INNES

AV Command

User manual

Feb. 2019

001G_en

1 Aim of document

The aim of this application note is to explain how to configure the media player to activate the support for AV Command functions.

2 Compatibility

- Gekkota : 4.10.10
- Screen Composer: 3.20.14

3 Introduction

3.1 AV Command profiles

The supported **AV Command profiles** are :

- Ethernet
- Serial
- DDC

The AV Command profile permits to the player to control some of the functions of the monitor without

- Pressing any key of the monitor remote control and
- Pressing any button of the monitor

3.2 AV Command monitor protocols

Each AV Command profile supports several **AV Command monitor protocols**:

- Standard (implemented by INNES following TV device user guide)
  - Samsung_m1
  - nec_m1
  - …
- Custom
  - Up to customer to implement it.

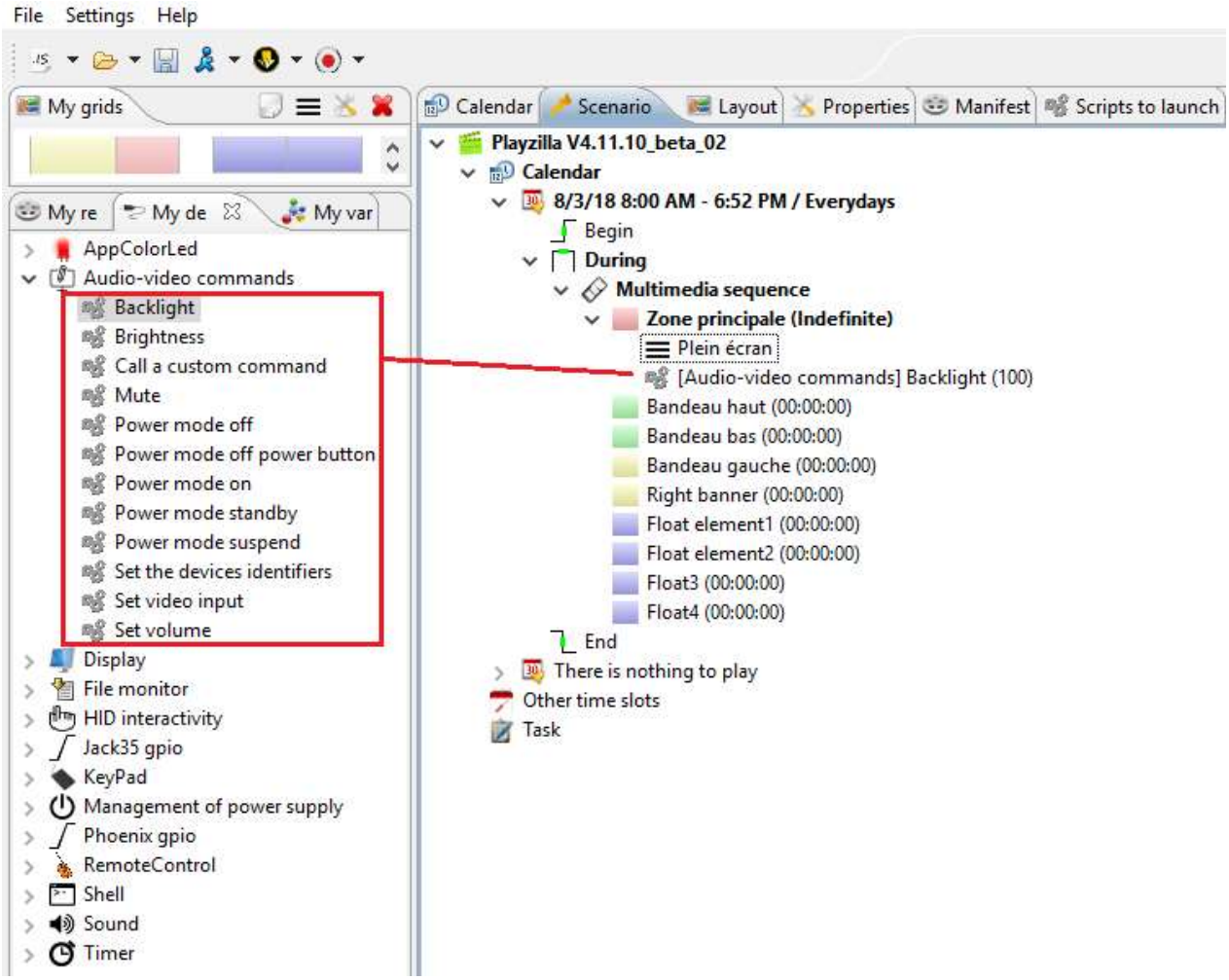
3.3 AV Commands

Each monitor protocol supports several **AV commands:**

- Standard
  - Standby
  - Brightness
  - Video input
  - Mute
  - …
- Custom
  - Up to customer to implement it and install it in Gekkota

3.4 AV Command & Screen Composer

The AV Commands are available in the tab **My Devices** in Screen Composer.



Now you have to activate the appropriate AV Command profile for each media player:

- DDC/CI
- LAN
- Serial

4 AV Command profiles

The **serial AV Commands** are used for monitor connected to the media player with a RS232 cable 'SUB-D9 to SUBD9' or 'SUB-D9 to Jack 3.5'

- Some USB to serial adapter (2 to 1, 4 to 1, …) may be supported for the device having no RS232 DTE connector. For example, the SMA300 device supports these adapters type:
  - FTDI,
  - PROFILIC, ….).
    Warning: In this case, only the first COM is supported.
    For more information, contact support@innes.pro.
- Warning: your monitor needs to be configured in serial mode to be able to support serial AV Commands. For more information, refer to your monitor user manual.

- In case your media player does not support your USB to serial adapter, you can use the LAN AV Commands instead.

The **LAN AV Commands are** used for monitors connected to the media player with an Ethernet cable.

- Warning: the monitor needs to be configured in Ethernet mode to be able to support LAN AV Commands. . For more information, refer to your monitor user manual.
- Check the port value used by your monitor to listen to the LAN AV Commands.

The DDC/CI **AV Commands** are used for monitors connected to the player with HDMI, VGA or DVI cable.

- Warning: the monitor needs to support the DDC/CI AV Commands
- The used video cable (HDMI, VGA or DVI) and the monitor connector has to support DDC/CI (pin DDC/SDA & DDC/SCL)

4.1 TV Commands

4.1.1 Standard TV commands

Standard AV Commands

- Standby
- Power mode
- Brightness
- Backlight
- Video input
- Mute
- Volume

Note: "mute" and "volume" AV Commands are effective whatever if the audio is transmitted to the monitor by an audio-video cable (HDMI) or a audio cable (jack35)

The list of standard AV commands are implemented with this string pattern:

- `power-mode_OFF`
- `power-mode_ON`
- `power-mode_STANDBY`
- `mute_ON`
- `mute_OFF`
- `brightness_0, brightness_1, ... , brightness_99, brightness_100`
- `backlight_0, backlight_1, ... , backlight_99, backlight_100`
- `volume_0, volume_1, ... , volume_0, volume_100`

*Refer to chapter Example of AV Command script for Samsung device: "Samsung_m1"

These are the standard input values

```
Based on MCCS 2.2 standard
https://milek7.pl/ddcbacklight/mccs.pdf

VIDEO_INPUT_NONE = 0
VIDEO_INPUT_RGB1 = 1
VIDEO_INPUT_RGB2 = 2
VIDEO_INPUT_TMDS1 = 3
VIDEO_INPUT_TMDS2 = 4
VIDEO_INPUT_COMPOSITE1 = 5
VIDEO_INPUT_COMPOSITE2 = 6
VIDEO_INPUT_SVIDEO1 = 7
```

