visual engineering
LIGHTWARE

Application Notes

Messaging and Controlling Functions with Lightware Devices

Table of Contents

Document Information

Document revision: 1.1
Release date: 24-08-2018
Editor: Laszlo Zsedenyi

Contact Us

sales@lightware.com
+36 1 255 3800

support@lightware.com
+36 1 255 3810

Lightware Visual Engineering LLC.
Peterdy 15, Budapest H-1071, Hungary

www.lightware.com

1

Introduction

The controlling and messaging features provided by Lightware Protocol #3 (LW3) is introduced in this document. Controlling Lightware devices or third-party devices remotely over the available interfaces are described by the detailed examples in the coming chapters.

The Purpose of this Document

When an A/V system is created, its devices and units are coming from different manufacturers – in most cases. One of the key elements during the installation is when the different devices start to communicate with each other – to get a parameterisable and reliable system for the user as a final result.

Besides, another important aspect is the ability for the remote controlling and/or automated controlling functions. When the most used functions can be executed by a button press or executed automatically, the user experience is improved a lot.

The messaging and controlling methods built in the Lightware Protocol #3 (LW3) can be used to implement the above mentioned functions. The LW3 protocol is implemented in almost all new products developed since 2012.

Device-dependence

As the devices contain different features the controlling and messaging features are also different. Certain functions cannot be implemented in all devices as they have different hardware/software specifications, thus, we emphasized the requirements in the examples.

General Rules (Lightware Protocol #3)

- All names and parameters are **case-sensitive**.
- The nodes are separated by a slash (‘/’) character.
- The node name can contain the elements of the English alphabet and numbers.
- Use the **TCP port no. 6107** when using LW3 protocol over Ethernet.
- When a command is issued by the device, the received response cannot be processed by the CPU.
- The path of a node has to contain all parent nodes from the root node.

Advantages of the Lightware Protocols

Complex controlling and monitoring the features of Lightware devices

Live protocol browsing by the freely available Lightware Device Controller (LDC) software

All-in-one commands

Transparent Ethernet

Serial-Ethernet bi-directional signal conversion

Monitoring the status of the connected devices, cables, and signals of each I/O port

Querying and changing the video parameters of the signals

Configurable analog audio parameters

1.1. TCP Message Sending via Ethernet

> INFO: The • symbol means the **space character** in the below sections.

Sending a TCP Message (ASCII-format)

The command is for sending a command message in ASCII-format. This method **allows** escaping the control characters, see the Escaping the Control Characters section.

Command and Response

- CALL•/MEDIA/ETHERNET:tcpMessage(<IP_address>:<port_no>=<message>)
- mO•/MEDIA/ETHERNET:tcpMessage

Example

- CALL /MEDIA/ETHERNET:tcpMessage(192.168.0.20:5555=C00)
- mO /MEDIA/ETHERNET:tcpMessage

The 'C00' message is sent to the indicated IP:port address.

Example with HEX codes

- CALL /MEDIA/ETHERNET:tcpMessage(192.168.0.20:5555=C00\x0a\x0d)
- mO /MEDIA/ETHERNET:tcpMessage

The 'C00' message with CrLf (Carriage return and Line feed) is sent to the indicated IP:port address. The \x sequence indicates the HEXA code; see more information in the Using Hexadecimal Codes section.

Sending a TCP Text (ASCII-format)

The command is for sending a text message in ASCII-format. This method **does not allow** escaping or inserting control characters.

Command and Response

- CALL•/MEDIA/ETHERNET:tcpText(<IP_address>:<port_no>=<text>)
- mO•/MEDIA/ETHERNET:tcpText

Example

- CALL /MEDIA/ETHERNET:tcpText(192.168.0.20:5555=open)
- mO /MEDIA/ETHERNET:tcpText

The 'open' text is sent to the indicated IP:port address.

Sending a TCP Binary Message (HEX-format)

The command is for sending a binary message in Hexadecimal format. This method **does not allow** escaping or inserting control characters.

Command and Response

- CALL•/MEDIA/ETHERNET:tcpBinary(<IP_address>:<port_no>=<HEX_message>)
- mO•/MEDIA/ETHERNET:tcpBinary

Example

- CALL /MEDIA/ETHERNET:tcpBinary(192.168.0.20:5555=433030)
- mO /MEDIA/ETHERNET:tcpBinary

The '433030' message is sent to the indicated IP:port address.

> INFO: There is no need to insert a space or other separator character between the binary messages.

1.2. UDP Message Sending via Ethernet

> INFO: The • symbol means the **space character** in the below sections.

