

INSTRUCTION MANUAL

CE-Series Universal Control Extenders

CE-IRS4, CE-IO4, CE-COM2, CE-REL8



IMPORTANT SAFETY INSTRUCTIONS

1. READ these instructions.
2. KEEP these instructions.
3. HEED all warnings.
4. FOLLOW all instructions.
5. DO NOT use this apparatus near water.
6. CLEAN ONLY with dry cloth.
7. DO NOT block any ventilation openings. Install in accordance with the manufacturer's instructions.
8. DO NOT install near any heat sources such as radiators, heat registers, stoves, or other apparatus (including amplifiers) that produce heat.
9. DO NOT defeat the safety purpose of the polarized or grounding type plug. A polarized plug has two blades with one wider than the other. A grounding type plug has two blades and a third grounding prong. The wider blade or the third prong are provided for your safety. If the provided plug does not fit into your outlet, consult an electrician for replacement of the obsolete outlet.
10. PROTECT the power cord from being walked on or pinched, particularly at plugs, convenience receptacles, and the point where they exit from the apparatus.
11. ONLY USE attachments/accessories specified by the manufacturer.



12. USE ONLY with a cart, stand, tripod, bracket, or table specified by the manufacturer, or sold with the apparatus. When a cart is used, use caution when moving the cart/apparatus combination to avoid injury from tip-over.
13. UNPLUG this apparatus during lightning storms or when unused for long periods of time.
14. REFER all servicing to qualified service personnel. Servicing is required when the apparatus has been damaged in any way, such as power-supply cord or plug is damaged, liquid has been spilled or objects have fallen into the apparatus, the apparatus has been exposed to rain or moisture, does not operate normally, or has been dropped.
15. DO NOT expose this apparatus to dripping or splashing and ensure that no objects filled with liquids, such as vases, are placed on the apparatus.
16. To completely disconnect this apparatus from the AC Mains, disconnect the power supply cord plug from the AC receptacle.
17. Where the mains plug or an appliance coupler is used as the disconnect device, the disconnect device shall remain readily operable.
18. DO NOT overload wall outlets or extension cords beyond their rated capacity as this can cause electric shock or fire.
19. Please use the HDMI cable with magnetic ring.



The exclamation point, within an equilateral triangle, is intended to alert the user to the presence of important operating and maintenance (servicing) instructions in the literature accompanying the product.



The lightning flash with arrowhead symbol within an equilateral triangle is intended to alert the user to the presence of uninsulated "dangerous voltage" within the product's enclosure that may be of sufficient magnitude to constitute a risk of electrical shock to persons.



ESD Warning: The icon to the left indicates text regarding potential danger associated with the discharge of static electricity from an outside source (such as human hands) into an integrated circuit, often resulting in damage to the circuit.

WARNING: To reduce the risk of fire or electrical shock, do not expose this apparatus to rain or moisture.

WARNING: No naked flame sources - such as candles - should be placed on the product.

WARNING: Equipment shall be connected to a MAINS socket outlet with a protective earthing connection.

WARNING: To reduce the risk of electric shock, grounding of the center pin of this plug must be maintained.

COPYRIGHT NOTICE

AMX© 2023, all rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of AMX. Copyright protection claimed extends to AMX hardware and software and includes all forms and matters copyrightable material and information now allowed by statutory or judicial law or herein after granted, including without limitation, material generated from the software programs which are displayed on the screen such as icons, screen display looks, etc. Reproduction or disassembly of embodied computer programs or algorithms is expressly prohibited.


LIABILITY NOTICE


No patent liability is assumed with respect to the use of information contained herein. While every precaution has been taken in the preparation of this publication, AMX assumes no responsibility for error or omissions. No liability is assumed for damages resulting from the use of the information contained herein. Further, this publication and features described herein are subject to change without notice.

AMX WARRANTY AND RETURN POLICY

The AMX Warranty and Return Policy and related documents can be viewed/downloaded at www.amx.com.

ESD WARNING

	<p>To avoid ESD (Electrostatic Discharge) damage to sensitive components, make sure you are properly grounded before touching any internal materials.</p> <p>When working with any equipment manufactured with electronic devices, proper ESD grounding procedures must be followed to make sure people, products, and tools are as free of static charges as possible. Grounding straps, conductive smocks, and conductive work mats are specifically designed for this purpose. These items should not be manufactured locally, since they are generally composed of highly resistive conductive materials to safely drain static discharges, without increasing an electrocution risk in the event of an accident.</p> <p>Anyone performing field maintenance on AMX equipment should use an appropriate ESD field service kit complete with at least a dissipative work mat with a ground cord and a UL listed adjustable wrist strap with another ground cord.</p>
---	---

	<p>WARNING: Do Not Open! Risk of Electrical Shock. Voltages in this equipment are hazardous to life. No user-serviceable parts inside. Refer all servicing to qualified service personnel.</p> <p>Place the equipment near a main power supply outlet and make sure that you can easily access the power breaker switch.</p>
---	---

WARNING:

This product is intended to be operated ONLY from the voltages listed on the back panel or the recommended, or included, power supply of the product. Operation from other voltages other than those indicated may cause irreversible damage to the product and void the product's warranty. The use of AC Plug Adapters is cautioned because it

can allow the product to be plugged into voltages in which the product was not designed to operate. If the product is equipped with a detachable power cord, use only the type provided with your product or by your local distributor and/or retailer. If you are unsure of the correct operational voltage, please contact your local distributor and/or retailer.

FCC AND CANADA EMC COMPLIANCE INFORMATION:

This device complies with part 15 of the FCC Rules.

Operation is subject to the following two conditions:

(1) This device may not cause harmful interference, and (2) this device must accept any interference received, including interference that may cause undesired operation.

NOTE: This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a commercial environment. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/TV technician for help.

Approved under the verification provision of FCC Part 15 as a Class A Digital Device.


Caution:

Changes or modifications not expressly approved by the manufacturer could void the user's authority to operate this device. CAN ICES-3 (B)/NMB-3 (B)

EU COMPLIANCE INFORMATION:

Eligible to bear the CE mark; Conforms to European Union Low Voltage Directive 2006/95/EC; European Union EMC Directive 2004/108/EC; European Union Restriction of Hazardous Substances Recast (RoHS2) Directive 2011/65/EU; European Union WEEE (recast) Directive 2012/19/EU; European Union Radio and Telecommunications Terminal Equipment (R&TTE) Directive 1999/5/EC

WEEE NOTICE:

	<p>This appliance is labeled in accordance with European Directive 2012/19/EU concerning waste of electrical and electronic equipment (WEEE). This label indicates that this product should not be disposed of with household waste. It should be deposited at an appropriate facility to enable recovery and recycling.</p>
---	--

Contents

IMPORTANT SAFETY INSTRUCTIONS	2
COPYRIGHT NOTICE	2
LIABILITY NOTICE	2
AMX WARRANTY AND RETURN POLICY	2
FCC AND CANADA EMC COMPLIANCE INFORMATION:	3
EU COMPLIANCE INFORMATION:	3
WEEE NOTICE:	3
CE-SERIES UNIVERSAL CONTROL EXTENDERS	6
OVERVIEW	6
COMMON FEATURES.....	6
LAN/POE PORT.....	7
DETAILED LED BEHAVIOR	7
LED PATTERNS.....	7
ID BUTTON	7
INSTALLATION	8
MOUNTING OPTIONS (RACK TRAYS AND MOUNTING BRACKETS).....	8
VSTYLE DIN RAIL CLIP (AMX-CAC0001)	8
AVB-VSTYLE-SURFACE-MNT (FG1010-722).....	8
AVB-VSTYLE-RMK (FG1010-720).....	8
AVB-VSTYLE-RMK-FILL-1U (FG1010-721).....	9
AVB-VSTYLE-POLE-MNT (FG1010-723)	9
SAFETY INSTRUCTIONS	10
PREPARING/CONNECTING CAPTIVE WIRES	10
LAN/POE PORT.....	10
NETWORK CONFIGURATION.....	11
DEFAULTS	11
NETWORK DISCOVERY	11
MANAGER DESKTOP	11
WEB SERVER CONTROLS	13
<i>Network</i>	13
<i>Device</i>	14
<i>Security</i>	23
<i>Administrator</i>	24
<i>Firmware</i>	25
CE-SERIES HCONTROL PROGRAMMING	26
INTRODUCTION	26
GETTING STARTED	26
BASIC SYNTAX AND PORTS.....	27
0-BASED VS 1-BASED PORT NUMBERING.....	28
PARAMETERS.....	28
COMMANDS.....	30
EVENTS.....	30

DEVICE SPECIFIC HCONTROL DETAILS	30
<i>CE-REL8</i>	30
<i>Parameters</i>	31
<i>CE-COM2</i>	31
<i>CE-IRS4</i>	33
<i>CE-IO4</i>	35
CE-SERIES 3RD-PARTY CONTROL PROTOCOL	37
OVERVIEW	37
TCP/IP CONNECTION INFORMATION	37
MESSAGE FORMAT	37
<i>Message IDs</i>	37
<i>Paths</i>	38
<i>Values / Arguments</i>	38
DEVICE SPECIFIC CONTROLS.....	39
<i>CE-REL8 Controls</i>	39
<i>CE-COM2 Controls</i>	39
<i>CE-IRS4 Controls</i>	41
<i>CE-IO4 Controls</i>	43

CE-Series Universal Control Extenders

Overview

The CE-Series of Universal Control Extenders provide port expansion over Ethernet for any control system. The HControl protocol provides a simple path and value approach to actuating the device functions. The use of the CE-Series Control Extender with other HControl-aware HARMAN products makes a seamless extension of those systems. The Ethernet connectivity helps break distance limitations and provide flexibility of installation.

CE-Series Control Extender boxes			
Name	SKU	Description	Page Reference
CE-IRS4	AMX-CEB001	Control Extender – 4 IR	
CE-REL8	AMX-CEB002	Control Extender – 8 Relay	
CE-COM2	AMX-CEB003	Control Extender – 2 COM	
CE-IO4	AMX-CEB004	Control Extender – 4 IO	

Common Features

CE-Series Common Features		
DIMENSIONS	1" x 4 1/16" x 4 3/32" 2.5 cm x 10.3 cm x 10.4 cm (1/4 RU Width x <1 RU Height)	
WEIGHT	1lb 454 g	
POWER REQUIREMENTS	PoE – 802.3af Class 0 Maximum	15.4W
	CE-IRS4	2.6W
	CE-REL8	4.1W
	CE-COM2	5.6W
	CE-IO4	14.7W
Enclosure	Powder-coated steel w/ grey matt finish	
Certifications	FCC Part 15 Class B, CE, and IEC 60950	
Environmental Requirements	<i>Operating Temperature:</i> 0° C (32° F) to 40° C (104° F) <i>Storage Temperature:</i> -10° C (14° F) to 60° C (140° F) <i>Operating Humidity:</i> 5% to 85% RH (non-condensing) <i>Heat Dissipation (Typical):</i> 36.9 BTU/hr Designed for indoor use only.	
FRONT PANEL COMPONENTS		
ID Pushbutton		
Status LED	A multi-colored LED indicating the device status (see fig ???)	
L/A LED	A network link/activity light	
REAR PANEL COMPONENTS		
Ethernet	RJ-45 connector provides IP communication and PoE	
Phoenix Connector	Module-specific captive wire connectors (See below)	

LAN/PoE Port

The LAN/PoE (RJ45) port on all CE Modules provides 10/100 Base-T network connectivity. Use standard Cat5/6/6E ethernet cable to connect the CE Module to the network.

The Ethernet Port LEDs show communication activity, connection status, speeds, and mode information:

- SPD (speed) - Yellow LED lights On when the connection speed is 100 Mbps and turns Off when the speed is 10 Mbps.
- L/A (link/activity) - Green LED lights On when the Ethernet cables are connected and terminated correctly, and blinks when receiving Ethernet data packets.

Detailed LED Behavior

The table below provides detailed descriptions of all blink patterns for the Status and A/L LEDs on the front panels of all CE modules.

LED PATTERNS

The CE-series features one tri-color LED for status.

Color	Frequency	Status
Green	Solid	Power, no connection
Green	1 Hz	HControl connection
Yellow	Solid	Booting
Cyan	3 Hz	Updating*
White	Solid	Locate mode
Red	1 Hz	Error, not running

* - Please do not disconnect from power when updating

ID Button

The ID button located on the front panel of the CE-Series module has two main functions: Reset and Locate.

Time	Function	Description
<10 seconds	Locate	An HControl Locate message is broadcast
10 – 20 seconds	Reset Configuration	The username, password, and relevant device settings are set back to defaults
>20 seconds	Revert to Factory Firmware	The CE module is restored to the factory firmware image with all default settings.

