

## RS-232 Switch



## TABLE OF CONTENTS

<b>Introduction</b> .....	<b>2</b>
RS-232 Switch .....	2
<b>Installation</b> .....	<b>3</b>
Before Installation .....	3
Connect the Controlling Devices .....	3
Connect the Receiving Devices .....	3
Power ON the RS-232 Switch .....	3
<b>Operation</b> .....	<b>4</b>
Operation .....	4
Cascading Multiple Switches .....	4
<b>Appendix</b> .....	<b>5</b>

## INTRODUCTION

### RS-232 Switch

The RS-232 Switch allows control of multiple RS-232 devices from a single or dual control point, acting as a gateway for all the connected devices.

This guide provides instructions on how to install and operate an RS-232 Switch.



## INSTALLATION

### Before Installation

Ensure all devices are powered OFF before installing.

#### Connect the Controlling Devices

- Connect the remote control unit to the RS-232 Switch via COM Port 1.
- If using a secondary remote control unit, connect it to the RS-232 Switch via COM Port 2.

**Note:** It is recommended that COM Port 1 be used for a remote control connected to a KVM Switch or other device that sends keepalive information, and COM Port 2 for a remote control connected to devices that do not send keepalive information..

#### Connect the Receiving Devices

- If using a KVM Switch or another KVM device such as a Combiner or Mini-Matrix, connect it to the RS-232 Switch via the dedicated COM Port 8.

**IMPORTANT NOTE:** COM Port 8 transmits raw keepalive information as is to COM Port 1 with no filters. This is to allow a KVM Switch to send keepalive information to the remote control.

- Connect up to 6 compatible devices to the RS-232 Switch via COM Ports 2-7.

**Note:** COM Ports 3, 4, and 5 are set to a baud rate of 9600..

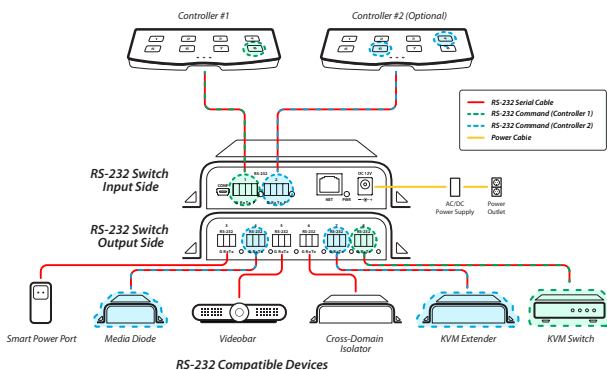
#### Power ON the RS-232 Switch

- Power ON the RS-232 devices.
- Power ON the RS-232 Switch by connecting a power supply to the 12V DC port.

## OPERATION

## Operation

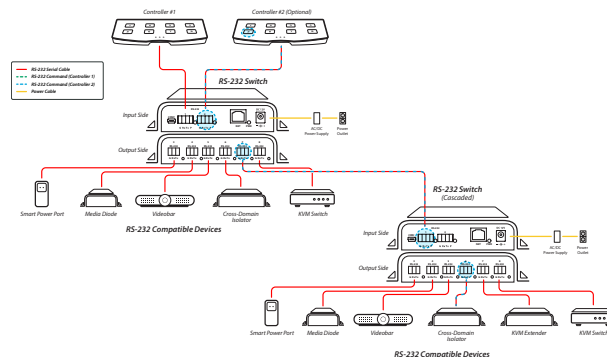
The RS-232 Switch receives commands from the connected controller(s), then sends them to the selected receiving devices via a dedicated cable for each device. These commands can be sent from any RS-232 compatible device, such as an HSL Remote Control Unit, and can be sent to one, multiple, or all the connected devices (including the sending device itself).



**Figure 1:** Controller 1 sends a command to devices connected to Ports 3 and 5; Controller 2 sends a command to devices connected to Ports 4 and 7.

## Cascading Multiple Switches

The RS-232 Switch can also cascade to other Switches, then send commands to devices connected to the cascaded Switches.



**Figure 2:** Controller 1 sends a command to devices on Ports 4 and 5 of a Switch cascaded to Port 8.

## APPENDIX

## A: RS-232 Command Structure:

The RS-232 Switch can send commands from one device to any or all other connected devices, including the sending device itself. This requires a specific command structure to address and route commands properly. The command structure is as follows:

### \$1<Header><Message data>\$D\$2

- **Start <\$1>**: Marks the beginning of the command
- **Header**: Defines the command's attributes and destination channels
- **Message Data**: The command itself
- **Carriage Return <\$D>**: Marks the end of the Message Data
- **Tail <\$2>**: Marks the end of the entire command

### Byte Stuffing:

The RS-232 interface uses <\$> as the control character to begin and end commands. When creating an RS-232 command, the header cannot contain <\$>, and any instances of <\$> in the message data must be written instead as <\$3>. For example, a command with the characters <C2\$3\$d> must be written as <C2\$3\$3d>.

## Header Structure:

### Message Attributes:

The command can be sent directly to a connected device or retransmitted through another connected RS-232 Switch.

- **<space><n>**: Transmit the command directly to a connected device. <n>= the destination channel(s).
- **R<n><space>**: Retransmit the command to another RS-232 Switch. <n>= the destination channel(s) of the cascaded Switch(es).

**Note:** Commands can be retransmitted through multiple levels of Switches that are cascaded together, but a command can only be sent to devices that are on the same level of cascaded. For example, a command cannot be sent to a device directly connected to the Switch and also retransmitted through a cascaded Switch to a different device.

### Destination Channels:

The destination channel(s) are represented as bits in an 8-bit byte, then entered in hexadecimal.

(See Appendix B for complete instructions on finding the values for destination channels)

- Example: Channels 3 and 5 = hex 14.

