "OccSensorGetResults",
  "method": "OccSensor.Get"
}
```

Feedback

```
{
    "id": "OccSensorGetResults",
    "result": {
    "OccupancySensorSettings":
    {
  "OccupancySensorIP/enable/off": true,
  "OccupancySensorIP/enable/on": true
},
    "OccupancySensor/OCS-900N-NUM": "2",
    "OccupancySensor/OCS-900N":
[
    {
    "ID": "OCS4567",
    "host":"192.168.50.110",
"port": 9000
"connected":true,
"state":"vacant"
    },
    {
    "ID": "OCS4567",
    "host":"192.168.50.110",
"port": 9000
"connected":true,
"state":"vacant"
    }
]
     },
    "jsonrpc": "2.0"
}
```

OccSensorAdd

Adds the OCS-900 as a sensor. Up to 2 AT-OCS-900Ns can be added.

Structure
`{"jsonrpc": "2.0","id": "X","method": "OccSensorAdd","params":{"host":"Y","port":Z}}`

Identifier	Request Object	Parameters
X	id	**Default:** OccSensorAddResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands
Y	host	IP address or hostname
Z	port	Port

Example

```
{
  "jsonrpc": "2.0",
  "id": "OccSensorAddResults",
  "method": "OccSensorAdd",
  "params": {
    "host": "192.168.23.218",
    "port": 9000
  }
}
```

Feedback

```
{
    "id": "OccSensorAddResults",
    "result": true,
    "jsonrpc": "2.0"
}
```

OccSensorDelete

Removes the AT-OCS-900N as a sensor.

Structure
{"jsonrpc":"2.0","id": "X","method": "OccSensorDelete","params": "Y"}

Identifier	Request Object	Parameters
X	id	**Default:** OccSensorDeleteResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands
Y	params	OCS Sensor ID

Example

```
{
  "jsonrpc": "2.0",
  "id": "OccSensorDeleteResults",
  "method": "OccSensorDelete",
  "params": "OCS1234"
}
```

Feedback

```
{
    "id": "OccSensorDeleteResults",
    "result": true,
    "jsonrpc": "2.0"
}
```

OccSensorIdentify

The LEDs of the AT-OCS-900N will blink to help identify a specific unit when there are more than one.

Structure
{"jsonrpc":"2.0","id": "X","method": "OccSensorIdentify","params": "Y"}

Identifier	Request Object	Parameters
X	id	**Default:** OccSensorIdentifyResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands
Y	params	OCS Sensor ID

Example

```
{
  "jsonrpc": "2.0",
  "id": "OccSensorIdentifyResults",
  "method": "OccSensorIdentify",
  "params": "OCS1234"
}
```

Feedback

```
{
    "id": "OccSensorIdentifyResults",
    "result": true,
    "jsonrpc": "2.0"
}
```

OccSensorSettings

Enables/Disables the AT-OCS-900N will send auto on/off commands.

Structure
{"jsonrpc": "2.0","id": "X","method": "OccSensorSettings","params":{"OccupancySensorIP/enable/on":Y,"OccupancySensorIP/enable/off":X}}

Identifier	Request Object	Parameters
X	id	**Default:** OccSensorSettingsResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands
Y	params	true, false

Example

```
{
  "jsonrpc": "2.0",
  "id": "OccSensorSettingsResults",
  "method": "OccSensorSettings",
  "params": {
    "OccupancySensorIP/enable/off": true,
    "OccupancySensorIP/enable/on": false
  }
}
```

Feedback

```
{
  "id": "OccSensorSettingsResults",
  "jsonrpc": "2.0",
  "result": {
    "OccupancySensorIP/enable/off": true,
    "OccupancySensorIP/enable/on": false
  }
}
```

Network.Get

Gets the current network settings.

Structure
`{"jsonrpc": "2.0","id": "X","method": "Network.Get"}`

Identifier	Request Object	Parameters
X	id	**Default:** NetworkGetResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands

Example

```
{
  "jsonrpc": "2.0",
  "id": "NetworkGetResults",
  "method": "Network.Get"
}
```

Feedback