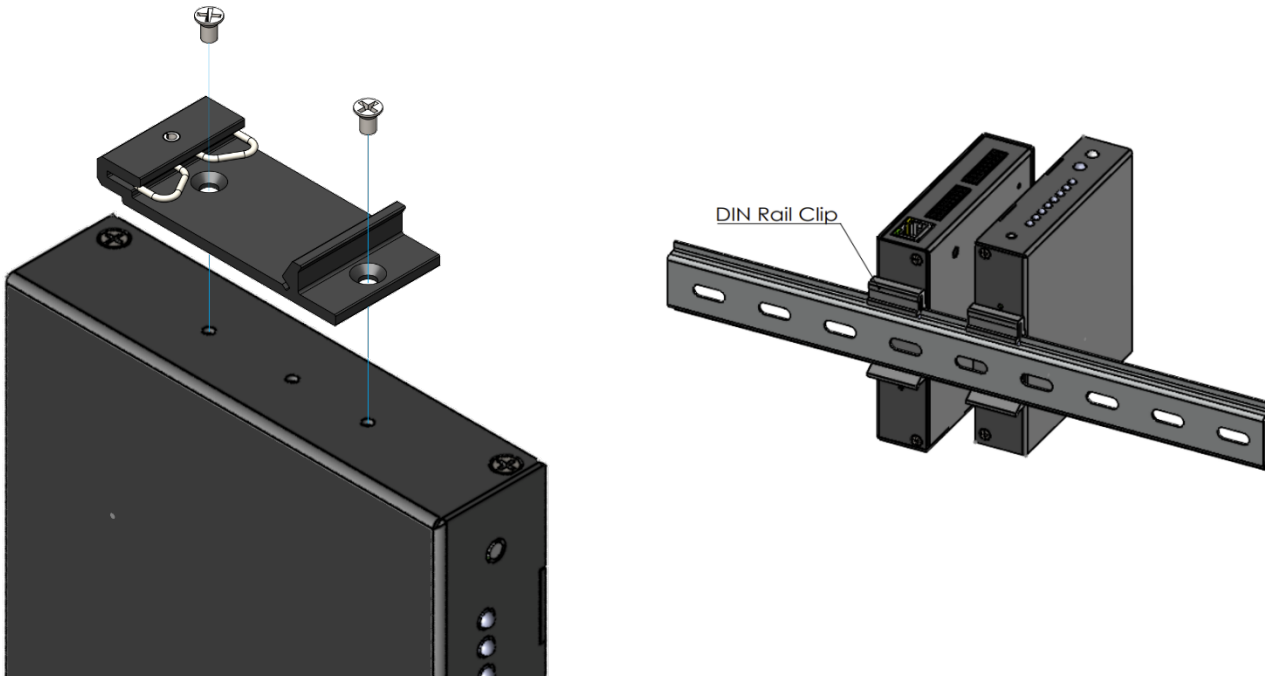
Installation

Mounting Options (Rack Trays and Mounting Brackets)

Refer to the documentation provided with each of the mounting kits for installation instructions (also available online at www.amx.com).

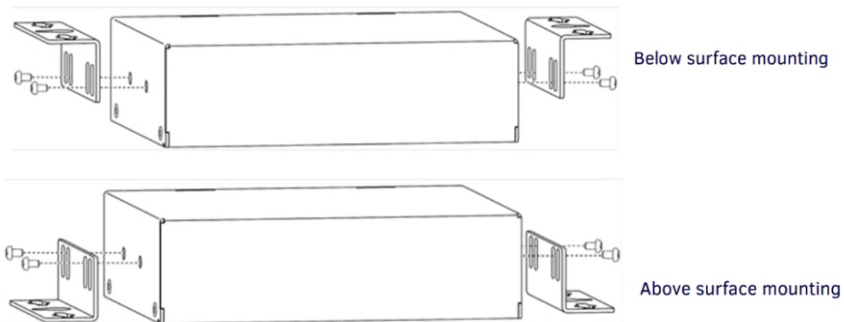
VSTYLE DIN Rail Clip (AMX-CAC0001)

Small VSTYLE compatible modules can be mounted on a DIN rail with the VSTYLE DIN Rail Clip



AVB-VSTYLE-SURFACE-MNT (FG1010-722)

V Style Single Module Surface Mount Brackets - Mount a single module to the wall, under a desk, etc. (FIG. 2).

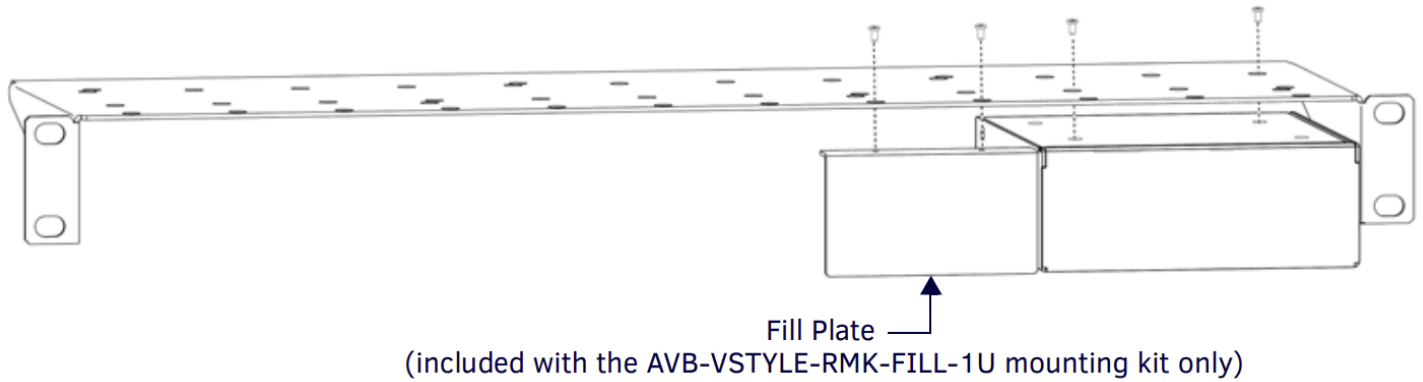


AVB-VSTYLE-RMK (FG1010-720)

V Style Module Tray Rack - Mount up to four modules side by side in a 1 RU space (FIG. 3).

AVB-VSTYLE-RMK-FILL-1U (FG1010-721)

V Style Module Tray Rack - Mount up to four modules side by side in a 1 RU space, with Fill Plates for instances when the entire tray is not full (FIG. 3).



Note that the tray illustrated in FIG. 3 is shown upside down for ease of installation. However, the tray can be rack-mounted in a rack either way.

Also note that the Fill Plate included in this illustration is included only in the AVB-VSTYLE-RMK-FILL-1U (FG1010-721).

NOTE: The AVB-VSTYLE-RMK and AVB-VSTYLE-RMK-FILL-1U mounting kits are appropriate for the CE-COM2, -IRS4, -IO8 and -REL8

AVB-VSTYLE-POLE-MNT (FG1010-723)

V Style Single Module Pole Mounting Kit - Suspend a single module to a pole (FIG. 4).

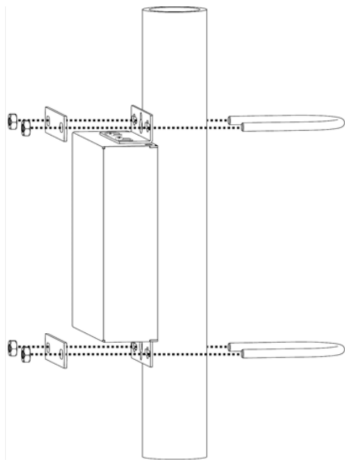


Fig xxx – AVB-VSTYLE-POLE-MNT

SAFETY INSTRUCTIONS

- For UL compliance, the CE family of ICSLan Device Control Boxes should be powered directly via any listed external IEC/UL 60950-1 2nd edition certified LPS PoE switch or injector, such as the AMX NXA-ENET8POE or PS-POE-AF.
- The CE-series family of ICSLan Device Control Boxes are intended for Network Environment 0 per IEC TR62101, and are to be connected only to PoE networks without routing to the outside plant.

Preparing/Connecting Captive Wires

1. Strip 0.25 inch of wire insulation off all wires.
2. Insert each wire into the appropriate opening on the connector according to the wiring diagrams and connector types described in this section.

Note: Do not tighten the screws excessively; doing so may strip the threads and damage the connector.

LAN/PoE Port

The LAN/PoE (RJ45) port on all CE Modules provides 10/100 BaseT network connectivity. Use standard Cat5/6/6E ethernet cable to connect the CE Module to the LAN. The Ethernet Port LEDs show communication activity, connection status, speeds, and mode information:

- SPD (speed) - Yellow LED lights On when the connection speed is 100 Mbps and turns Off when the speed is 10 Mbps.
- L/A (link/activity) - Green LED lights On when the Ethernet cables are connected and terminated correctly, and blinks when receiving Ethernet data packets.

Network Configuration

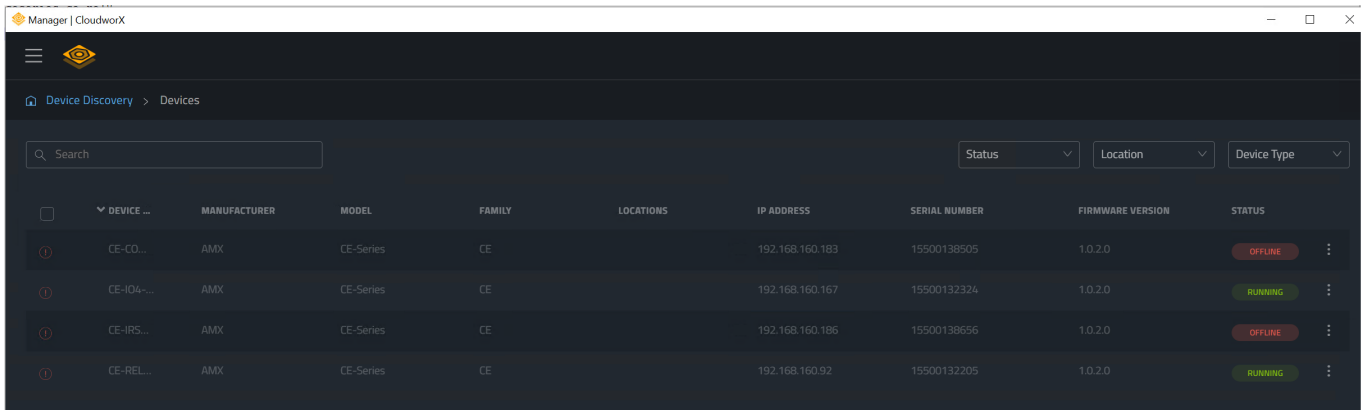
Defaults

The CE- family of Control Extenders defaults to using DHCP to obtain a network address. If a DHCP server is unavailable, the CE will use a link-local address.

Network Discovery

The CE family of Control Extenders use the HARMAN HControl Protocol for communication and network discovery. HControl aware software and devices are able to detect other HControl devices on their subnet. Software such as Manager can be used to discover the IP address that has been assigned via DHCP.

Manager Desktop



The screenshot shows the Manager Desktop interface with a table of discovered devices. The table has columns for Device ID, Device Name, Manufacturer, Model, Family, Locations, IP Address, Serial Number, Firmware Version, and Status. There are four rows of data, each with a status button (OFFLINE or RUNNING) and a menu icon.