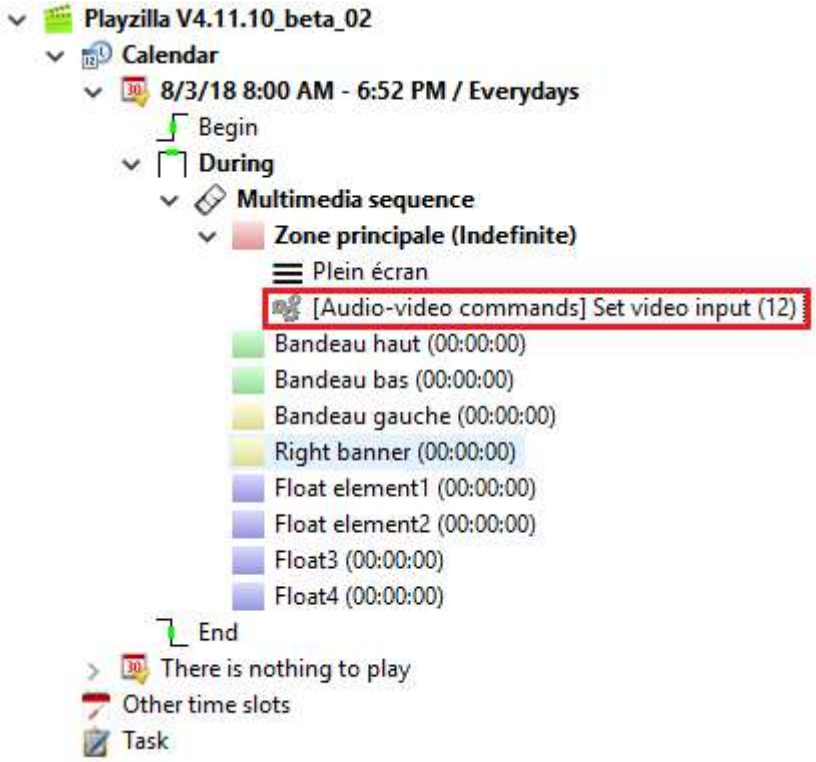
```
VIDEO_INPUT_SVIDEO2 = 8
VIDEO_INPUT_TUNER1 = 9
VIDEO_INPUT_TUNER2 = 10
VIDEO_INPUT_TUNER3 = 11
VIDEO_INPUT_COMPONENT1 = 12
VIDEO_INPUT_COMPONENT2 = 13
VIDEO_INPUT_COMPONENT3 = 14
VIDEO_INPUT_DISPLAY_PORT1 = 15
VIDEO_INPUT_DISPLAY_PORT2 = 16
VIDEO_INPUT_TMDS3 = 17
VIDEO_INPUT_TMDS4 = 18

Others

/** HDMI inputs */
VIDEO_INPUT_HDMI1   = 101
VIDEO_INPUT_HDMI2   = 102
VIDEO_INPUT_HDMI3   = 103
VIDEO_INPUT_HDMI4   = 104
VIDEO_INPUT_HDMI5   = 105
VIDEO_INPUT_HDMI6   = 106
VIDEO_INPUT_HDMI7   = 107
VIDEO_INPUT_HDMI8   = 108
/** VGA inputs */
VIDEO_INPUT_VGA1   = 111
VIDEO_INPUT_VGA2   = 112
VIDEO_INPUT_VGA3   = 113
VIDEO_INPUT_VGA4   = 114
/** DVI inputs */
VIDEO_INPUT_DVI1   = 121
VIDEO_INPUT_DVI2   = 122
VIDEO_INPUT_DVI3   = 123
VIDEO_INPUT_DVI4   = 124
/** RGB inputs */
VIDEO_INPUT_RGB3   = 133
VIDEO_INPUT_RGB4   = 134
/** COMPONENT input */
VIDEO_INPUT_COMPONENT4   = 144
/** VIDEO input */
VIDEO_INPUT_VIDEO1   = 151
VIDEO_INPUT_VIDEO2   = 152
VIDEO_INPUT_VIDEO3   = 153
VIDEO_INPUT_VIDEO4   = 154
/** SVIDEO input */
VIDEO_INPUT_SVIDEO3   = 163
VIDEO_INPUT_SVIDEO4   = 164
/** COMPOSITE input */
VIDEO_INPUT_COMPOSITE3   = 173
VIDEO_INPUT_COMPOSITE4   = 174
/** PC input */
VIDEO_INPUT_PC1   = 181
VIDEO_INPUT_PC2   = 182
VIDEO_INPUT_PC3   = 183
VIDEO_INPUT_PC4   = 184
/** DTV input */
VIDEO_INPUT_DTV1   = 191
VIDEO_INPUT_DTV2   = 192
VIDEO_INPUT_DTV3   = 193
VIDEO_INPUT_DTV4   = 194
/** OTHER input */
VIDEO_INPUT_TV   = 200
VIDEO_INPUT_TV1   = 201
VIDEO_INPUT_TV2   = 202
```

```
VIDEO_INPUT_SCART1    = 211
VIDEO_INPUT_SCART2    = 212

VIDEO_INPUT_TUNER4    = 224
/** DIPLAY PORT inputs */
VIDEO_INPUT_DISPLAY_PORT3    = 233
VIDEO_INPUT_DISPLAY_PORT4    = 234
VIDEO_INPUT_DISPLAY_PORT5    = 235
VIDEO_INPUT_DISPLAY_PORT6    = 236
VIDEO_INPUT_DISPLAY_PORT7    = 237
VIDEO_INPUT_DISPLAY_PORT8    = 238
VIDEO_INPUT_OTHER1    = 241
VIDEO_INPUT_OTHER2    = 242
VIDEO_INPUT_OTHER3    = 243
```

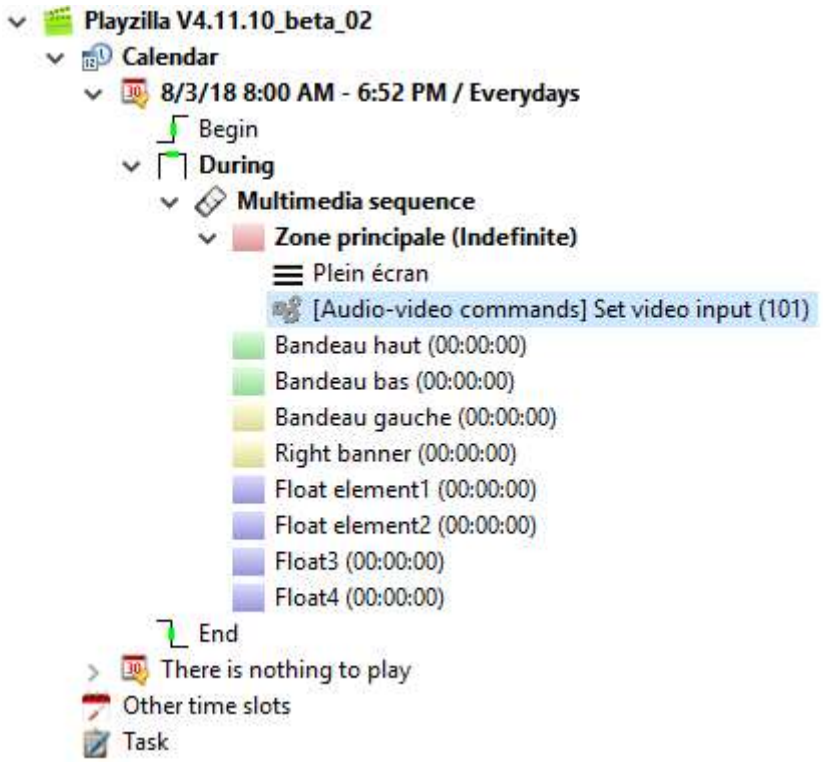
4.1.2 Screen Composer & AV commands video-input

You have to use the integer values in Screen Composer:

- *ex:* VIDEO_INPUT_COMPONENT1



- *ex:* VIDEO_INPUT_HDMI1

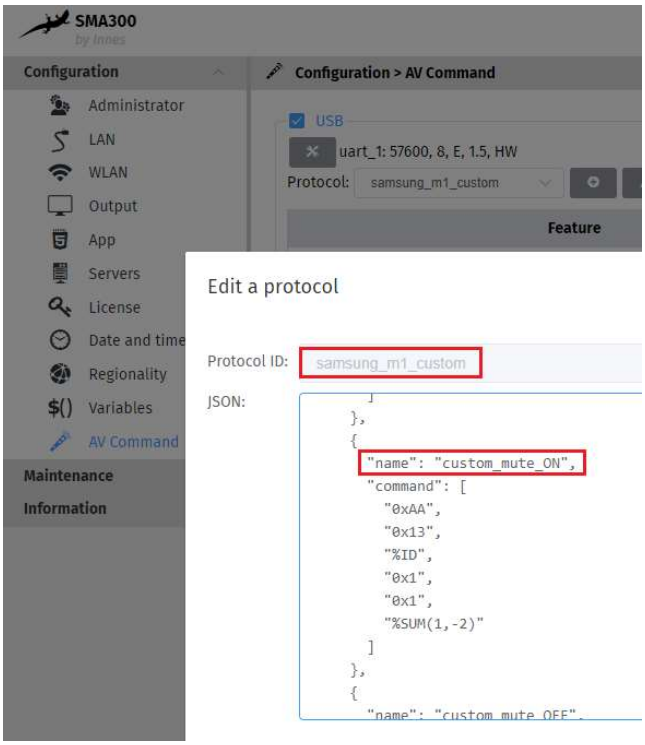


4.1.3 Custom TV commands

To use a custom-command, clone a protocol and modify it or create a new one.

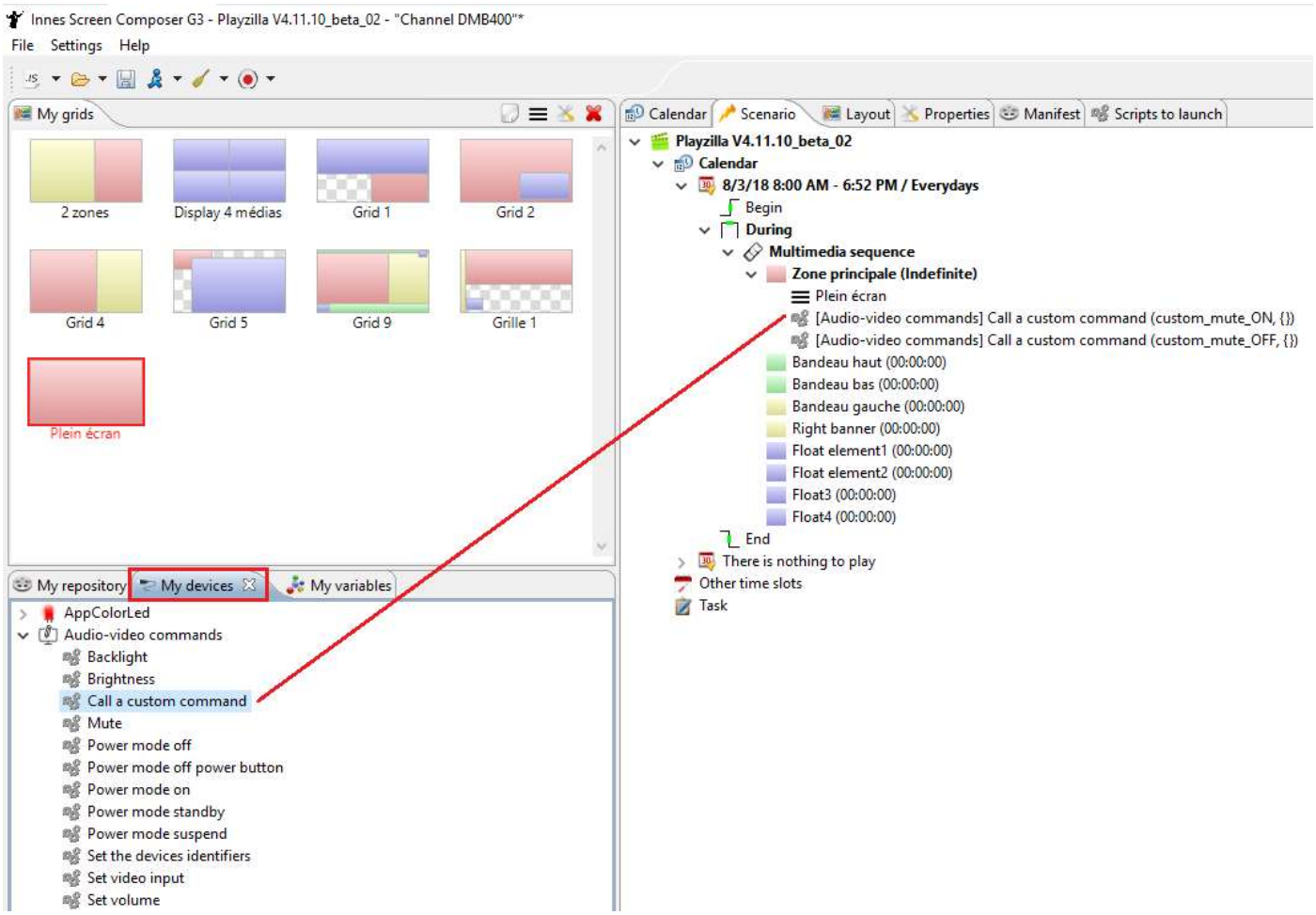
Warning: you have to respect the JSON format syntax !!!