Sending a UDP Message (ASCII-format)

The command is for sending a UDP message in ASCII-format. This method **allows** escaping the control characters, see the Escaping the Control Characters section.

Command and Response

- CALL•/MEDIA/ETHERNET:udpMessage(<IP_address>:<port_no>=<message>)
- mO•/MEDIA/ETHERNET:udpMessage

Example

- CALL /MEDIA/ETHERNET:udpMessage(192.168.0.20:9988=C00)
- mO /MEDIA/ETHERNET:udpMessage

The 'C00' message is sent to the indicated IP:port address.

Example with HEX codes

- CALL /MEDIA/ETHERNET:udpMessage(192.168.0.20:9988=C00\x0a\x0d)
- mO /MEDIA/ETHERNET:udpMessage

The 'C00' message with CrLf (Carriage return and Line feed) is sent to the indicated IP:port address. The \x sequence indicates the HEXA code; see more information in the Using Hexadecimal Codes section.

Sending a TCP Text (ASCII-format)

The command is for sending a text message in ASCII-format via UDP-protocol. This method **does not allow** escaping or inserting control characters.

Command and Response

- CALL•/MEDIA/ETHERNET:udpText(<IP_address>:<port_no>=<text>)
- mO•/MEDIA/ETHERNET:udpText

Example

- CALL /MEDIA/ETHERNET:udpText(192.168.0.20:9988=open)
- mO /MEDIA/ETHERNET:udpText

The 'open' text is sent to the indicated IP:port address.

Sending a UDP Binary Message (HEX-format)

The command is for sending a binary message in Hexadecimal format via UDP protocol. This method **does not allow** escaping or inserting control characters.

Command and Response

- CALL•/MEDIA/ETHERNET:udpBinary(<IP_address>:<port_no>=<HEX_message>)
- mO•/MEDIA/ETHERNET:udpBinary

Example

- CALL /MEDIA/ETHERNET:udpBinary(192.168.0.20:9988=433030)
- mO /MEDIA/ETHERNET:udpBinary

The '433030' message is sent to the indicated IP:port address.

> INFO: There is no need to insert a space or other separator character between the binary messages.

1.3. Message Sending via RS-232 Serial Port

> INFO: The • symbol means the space character in the below sections.

Sending a Message (ASCII-format)

The command is for sending a command message in ASCII-format. This method **allows** escaping the control characters, see the Escaping the Control Characters section.

Command and Response

- CALL•/MEDIA/UART/P1:sendMessage(<message>)
- mO•/MEDIA/UART/P1:sendMessage

Example

- CALL /MEDIA/UART/P1:sendMessage(PWR0)
- mO /MEDIA/UART/P1:sendMessage

The 'PWR0' message is sent out via the P1 serial port.

Sending a Text (ASCII-format)

The command is for sending a command message in ASCII-format. This method **does not allow** escaping the control characters.

Command and Response

- CALL•/MEDIA/UART/P1:sendText(<message>)
- mO•/MEDIA/UART/P1:sendText

Example

- CALL /MEDIA/UART/P1:sendText(open)
- mO /MEDIA/UART/P1:sendText

The 'open' text is sent out via the P1 serial port.

Sending a Binary Message (HEX-format)

The command is for sending a command message in Hexadecimal-format. This method **does not allow** escaping the control characters.

Command and Response

- CALL•/MEDIA/UART/P1:sendBinaryMessage(<message>)
- mO•/MEDIA/UART/P1:sendBinaryMessage

Example

- CALL /MEDIA/UART/P1:sendBinaryMessage(433030)
- mO /MEDIA/UART/P1:sendBinaryMessage

The '433030' message is sent out via the P1 serial port.

1.4. Escaping the Control Characters

> DEFINITION: An escape sequence is a sequence of characters that does not represent itself when used inside a character or string literal, but is translated into another character or a sequence of characters.

Property values and method parameters can contain characters that are used as control characters in the protocol. They must be escaped. The escape character is the backslash (‘\’) and escaping means injecting a backslash before the given character (like in C language).

Control characters are the followings: \ { } # % ( ) \r \n \t

A typical usage when a message is sent and it contains such a character that must be escaped.

Example

The original message: `CALL /MEDIA/UART/P1:sendMessage(Set(01))`

The escaped message: `CALL /MEDIA/UART/P1:sendMessage(Set\(01\))`

The above case is a typical example: the Lightware device is directed to send out a message over one of its port. The round brackets in the message are escaped.