DEVICE ID	DEVICE NAME	MANUFACTURER	MODEL	FAMILY	LOCATIONS	IP ADDRESS	SERIAL NUMBER	FIRMWARE VERSION	STATUS
1	CE-CO...	AMX	CE-Series	CE		192.168.160.183	15500138505	1.0.2.0	OFFLINE
1	CE-IO4...	AMX	CE-Series	CE		192.168.160.167	15500132324	1.0.2.0	RUNNING
1	CE-IRS...	AMX	CE-Series	CE		192.168.160.186	15500138656	1.0.2.0	OFFLINE
1	CE-REL...	AMX	CE-Series	CE		192.168.160.92	15500132205	1.0.2.0	RUNNING

Additionally, each CE-Series module has an ID button that, when tapped, causes the module to send a broadcast message to facilitate discovery. The discovery beacon is sent as a UDP packet on port 4197 to 255.255.255.255, which is the universal broadcast address. If you use a laptop on the same switch as the CE device and use packet capture software like Wireshark, you can see the broadcast message as well as any other HControl devices that reply:

*Ethernet (port 4197)

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

Apply a display filter ... <Ctrl-/>

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.160.174	255.255.255.255	UDP	101	4197 → 4197 Len=59
2	0.000032	192.168.160.174	192.168.160.255	UDP	101	4197 → 4197 Len=59

> Frame 1: 101 bytes on wire (808 bits), 101 bytes captured (808 bits) on interface \Device\NPF_{7DD751B0-EA79-43D8-89C3-4624DC1D2204}, id 0
 > Ethernet II, Src: Phast_aa:ef:44 (00:60:9f:aa:ef:44), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
 > Internet Protocol Version 4, Src: 192.168.160.174, Dst: 255.255.255.255
 > User Datagram Protocol, Src Port: 4197, Dst Port: 4197
 ✓ Data (59 bytes)
 Data: 646973636f207b227265706c79706f7274223a343139372c22706172616d73223a5b2269...
 [Length: 59]

```

0000 ff ff ff ff ff ff 00 60 9f aa ef 44 08 00 45 00
0010 00 57 94 fb 40 00 40 11 44 44 c0 a8 a0 ae ff ff
0020 ff ff 10 65 10 65 00 43 57 05 64 69 73 63 6f 20
0030 7b 22 72 65 70 6c 79 70 6f 72 74 22 3a 34 31 39
0040 37 2c 22 70 61 72 61 6d 73 22 3a 5b 22 69 70 22
0050 2c 22 6e 61 6d 65 22 2c 22 6c 6f 63 61 74 69 6f
0060 6e 22 5d 7d 0a
  
```

Data (data.data), 59 bytes | Packets: 2 · Displayed: 2 (100.0%) · Dropped: 0 (0.0%) | Profile: Default

In this example, the CE module sent the discovery broadcast from 192.168.160.174 when the ID button was tapped.

Web Server Controls

Once you know the IP address that has been assigned to the CE box, all configuration settings are available through the built in web server. Navigate to the target address, use the default credentials of:

Default Credentials	
Username	admin
Password	password

Note: You will be forced to choose a new username and password upon login. Follow the prompts to set a new username and password for the device.

Note: On the first login, your browser will likely ask you to trust the CE module's self-signed certificate. This is required to access the module's web pages. You are able to provide a new certificate later if desired.

Network

The Network tab displays the current network settings and allow the user to set new network settings.

The screenshot shows a web browser window with the URL `https://192.168.160.92/setting.html`. The page title is "CE-REL8-AB0FD0 | CE-REL8-AB0FD0 | 192.168.160.92 | 15500132205". The navigation menu includes "Network", "Device", "Security", "Administrator", and "Firmware". The "Network" tab is active, showing two panels: "General" and "IPv4 Address".

General Panel:

- MAC Address: 00:60:9F:AB:0F:D0
- IP Hostname: CE-REL8-AB0FD0

IPv4 Address Panel:

- Mode: DHCP (selected), Specific IP Address
- IP Address: 192.168.160.92
- Subnet Mask: 255.255.255.0
- Gateway: 192.168.160.1
- Preferred DNS: 192.168.160.1
- Alternate DNS: 0.0.0.0

Buttons: Cancel, Accept

Footer: AMX HARMAN © 2022 HARMAN International. All Rights Reserved.

The MAC address field is read only and displays the preprogrammed MAC address of the CE box.

The IP address settings allow the choice of DHCP addressing or static addressing.

If DHCP is chosen, the IP Address and subnet mask setting are read-only as they will be determined by interaction with the DHCP server. If a DHCP server is not present, a fallback link local address will be assigned.

If Static is chosen, all fields related to the IP address are editable. Choose the IP address, subnet mask, gateway and DNS server that is appropriate for the network the CE box is installed on.

Device

This is the device-specific web page content for each type of CE-series box. Different content is displayed for each.

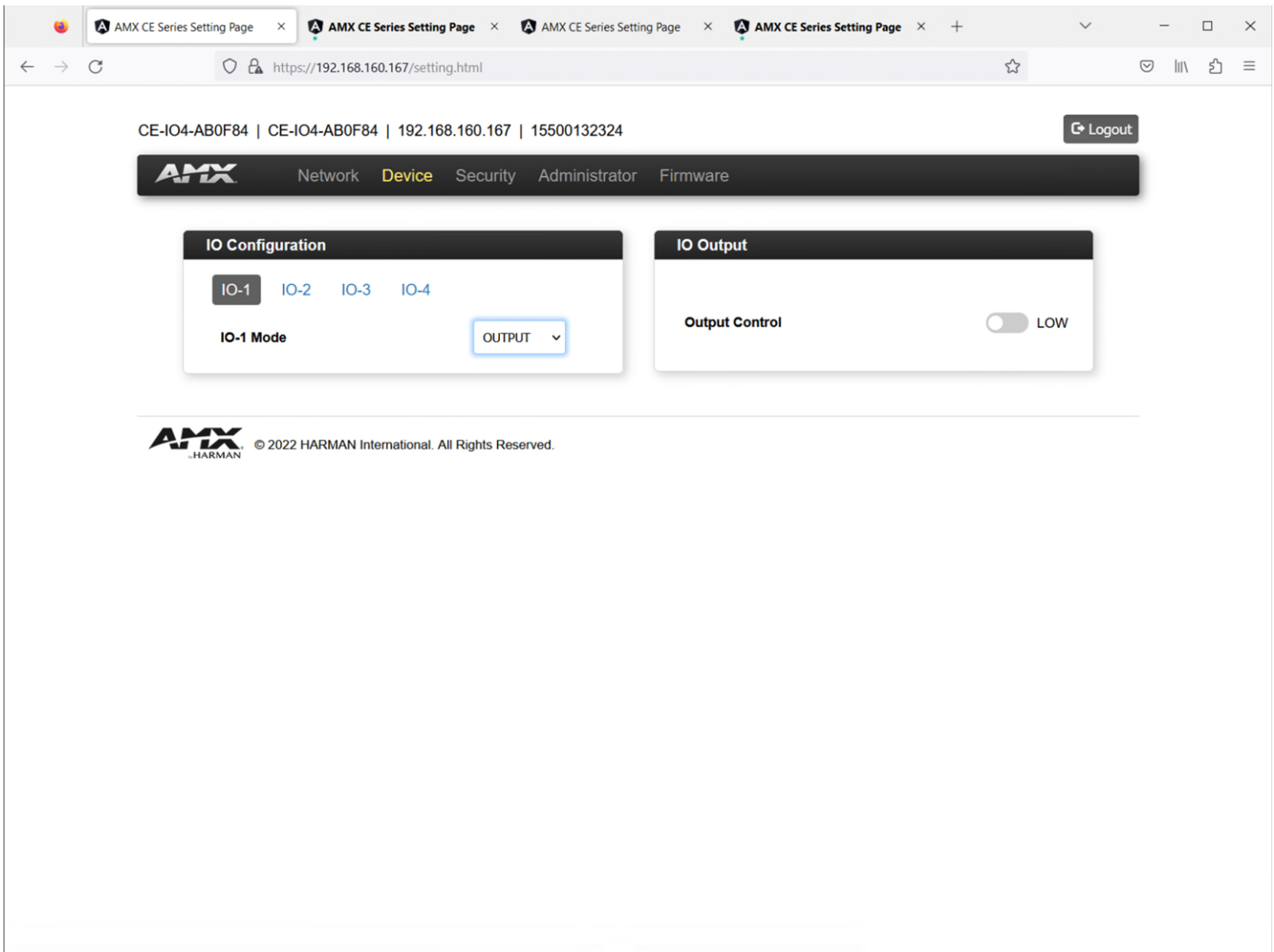
The screenshot displays the web interface for configuring a CE-REL8 - 8 Relay Module. The browser address bar shows the URL `https://192.168.160.92/setting.html`. The page header includes the AMX logo and navigation tabs for Network, Device (selected), Security, Administrator, and Firmware. The device information is shown as CE-REL8-AB0FD0 | CE-REL8-AB0FD0 | 192.168.160.92 | 15500132205, with a Logout button. The main content area is divided into two sections: Relay 1~4 Config and Relay 5~8 Config. Each section lists the enable status for its respective relays using toggle switches.

Relay	Enable Status
Relay-1	OFF
Relay-2	OFF
Relay-3	OFF
Relay-4	OFF
Relay-5	OFF
Relay-6	OFF
Relay-7	ON
Relay-8	OFF

AMX HARMAN © 2022 HARMAN International. All Rights Reserved.

The status of each relay is displayed in the CE-REL8's Device page. Clicking on the label or slider for a particular relay will toggle the status of the relay. As pictured, Relay 7 is engaged.

CE-IO4 (configured as Output)



First, select from the list of I/O ports to see the status. Those labels act as radio buttons. When one is selected, all the settings shown are for that particular I/O. Each I/O is individually configurable.

If the I/O is set as an output, there is only one further setting available. Similar to the Relay, the Output can be driven high or low by the toggle switch represented on this page. The current status is displayed this way.

- **IO Output: Output Control** (sets output status)
The current setting for the digital output state: low or high

CE-IO4 (Configured as Input)

The screenshot displays the AMX CE Series Setting Page for CE-IO4 (Configured as Input). The page is accessed via a browser at <https://192.168.160.167/setting.html>. The device ID is CE-IO4-AB0F84 | CE-IO4-AB0F84 | 192.168.160.167 | 15500132324. The page features a navigation bar with the AMX logo and tabs for Network, Device (selected), Security, Administrator, and Firmware. A Logout button is located in the top right corner.

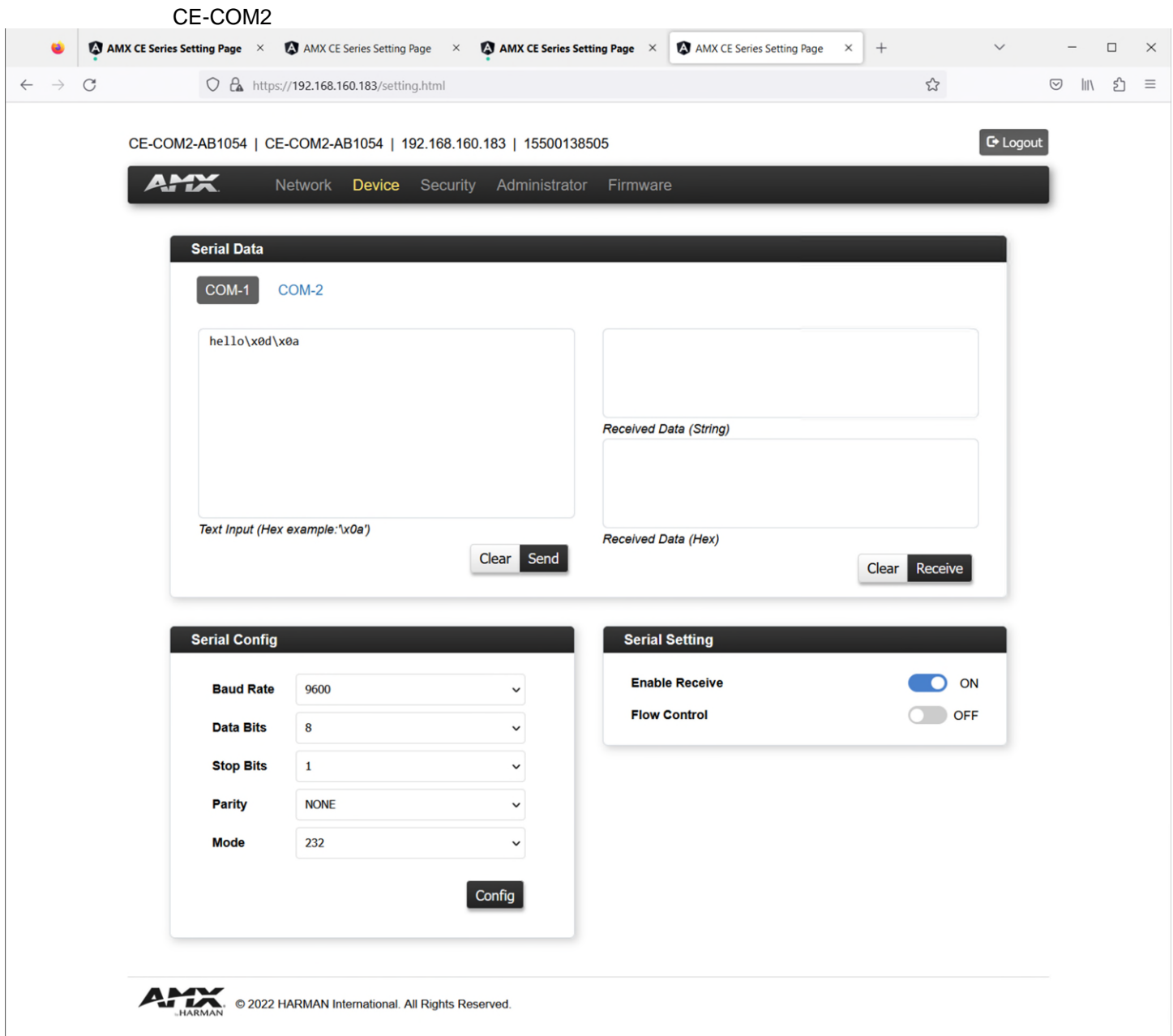
The main content area is divided into four sections:

- IO Configuration:** Shows tabs for IO-1, IO-2 (selected), IO-3, and IO-4. The IO-2 Mode is set to INPUT.
- IO Input:** Shows Input Mode set to BOTH and Pull Up Enable set to ON.
- Digital Setting:** Shows Input State (High/Low) set to Low, Debounce Time (5~250 ms) set to 75, High Threshold (0.0~10.0 Volts) set to 2.7, and Low Threshold (0.0~10.0 Volts) set to 0.8.
- Analog Setting:** Shows Input Voltage (Volts) set to 0.1 Volts and Trigger Delta (0.1~4.9 Volts) set to 0.2.