```
{
  "id": "NetworkGetResults",
  "jsonrpc": "2.0",
  "result": {
    "gateway": "192.168.23.1",
    "ip_mode": "dhcp",
    "ip_port": 23,
    "ipaddr": "192.168.23.214",
    "mac": "34:1b:22:80:11:44",
    "netmask": "255.255.255.0"
  }
}
```

Network.Set

Sets the network settings for the switcher.

Structure
{"jsonrpc": "2.0","id": "X","method": "Network.Set","params":{"ip_mode": "V","ipaddr": "Y","netmask": "Z", "gateway": "I","ip_port": W}}

Identifier	Request Object	Parameters
X	id	**Default:** NetworkSetResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands
V	ip_mode	static, dhcp DHCP is selected by default
Y	ipaddr	IP address (e.g. 169.254.9.9)
Z	netmask	Netmask (e.g. 255.255.0.0)
I	gateway	Gateway (e.g. 192.168.2.254)
W	ip port	Port number

Example

```
{
  "jsonrpc": "2.0",
  "id": "NetworkSetResults",
  "method": "Network.Set",
  "params": {
    "ip_mode": "static",
    "ipaddr": "192.168.23.214",
    "netmask": "255.255.255.0",
    "gateway": "192.168.23.1",
    "ip_port": 23
  }
}
```

Feedback

```
{
  "id": "NetworkSetResults",
  "jsonrpc": "2.0",
  "result": {
    "gateway": "192.168.23.1",
    "ip_mode": "static",
    "ip_port": 23,
    "ipaddr": "192.168.23.214",
    "netmask": "255.255.255.0"
  }
}
```

NetworkHostname.Set

Sets the unit's host name.

Structure
{"jsonrpc": "2.0","id": "X","method": "NetworkHostname.Set","params":"Y"}

Identifier	Request Object	Parameters
X	id	**Default:** NetworkHostnameSetResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands
Y	params	xxx (user created host name), nil Default is nil

Example

```
{
  "jsonrpc": "2.0",
  "id": "NetworkHostnameSetResults",
  "method": "NetworkHostname.Set",
  "params": "AT-OME-SW21TX"
}
```

Feedback

```
{
      "id": "NetworkHostnameSetResults",
      "jsonrpc": "2.0",
      "result": {
            "hostname": "AT-OME-SW21TX"
      }
}
```

NetworkHostname.Get

Gets the unit's host name.

Structure
{"jsonrpc": "2.0","id": "X","method": "NetworkHostname.Get"}

Identifier	Request Object	Parameters
X	id	**Default:** NetworkHostnameGetResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands

Example

```
{
  "jsonrpc": "2.0",
  "id": "NetworkHostnameGetResults",
  "method": "NetworkHostname.Get"
}
```

Feedback

```
{
  "id": "NetworkHostnameGetResults",
  "jsonrpc": "2.0",
  "result": {
    "hostname": "AT-OME-SW21TX-1"
  }
}
```

NetworkCtlProtocolsEnable.Set

Sets the network control protocol was enabled or disabled.

Structure
`{"jsonrpc": "2.0","id": "X","method": "NetworkCtlProtocolsEnable.Set","params":{"protocol":"Y","enabled": Z}}`

Identifier	Request Object	Parameters
X	id	**Default:** NetworkCtlProtocolsEnableSetResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands
Y	host	telnet, ssh
Z	port	true, false default is true

Example

```
{
  "jsonrpc": "2.0",
  "id": "NetworkCtlProtocolsEnableSetResults",
  "method": "NetworkCtlProtocolsEnable.Set",
  "params": {
    "protocol": "ssh",
    "enabled": true
  }
}
```

Feedback

```
{
  "id": "NetworkCtlProtocolsEnableSetResults",
  "jsonrpc": "2.0",
  "result": {
    "enabled": true,
    "protocol": "ssh"
  }
}
```

NetworkCtlProtocols.Get

Displays the network control protocol.

Structure
{"jsonrpc": "2.0","id": "X","method": "NetworkCtlProtocols.Get"}

Identifier	Request Object	Parameters
X	id	**Default:** NetworkCtlProtocolsGetResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands

Example

```
{
  "jsonrpc": "2.0",
  "id": "NetworkCtlProtocolsGetResults",
  "method": "NetworkCtlProtocols.Get"
}
```

Feedback