1.5. Using Hexadecimal Codes

Hexadecimal codes can be inserted in the ASCII message when using:

sendMessage command: `CALL /MEDIA/UART/P1:sendMessage(C00\x0D)`

tcpMessage command: `CALL /MEDIA/ETHERNET:tcpMessage(C00\x0D)`

udpMessage command: `CALL /MEDIA/ETHERNET:udpMessage(C00\x0D)`

- **C00**: the message.
- **\x**: indicates that the following is a hexadecimal code.
- **0D**: the hexadecimal code (Carriage Return).

1.6. What is also Important

Response is not processed

When a command is sent from the Lightware device, the response coming from the other device cannot be processed.

The settings of the communication ports

Pay attention to the TCP/IP port no. (and have it opened) or the RS-232 port settings port settings in the connected devices.

Serial port mode setting

The serial port mode can be set differently in the devices: in certain devices the mode can be set individually, in other devices the mode setting is common and affects all serial ports.

Same subnet

The Ethernet devices must be in the same subnet.

Ping the device

If you have problems with accessing a device over Ethernet, try to check the connection e.g. by PING the IP address.

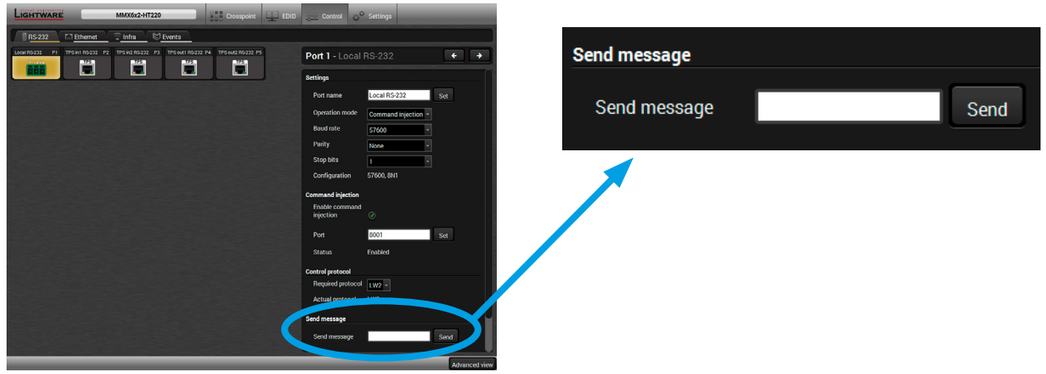
Input/Output port numbering

Please pay attention to the I/O port numbering when editing the commands. The Appendix chapter in each User's Manual contains the exact I/O port numbering of the product.

1.7. Integration in the Lightware Device Controller Software

Control Tab

The RS-232 tab in the Control menu contains the below indicated section:

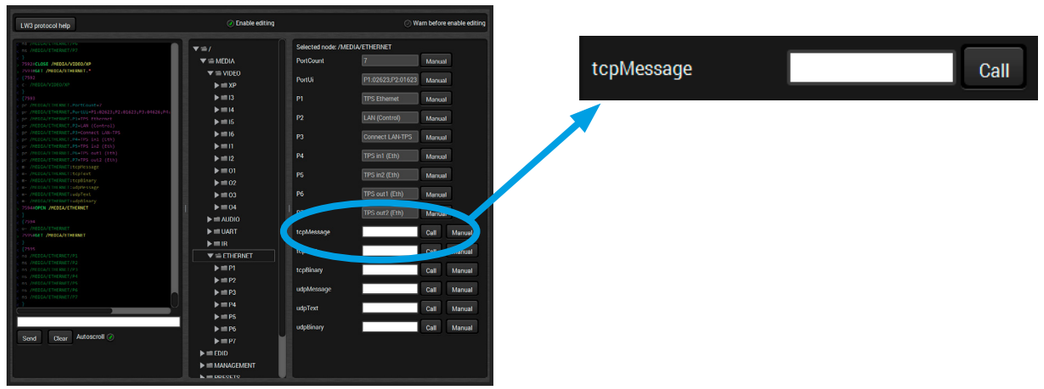


The message in the field can be sent out via the current RS-232 port. Response cannot be seen in the surface.

> **ATTENTION!** The escaping is done automatically when sending a message via this surface. When the command is an LW3 message it has to be closed by Carriage return and Line feed, e.g:
> CALL /MEDIA/VIDEO/XP:switch(I1:O1)\x0d\x0a.