## APPENDIX

## Header/Tail Stripping

When a command is sent into the RS-232 Switch, it strips the header and tail before sending it to the destination device. If multiple Switches are cascaded together, each Switch only strips the first header in the command.

### Examples:

- **Direct Connection:** To send the command **GO** to devices directly connected to Channels 3 and 5:

**\$1 14GO\$D\$2**

- The Switch strips the header **<\$1 14>** and tail **<\$2>** from the command before sending it to the connected devices, so the devices receive the message **GO**.

- **Cascaded Switch:** To retransmit the command **GO** to devices connected to Channels 3 and 5 of a cascaded Switch (Switch 2), which is connected to Channel 4 of the transmitting Switch (Switch 1):

**\$1R08 14GO\$D\$2**

- Switch 1 strips the header **<R08>** from the command before sending it to Switch 2, so Switch 2 receives the command **\$1 14GO\$D\$2**.
- Switch 2 strips the header **<\$1 14>** and the tail **<\$2>** from the command before sending it to the connected devices, so the devices receive the message **GO**.

- **Multiple Levels of Cascading:** To retransmit the command **GO** to devices connected to Channels 3 and 5 of a cascaded Switch (Switch 3), which is connected to Channel 4 of a cascaded Switch (Switch 2), which is connected to Channel 1 of the transmitting Switch (Switch 1):  
**\$1R01R08 14GO\$D\$2**

- Switch 1 strips the header **<R01>** from the command before sending it to Switch 2, so Switch 2 receives the command **\$R08 14GO\$D\$2**.
- Switch 2 strips the header **<R08>** from the command before sending it to Switch 3, so Switch 3 receives the command **\$1 14GO\$D\$2**.
- Switch 3 strips the header **<\$1 14>** and the tail **<\$2>** from the command before sending it to the connected devices, so the devices receive the message **GO**.

## APPENDIX

**B: Determining Destination Channel Values**

When creating the Header for an RS-232 command, the destination channel is expressed in a 2-digit hexadecimal value.

To find this value, perform the following:

**Step 1 – Express the Destination Channels as an 8-Digit Byte**

Read the channels from right to left, with Channel 1 being the bit furthest to the right and Channel 8 being the bit furthest to the left, expressing them as either “1” for an active channel or “0” for an inactive channel. All 8 channels are represented together as an 8-bit byte.

**Example:** For a command sent to Channels 3 and 5, the 3rd and 5th bit in the byte would be active (1), and all other bits would be inactive (0). Therefore, expressing the values from right to left, the 8-bit byte would be **00010100**.

**Step 2 – Convert the Byte to Decimal**

Reading from right to left, count each active channel in the 8-bit byte by converting each digit to an exponent of 2, with the first digit on the right being 1 (2 to the 0 power) and the leftmost digit being 128 (2 to the 7th power). Once these digits are converted, add them together.

**Example:** For a command sent to Channels 3 and 5, the 8-bit byte **00010100** has a 1 on the 3rd and 5th digits from right to left. The decimal value of the 3rd digit is 4 (2 to the 2nd), and the decimal value of the 5th digit is 16 (2 to the 4th). Therefore, the sum of the active channels is 16+4, which equals **20**.

**Step 3 – Convert the Decimal Value to Hexadecimal**

Converting from decimal to hexadecimal is done most easily with a calculator, but the decimal and hex values for each individual channel are as follows:

Channel	Byte	Decimal	Hex Expression
1	00000001	1	1
2	00000010	2	2
3	00000100	4	4
4	00001000	8	8
5	00010000	16	10
6	00100000	32	20
7	01000000	64	40
8	10000000	128	80

**Example:** For a command sent to Channels 3 and 5, the 8-bit byte 00010100 converts to decimal as 20. This converts to hexadecimal as 1 increment of 16 and 4 increments of 1; therefore, the value is expressed in hexadecimal as 14.

Following the syntax described in Appendix A, the complete structure to send the command “GO” to Channels 3 and 5 would be:

**\$1 14GO\$A\$D\$2**

## APPENDIX

### C: Cascading Controllers

In addition to being able to send commands through cascaded RS-232 Switchers, the Switcher can also receive and transmit commands sent through cascaded controllers.

In order to send a command through a cascaded controller, the command must end with **\r** to perform a carriage return.

**Example:** To send the command **GO** from a controller, cascaded through a different controller, to Channels 3 and 5 of the RS-232 Switcher, a carriage return must be added to the end of the command structure. Following the syntax described in Appendix A, the complete structure would be:

**\$1 14GO\$A\$D\$2\r**

For complete details on sending commands through an HSL Programmable Remote Control Unit, consult the User Manual, which can be found here:

[https://highseclabs.com/wp-content/uploads/2026/01/HLT36829\\_HSL\\_UM\\_Programmable\\_RC\\_Rev1.1.pdf](https://highseclabs.com/wp-content/uploads/2026/01/HLT36829_HSL_UM_Programmable_RC_Rev1.1.pdf)

# Highseclabs.com

For more information about HSL's solutions, please contact:

**HighSecLabs Inc.**

905 James Record Road STE A,  
Huntsville AL, 35824

**HSL Support**

256-203-3036  
<https://highseclabs.com/contact/>

**Sales**

[Sales@highseclabs.com](mailto:Sales@highseclabs.com)

©2025 All rights reserved. HSL logo and product names are trademarks or service trademarks of HighSecLabs Ltd (HSL). All other marks are the property of their respective owners. Images for demonstration purposes only. This document may contain confidential and/or proprietary information of HSL Corporation, and its receipt or possession does not convey any right to reproduce, disclose its contents, or to manufacture or sell anything that it may describe. Reproduction, disclosure, or use without specific authorization from HSL Corporation is strictly prohibited.