Warning: Innes does not bring support to customer on the creation of their custom protocols.




```
{"name": "custom_mute_ON", "command": ["0xAA", "0x13", "%ID",
"0x1", "0x0",    "%SUM(1,-2)"]      }
```

How to use it in Screen Composer



ser :

4.2 List of monitors protocols (supported by default)

These are the monitors protocols supported by default in serial and in Ethernet.

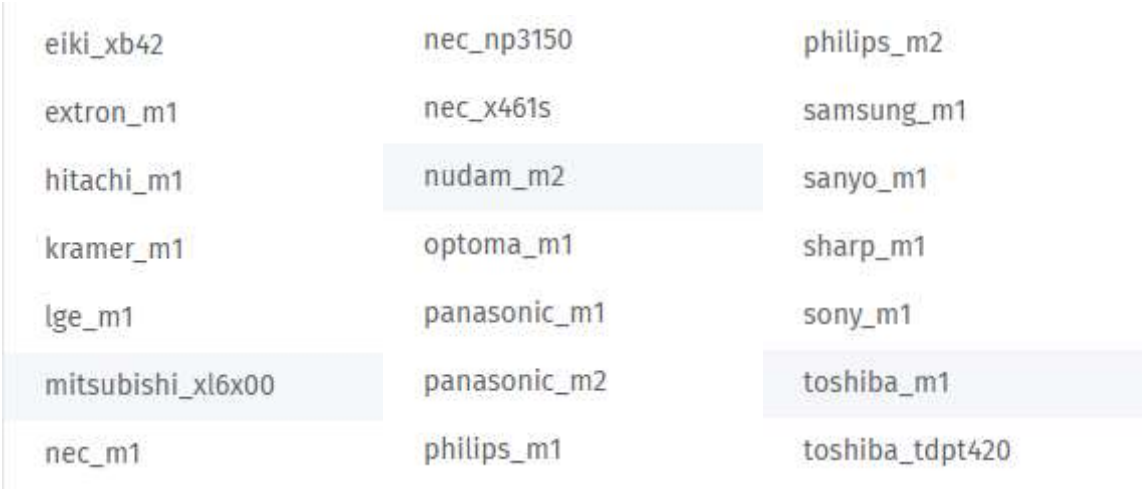
Note: The AV Command profiles serial and Ethernet are not activated by default. Indeed, some specific preferences need to be programmed in the player. Please refer to next chapter Configuration for more information.

4.2.1 RS232 (serial)

These are the supported monitor protocols with the AV Command serial profile

- eiki_xb42
- nec_np3150
- philips_m2
- extron_m1
- nec_x461s
- samsung_m1
- hitachi_m1
- nudam_m2
- sanyo_m1
- optoma_m1

- sharp_m1
- lge_m1
- panasonic_m1
- sony_m1
- mitsubishi_xl6x00
- panasonic_m2
- toshiba_m1
- nec_m1
- philips_m1
- toshiba_tdpt420
- kramer_m1



eiki_xb42	nec_np3150	philips_m2
extron_m1	nec_x461s	samsung_m1
hitachi_m1	nudam_m2	sanyo_m1
kramer_m1	optoma_m1	sharp_m1
lge_m1	panasonic_m1	sony_m1
mitsubishi_xl6x00	panasonic_m2	toshiba_m1
nec_m1	philips_m1	toshiba_tdpt420

4.2.2 Ethernet (TCP/IP)

These are the supported monitor protocols with the AV Command LAN profile

- eiki_xb42
- nec_np3150
- philips_m2
- extron_m1
- nec_x461s
- samsung_m1
- hitachi_m1
- nudam_m2
- sanyo_m1
- optoma_m1
- sharp_m1
- lge_m1
- panasonic_m1
- sony_m1
- mitsubishi_xl6x00
- panasonic_m2
- toshiba_m1
- nec_m1
- philips_m1
- toshiba_tdpt420
- kramer_m1

eiki_xb42	nec_np3150	philips_m2
extron_m1	nec_x461s	samsung_m1
hitachi_m1	nudam_m2	sanyo_m1
kramer_m1	optoma_m1	sharp_m1
lge_m1	panasonic_m1	sony_m1
mitsubishi_xl6x00	panasonic_m2	toshiba_m1
nec_m1	philips_m1	toshiba_tdpt420

4.2.3 DDC (over HDMI/DVI/VGA)

Supports DDC over HDMI, DVI and VGA connector

- DDC-EDID
- DDC-CI v1 / MCCS v2.2

Most of devices support DDC-CI but only few of them supports fully MCCS v2.2.

If your screen does not support MCCS, some AV Command could not work properly.

Note: Most of screen does not support yet MCCS v3

4.3 Configuration

These AV Command profiles are supported:

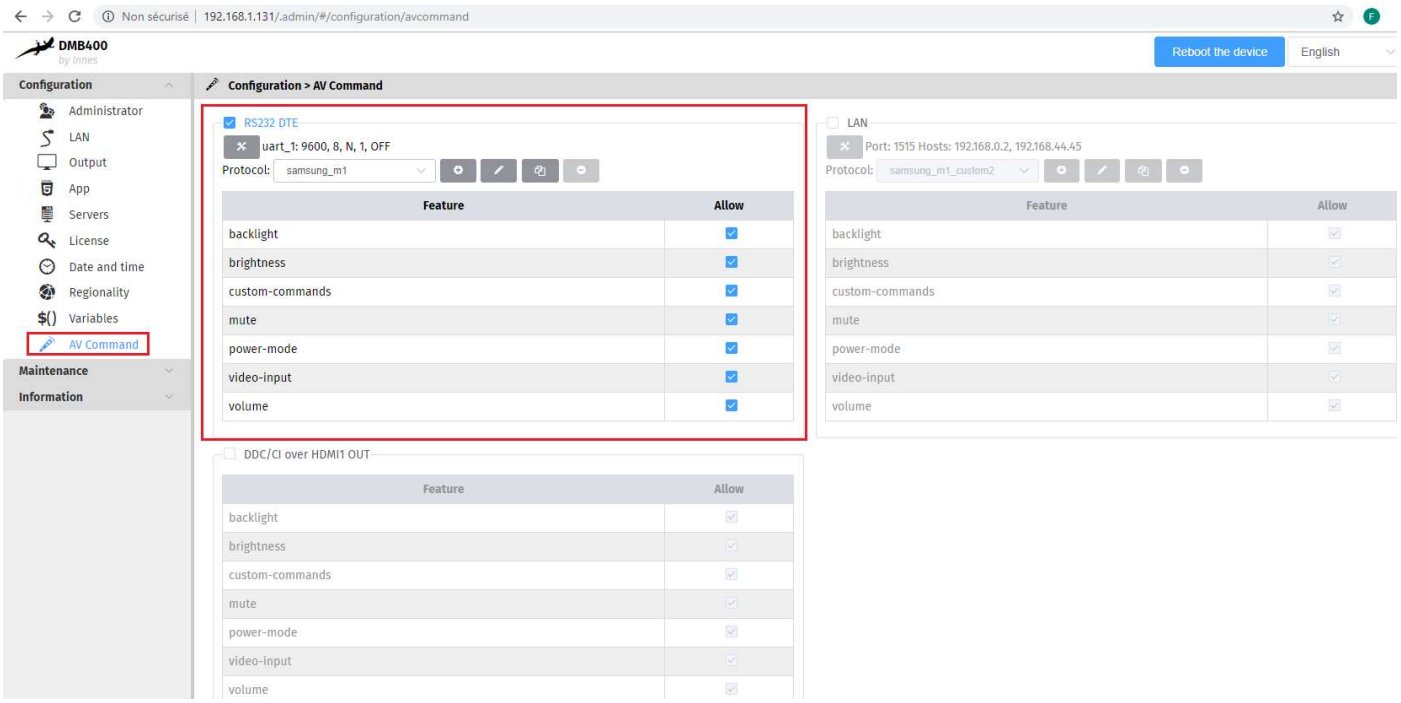
- "**Ethernet**" AV Command profile
  - Player is connected to the IP network (for publishing)
  - Monitor connected to the IP network and configured to receive TCP/IP TV commands
  - Player is connected to the monitor with AV cables
  - Daisy chain: head TV is connected to the player with Ethernet, other TV in serial (TVs belong to same manufacturer series)
- "**Serial**" AV Command profile
  - Player is connected to the IP network (for publishing)
  - Player supporting RS232 DTE connector is connected to the monitor with serial cable
    - *Note in some case, RS232 input connector on Monitor is a jack 3.5" format and requires a specific cable*
    - *Gekkota does not support adapter serial to USB except SMA300 – V3.12.31)*
  - Player SMA300 supporting serial to USB adaptor (vendor id "profilic" of "ftdi"), serial connector side is connected to the Monitor with serial cable.
    - *See Gekkota application note "Driver USB to serial: supported devices vendor ID" for more information*
  - Player is connected to the Monitor with AV cables
  - Daisy chain: head TV is connected to the player with serial cable and other TV are also connected in serial IN/OUT (TV are belonging to same manufacturer series)
- "**DDC**" AV Command profile
  - Player is connected to the IP network (for publishing)
  - Player is connected to the Monitor (supporting DDC) with VGA/HDMI/DVI cables (supporting DDC)