Advanced View

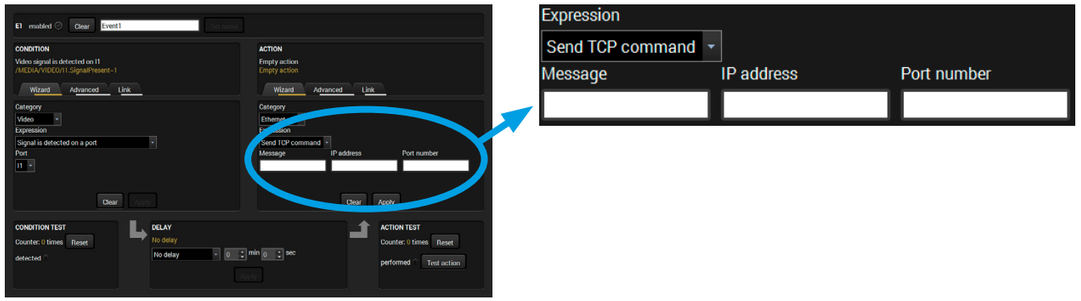
The Advanced View is suitable for not only testing the message sending feature but learning how to use and parametrize the LW3 commands; see the Terminal window also where you can type the commands.



> **ATTENTION!** The escaping is done automatically when sending a message via this surface. When the command is an LW3 message it has to be closed by Carriage return and Line feed, e.g:
> CALL /MEDIA/VIDEO/XP:switch(I1:O1)\x0d\x0a.

Event Manager

The message sending function is available also in the Event manager by defining an Action: sending a message. The feature is available in the Wizard and the Advanced surface also.

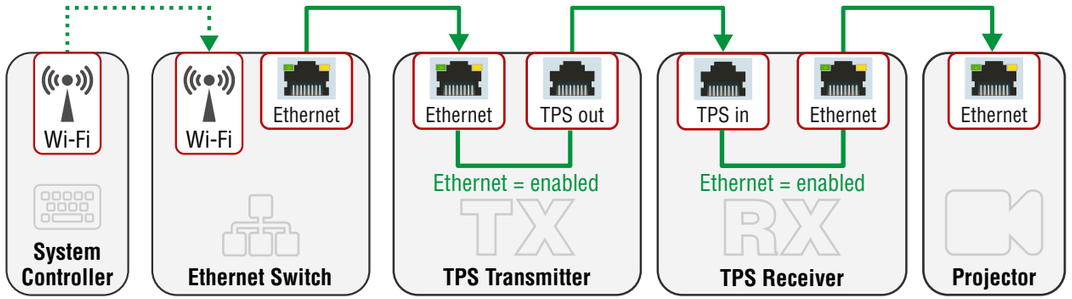


> **ATTENTION!** The escaping is done automatically when sending a message via this surface. When the command is an LW3 message it has to be closed by Carriage return and Line feed, e.g:
> CALL /MEDIA/VIDEO/XP:switch(I1:O1)\x0d\x0a.

1.8. Ethernet Messaging

What is Happening?

The **Projector** is switched on by the control command coming from the **System Controller**.



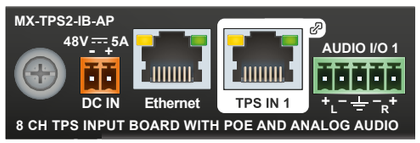
The Working Method

All the devices in this system are connected to the same Ethernet. However, the **Receiver** and the **Projector** are not connected to the network directly, the **TPS signal carries the Ethernet,** therefore, both devices can be accessed and addressed. Thus, the **System Controller** can send the control command directly to the **Projector**.

Preparations and Settings

- The devices (System Controller, Transmitter, Receiver, Projector) must be in the **same subnet** with configured and known IP address.
- The Ethernet has to be **enabled** at the TPS and Ethernet ports. This feature is also supported by the Basic TPS extenders (e.g.TPS-97 series).

INFO: The Ethernet for the TPS port has to be connected externally in the case of certain Lightware devices. E.g. there is a separate TPS Ethernet connector in **MX-TPS I/O boards**. Connect that port also to the Ethernet.



The command Sent by the System Controller (in HEX-format)

- 4330300D

INFO: According to the documentation of the projector the command (C00<CR>) has to be sent in hexadecimal format.

Explanation

- **433030:** The hexadecimal code of the "Power on" command.
- **0D:** The hexadecimal code of the **Carriage return** (requirement of the Projector)
- **192.168.0.50:9715: The IP address** and the **control TPC/IP port number** of the **Projector** (Projector's requirement). These are not shown in the above command since the format and the addressing depend on the **System Controller**.

The command Sent by the System Controller (in ASCII-format)

- C00\r

INFO: The ASCII format of the command is mentioned just to demonstrate another messaging option.

