

RS232 Syntax from ZyPer Management Platform



Intelligent AV Distribution

ZeeVee, Inc.
295 Foster Street, Suite 200
Littleton, MA 01460 USA
April 19, 2022

Table of Contents

Introduction.....	3
RS232.....	3

Introduction

It is possible to send an RS232 text string from the ZyPer Management Platform to any connected ZyPer4K or ZyPerUHD endpoint on the network. (Encoder or Decoder) This function can be done from the ZMP command line interface or via a 3rd party control system. Details on syntax are shown below:

RS232

The Management Platform (ZMP) must first be linked to the specific endpoint to send RS232 information. This can be done with either the dataConnect or switch command.

Examples:

```
dataConnect Dec1 server rs232 tunnelPort 1234
```

```
switch Dec1 server rs232
```

Note: The feature of dataConnect was added to allow a third party to connect to the ZMP server with a specific port and pass raw or telnet API commands (depending on the mode) to the server and port which is designated for a particular encoder or decoder.

When using any control system; that system is actually talking to our MP and not to any specific endpoint.

When sending RS232 commands to an encoder or decoder via the MP you must follow very specific syntax.

The ZeeVee command is: send <decoder_name> rs232 text

Here are examples on this. (Assume decoder name is *Dec1*)

Input command: send Dec1 rs232 Hello

Received at Dec1: Hello (Note, no line feed or carriage return)

Input command: send Dec1 rs232 Hello\r\n

Received at Dec1: Hello (with carriage return and line feed)

Input command: send Dec1 rs232 Hello World

Received at Dec1: Nothing. You get an error. Bad syntax. You cannot have a space between hello and world.

Input command: send Dec1 rs232 Hello_World

Received at Dec1: Hello_World (Note, no line feed or carriage return)

Input command: send Dec1 rs232 "Hello World"

Received at Dec1: Hello World (Note, no line feed or carriage return)

Input command: send Dec1 rs232 "Hello World"\r\n

Received at Dec1: Nothing. You get an error. Bad syntax. Token \r\n is invalid.

You need to contain the line feed and carriage return symbols inside the quotes in this case.

Input command: send Dec1 rs232 "Hello World\r\n"

Received at Dec1: Hello World (with carriage return and line feed)

Text can also be Hexadecimal Code as shown below:

Input command:

```
send Dec1 rs232 \x48\x65\x6c\x6c\x6f\x20\x57\x6f\x72\x6c\x64\x0A\x0D
```

Received at Dec1: Hello World (with carriage return and line feed)

The Management Platform also has the ability to receive RS232 communications that were input into a ZyPer endpoint. To view any such RS232 string you use the "show responses" command.

Example:

```
Zyper$ show responses DEC1 rs232 since 0
```

```
device(d8:80:39:59:bf:57);
```

```
  device.rs232Response.0; string="Have a great day!\x0D"
```

```
  device.rs232Response.1; string="\x0A"
```

```
lastChangeId(2);
```

```
Success
```

Note that the ZyPer Management Platform can only show the user the responses. It has no means to act on any RS232 string it receives. This would require a 3rd party control system.

Important Note

When using Crestron as the control system, you need to append an extra \ symbol before the Carriage return symbol. Otherwise carriage return does not work.

Example using Crestron to turn on/off LG display.

LG TV

ON

```
send DecoderName rs232 \x6B\x61\x20\x30\x31\x20\x30\x31\\x0D
```

OFF

```
send DecoderName rs232 \x6B\x61\x20\x30\x30\x20\x30\x30\\x0D
```