4.3.1 Serial configuration

4.3.1.1 User preferences for Serial AV Command profile

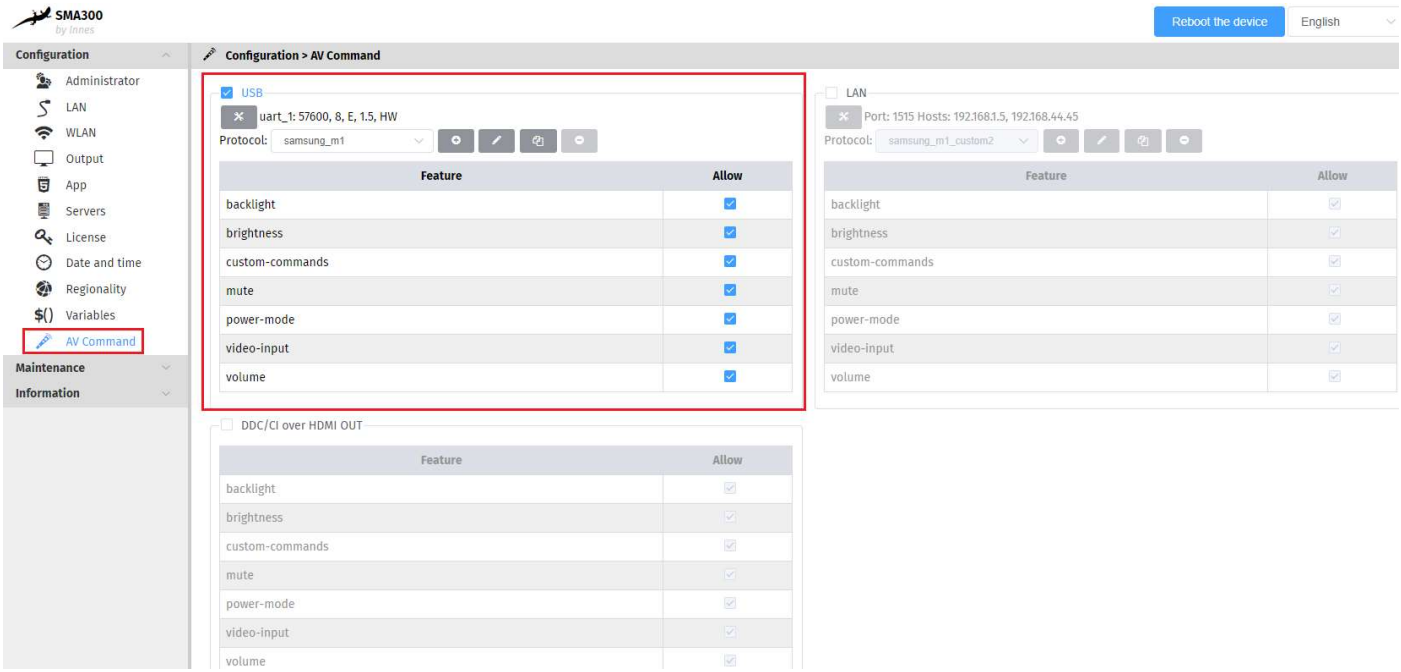
Use the device WebUI to configure the serial interface for AVCommand and to select the monitor protocol type.

Ex: **DMB400**



.

Ex: **SMA300 with a serial to USB adapter**

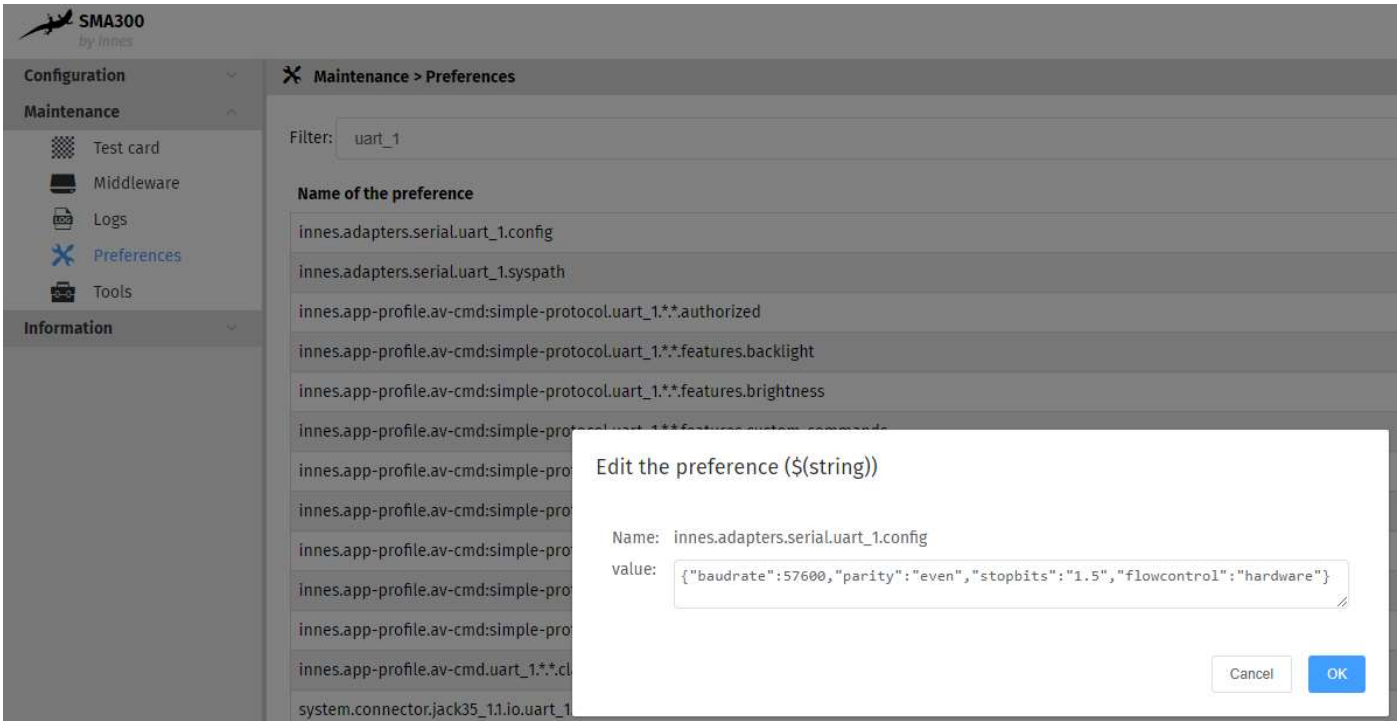


Warning: some serial AV Command may not be supported by your monitor. For more information, refer to your monitor manual.

The serial interface is already configured with the baudrate often used by the monitors.

innes.adapters.serial.uart_1.config : {"baudrate":9600,"parity":"none","stopbits":"1","flowcontrol":"off"}

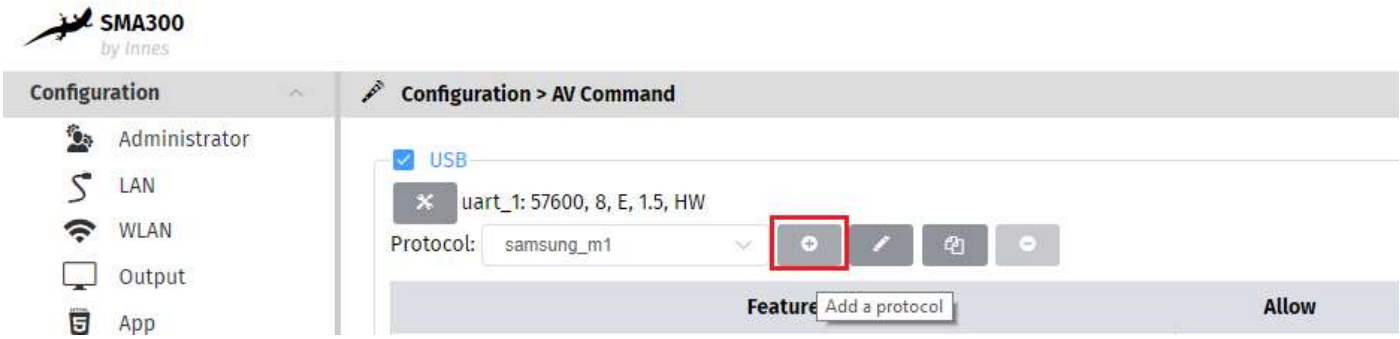
But it can be configured also by the customer.