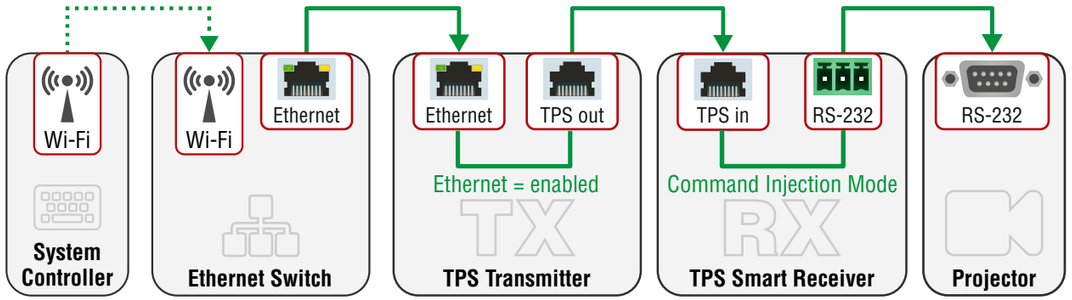
Explanation

- **C00:** The **"Power on"** command.
- **\r: Carriage return -** necessary to close the command (Projector's requirement).
- **192.168.0.50:9715: The IP address** and the **control TPC/IP port number** of the **Projector** (Projector's requirement). These are not shown in the above command since the format and the addressing depend on the **System Controller**.

1.9. Ethernet-Serial Messaging

What is Happening?

The **Projector** is switched on by the control command coming from the **System Controller**.



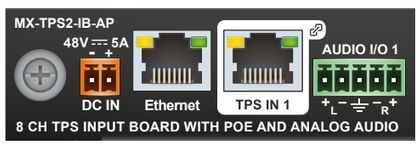
The Working Method

The Ethernet-Serial messaging is a bit similar to the previous case: an Ethernet message is sent. The key of this setup is the **Receiver's RS-232 serial port**. When the port is switched to **Command Injection Mode**, the port will act like an Ethernet-Serial bidirectional converter. The serial port gets a virtual **TCP port number** which can be addressed, thus, the desired command is sent directly to the **local RS-232** port of the **Receiver**.

Preparations and Settings

- The connected Ethernet devices (System Controller, TPS Transmitter, TPS Receiver) must be in the **same subnet** with configured and known IP address.
- The Ethernet has to be **enabled** at the TPS and Ethernet ports.
- The **RS-232 port settings** must be the same in the connected serial devices.
- The RS-232 port of the **Receiver** must be in **Command Injection Mode**. The Basic TPS extenders (like the TPS95 or TPS97 series) do not support that feature.
- Pay attention to the correct **serial cabling** (connector pinout).

> INFO: The Ethernet has to be connected externally in the case of certain Lightware devices. E.g. there is a separate TPS Ethernet connector in **MX-TPS I/O boards**. Connect that port also to the Ethernet.



The command Sent by the System Controller (in HEX-format)

- 4330300D

> INFO: According to the documentation of the projector the command (C00<CR>) has to be sent in hexadecimal format.

Explanation

- **433030:** The IP address of the Smart TPS Receiver.
- **0D:** The hexadecimal code of the **Carriage return** (requirement of the Projector)
- **192.168.0.40:8001: The IP address** of the **Receiver** and the virtual **TCP port number** of the serial port. These are not shown in the above command since the command format and the addressing depend on the **System Controller**.

The command Sent by the System Controller (in ASCII-format)

- C00\r

> INFO: The ASCII format of the command is mentioned just to demonstrate another messaging option.

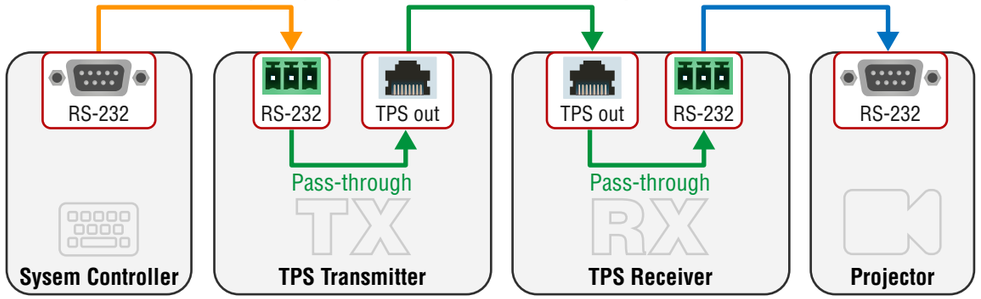
Explanation

- **C00:** The **"Power on"** command.
- **\r: Carriage return -** necessary to close the command (Projector's requirement).
- **192.168.0.50:9715: The IP address** and the **control TPC/IP port number** of the **Projector** (Projector's requirement). These are not shown in the above command since the format and the addressing depend on the **System Controller**.