```
{
  "id": "NetworkCtlProtocolsGetResults",
  "jsonrpc": "2.0",
  "result": {
    "ssh": {
      "enabled": true,
      "timeout": "off"
    },
    "telnet": {
      "enabled": true,
      "timeout": "off"
    }
  }
}
```

Platform.FactoryReset

Factory resets the unit.

Structure
{"jsonrpc": "2.0","id": "X","method": "Platform.FactoryReset"}

Identifier	Request Object	Parameters
X	id	**Default:** PlatformFactoryResetResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands

Example

```
{
  "jsonrpc": "2.0",
  "id": "PlatformFactoryResetResults",
  "method": "Platform.FactoryReset"
}
```

Feedback

```
{
    "id": "PlatformFactoryResetResults",
    "result": true,
    "jsonrpc": "2.0"
}
```

Platform.Reboot

Reboots the unit.

Structure
{"jsonrpc": "2.0","id": "X","method": "Platform.Reboot"}

Identifier	Request Object	Parameters
X	id	**Default:** PlatformRebootResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands

Example

```
{
  "jsonrpc": "2.0",
  "id": "PlatformRebootResults",
  "method": "Platform.Reboot"
}
```

Feedback

```
{
    "id": "PlatformRebootResults",
    "result": true,
    "jsonrpc": "2.0"
}
```

System.Get

Displays the current system's information.

Structure
{"jsonrpc": "2.0","id": "X","method": "System.Get"}

Identifier	Request Object	Parameters
X	id	**Default:** SystemGetResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands

Example

```
{
  "jsonrpc": "2.0",
  "id": "SystemGetResults",
  "method": "System.Get"
}
```

Feedback

```
{
  "id": "SystemGetResults",
  "jsonrpc": "2.0",
  "result": {
    "FwVer": "0.0.0.18",
    "HwVer": "V0.1",
    "OnTime": "0-18:49:26",
    "SerialNumber": "",
    "USB-C": "1.3.1",
    "ValensVer": "7.5.23",
    "blinkled": false,
    "frontpanel": false,
    "model": "AT-OME-SW21TX",
    "network": {
      "hostname": "AT-OME-SW21TX-1",
      "ipaddress": "192.168.23.214",
      "macaddress": "34:1b:22:80:11:44"
    },
    "standby": false
  }
}
```

SystemFrontPanel.Set

Enables/disables the front panel buttons to help installer identify the device.

Structure
`{"jsonrpc": "2.0","id": "X","method": "SystemFrontPanel.Set","params": Y}`

Identifier	Request Object	Parameters
X	id	**Default:** SystemFrontPanelSetResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands
Y	params	true, false

Example

```
{
  "jsonrpc": "2.0",
  "id": "SystemFrontPanelSetResults",
  "method": "SystemFrontPanel.Set",
  "params": true
}
```

Feedback

```
{
    "id": "SystemFrontPanelResults",
    "result": {
     "frontpanel": true
     },
    "jsonrpc": "2.0"
}
```

SystemHdbt.Get

Get the System HDBaseT Info.

Structure
{"jsonrpc": "2.0","id": "X","method": "SystemHdbt.Get"}

Identifier	Request Object	Parameters
X	id	**Default:** SystemHdbtGetResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands

Example

```
{
  "jsonrpc": "2.0",
  "id": "SystemHdbtGetResults",
  "method": "SystemHdbt.Get"
}
```

Feedback

```
{
  "id": "SystemHdbtGetResults",
  "jsonrpc": "2.0",
  "result": {
"TxVersion": "V1.0.0",
"RxVersion": "V1.0.0",
"Tmdsclk": "148 Mhz",
"CableLength": "20 meters",
"VideoQ": "Pass",
"QualityA": "Pass",
"QualityB": "Pass",
"QualityC": "Pass",
"QualityD": "Pass",
  }
}
```

SystemHdbtStart.Set

Set the System HDBaseT information to start or stop.

Structure
{"jsonrpc": "2.0","id": "X","method": "SystemHdbtStart.Set","params": Y}

Identifier	Request Object	Parameters
X	id	**Default:** SystemHdbtStartSetResults ***Optional*** Type in a user id, this will return with each feedback, to provide a way to see the difference between commands
Y	params	true, false

Example

```
{
  "jsonrpc": "2.0",
  "id": "SystemHdbtStartSetResults",
  "method": "SystemHdbtStart.Set",
  "params": true
}
```

Feedback

```
{
    "id": "SystemHdbtStartSetResults",
    "result": {
     "Start": true,
 },
    "jsonrpc": "2.0"
}
```