The footer contains the AMX logo and the text: © 2022 HARMAN International. All Rights Reserved.

If the I/O is configured as an Input, there are many more settings available:

- **I/O input mode:** Analog, Digital, or Both
Depending on the choice made from this pulldown, the following additional settings will appear:
- **Digital Settings: Input State** (feedback for the input)
The current reading for the digital input state: low or high
- **Digital Settings: Debounce Time** in ms
This is the polling interval for the input state. This prevents flooding of updates from unstable sources.
- **Digital Settings: High Threshold** in volts
This sets the boundary for what is considered 'high' for an input.
- **Digital Settings: Low Threshold** in volts
This sets the boundary for what is considered 'low' for an input
Note: If the voltage is between the high and low threshold, the previous input state will be unchanged.
- **Analog Settings: Input Voltage**
This is the current voltage detected on the Analog input pin
- **Analog Settings: Trigger Delta in Volts**
This is the magnitude of change that must occur before a new value is reported.



The CE-COM2 has two configurable COM ports. First, select the COM port you wish to view or set the parameters for (COM-1 or COM2)

The **Serial Config** settings contain the standard COM port settings. Match these settings for those specified on the target serial-controlled device. Once set, press the Config button to commit these changes.

The **Serial Setting** dialog contains two controls: Enable Receive and Flow Control. Enable Receive determines if any messages received by that COM port should trigger an event for any clients that have subscribed. If receiving a response is not desired, turning Enable Receive off can reduce traffic and control system overhead for handling these events. The Flow Control switch enables or disables the use of the hardware handshaking pins.

The **Serial Data** dialog shows the last received message in both ASCII encoding and hexadecimal representation. The Text input field allows you to send a serial message through the selected COM port. This allows for testing of the COM port and wiring without using a control system.

In the example below, the string *hello* followed by a carriage return and line feed are ready to be sent from COM-1

The interface is titled "Serial Data" and has a dark header. Below the header, there are two buttons: "COM-1" (selected) and "COM-2". On the left, there is a large text input field containing the text "hello\r\n". Below this field is the label "Text Input (Hex example: '\x0a')". To the right of the input field are two buttons: "Clear" and "Send". On the right side of the interface, there are two empty text boxes. The top one is labeled "Received Data (String)" and the bottom one is labeled "Received Data (Hex)". Below these boxes are two buttons: "Clear" and "Receive".

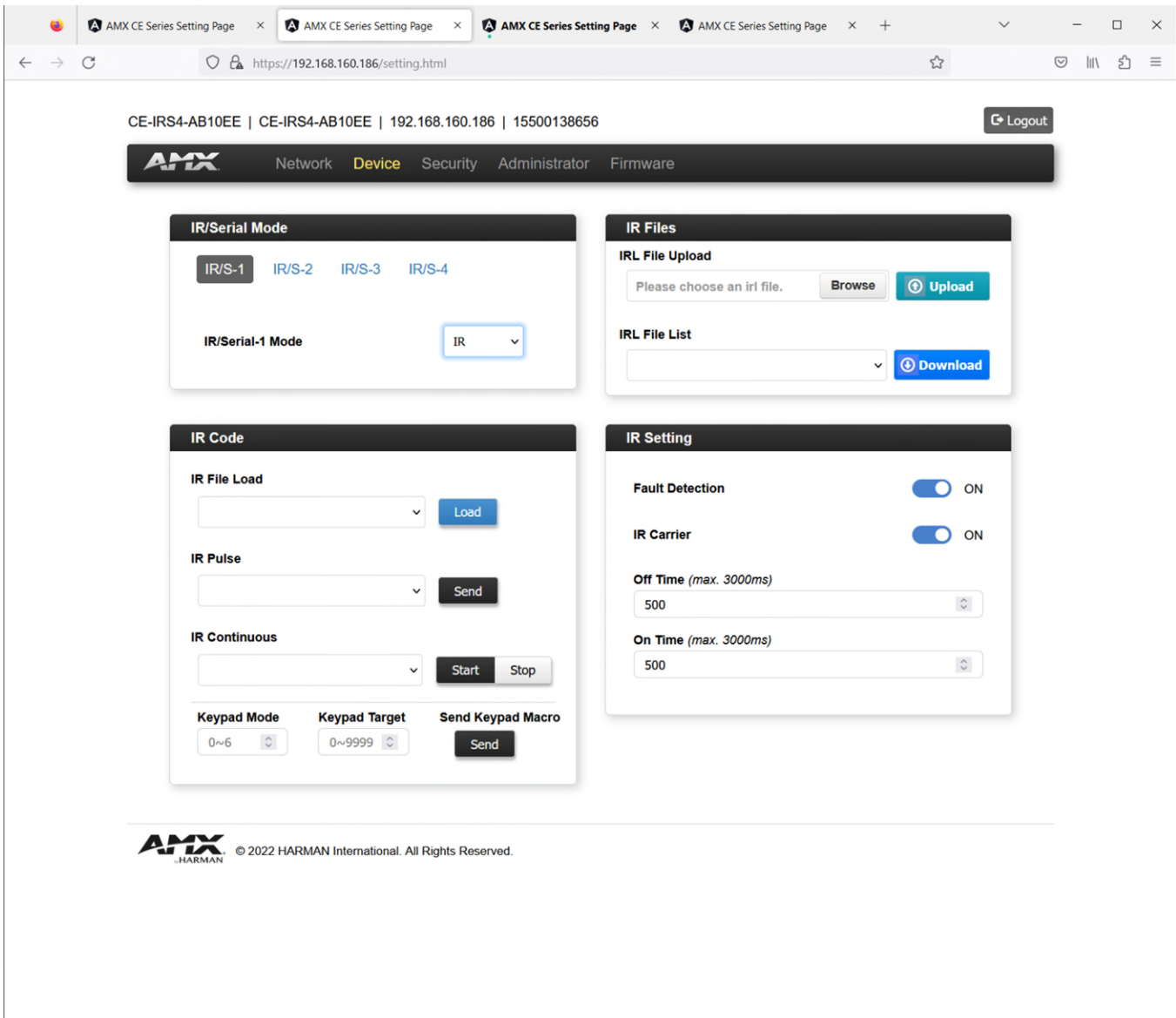
Clicking **Send** will transmit the message in the Text Input window out of the COM port selected using the present RS-232 protocol settings for that port.

For demonstration purposes, I wired COM-1 to COM-2 to show the Receive function. To see the last message received by the CE-COM2's COM port, click the **Receive** button.

The interface is titled "Serial Data" and has a dark header. Below the header, there are two buttons: "COM-1" and "COM-2" (selected). On the left, there is an empty text input field with the label "Text Input (Hex example: '\x0a')". To the right of the input field are two buttons: "Clear" and "Send". On the right side of the interface, the top text box labeled "Received Data (String)" now contains the text "hello\r\n". The bottom text box labeled "Received Data (Hex)" contains the text "\x68 \x65 \x6c \x6c \x6f \x0d \x0a". Below these boxes are two buttons: "Clear" and "Receive".

The last message received will be represented as mixed ASCII & hexadecimal in the **Received Data (String)** window, and as strictly hexadecimal for the **Received Data (Hex)** window.

CE-IRS4



The CE-IRS4 has 4 individually configurable outputs.

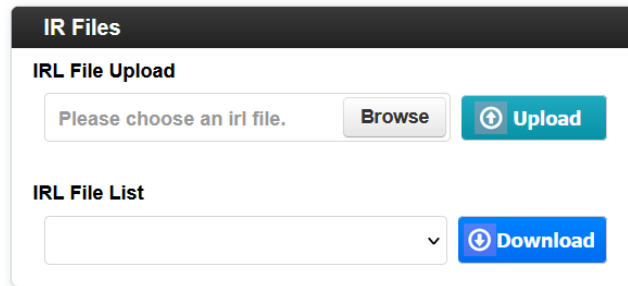
First, select the output you want to configure. The labels **IR/S-1**, **IR/S-2**, **IR/S-3**, and **IR/S-4** act as radio buttons allowing you to select one particular IR/Serial port to configure. In the above example, IR/S-1 is selected.

Next select the desired mode using the pulldown labelled **IR/Serial-n Mode**. The modes are: **IR**, **Serial**, and **Data**.

- The **IR** mode expects to drive an IR emitter (CC-NIRC). This emitter should be placed on the IR receptive area of the controlled device.
- The **Serial** mode expect to be wired directly to a controlled device, but still uses the IR waveform as control.
- The **Data** mode works as a 1-way RS-232 port. The effective distance and available baud rates are reduced from a dedicated RS-232 port.

IR Mode

If the port selected is in IR modes, the following dialog boxes will appear: IR Files, IR Code, and IR Setting

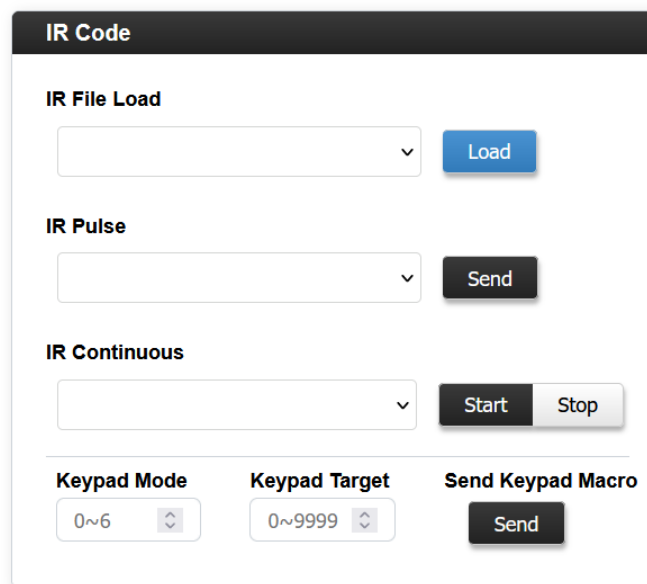


The screenshot shows a dialog box titled "IR Files". It contains two main sections: "IRL File Upload" and "IRL File List". The "IRL File Upload" section has a text input field with the placeholder text "Please choose an irl file.", a "Browse" button, and an "Upload" button. The "IRL File List" section has a dropdown menu and a "Download" button.

In **IR Files**, you may upload multiple AMX .irl files. These .irl files contain the IR data for each button that has been captured.

The **IR File List** allows you to browse all of the loaded .irl files present on this module. You may also back up the .irl file by selecting the file and pressing the **Download** button.

The **IR Code** window has several setting and controls:



The screenshot shows a dialog box titled "IR Code". It contains several sections: "IR File Load" with a dropdown menu and a "Load" button; "IR Pulse" with a dropdown menu and a "Send" button; "IR Continuous" with a dropdown menu and "Start" and "Stop" buttons; "Keypad Mode" with a dropdown menu showing "0~6"; "Keypad Target" with a dropdown menu showing "0~9999"; and "Send Keypad Macro" with a "Send" button.