1.10. Serial-Serial Messaging

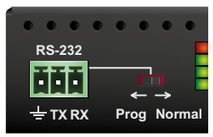
The Simple Setup – What is Happening?

The **Projector** is switched on by the proper control command coming from the **System Controller**.



The Working Method

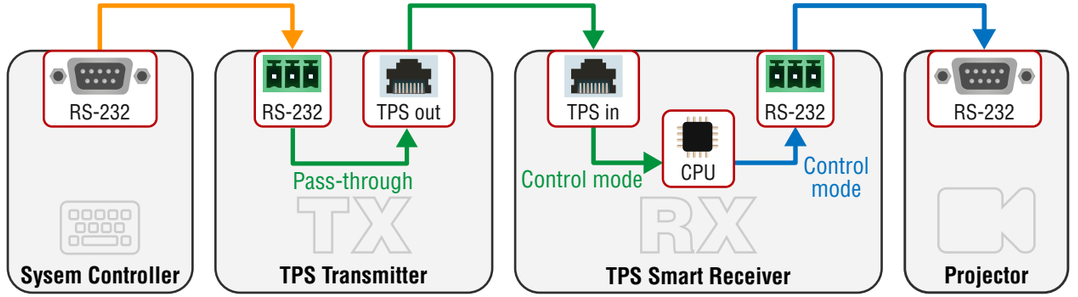
Above setup contains two Basic TPS extenders (e.g. TPS-97 series). The RS-232 ports are set to **Passthrough**, (**Normal** option of the mode selector). There is a serial communication line between the Transmitter's local RS-232 port and the Receiver's RS-232 port (green line) – think about it like a tube. The command sent by the **System Controller** is sent out over the local RS-232 port of the **Receiver** (blue line).



Preparations and Settings

- The **RS-232 port settings** of the connected serial devices (e.g. the System Controller and the Projector) must be the same.
- The **RS-232 ports** of the Extenders must be in **Normal** mode. The Basic TPS extenders (e.g. TPS-97 series) support that feature.
- Pay attention to the correct **serial cabling** (connector pinout).

The Advanced Setup



The Working Method

This kind of setup is a bit tricky since there is no direct connection between the **System Controller** and the **Projector**. There is a serial communication line between the Transmitter's Local RS-232 port and the Receiver's CPU (green line) - think about it like a tube. The command sent by the **System Controller** actually makes the **Receiver** send out the "**Power on**" command over its local RS-232 port (blue line).

Preparations and Settings

- The **RS-232 port settings** of the Transmitter's ports and the System Controller must be the same.
- The **RS-232 port settings** of the Receiver's ports and the Projector must be the same (but they do not have to match with the Transmitter side).
- The **Transmitter's** RS-232 port must be in **Pass-through** mode. The Basic TPS extenders (e.g. TPS-97 series) also support that feature.
- The **Receiver's** RS-232 port must be in **Control Mode**. The Smart TPS extenders (e.g. the TPS-100 and TPS-200 series) support that feature.
- Pay attention to the correct **serial cabling** (connector pinout).

The Command Sent by the System Controller

▸ CALL /MEDIA/UART/P1:sendMessage(C00\\r)\r\n

Explanation

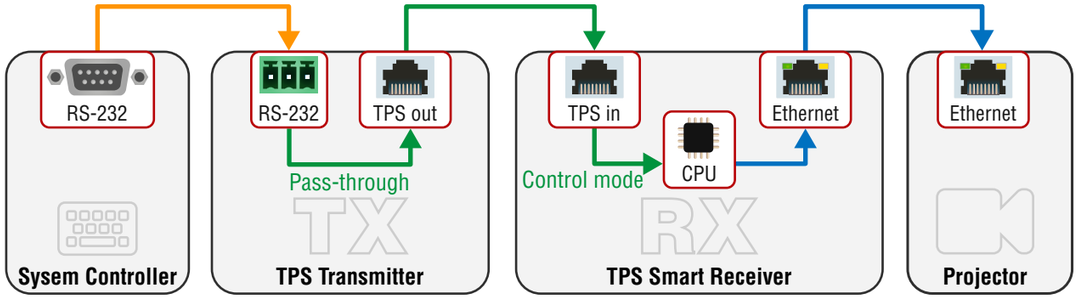
The above command is sent from the **System Controller** to the **Receiver's CPU**. The **C00\\r** string is sent from the **Receiver's CPU** to the **Projector**.