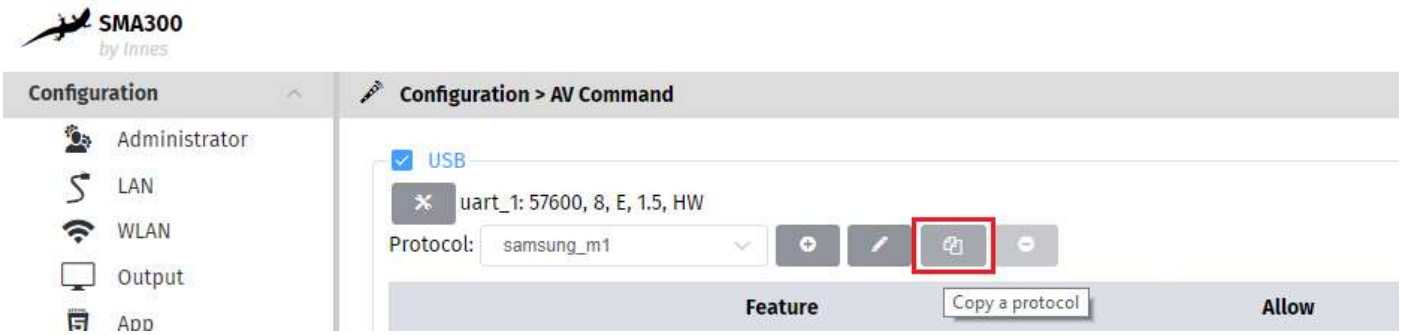
Example of user preferences value after programmation in the WebUI:

```
innes.app-profile.av-cmd.uart_1.*.*.class-name = simple-protocol
innes.app-profile.av-cmd:simple-protocol.uart_1.*.*.authorized = true
innes.app-profile.av-cmd:simple-protocol.uart_1.*.*.protocol = samsung_m1

innes.app-profile.av-cmd:simple-protocol.uart_2.*.*.authorized = false
innes.app-profile.av-cmd:simple-protocol.uart_3.*.*.authorized = false
innes.app-profile.av-cmd:simple-protocol.uart_4.*.*.authorized = false

for LCAN, set the preference
innes.adapters.serial.uart_1.syspath = /dev/ttyS0
for Gekkota_RT (player Windows), set the preference
innes.adapters.serial.uart_1.port = COM1
(or COM2, …, COM5, COM6 according the peripheral name installed on your PC)
```

It is possible to add a new custom protocol with USB injection of a configuration script or by connecting to the device Web UI.



Is is possible also to clone a protocol in order to customize it:



To inactivate the AV command Ethernet profile :

```
innes.app-profile.av-cmd:simple-protocol.network.*.*.authorized = false
```

User preferences for AV Command Serial profile: COM2, COM3, COM4

Some TV screen with rack-able system embedding Gekkota_RT do reserve COM1 for internal bus. So most of the time the external RS232 has to be done through COM2.

For the same reason Gekkota RT embedded on Windows player can support several ports COM, especially if an adapter USB to RS232 is installed. In this case, do activate the uart_<i> according to where the RS232 is plugged and inactivate all others.

```
innes.app-profile.av-cmd.uart_2.*.*.class-name = simple-protocol
innes.app-profile.av-cmd:simple-protocol.uart_2.*.*.authorized = true
innes.app-profile.av-cmd:simple-protocol.uart_2.*.*.protocol = samsung_m1

innes.app-profile.av-cmd:simple-protocol.uart_1.*.*.authorized = false
innes.app-profile.av-cmd:simple-protocol.uart_3.*.*.authorized = false
innes.app-profile.av-cmd:simple-protocol.uart_4.*.*.authorized = false
```

4.3.1.2 Monitor serial configuration

Monitor requires to be configured in serial mode according to:

For example

SAMSUNG

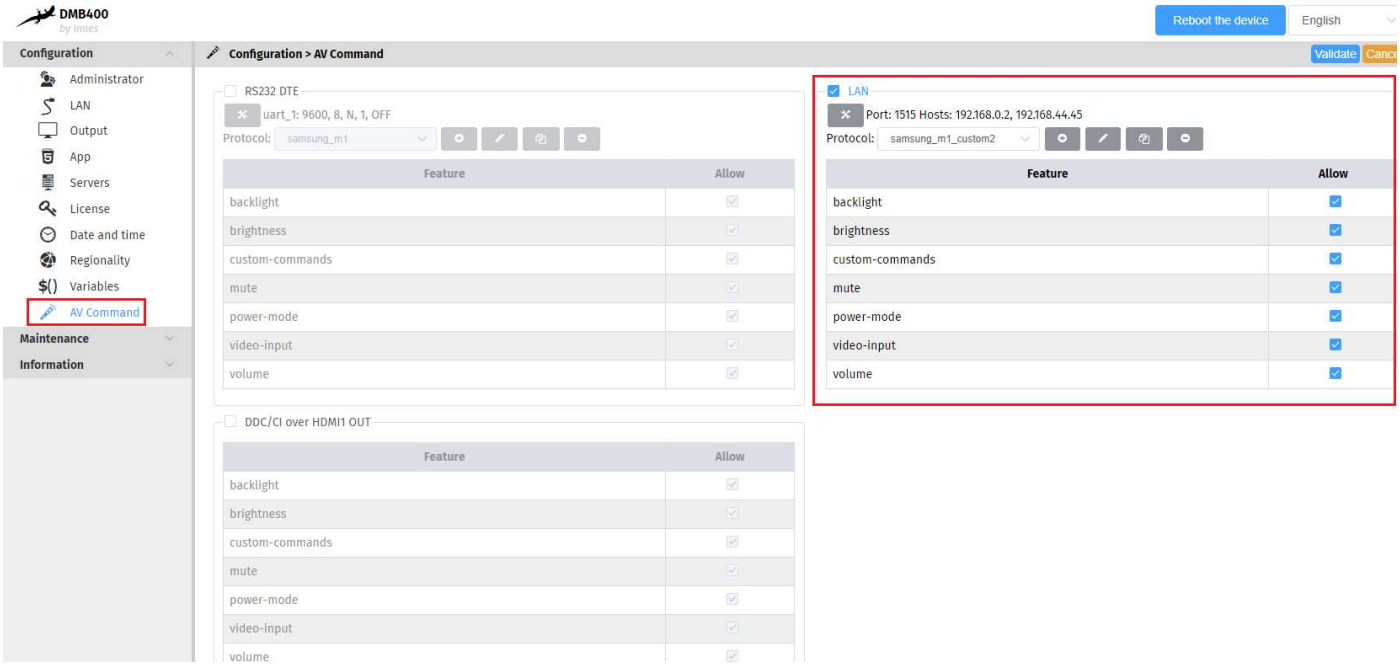
Samsung PE46C

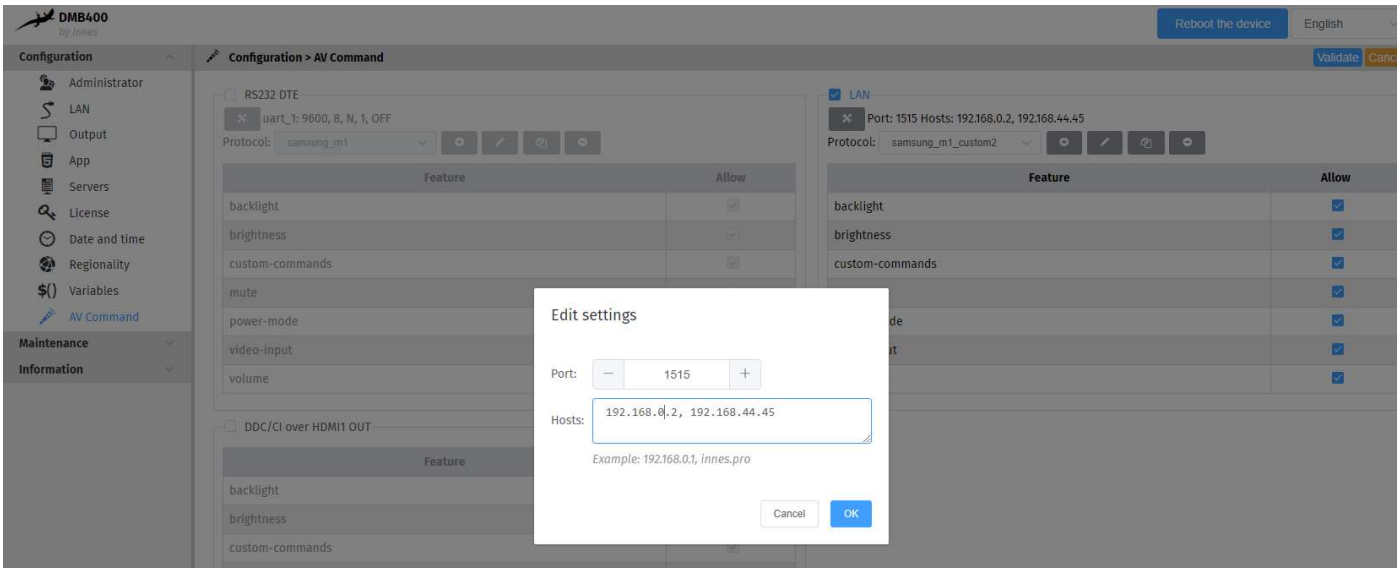
- Menu Multi-Control
  - Connection MDC
    - Select "**RS232C MDC**"

Connect RS232 serial cable between Monitor and the RS232 DTE connector of the player

4.3.2 Ethernet configuration

Use the device WebUI to configure the LAN interface for AVCommand and to select the monitor protocol type.





Warning: some LAN AV Command may not be supported by your monitor. For more information, refer to your monitor user manual.

4.3.2.1 User preferences for AV Command Ethernet profile

Associated user preferences:

```
innes.app-profile.av-cmd.network.*.*.class-name = simple-protocol
innes.app-profile.av-cmd:simple-protocol.network.*.*.authorized = true
innes.app-profile.av-cmd:simple-protocol.network.*.*.protocol = samsung_m1
innes.app-profile.av-cmd:simple-protocol.network.*.*.tcp.host = 192.168.1.10
innes.app-profile.av-cmd:simple-protocol.network.*.*.tcp.port = 1515
```

- "samsung_m1" is corresponding to one of the available device protocols available in Gekkota 3.12.26*.
- 192.168.1.10 is corresponding to the IP address of the Monitor in the local network
- 1515 is corresponding to the port listened by the monitor. Refer to the monitor user manual to know the value of this port

It is possible to add a new custom protocol with a USB injection of a configuration script or with the device Web UI.