- The **IR File Load** allows you to select the actual .irl file that will be in use from all the .irl files that have been loaded. This will load all the captured buttons that will be available to this web page and the connected control system. Only one IR file is active at a time.
- **IR Pulse** allows you to select from a list of loaded captured IR buttons and send the button for testing that function. The pulses are queued, so pressing the button repeated will send a pulse for each press. The timing for the pulse is in the IR Setting window.
- **IR Continuous** allows you to send an IR code for an arbitrarily long time. The IR code is emitted when Start is pressed and ceases when Stop is pressed.
- **Keypad Mode** defines the pattern of IR codes that are queued for the purpose of sending a multidigit number with an IR Remote.
- **Keypad Target** is the series of numbers that will be sent when **Send** is pressed
- **Send Keypad Macro** will send the sequence of numbers in the pattern specified by **Keypad Mode**

The **Keypad Mode** values are based on the NetLinx NX implementation of an IR Port. The available Keypad modes are:

Mode 0: No leading zeros, use Enter key, up to 3 digits

- “3” transmits the IR code as 3-enter.
- “34” transmits the IR code as 3-4-enter.
- “343” transmits the IR code as 3-4-3-enter.

Mode 1: Use leading zeros, use Enter key, always 3 digits

- “3” transmits the IR code as 0-0-3-enter.
- “34” transmits the IR code as 0-3-4-enter.
- “343” transmits the IR code as 3-4-3-enter.

Mode 2: No leading zeros, no Enter key, up to 3 digits

- “3” transmits the IR code as 0-0-3.
- “34” transmits the IR code as 0-3-4.
- “343” transmits the IR code as 3-4-3.

Mode 3: Use leading zeros or Hundred key, No Enter key, up to 3 digits

- “3” transmits the IR code as 0-3.
- “34” transmits the IR code as 3-4.
- “343” transmits the IR code as 100-100-100-4-3

Mode 4: Legacy Keypad mode, values 0-199 only, use Enter if present

- “3” transmits the IR code as 0-0-3-enter.
- “34” transmits the IR code as 0-3-4-enter.
- “343” returns an error message

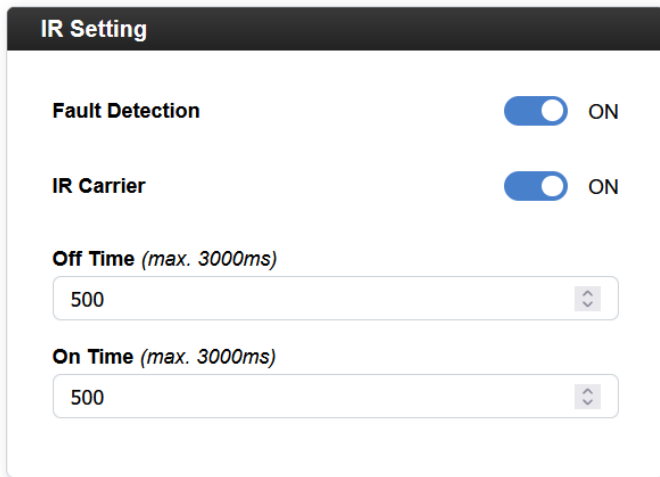
Mode 5: Use leading zeros, use Enter key, always 4 digits

- “3” transmits the IR code as 0-0-0-3-enter.
- “34” transmits the IR code as 0-0-3-4-enter.
- “343” transmits the IR code as 0-3-4-3-enter.
- “3434” transmits the IR code as 3-4-3-4-enter.

Mode 6: Use leading zeros, no Enter key, always 4 digits

- “3” transmits the IR code as 0-0-0-3.
- “34” transmits the IR code as 0-0-3-4.
- “343” transmits the IR code as 0-3-4-3.
- “3434” transmits the IR code as 3-4-3-4.

The **IR Setting** window has the following controls:



The screenshot shows a window titled "IR Setting" with a dark header. Below the header, there are four settings:

- Fault Detection**: A blue toggle switch is turned on, followed by the text "ON".
- IR Carrier**: A blue toggle switch is turned on, followed by the text "ON".
- Off Time (max. 3000ms)**: A text input field contains the number "500" and a small up/down arrow icon on the right.
- On Time (max. 3000ms)**: A text input field contains the number "500" and a small up/down arrow icon on the right.

- **Fault Detection** – If an IR code is sent but no IR emitter is connected, a fault will be registered
- **IRCarrier** – This setting determines whether the IR carrier frequency (typically 41kHz) is present in the output signal. This should always be on for use with an IR emitter. Some wired IR connections may require removal of the IR carrier frequency
- **Off Time** – For queued IR pulses such as Keypad Macros and IR Pulse this setting determines the time between queued IR pulses
- **On Time** – For queued IR pulses such as Keypad Macros and IR Pulse this setting determines the total transmission time for the IR pulse

Security

CE-IRS4-AB2408 | CE-IRS4-AB2408 | 192.168.160.199 | 15500074173 Logout

AMX Network Device **Security** Administrator Firmware

Certificates

Server Certificate

Please choose server cert

Server Key

Please choose server key

Security Setting

H-Control Secure Connection OFF

AMX HARMAN © 2022 HARMAN International. All Rights Reserved.

The Security tab allows the upload of certificates for HTTPS and secure HControl functionality.

The CE-series modules will have a self-signed certificate installed from the factory. To add your own certificate, use the **Browse** button under Server Certificate to choose the file and **Upload** to install it. The expected certificate format is an X.509 certificate (PEM file).

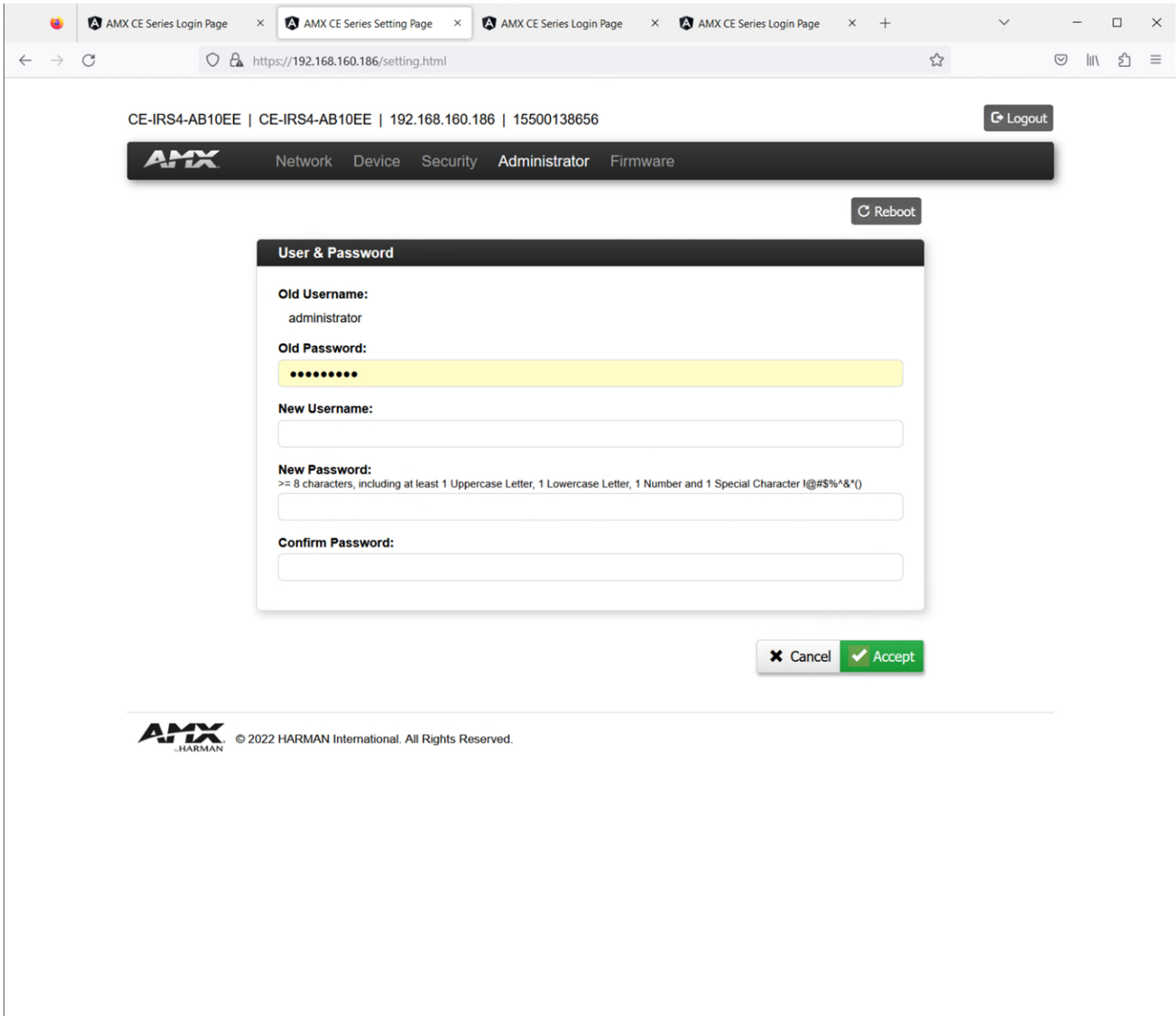
To change the CE-

The **HControl Secure Connection** switch allows you to enforce the use of Secure HControl. With the switch set to 'off', both secure and plain text connections are allowed. With the switch set to 'on', only a TLS-based connection is accepted.

When using Secure HControl, it is recommended to disable **HControl** on the Security tab in the web interface of the connected MUSE. This forces the MUSE HControl router to use TLS when connecting in Secure HControl mode.

Administrator

The Administrator tab has two functions: setting the module's credentials and rebooting.



The screenshot displays a web browser window with the URL `https://192.168.160.186/setting.html`. The page header shows the device information: `CE-IRS4-AB10EE | CE-IRS4-AB10EE | 192.168.160.186 | 15500138656` and a `Logout` button. The navigation menu includes `Network`, `Device`, `Security`, `Administrator` (selected), and `Firmware`. A `Reboot` button is located in the upper right corner. The main content area features a `User & Password` form with the following fields:

- Old Username:** administrator
- Old Password:** [masked with dots]
- New Username:** [text input field]
- New Password:** [text input field with a note: `>= 8 characters, including at least 1 Uppercase Letter, 1 Lowercase Letter, 1 Number and 1 Special Character !@#%*&'()`]
- Confirm Password:** [text input field]

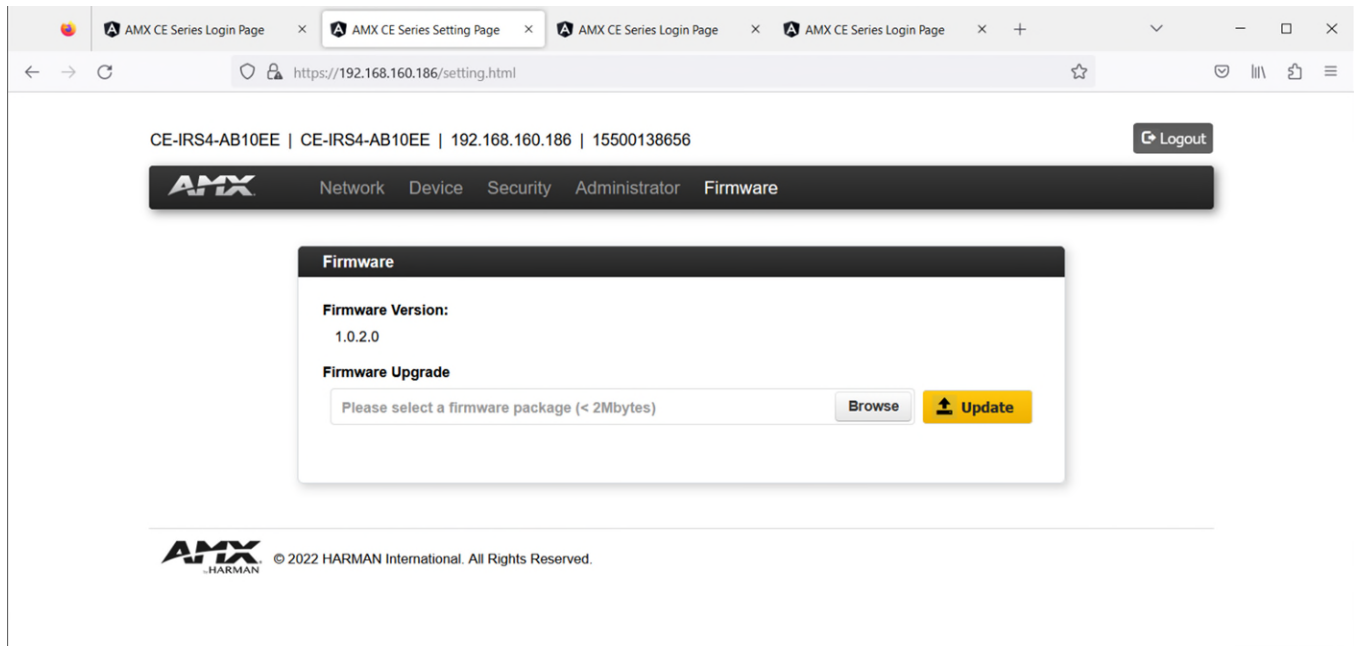
At the bottom of the form are `Cancel` and `Accept` buttons. The footer contains the AMX logo and the text: `© 2022 HARMAN International. All Rights Reserved.`

To set a new username and password, fill out the **User & Password** form.