- **CALL /MEDIA/UART/P1:sendMessage:** Calling the serial message sending method.
- **P1:** the message is sent out via the **P1 (local) RS-232 port**.
- **C00:** The **"Power on"** command.
- **\\r: Carriage return -** necessary to close the command (Projector's requirement). The "\r" control character has to be escaped by the extra "\" character (LW3 requirement).
- **\r\n: Carriage return** and **Line feed** - necessary to close the LW3 command (LW3 requirement).

1.11. Serial-Ethernet Messaging

What is Happening?

The **Projector** is switched on by the proper control command coming from the **System Controller**.



The Working Method

This kind of setup is similar to the previous case. There is no direct connection between the **System Controller** and the **Projector**. There is a serial communication line between the Transmitter's Local RS-232 port and the Receiver's CPU (green line) - think about it like a tube. The command sent by the **System Controller** actually makes the **Receiver** send out the "**Power on**" command via Ethernet (blue line).

Preparations and Settings

- The **RS-232 port settings** of the Transmitter's ports and the System Controller must be the same.
- The **RS-232 port settings** of the Receiver's ports and the Projector must be the same (but they do not have to match with the Transmitter side).
- The **Transmitter's** RS-232 port must be in **Pass-through** mode. The Basic TPS extenders (e.g. TPS-97 series) also support that feature.
- The **Receiver's** RS-232 port must be in **Control Mode**. The Smart TPS extenders (e.g. the TPS-100 and TPS-200 series) support that feature.
- The **local Ethernet** must be enabled in the **Receiver**.
- Pay attention to the correct **serial cabling** (connector pinout).

The Command Sent by the System Controller

- CALL /MEDIA/ETHERNET:tcpMessage(192.168.0.50:9715=C00\\r)\r\n

Explanation

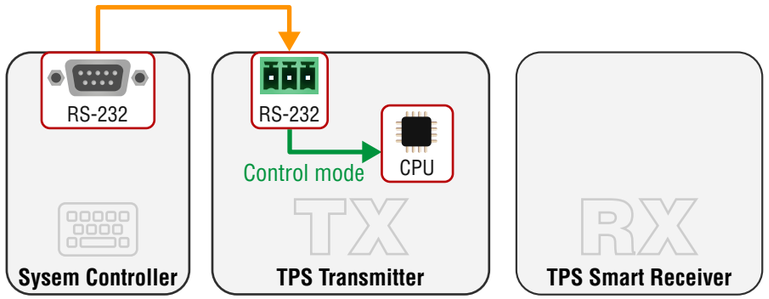
The above command is sent from the **System Controller** to the **Receiver's CPU**. The **C00\\r** string is sent from the **Receiver's CPU** to the **Projector**.

- **CALL /MEDIA/ETHERNET:tcpMessage:** Calling the TCP message sending method.
- **192.168.0.50:** The IP address of the Projector.
- **9715:** The port number (Projector's requirement)
- **C00:** The **"Power on"** command.
- **\\r: Carriage return -** necessary to close the command (Projector's requirement). The "\r" control character has to be escaped by the extra "\" character (LW3 requirement).
- **\r\n: Carriage return** and **Line feed** - necessary to close the LW3 command (LW3 requirement).

1.12. Device Control over the Serial Port

What is Happening?

Local control – the **Transmitter** is controlled by the **System Controller.**



The Working Method

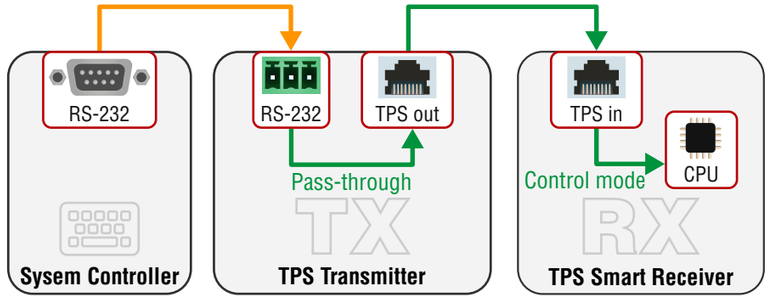
There is a simple RS-232 serial connection between the System Controller and the Transmitter. The commands sent by the System Controller are processed by the Transmitter.