Is is possible also to clone a protocol in order to customize it:

4.3.2.2 Set the monitor in Ethernet configuration

Monitor requires to be configured in Ethernet mode according to:

For example

Samsung PE46C

- Menu Multi-Control
  - Connection MDC
    - Select "**RJ45 MDC**"
- Menu Multi-Control
  - Network settings:
    - IP configuration:
      - IP address:
        - Static: 192.168.52.10 (or DHCP : **192.168.1.10**)
      - Mask : 255.255.128.0
      - Gateway : 192.168.0.1

Connect the Monitor to the local network with an Ethernet cable

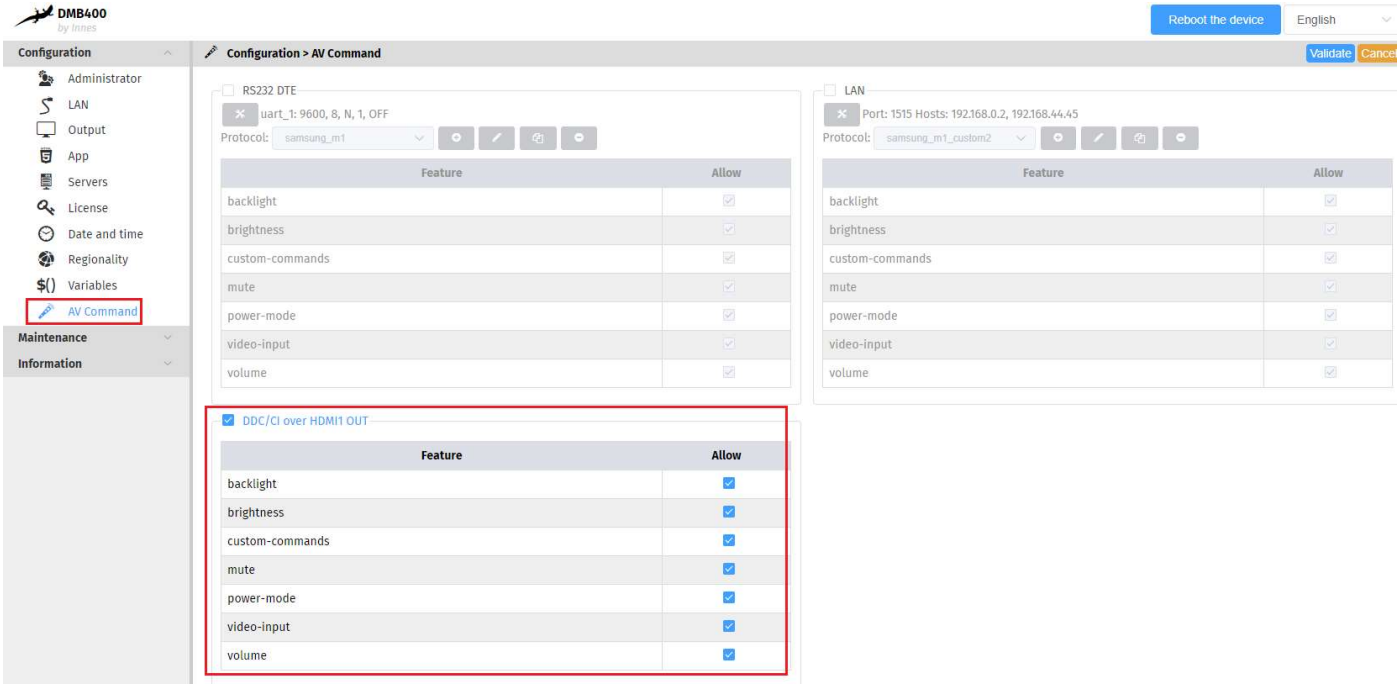
Note:

It is required to wait for a while before the TV is ready to communicate over Ethernet after these different use cases (this delay is depending on the device manufacturer)

- *monitor electric unplug*
- *Ethernet cable unplug*
- *IP socket reinit (for example when 2 players are using AV Command to access the same monitor)*
- *After player first publishing*

4.3.3 DDC configuration

Use the device WebUI to configure the DDC/CI interface for AV Command.



Warning: some DDC/CI AV Command may not be supported by your monitor. For more information, refer to your monitor user manual.

Associated user preferences

```
innes.app-profile.av-cmd.i2c_1.<connector>.authorized= true
```

.<connector>: can be

- *.*
- vga_1,
- hdmi_1,
- dvi_1

- For SMT210, set the preference below with the value ‘true’ (instead of previous one):

```
innes.app-profile.monitor-monitoring.gpu_1.lcd-module_1.authorized = true
```

5 Install custom monitor AV Command protocol

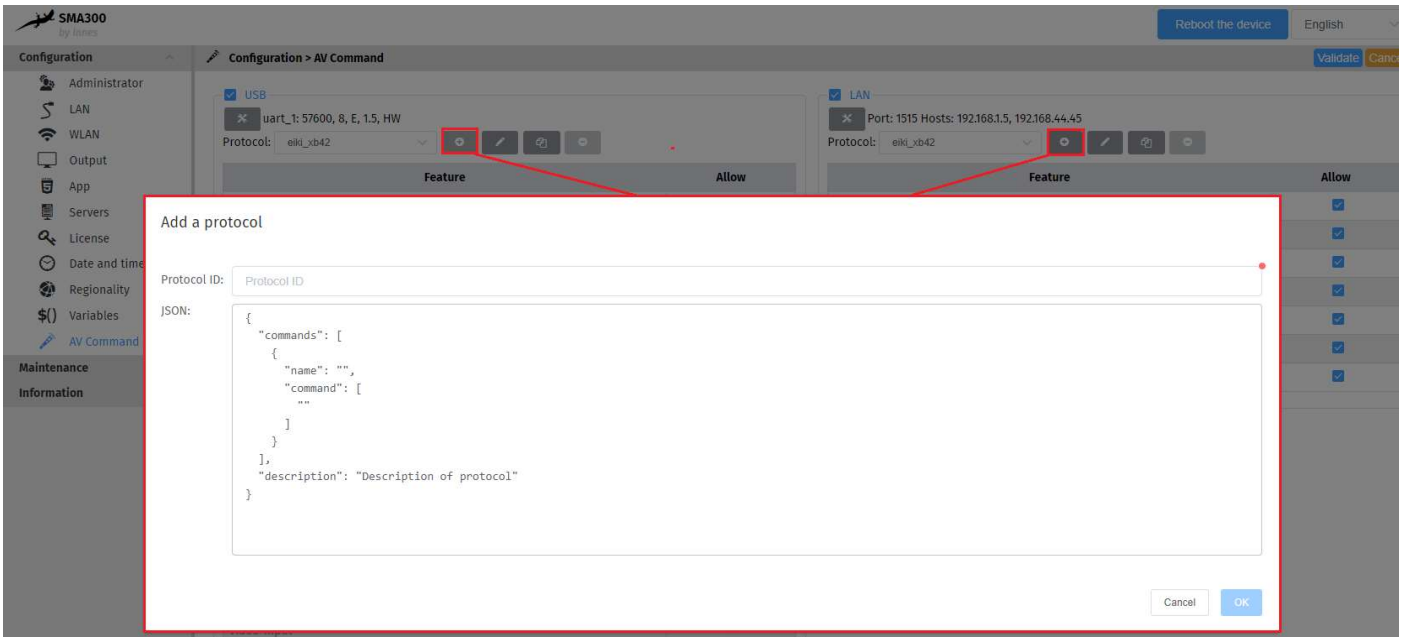
It is possible to install a custom monitor AV Command protocol for

- Serial profile
- Ethernet profile

It is possible to build and install custom monitor protocol by

- Using auto-configuration file
- Using WebUI

5.1 Building a AV Command monitor protocol using the media player WebUI



5.2 Building a custom AV Command monitor protocol using auto-configuration script

In order to install a new monitor protocol in your player

- Open the auto-configuration script V1.10.19 (or above) and follow the different step below
  - *Available in CDROM or on Innes support site (Gekkota application note auto-configuration)*

5.2.1 Build your own device protocol directly inside the auto-configuration file

- On the base of the tiny script below, write your own monitor protocol inside the template script. You can implement the different TV commands by picking up some existing TV commands names in JSON file (example in chapter Example of AV Command script for Samsung device: "Samsung_m1") and porting the appropriate bytes sequence for each TV commands corresponding to your monitor (refer to your monitor user manual)
- The custom protocol are not provided by INNES
- Don't use an existing name of default device protocol for your custom device protocol else the installation will failed
- In case Gekkota upgrade the device TV procotol custom name are kept and priority in case name conflict.
- Attention: the Javascript object is case sensitive; These are the good practises to port the JSON file into auto-configuration file:
  - Copy paste the JSON content (respect the parity of {} and [])
  - Add a header and choose a appropriate name for device TV and custom protocol name
    `let TVDeviceName="toshiba_custom";`
    `let TVDeviceProtocol=`
  - Change "name" and "description"
  - Add the ; character at the end of file
  - Implement the byte sequence according to your monitor in all the TV commands (the number of byte in the sequence can be different depending on manufacturer and TV commands types)
  - Do use the Javascript character " (do not use the MS Windows one " !!!)
  - In case installation error, you should be able to read error status in device-status (if it is well configured in WebUI)

```
let TVDeviceName="toshiba_custom";
let TVDeviceProtocol=
{
    "name":"toshiba_custom_info",
    "description":"Commands (on,off) for display TOSHIBA type AV/RV625D",
```

```
        "commands":
        [
                {
                        "name":"power-mode_ON",
                        "command":[
                                "0x77","0x30","0x34","0x20","0x00","0x20","0x01","0x0D"
                        ]
                },
                {
                        "name":"power-mode_OFF",
                        "command":[
                                "0x77","0x30","0x34","0x20","0x00","0x20","0x00","0x0D"
                        ]
                }
        ]
}
;
```

```
let TVDeviceName2="samsung_m1_custom";
let TVDeviceProtocol2=
{
   "name":"samsung_m1_custom_info",
   "description":"Commands (on,off,mute)",
   "commands":
   [
      {
         "name":"power-mode_OFF",
         "command":["0xAA","0xF9","0xFE","0x1","0x1","0xF9"]
      },
      {
         "name":"power-mode_ON",
         "command":["0xAA","0xF9","0xFE","0x1","0x0","0xF8"]
      },
      {
         "name":"power-mode_STANDBY",
         "command":["0xAA","0x4A","0xFE","0x1","0x1","0x4A"]
      },
      {
         "name":"mute_ON",
         "command":["0xAA","0x13","0xFE","0x1","0x1","0x13"]
      },
      {
         "name":"mute_OFF",
         "command":["0xAA","0x13","0xFE","0x1","0x0","0x12"]
      }
   ]
}
;
```