Note: Only one set of credentials exist within the CE-series module. Setting a new username and password will remove the previous set.

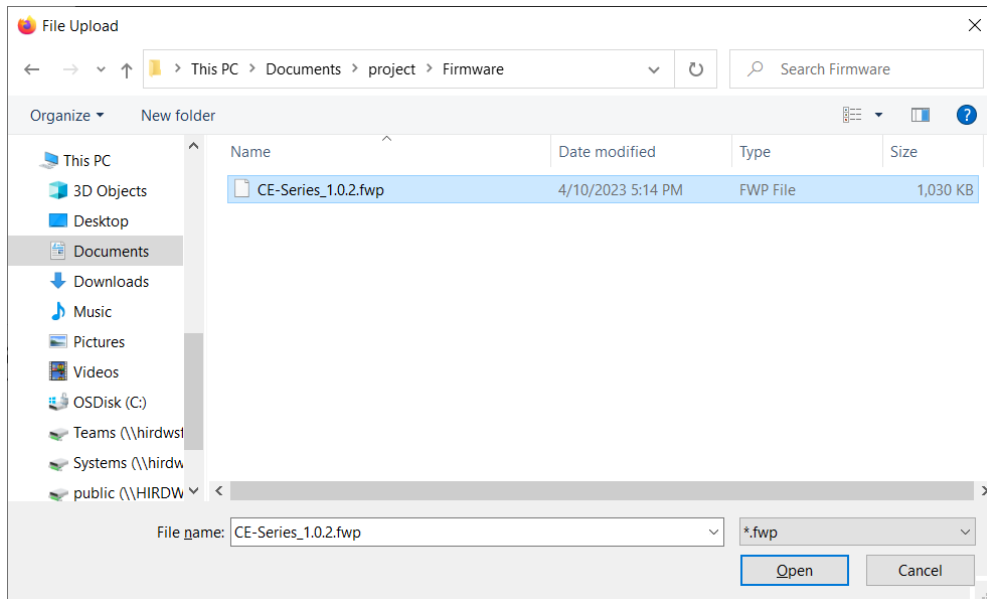
To reboot the module, click the **Reboot** button on the upper right portion of the window.

Firmware



The Firmware tab displays the current firmware version of the CE box.

To update the firmware of the CE-series Control Extender, Select the Firmware Tab then click **Browse...** to select the firmware image you wish to load. The firmware file extension for a firmware update package is **.fwp**.



Once selected, click **Update** to commit this firmware change.

Note: Please do not remove power from the CE module during this process.

CE-Series HControl Programming

Introduction

The CE-Series control extenders support the HControl protocol, HARMAN Pro’s standard discovery, control and messaging protocol used on all newer Audio, Video and Control products. This means they can be natively configured and controlled within any MUSE script or Automator flow without the need for any additional drivers or modules.

Getting Started

For traditional coding, it’s assumed that a user has access to a development environment for Python, JavaScript and/or Groovy. The easiest way to get started is by using Microsoft® Visual Studio Code (VSCoDe) development environment with the Mojo Device Plugin for VSCoDe installed.

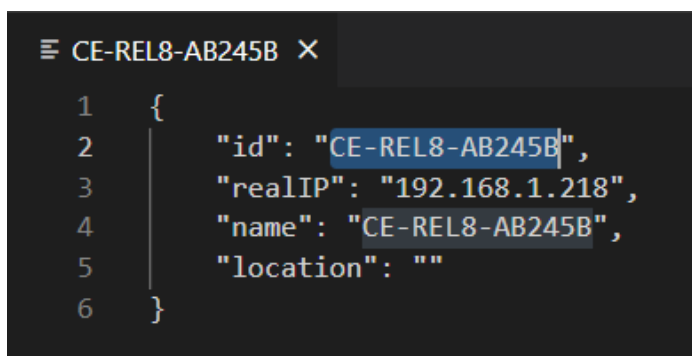
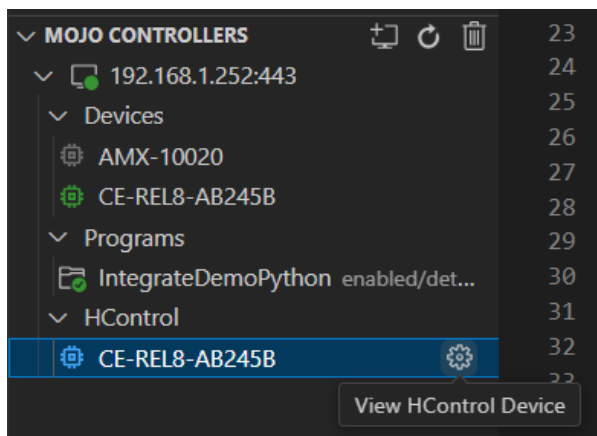
For drag-and-drop coding, the user should have the MUSE Automator application installed.

To use any CE-Series control extender on a MUSE controller, they must first be configured as a device on that controller. The easiest way to do this is from the MUSE controllers Devices web page under the System menu item.

IMPORTANT NOTE:

The Instance Id used when creating the device from the MUSE Device Web Page **MUST MATCH** the ID of the particular CE Control Extender you’re connecting to.

You can find the CE Control Extender’s ID under the HControl section of the MOJO CONTROLLERS side bar in Microsoft Visual Studio Code.



Copy and paste this “id” to the Instance id field when creating the device on the MUSE controller:

Create a New Device

Instance Id:*

CE-REL8-AB245B

Driver ID:*

com.amx.mojo.hcontrol.discovered

Name:

Room Relays

Description:

Relay control for screen and shades

✕ Cancel

✓ Accept

Refer to the MUSE Automation Controllers Programming Guide to get more details on installing and setting up VS Code and MUSE Automator and on how to connect an HControl device to a MUSE controller.

The programming examples below assume that one of each CE-Series control extender has been configured as a device on a MUSE controller with the names: “dvREL8”, “dvCOM2”, “dvIO4”, and “dvIRS4”

The examples below are provided in Python. All commands, parameters and general coding structure is the same when coding in JavaScript and Groovy. The only differences are the syntax differences between the languages, like the use of curly braces to identify coding blocks, semi-colons at the end of lines, and (for JavaScript) the use of the “var” keyword to declare variables.

Basic Syntax and Ports

Coders ‘write’ to the CE Control Extenders using Parameters and Commands. Parameters and Commands are directed at a Port. The generic syntax is:

```
<device>.<port>.<parameter>
```

Or

```
<device>.<port>.<command>(<zero or more args>)
```

<device> is the name of the device used when setting up the device on the MUSE controller as described in the Getting Started section above (e.g. dvREL8, dvCOM2, dvIRS4, dvIO4).

The <port> is the name of the specific port on the device. All CE Control Extenders have only a single port type, but listing the port is still required. This table shows the port names for each CE Control Extender.

CE Type	Port Name	Description
CE-REL8	relay	Array of individual relay ports
CE-COM2	serial	Array of individual serial ports

CE-IRS4	ir	Array of individual ir ports
CE-IO4	io	Array of individual io ports

The Parameters and Commands for each CE type are described in the Device Specific HControl Details section below.

0-based vs 1-based port numbering

HControl uses a 1-based numbering system for all ports. Therefore all ports on the CE-Series control extenders are numbered starting with 1. You can see this on the silkscreen on each of the boxes and when configuring ports from the AVX Manager software.

To simplify the programming for the multiple ports on each of the CE control extenders, the ports are structured as arrays when coding on MUSE. For example, rather than having eight uniquely named ports on a CE-REL8 from “relay1” through “relay8”, you identify each port by an index into an array.

Because Python, JavaScript and Groovy all use 0-based arrays, you must use array index 0 to access the port numbered “1” on the silkscreen, and array index 7 to access the port numbered “8”. In other words, you will always use an array index that is ONE LESS than the port number you wish to access.

All CE Control Extender ports are arrays, so you must use an array index on the port name. So the general syntax is:

```
<device>.<port_name>[<0-based index>].<parameter>
```

Or

```
<device>.<port_name>[<0-based index>].<command>(<zero or more args>)
```

Parameters

Parameters are used to set and get individual properties of a device that hold their state, for example the position of a relay on the CE-REL8.

Parameters contain the following properties that can be read and (if writable) written.

Property	Data Type	R/W	Description
value	Varies	R/W	The value of the parameter
normalized	Float	R/W	A float value between 0 and 1, inclusive, based on the range of the value
min	Varies	R	The minimum value of a numerical parameter
max	Varies	R	The maximum value of a numerical parameter
defaultvalue	Varies	R	The default value of a parameter
type	Varies	R	The data type of this specific parameter
enums	Array	R	For an enumeration, the specific data points available

Getting and Setting Parameters:

Parameters in HControl can be directly read by setting a variable equal to the parameter value and directly written (if read/write) by setting the parameter value equal to a new value.

For example, you can access the state parameter of relay 1 with this code:

```
Relay1_on = dvREL8.relay[0].state.value
```

```
dvREL8.relay[0].state.value = True
```

The following table lists the contents of each of these parameter properties depending on the data type of the parameter:

Property	Boolean	Integer/Float	Enumeration	String	Byte Array
value	true/false	<number>	<string>**	<string>	byte[]
normalized	0 or 1	0 to 1 inclusive	0 to 1 in steps of 1/(# enums - 1)	Invalid	Invalid
min	0	Min accepted value	0	Min string length	Min array length
max	1	Max accepted value	# enums - 1	Max string length	Max array length
defaultvalue	*	*	*	*	*
type	'boolean'	'integer' or 'float'	'enum'	'string'	'byte_array'
enums	Empty Array	Empty Array	Array of enum strings	Empty Array	Empty Array

* The **defaultValue** field will be the same data type as the Parameter and will contain a valid data entry if there is a default for that parameter, or it will be undefined if there is no default value.

** Reading enumerations will always return a string. Writing/setting can be done with either a string or the zero-based integer that represents the index into the array of enum strings. For example, setting an enum to 0 will set it the value in <parameter>.enums[0].

Watching for Parameter changes:

If you want to trigger a function when a parameter changes, you can set a “watch” on that parameter, passing in the name of the function you want to call when the parameter changes.

Here’s an example of using a watch on a relay port:

```
handle_relay_state_change(update):
    print("Relay 1 new state is ", update.value)
dvREL8.relay[0].state.watch(handle_relay_state_change)
```

When the watch function is called, it will be passed a **Parameter Update Structure** with details of the parameter and the new value. The contents of the **Parameter Update Structure** (the update variable passed into the example above) is:

Field	Data Type	Description
path	String	The property of the device that this event refers to
id	String	A shortened version path. For ICSP, only the button number is conveyed
value	Varies	The event data. For button events, this would be a boolean
newValue	Varies	Identical to value
oldValue	Varies	The data value before the event was processed
normalized	float	A float value between 0 and 1, inclusive, based on the range of the value
source	Object Ref	The object reference for the specific parameter that was updated

For parameters, the **path** field contains a '/' delimited string containing the port name, 0-based port number and parameter name:

<port name>/<port number>/<parameter>

So in the watch example above, update.path would contain the string “relay/0/state”. If watch events are set on multiple ports and/or parameters pointing to the same handler function, you can parse the path string to determine the port and parameter that changed.

The data type of the **value**, **newValue** and **oldValue** fields will always be the same as the data type of the parameter being watched.

Commands

Commands in HControl are called directly using the name of the command with the necessary arguments. Here’s an example of using the CE-COM2’s send command:

```
dvCOM2.serial[0].send(“Hello”)
```

Events

Events in HControl are sent from the CE Controller Extender when a stateless device event occurs. Good examples of this are when data is received on a COM2 serial port.

To trigger a function when an event occurs, you can set a “listen” on that event, passing in the name of the function you want to call when the event happens. Here’s an example using a listen on a serial port receive event:

```
def handle_received_serial(event):  
    # process incoming string found inside the event variable  
  
dvCOM2.serial[0].receive.listen(handle_received_serial)
```

When the function is called it will be passed an **Event Update Structure** with details on the event. The contents of the **Event Update Structure** (the event variable passed into the example above) is:

Field	Data Type	Description
path	String	The property of the device that this event refers to
id	String	A shortened version path. For ICSP, only the button number is conveyed
arguments	JSON	The data payload of the event, dependent on the specific event
source	Object Ref	The object reference for the specific parameter that was updated

For events, the **path** field contains a ‘/’ delimited string containing the port name, 0-based port number and the event name:

<port name>/<port number>/<event>

So in the listen example above, event.path would contain the string “serial/0/receive”. If listen events are set on multiple ports and/or events pointing to the same handler function, you can parse the path string to determine the port and event that triggered the function call.