Preparations and Settings

- This control feature is not supported by the **TPS Basic Extenders** (e.g. TPS-97 series).
- The **RS-232 port settings** of the Transmitter's ports and the System Controller must be the same.
- The **Transmitter's** RS-232 port must be in **Control** mode.
- Pay attention to the correct **serial cabling** (connector pinout).
- Pay attention to the current **protocol setting** of the serial port: **LW2** or **LW3**. LW3 commands are interpreted only when the current setting is **LW3.**

What is Happening?

Remote control – the **Receiver** is controlled by the **System Controller**.



The Working Method

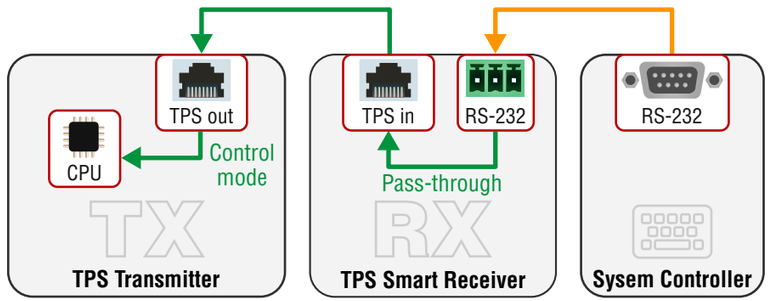
There is a serial communication line between the Transmitter's Local RS-232 port and the Receiver's CPU (green line) – think about it like a tube. This way, the **System Controller** can control directly the **Receiver** by sending commands.

Preparations and Settings

- The **RS-232 port settings** of the Transmitter's ports and the System Controller must be the same.
- The **Transmitter's** RS-232 port must be in **Pass-through** mode. The Basic TPS extenders (e.g. TPS-97 series) also support that feature.
- The **Receiver's** RS-232 port must be in **Control Mode**. The Smart TPS extenders (e.g. the TPS-100 and TPS-200 series) support that feature.
- Pay attention to the correct **serial cabling** (connector pinout).
- Pay attention to the current **protocol setting** of the serial port: **LW2** or **LW3**. LW3 commands are interpreted only when the current setting is **LW3.**

The Opposite Direction

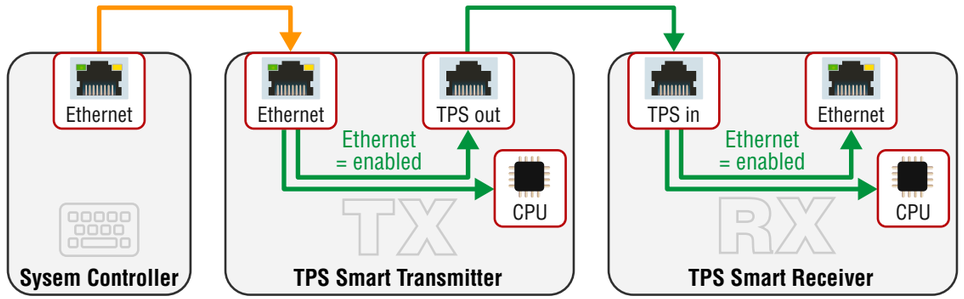
The layout is almost the same as above but in this case, the **Controller** is connected to the **Receiver**. The **Receiver's RS-232 port** must be in **Pass-through mode** and the **Transmitter's RS-232 port** must be in **Control mode**. This way, the **Transmitter** can be controlled by the **System Controller**.



1.13. Device Control over Ethernet (via TPS)

What is Happening?

The **Transmitter** and the **Receiver** can be controlled by the **System Controller**.



The Working Method

The devices are connected via Ethernet. This way, the **System Controller** can control directly the **Transmitter** and the **Receiver**.

Preparations and Settings

- The devices (System Controller, Transmitter, Receiver) must be in the **same subnet** with configured and known IP address.
- The Ethernet has to be **enabled** at the TPS and Ethernet ports. This feature is also supported by the Basic TPS extenders (e.g.TPS-97 series).
- Please note that only **Smart TPS Extenders** (e.g. TPS-100 or TPS-200 series) can be controlled, **Basic TPS Extenders** (e.g. TPS-97 series) support only pass-through Ethernet transmission. E.g. if the above setup contains a **Basic TPS Transmitter** and a **Smart TPS Receiver** only the **Receiver** could be controlled over Ethernet.

Controlling

The smart extenders accept:

- **LW2 commands** over TCP/IP port no. **10001**.
- **LW3 commands** over TCP/IP port no. **6107**.

The command must be closed by Carriage Return and Line Feed.

> INFO: For the supported commands please check the corresponding Programmer's Reference chapters in the User's Manual of the device.