With 0xFE = Samsung broadcast id

5.2.1.1 Configure auto-configuration file to install your script for serial profile

In order to install your script in serial profile, in auto-configuration script, activate the line

- `let avCmdSerial = AvCmdGetProfile("uart_1");`
- `AvCmdInstallProtocol(avCmdSerial,TVDeviceName,TVDeviceProtocol);`
- `AvCmdActivateProfile(avCmdSerial);`

Note: several custom AV Command monitor protocol can be installed in Gekkota but only one can be active. Even if all your TV device custom protocol are installed, you have to install your TV device custom protocol again before to activate it.

```
// ------------------------------------------------------------------------------
// ---- AVCmd: init installation for custom TV device protocol for serial profile
// ------------------------------------------------------------------------------
// ---- Get the "av-cmd" from the uart_1 profile : uncomment the line after
// ---- Double check which player COM number (COM1 on uart1, COM2 on uart2... ) is used to control by
serial cable the Monitor and set the uart_n according to
let avCmdSerial = AvCmdGetProfile("uart_1");
```

```
//let avCmdSerial = AvCmdGetProfile("uart_2");
//let avCmdSerial = AvCmdGetProfile("uart_3");

// ------------------------------------------------------------------------------
// ---- AVCmd: activate and install a custom TV device protocol for serial profile
// ------------------------------------------------------------------------------
// ---- Install a protocol for the serial profile: uncomment the line after
AvCmdInstallProtocol(avCmdSerial,TVDeviceName,TVDeviceProtocol);
// ---- Activate the serial profile : uncomment the line after
AvCmdActivateProfile(avCmdSerial);

// ------------------------------------------------------------------------------
// ---- AVCmd: inactivate and uninstall a custom TV device protocol for serial profile
// ------------------------------------------------------------------------------
// ---- Inactivate the serial profile : uncomment the line after
//AvCmdDesactivateProfile(avCmdSerial);
// ---- Uninstall custom device protocol for the serial profile: uncomment the line after
//AvCmdDesinstallProtocol(avCmdSerial,TVDeviceName);
```

Save the file and put it in an empty USB key with the according naming format of your needs

- universal file name:
    - 000000000000.js

See Gekkota application note auto-configuration for more information

5.2.1.2 Configure auto-configuration file to install your script for Ethernet profile

In order to install your script in serial profile, in auto-configuration script, activate the lines

- `let avCmdTCPIP = AvCmdGetProfile("network");`
- `AvCmdInstallProtocol(avCmdTCPIP,TVDeviceName,TVDeviceProtocol);`
- `AvCmdActivateProfile(avCmdTCPIP);`

Note: several TV device custom protocol can be installed in Gekkota but only one can be active. Even if all your TV device custom protocol are installed, you have to install your TV device custom protocol again before to activate it.

```
/ ------------------------------------------------------------------------------
// ---- AVCmd: init installation for custom TV device protocol for Ethernet profile
// ------------------------------------------------------------------------------
// ---- Get the "av-cmd" from Ethernet profile: uncomment the line after
let avCmdTCPIP = AvCmdGetProfile("network");

// ------------------------------------------------------------------------------
// ---- AVCmd: activate and install a custom TV device protocol for Ethernet profile
// ------------------------------------------------------------------------------
// ---- Install a device protocol for the Ethernet profile : uncomment the line after
AvCmdInstallProtocol(avCmdTCPIP,TVDeviceName,TVDeviceProtocol);
// ---- Activate the TCP/IP profile : uncomment the line after
AvCmdActivateProfile(avCmdTCPIP);


// ------------------------------------------------------------------------------
// ---- AVCmd: inactivate and uninstall a custom TV device protocol for Ethernet profile
// ------------------------------------------------------------------------------
// ---- Inactivate the Ethernet : uncomment the line after
//AvCmdDesactivateProfile(avCmdTCPIP);
// ---- Uninstall custom device protocol for the Ethernet profile : uncomment the line after
//AvCmdDesinstallProtocol(avCmdTCPIP,TVDeviceName);
```

Save the file and put it in an empty USB stick with the according naming format of your needs

- universal file name:
    - 000000000000.js

See Gekkota application note auto-configuration for more information

5.2.2 Inject USB stick containing the auto-configuration script

Once USB stick injected, follow the indication on the TV screen connected to the player.

In case error is raised after script installation, it is probably due to a Javascript error inserted in the script. Please double check the implementation of your script by following the good practises

Internal Innes for support:
once installed, the script is installed with the those supported by defaut in Gekkota

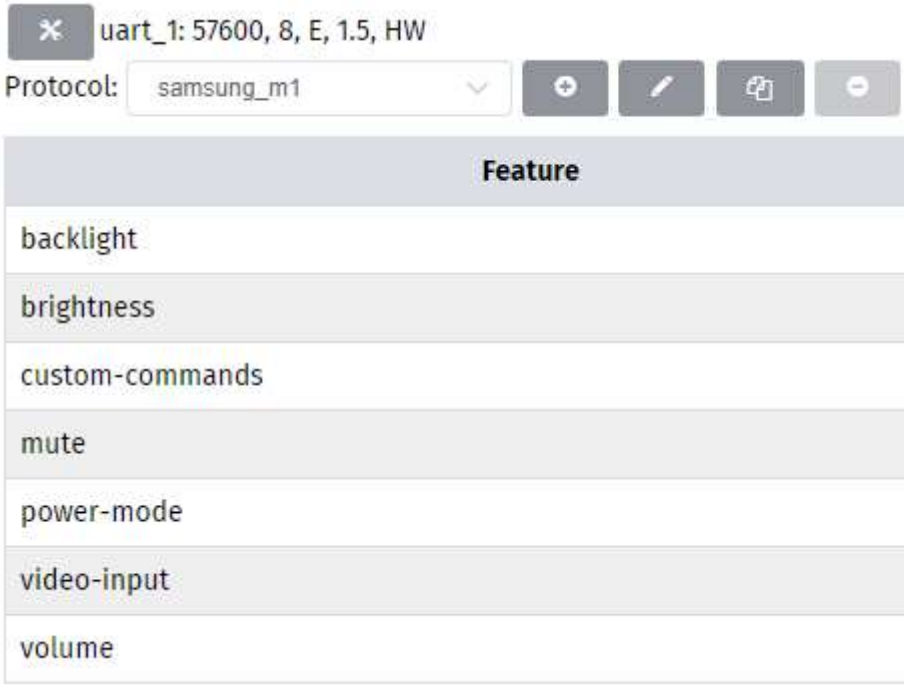
- `/usr/playzilla/profile/res/system/av-cmd/serial`
- `/usr/playzilla/profile/res/system/av-cmd/tcp`

These path are available by installing a debug patch. Please contact support@innes.pro for more information

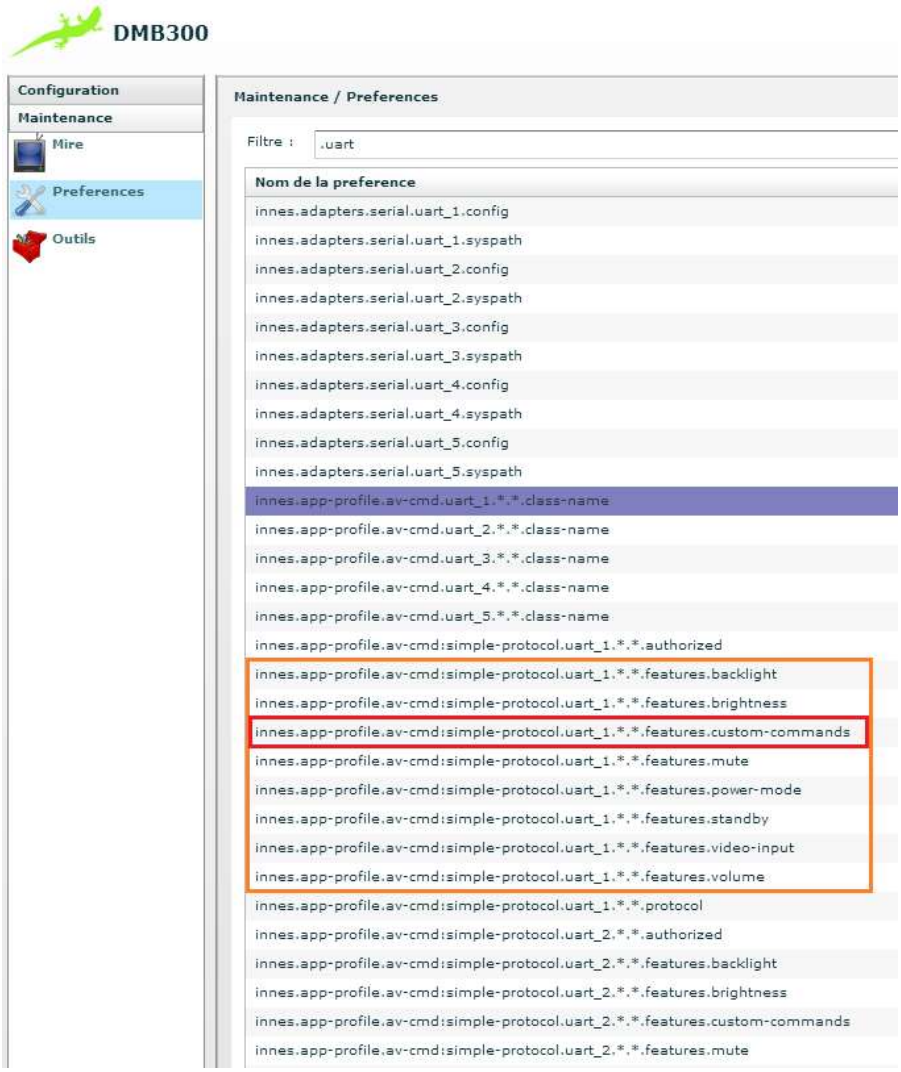
6 Inactivate some AV Commands for a standard protocol

When a profile serial or Ethernet is activated, by default all the TV commands types are activated (value is true).