The contents of the **arguments** field contains all the relevant information pertaining to an event and varies depending on the specific CE device and event. Details on the contents of this field for events on CE Control Extenders can be found below.

Device Specific HControl Details

CE-REL8

The CE-REL8 is the simplest device, containing only one parameter for control. There are no commands or events.

Parameters

Parameter list:

Parameter Name	Read/Write	Data Type	Description
state	Read/Write	boolean	Engage the relay

Examples:

Turn relay 1 on

```
dvREL8.relay[0].state.value = True
```

To watch a relay state, use:

```
handle_relay_state_change(update):  
    print("Relay 1 new state is ", update.value)  
dvREL8.relay[0].state.watch(handle_relay_state_change)
```

CE-COM2

The CE-COM2 has the following commands:

Command	Arguments	Date Type	Description
send	data	String	Send data out the port
setFlowControl	mode	Enumeration	NONE or HARDWARE
setCommParams	baudRate	Enumeration	1200, 4800, 9600, 19200, 38400, 57600, 115200
	dataBits	Integer	7 or 8
	stopBits	Integer	1 or 2
	parity	Enumeration	NONE, EVEN, ODD
	mode	Enumeration	232, 422, 485
enableReceive	-	-	Listen for message on the port
disableReceive	-	-	Stop listening for messages
flushReceiveBuffer	-	-	Clear incoming buffer

Examples:

To send a string out of COM Port 1:

```
dvCOM2.serial[0].send("Hello, world")
```

To send the string: 'This string contains "quoted text" and ° binary data'

```
dvCOM2.serial[0].send("This string contains \"quoted text\" and \xB0 binary data")
```

Notice that the double-quote characters within the string must be escaped and that the degree symbol can be sent as binary using hex code escape sequence `\xB0`

To disable the hardware handshaking available on COM Port 1, send:

```
dvCOM2.serial[0].setFlowControl("NONE")
```

To clear the receive buffer on COM Port 2, send:

```
dvCOM2.serial[1].flushReceiveBuffer()
```

To configure COM port 1 to the most common transmission settings, send:

```
dvCOM2.serial[0].setCommParams("9600", 8, 1, "NONE", "232")
```

Events

Event Name	Argument Keys	Data Type	Description
receive	"data"	UTF-8 encoded String	Received data from a connected device

Examples:

To enable receiving strings that are sent to the COM port from the controlled device it is required to enable receive strings on the port, and then to set a listen event handler for the **receive** event. For example:

```
dvCOM2.serial[0].enableReceive()
def on_serial1_receive(event):
    received_string = str(event.arguments['data'].decode('utf-8'))
dvCOM2.serial[0].receive.listen(on_serial1_receive)
```

Note that you must use the **arguments** field of the **Event Update Structure** that's passed to the event handler function (the event variable above) and access the "data" element to retrieve the UTF-8 encoded data received on the serial port.

Parameters

Parameter	Read/Write	Data Type	Values	Default
baudRate	Read Only	Enumeration	1200, 4800, 9600, 19200, 38400, 57600, 115200	9600
dataBits	Read Only	Integer	7 or 8	8
mode	Read Only	Enumeration	232, 422, 485	232
parity	Read Only	Enumeration	NONE, EVEN, ODD	EVEN
stopBits	Read Only	Integer	1 or 2	1

Examples:

All of the COM2 parameters are read-only. These parameters are configured using the setCommParams command above.

Here's an example of reading one of the parameters:

```
data_bits = dvCOM2.serial[0].dataBits.value
```

To watch for changes on one of these parameters, use:

```
handle_databits_change(update):
    print("Serial 1 new data bits is ", update.value)
```



```
dvCOM2.serial[0].dataBits.watch(handle_databits_change)
```

CE-IRS4

Commands

Path	Arguments	Data Type	Description
bufferedSendIr	code	Integer	Queue a timed pulse of IR by index
bufferedSendNamedIr	code	String	Queue a timed pulse invoked by name
clearAndSendIr	code	Integer	Clear the queue, then send the new IR pulse
clearAndSendNamedIr	code	String	Clear the queue, then send the named IR pulse
keypadMacro	code	Integer	Queue digits based on the keypadMode pattern
keypadMode	mode	Integer	Set the pattern for keypad digit transmission
loadIrFile	file	String	The .irl filename to use
offIr	-	-	Turn off IR on the port
onIr	code	Integer	Send IR code continuously by #
onNamedIr	code	String	Send IR code continuously by name
setOffTime	millis	Integer	Set the queue timing interval in ms
setOnTime	millis	Integer	Set the queue active interval in ms
enableFaultDetection	-	-	Alert if no IR LED detected
disableFaultDetection	-	-	Do not alert
setCommParams	baudRate	Enumeration	1200, 4800, 9600, or 19200
	dataBits	Integer	7 or 8
	stopBits	Integer	1 or 2
	parity	Enumeration	NONE, ODD, or EVEN
send	data	String	Use IR as 1-way COM port

Examples:

To send on IR port 3 the IR code in position 1 for a pre-determined pulse time (or add to the queue if there are already bufferedSendIR or bufferedSendNamedIR or keypadMacro commands active) send:

```
dvIRS4.ir[2].bufferedSendIr(1)
```

To send that same code if it is named 'PLAY' in the .irl file, send:

```
dvIRS4.ir[2].bufferedSendNamedIr("PLAY")
```

To send the codes on the number pad of the remote for station 345, send:

```
dvIRS4.ir[2].keypadMacro(345)
```

To change the pattern used to execute keypad macros, use:

```
dvIRS4.ir[2].keypadMode(2)
```

See the [Keypad Mode](#) section of the web server documentation for pattern information.

To load a new IR from the collection of .irl files uploaded to the CE-IRS4 configuration web page, send:

```
dvIRS4.ir[2].loadIrFile("samsung01.irl")
```

When using buffered (queued) IR, to configure a 1 second interval between IR pulses, send:

```
dvIRS4.ir[2].setOnTime(1000)
```

When using buffered (queued) IR, to configure the active pulse width to 0.5 second, send:

```
dvIRS4.ir[2].setOffTime(500)
```

Parameters

Path	Read/Write	Data Type	Values	Default
carrier	Read/Write	Boolean	true or false	True
mode	Read/Write	Enumeration	IR, SERIAL, or DATA	IR
baudRate	Read Only	Enumeration	1200, 4800, 9600, 19200, 38400, 57600, 115200	9600
dataBits	Read Only	Integer	7 or 8	8
parity	Read Only	Enumeration	NONE, EVEN, ODD	NONE
stopBits	Read Only	Integer	1 or 2	1

Examples:

To set the IR port to use the carrier frequency specified in the .irl file (typically 41Hz modulation of the near infrared signal), send:

```
dvIRS4.ir[2].carrier.value = True
```

To set the IR port mode to use as a 1-way COM port, send:

```
dvIRS4.ir[2].mode.value = "DATA" # turn off carrier to use this mode
```

The COM related parameters are read-only. See the CE-COM2 Parameters for syntax examples.

Events

Event	Argument Keys	Data Type	Description
fault	"data"	String	Alerts to missing or reversed IR bud

Examples:

The IRS4 will send a fault event if it detects certain fault conditions like a missing or reversed IR bud when attempting to send an IR message out a port. You must subscribe to this event to receive fault messages.

```
handle_ir_faults(event):  
    print("received fault: ", event.arguments['data'])
```

```
dvIRS4.ir[3].fault.listen(handle_ir_faults)
```

CE-IO4

Parameters

Parameter	Read/Write	Data Type	Values	Default
debounceTimeMilliseconds	Read/Write	Integer	5-250	75
mode	Read/Write	Enumeration	INPUT or OUTPUT	INPUT
output	Read/Write	Boolean	true or false	false
debounceMinDelta	Read/Write	Float	0.1 - 4.9	0.2
inputMode	Read/Write	Enumeration	DIGITAL, ANALOG, BOTH	BOTH
digitalInputLowMax	Read/Write	Float	0.0 - 9.9	0.8
digitalInput2KPullup	Read/Write	Boolean	true or false	false
analogInput	Read Only	Float	0.0 - 10.0	-
digitalInput	Read Only	Boolean	true or false	-
digitalInputHighMin	Read/Write	Float	0.1 - 10.0	2.7

Examples:

Set the debounce time (minimum time between updates) to 123ms:

```
dvIO4.io[0].debounceTimeMilliseconds.value = 123
```

Set the mode for the particular input (input or output):

```
dvIO4.io[0].mode.value = "INPUT"
```

```
dvIO4.io[0].mode.value = "OUTPUT"
```

Set the state of an IO that is in output mode to output a 'high' or 'low' voltage:

```
dvIO4.io[0].output.value = True
```

```
dvIO4.io[0].output.value = False
```

Set the updates that an IO in input mode will generate:

```
dvIO4.io[0].inputMode.value = "DIGITAL"
```

```
dvIO4.io[0].inputMode.value = "ANALOG"
```

```
dvIO4.io[0].inputMode.value = "BOTH"
```

Get the state of an IO that is in digital input mode (inputMode set to DIGITAL or BOTH):

```
io1_state = dvIO4.io[0].digitalInput.value
```

To watch for state changes on this digital input:

```
handle_io_state_change(update)
```

```
print("IO state change to: ", update.value)
dvIO4.io[0].digitalInput.watch(handle_io_state_change)
```

Get the state of an IO that is in analog input mode (inputMode set to ANALOG or BOTH):

```
Io1_level = dvIO4.io[0].analogInput.value
```

You can set a watch event on the analog input in a similar fashion as the digital input above. Because the analog input may be constantly changing, you can set the minimum amount of voltage change that will cause a watch to trigger. To set this minimum threshold to 0.5v:

```
dvIO4.io[0].debounceMinDelta.value = 0.5
```

Set the maximum threshold voltage that is considered to be a digital 'low':

```
dvIO4.io[0].digitalInputLowMax.value = 0.1
```

Set the minimum threshold voltage that is considered to be a digital 'high':

```
dvIO4.io[0].digitalInputHighMin.value = 1.1
```

Set the state of the pull-up resistor used in the input configuration:

```
dvIO4.io[0].digitalInput2KPullup.value = True
dvIO4.io[0].digitalInput2KPullup.value = False
```

CE-Series 3rd-Party Control Protocol

Overview

The CE-Series 3rd-Party Control Protocol is a simplified set of messages that can be used by a 3rd-party control system (i.e. a control system that does not support the native HControl protocol) that provides full configuration and control of the CE control extenders over a simple socket connection.

TCP/IP Connection Information

The CE-Series Control Protocol occurs on TCP port 44197. It utilizes a raw TCP socket as opposed to a telnet or ssh connection.

Message Format

Messages are ASCII based and delimited with the newline (\n) character. The general format of a message is a **Message ID**, followed by a **path**, followed by zero or more **arguments**.

```
<message_id> <path> <argument(s)>\n
```

Each element of the message and all arguments are delineated by a single space. Each of these three elements of an Open Control message are defined below:

Message IDs

The message ID is one of the following:

Message ID	Direction	Success Response	Failure Response	Purpose
set	to Device	update	error	Set the value / state of a parameter
get	to Device	update	error	Read the current value / state of a parameter
subscribe	to Device	update	error	Subscribe to parameter change notifications
unsubscribe	to Device	None	error	Unsubscribe from parameter change notifications
update	to Controller	N/A	N/A	Response to set , get and subscribe messages
exec	to Device	None	error	Initiate an action that has no state or that requires more than one argument
event	to Controller	N/A	N/A	Triggered by stateless device events (e.g. received strings)
error	to Controller	N/A	N/A	Sent in response to all "controller to device" messages that fail

Note that **set** and **update** messages include only a single value parameter.

Some products provide **exec** commands (paths) that allow setting multiple parameters with a single message in the form:

```
exec <path> <arg1> <arg2> ... \n.
```

The **exec** command does not provide a response on success, but the user will receive individual **update** messages for any subscribed parameters that are changed.

The **subscribe** command will subscribe to any changes on a parameter or event. If the target is a parameter, it will also immediately generate an **update** message to give the subscriber the current value.

Unless there is an error, the **set** and **get** messages will always return an **update** message, even if you are not subscribed to that parameter.