It is possible to manually inactivate (or reactivate) the TV commands types:



- Features.backlight
- Features.brightness
- Features.mute
- Features.power-mode
- Features.standby
- Features.video-input
- Features.volume
- Features.custom-commands



6.1 Custom TV commands

Custom TV commands can be implemented only in custom device protocol.

A TV command becomes custom when its name is different of TV commands names listed in chapter:

Example of AV Command script for Samsung device: “Samsung_m1”

Note: Standard device protocol does not embed custom TV commands.

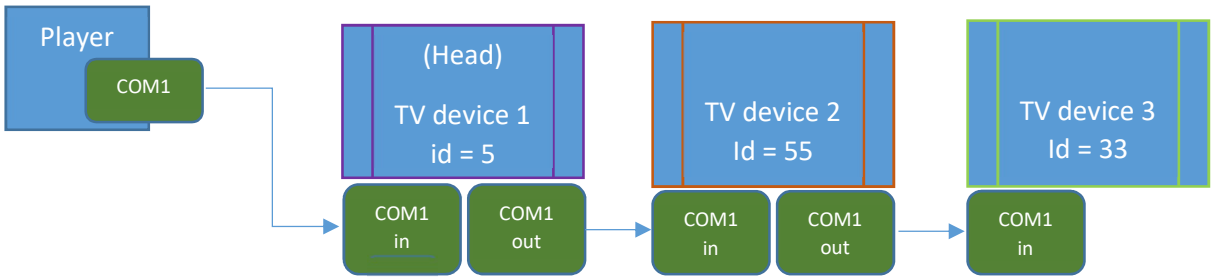
7 Daisy chain

Daisy chain is a chain of TV devices connected each other with RS232 cable and with the same manufacturer serie.

- The head TV device can be addressed as well by serial or Ethernet
- Each TV device can have a broadcast_id different:
  - Ex:
    - TV device 1: Samsung
      - Broadcast Id = 5
    - TV device 2: Samsung
      - Broadcast Id = 55
    - TV device 3 Samsung
      - Broadcast Id = 33
- Two mode permitting to send RS232 TV commands to TV device:
  - Mode broadcast
    - Meaning the TV command is applied by all the TV connected each other by RS232
  - Mode with TV device broadcast id
    - Meaning the TV command is applied only to TV
      - connected each other by RS232
      - having some specific id

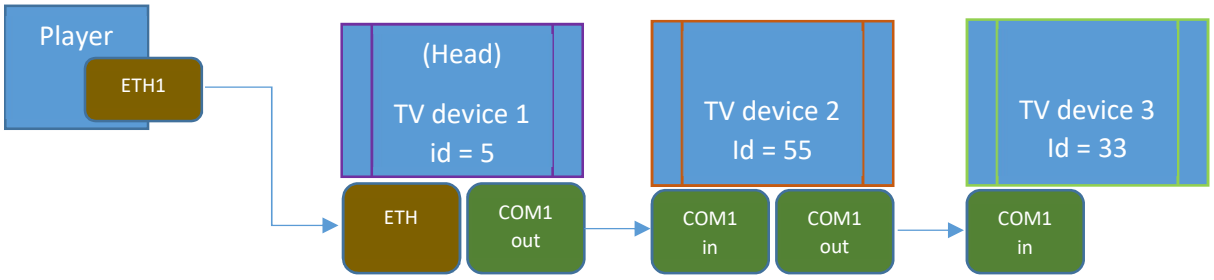
7.1 Head TV device: connected to the player in serial

Ex: TV device are Samsung manufacturer



7.2 Head TV device: connected to the player in Ethernet

Ex: TV device are NEC manufacturer



7.3 Broadcast id (Not implemented)

The broadcast id can be configured in the TV device (same menu as TV device Ethernet or serial configuration)

When the TV command is using manufacturer broadcast id, the TV command is propagated through all the TV device but only the TV device with the proper broadcast id apply the TV command.

Note: TV devices of the daisy chain are same manufacturer series

7.4 Broadcast

When the TV command is using manufacturer broadcast byte value (ex 0xFE for Samsung), the TV command is propagated through all the TV device (of same manufacturer) connected with RS232 cable, and all the TV apply the TV command.

Note: TV devices of the daisy chain are same manufacturer series

8 Appendix

8.1 serial profile AV Command protocols for specific monitors (embedding Gekkota_RT 3.12.57 on a rack-able module)

The AV Commands are supported for some for monitor equipped with rack-able OPS module, in which the serial communication is done between rack-able PC system and monitor. Given that a propriety protocol is used on this bus, the player has to send a specific binary frame to completely standby the TV Screen. These are the different typical connection:

- Through the OPS interface
  - For example : **NEC V 423** compatible OPS
- Through the HDMI cable (MagicInfo video input)
  - For example the **Samsung 480MX-3**, rack-able system (not compatible with OPS interface)
    - To activate MagicInfo, switch off the TV and switch on then press MagicInfo on remote control
    - HDMI cable compatible
      - HDMI cable “standard”: AWV E258864 STYLE 20276
      - Note: for this use case, do not use “HDMI cable supporting Ethernet” !!
- Through an external serial link

For information:

1. *The rack-able PC systems are not always compatible with OPS interface (ex: **Samsung PE46C**)*
2. *Some recent TV screen compatible OPS can support the AV Command serial profile (ex: Samsung serie ME)*

8.1.1 TV Commands

The supported TV commands are those used in INNES products to control the TV screen

- **TV screen standby**
- **TV screen wake up**

- Compatibility
  - Screen Composer G3
  - PlugnCast G2/G3

8.1.2 List of TV device internal serial protocols

- “**samsung internal**”
  - for Samsung series MDC2 and MDC3

- “**nec-internal**”
  - for NEC V 423 series for example

Note:

- *The TV screen standby/wake up through AV command replaces the legacy protocol implemented in previous version "display-output" automatically with the appropriate AV Command serial profile Samsung or NEC. That means that customer using Gekkota_RT 3.12.24 (or previous version) should install 3.12.26 without regression on TV standby/TV wake up.*
- *The preference installation is done while the migration from version 3.12.25 (or lower) to 3.12.26 (or upper). Before migration, switch OFF then switch ON TV Screen (it could be needed in some case to restore player factory settings before migrations)*
- *Pre-requisite: while the migration, some the serial internal communication between rack-able system and TV screen has to work properly. For example for Samsung 480MX-3 the TV screen video input has to be "MagicInfo". If not, the user preference will not be installed properly.*
- *In case regression (meaning that TV standby and TV wake up does not work properly because migration has failed), install manually by connecting to the WebUI:*
    - *Samsung 480MX-3*

```
innes.app-profile.av-cmd.uart_1.*.*.class-name = samsung-internal
innes.app-profile.av-cmd:samsung-internal.uart_1.*.*.authorized = true
innes.app-profile.display-output.*.*.*.features.standby = false
```

    - *NEC V 423*

```
innes.app-profile.av-cmd.uart_1.*.*.class-name = nec-internal
innes.app-profile.av-cmd:nec-internal.uart_1.*.*.authorized = true
innes.app-profile.display-output.*.*.*.features.standby = false
```

8.1.3 Configuration of TV device internal protocol

8.1.3.1 Samsung series with rack-able module

8.1.3.1.1 User preferences automatic activation

The AV Command specific serial profile is automatically activated when Gekkota RT 3.12.26 (or above) is installed.

In case issue with complete Samsung TV standby/wake-up (meaning that the migration has failed for any unexpected reason), set the user preferences below.

```
innes.app-profile.av-cmd.uart_1.*.*.class-name = samsung-internal
innes.app-profile.av-cmd:samsung-internal.uart_1.*.*.authorized = true
innes.app-profile.display-output.*.*.*.features.standby = false

innes.app-profile.av-cmd:Samsung-internal.uart_1.*.*.features.power-mode = true
innes.app-profile.av-cmd:Samsung-internal.uart_1.*.*.features.standby = true
```

These preferences can be inactivated to inactivate the specific functions linked to

- **Power mode**

```
innes.app-profile.av-cmd:samsung-internal.uart_1.*.*.features.power-mode = false
```

- **Standby mode**

```
innes.app-profile.av-cmd:samsung-internal.uart_1.*.*.features.standby = false
```

8.1.3.2 NEC series with rack-able module

8.1.3.2.1 User preferences automatic activation

The AV Command internal serial protocol automatically activated once fwhen Gekkota RT 3.12.26 (or above) is installed.

In case issue with complete TV NEC standby/wake-up (meaning that the migration has failed for any unexpected reason), set the user preferences below.

```
innes.app-profile.av-cmd.uart_1.*.*.class-name = nec-internal
innes.app-profile.av-cmd:nec-internal.uart_1.*.*.authorized = true
innes.app-profile.display-output.*.*.*.features.standby = false

innes.app-profile.av-cmd:nec-internal.uart_1.*.*.features.power-mode = true
innes.app-profile.av-cmd:nec-internal.uart_1.*.*.features.standby = true
```

These preferences can be inactivated to inactivate the specific functions linked to

- **Power mode**

```
innes.app-profile.av-cmd:nec-internal.uart_1.*.*.features.power-mode = false
```

- **Standby mode**

```
innes.app-profile.av-cmd:nec-internal.uart_1.*.*.features.standby = false
```