The **event** message may return more than one value parameter depending on the event. It's important to use the `\n` message delimiter to ensure that all parameters are read.

The **error** message is sent if the CE device is unable to process a command, for example if the path is not recognized or if a parameter is missing or of the wrong type.

Paths

A path is a string identification of a specific parameter, command, or event. The path reflects a hierarchy of elements within a device from general to specific, with each element separated by a forward-slash (/). For working with the control ports on the CE boxes, the path consists of three elements:

```
/<port type>/<port number (1-based)>/<parameter, command or configuration item(s)>
```

For example, the path for the baud rate parameter on the first serial port of a CE-COM2 is:

```
/serial/1/baudRate
```

Note that the leading / is optional in all paths ('/serial/1/baudRate' is equivalent to 'serial/1/baudRate'), however any **update**, **event** or **error** message sent from the device will include the leading / in the path.

Values / Arguments

The values or arguments follow the path and are separated from each other by a single space. The exact arguments needed will depend on the message ID and the parameter or command being used. Arguments can be one of the following types:

Boolean

Boolean values should be either true or false with no surrounding quotes

Integer/Floating point

Numbers are represented in simple decimal notation with no surround quotes. For example: 42 or 3.1416

Enumerations

Enumerations are represented using the String format.

String

Strings may be quoted or unquoted. If the string contains any whitespace, it must be quoted.

Printable characters are ASCII values ($32 < c < 127$) and are sent as the character itself.

Non-printable characters must be escaped with a back-slash (\). The following escape sequences are supported (escape sequences are per-byte):

Description	Escape	ASCII Value (decimal)	Notes
Backslash character: \	\\	92	Necessary because the \ character is used to denote escapes.
Double quote: "	\"	34	To differentiate from quotation of the string itself.
Data (non-printable ASCII)	\x	($0 \leq c < 32$) and ($127 \leq c \leq 255$)	To allow for binary data to be present in a sting. The escaped value is in hexadecimal.

Binary Blob (Array of Bytes)

Binary data is sent as a quoted string utilizing the `\x` escape sequence for each byte.

Note that it is up to the user to know when data is to be interpreted as binary and therefore, escaped.

Device Specific Controls

NOTE that in the tables below the # character is used as a place holder for the specific port number. The # character MUST be replaced by the specific 1-based port number that is the intended target for the command. There is no wild-card character to address all ports. The examples provided for each product show correct usage.

CE-REL8 Controls

The CE-REL8 is the simplest device, containing only one parameter for control.

Parameters

Path	Read/Write	Data Type	Description
/relay/#/state	Read/write	Boolean	Open or Close the relay

Examples:

Turn relay 1 on

```
set /relay/1/state true\n
```

To subscribe to a relay state, use:

```
Subscribe /relay/1/state\n
```

Once subscribed you would receive an **update** message with the new state on any state change, for example:

```
update /relay/1/state true\n
```

CE-COM2 Controls

Commands

Path	Arguments	Date Type	Description
/serial/#/send	data	String	Send data out the port
/serial/#/setFlowControl	mode	Enumeration	NONE or HARDWARE
/serial/#/setCommParams	baudRate	Enumeration	1200, 4800, 9600, 19200, 38400, 57600, 115200
	dataBits	Integer	7 or 8
	stopBits	Integer	1 or 2
	parity	Enumeration	NONE, EVEN, ODD
	mode	Enumeration	232, 422, 485
/serial/#/enableReceive	-	-	Listen for message on the port
/serial/#/disableReceive	-	-	Stop listening for messages
/serial/#/flushReceiveBuffer	-	-	Clear incoming buffer

Examples:

To send a string out of COM Port 1:

```
exec /serial/1/send "Hello, world"\n
```

To send the string: 'This string contains "quoted text" and ° binary data'

```
exec /serial/1/send "This string contains \"quoted text\" and \xB0 binary data"\n
```

Notice that the double-quote characters within the string must be escaped and that the degree symbol can be sent as binary using hex code escape sequence `\xB0`

To disable the hardware handshaking available on COM Port 1, send:

```
exec /serial/1/setFlowControl NONE\n
```

To clear the receive buffer on COM Port 2, send:

```
exec /serial/2/flushReceiveBuffer\n
```

To configure the first COM port to the most common transmission settings, send:

```
exec /serial/1/setCommParams 9600 8 1 NONE 232\n
```

Events

Path	Arguments	Data Type	Description
/serial/#/receive	data	String	Received data from a connected device

Examples:

To enable strings sent to the COM port from the controlled device to be received by the control system it is required to BOTH enable receive strings on the port AND subscribe to the receive event. For example:

```
exec /serial/1/enableReceive\n
subscribe /serial/1/receive\n
```

Note that without the **subscribe** message, the strings will not be transmitted to the controller.

If a device connected to COM Port 1 were to send a string "Powering Off", that would be received at the controller as an **event** message as follows:

```
event /serial/1/receive "Powering Off"\n
```

Parameters

Path	Read/Write	Data Type	Values
/serial/#/baudRate	Read Only	Enumeration	1200, 4800, 9600, 19200, 38400, 57600, 115200
/serial/#/dataBits	Read Only	Integer	7 or 8
/serial/#/mode	Read Only	Enumeration	232, 422, 485
/serial/#/parity	Read Only	Enumeration	NONE, EVEN, ODD
/serial/#/stopBits	Read Only	Integer	1 or 2

Examples:

All of the COM2 parameters are read-only. These parameters are configured using the setCommParams command above.

Use the **get** message to read any of the parameters. The current value of the parameter will be returned in an **update** message:

```
get /serial/1/dataBits\n
update /serial/1/dataBits 8\n
```

When a parameter changes, an **update** message will also be sent to the controller for any parameter that has been subscribed to.

```
subscribe /serial/1/dataBits\n
```

CE-IRS4 Controls

Commands

Path	Arguments	Data Type	Description
/ir/#/bufferedSendIr	code	Integer	Queue a timed pulse of IR by index
/ir/#/bufferedSendNamedIr	code	String	Queue a timed pulse invoked by name
/ir/#/clearAndSendIr	code	Integer	Clear the queue, then send the new IR pulse
/ir/#/clearAndSendNamedIr	code	String	Clear the queue, then send the named IR pulse
/ir/#/keypadMacro	code	Integer	Queue digits based on the keypadMode pattern
/ir/#/keypadMode	mode	Integer	Set the pattern for keypad digit transmission
/ir/#/loadIrFile	file	String	The .irl filename to use
/ir/#/offIr	-	-	Turn off IR on the port
/ir/#/onIr	code	Integer	Send IR code continuously by #
/ir/#/onNamedIr	code	String	Send IR code continuously by name
/ir/#/setOffTime	millis	Integer	Set the queue timing interval in ms
/ir/#/setOnTime	millis	Integer	Set the queue active interval in ms
/ir/#/enableFaultDetection	-	-	Alert if no IR LED detected
/ir/#/disableFaultDetection	-	-	Do not alert
/ir/#/setCommParams	baudRate	Enumeration	1200, 4800, 9600, or 19200
	dataBits	Integer	7 or 8
	stopBits	Integer	1 or 2
	parity	Enumeration	NONE, ODD, or EVEN
/ir/#/send	data	String	Use IR as 1-way COM port

Examples:

To send on IR port 3 the IR code in position 1 for a pre-determined pulse time (or add to the queue if there are already bufferedSendIR or bufferedSendNamedIR or keypadMacro commands active) send:

```
exec /ir/3/bufferedSendIr 1\n
```

To send that same code if it is named 'PLAY' in the .irl file, send:

```
exec /ir/3/bufferedSendNamedIr "PLAY"\n
```

To send the codes on the number pad of the remote for station 345, send:

```
exec /ir/3/keypadMacro 345\n
```

To change the pattern used to execute keypad macros, use:

```
exec /ir/3/keypadMode 2\n
```

See the [Keypad Mode](#) section of the web server documentation for pattern information.

To load a new IR from the collection of .irl files uploaded to the CE-IRS4 configuration web page, send:

```
exec /ir/3/loadIrFile "samsung01.irl"\n
```

When using buffered (queued) IR, to configure a 1 second interval between IR pulses, send:

```
exec /ir/3/setOnTime 1000\n
```

When using buffered (queued) IR, to configure the active pulse width to 0.5 second, send:

```
exec /ir/3/setOffTime 500\n
```

Parameters

Path	Read/Write	Data Type	Values
/ir#/carrier	Read/Write	Boolean	true or false
/ir#/mode	Read/Write	Enumeration	IR, SERIAL, or DATA
/ir#/baudRate	Read Only	Enumeration	1200, 4800, 9600, 19200, 38400, 57600, 115200
/ir#/dataBits	Read Only	Integer	7 or 8
/ir#/parity	Read Only	Enumeration	NONE, EVEN, ODD
/ir#/stopBits	Read Only	Integer	1 or 2

Examples:

To set the IR port to use the carrier frequency specified in the .irl file (typically 41Hz modulation of the near infrared signal), send:

```
set /ir/3/carrier true\n
```

To set the IR port mode to use as a COM port, send:

```
set /ir/3/mode DATA\n
```

For the COM related parameters, see the CE-COM2 Parameters for syntax examples.

Events

Path	Arguments	Data Type	Description
/ir/#/fault	data	String	Alerts to missing or reversed IR bud

Examples:

The IRS4 will send a fault event if it detects certain fault conditions like a missing or reversed IR bud when attempting to send an IR message out a port. You must subscribe to this event to receive fault messages.

```
subscribe /ir/1/fault\n
```

If a fault is detected, the IRS4 will send an **event** message with a string argument describing the fault.

```
event /ir/1/fault <string>\n
```

CE-IO4 Controls

Parameters

Path	Read/Write	Data Type	Values
/io#/debounceTimeMilliseconds	Read/Write	Integer	5-250
/io#/mode	Read/Write	Enumeration	INPUT or OUTPUT
/io#/output	Read/Write	Boolean	true or false
/io#/debounceMinDelta	Read/Write	Float	0.1 - 4.9
/io#/inputMode	Read/Write	Enumeration	DIGITAL, ANALOG, BOTH
/io#/digitalInputLowMax	Read/Write	Float	0.0 - 9.9
/io#/digitalInput2KPullup	Read/Write	Boolean	true or false
/io#/analogInput	Read Only	Float	0.0 - 10.0
/io#/digitalInput	Read Only	Boolean	true or false
/io#/digitalInputHighMin	Read/Write	Float	0.1 - 10.0

Examples:

Set the debounce time (minimum time between updates) to 123ms:

```
set /io/1/debounceTimeMilliseconds 123\n
```

Set the mode for the particular input (input or output):

```
set /io/1/mode INPUT\n
```

```
set /io/1/mode OUTPUT\n
```

Set the state of an IO that is in output mode to output a 'high' or 'low' voltage:

```
set /io/1/output true\n
```

```
set /io/1/output false\n
```

Set the updates that an IO in input mode will generate:

```
set /io/1/inputMode DIGITAL\n
```

```
set /io/1/inputMode ANALOG\n
set /io/1/inputMode BOTH\n
```

Get the state of an IO that is in digital input mode (inputMode set to DIGITAL or BOTH):

```
get /io/1/digitalInput\n
```

The response would be as follows assuming the input is “high”:

```
update /io/1/digitalInput true\n
```

Get the state of an IO that is in analog input mode (inputMode set to ANALOG or BOTH):

```
get /io/1/analogInput\n
```

An example response would be:

```
update /io/1/analogInput 5.4\n
```

Set the minimum amount of voltage change that will cause an update to 0.5v:

```
set /io/1/debounceMinDelta 0.5\n
```

Set the threshold voltage that is considered to be a digital ‘low’:

```
set io/1/digitalInputLowMax 0.1\n
```

Set the threshold voltage that is considered to be a digital ‘high’:

```
set io/1/digitalInputHighMin 1.1\n
```

Set the state of the pull-up resistor used in the input configuration:

```
set io/1/digitalInput2KPullup true\n
set io/1/digitalInput2KPullup false\n
```



© 2026 HARMAN. All rights reserved. SmartScale, NetLinx, Enova, AMX, AV FOR AN IT WORLD, and HARMAN, and their respective logos are registered trademarks of HARMAN. Oracle, Java and any other company or brand name referenced may be trademarks/registered trademarks of their respective companies. AMX does not assume responsibility for errors or omissions. AMX also reserves the right to alter specifications without prior notice at any time. The AMX Warranty and Return Policy and related documents can be viewed/downloaded at www.amx.com.
3000 RESEARCH DRIVE, RICHARDSON, TX 75082 AMX.com | 800.222.0193 | 469.624.8000 | +1.469.624.7400 | fax 469.624.7153
Last Revised: 2026-03-06