SYN-CTL-HUB

Application Programming Interface

Document Revision	V2.0
Document Date	January 2026
Supported Firmware	2.0.4 and above



Contents

0. Device Communication 4
0.1 Overview 4
0.2 Network Connection 4
0.3 Command Delimiters for Sent Commands 4
0.4 RS232 Proxy Mode 4
1 Serial Port 5
1.1 Send Data Over UART (ACT Format) 5
1.2 Set UART Properties 5
1.3 Enable or Disable UART Proxy Function 6
1.4 Get UART Proxy Function Status 6
1.5 Set UART Proxy Mapping Port 6
1.6 Get UART Proxy Mapped Port 7
2 System Commands 7
2.1 Factory Reset 7
2.2 Device Reboot 8
2.3 Get Device Serial Number 8
2.4 Get Device Firmware Version 8
2.5 Get Device Model 9
2.6 Telnet Keepalive Heartbeat Mechanism 9
2.7 Set Telnet Keepalive Timeout 10
2.8 Get Telnet Keepalive Timeout Setting 10
2.9 Set Device to Report Proactively 11
2.10 Get Active Reporting Status 11
3 Network Configuration 12
3.1 Set Static IP 12
3.2 Set DHCP IP Address 12
3.3 Get IP Address 13
4 Services Configuration 13
4.1 Set Web Login Password 13
4.2 Set Telnet Service Port 14
4.3 Get Telnet Service Port 14
5 Infrared Commands 15
5.1 Send Infrared Data 15
6 GPIO Commands 15
6.1 GPIO Input Status Report 15
6.2 GPIO Input Status Query 16

7 Relay Commands 16
7.1 Relay Control 16
7.2 Get Relay Status 17

0. Device Communication

0.1 Overview

The following contains the connection and commands to control the SYN-CTL-HUB. By following the content contained in this document, the device can be controlled and configured via 3rd party control system.

⚠ IMPORTANT NOTE: Due to differences between WyreStorm device model versions within a series, some commands have different parameters based on the model and version. These differences are noted where applicable and should be followed, as sending an incorrect parameter may cause the unit to lock up and become inoperable.

0.2 Network Connection

The default IP address is **169.254.1.51** with subnet mask **255.255.255.0**, and the default port number is 23.

0.3 Command Delimiters for Sent Commands

All commands should be delimited with carriage return and line feed.

Accepted delimiter characters are:

Character	Shorthand	Hex Notation	Escape Notation	Decimal Notation
Line Feed	LF	0A	\n	10
Carriage Return + Line Feed	CR LF	0D 0A	\r\n	13 10

Please note, most 3rd party control software will either append these characters automatically or an option to specify them will be present. It is important that the last delimited character is LF and not CR.

0.4 RS232 Proxy Mode

RS232 “Proxy mode” allows you to send strings directly to your device. It is enabled by default, you can connect to port number 5001 via TCP Client (ie. 169.254.1.51:5001). Make sure you setup correct Baud Rate (or other RS232 port parameters) via API and that the wiring to the device is accurate. Proxy mode offers direct 2-way communication to the device via 3rd party controller or software, to make troubleshooting easier.

1 Serial Port

1.1 Send Data Over UART (ACT Format)

Send serial data over a specified UART port. Supports ASCII string or hexadecimal formats. The entire command must not exceed 4000 characters. This type of RS232 communication is deprecated by “Proxy Mode”

Command Structure
ACT UART_SEND <port_id> <format> <data>[CR/LF]

Response Structure
UART_SEND <port_id> <format> <data>[CR/LF]

Command Example

```
ACT UART_SEND 1 ASCII "hello world"[CR/LF]
“OR”
UART_SEND 1 ASCII "hello \"world\""[CR/LF]
“OR”
ACT UART_SEND 1 ASCII hello[CR/LF]
“OR”
ACT UART_SEND 1 HEX 68 65 6C 6C 6F[CR/LF]
```

Response Example

```
UART_SEND 1 ASCII "hello \"uart\" test"[CR/LF]
“OR”
ACT UART_SEND 1 ASCII "hello \"world\""
“OR”
UART_SEND 1 ASCII hello
“OR”
UART_SEND 1 HEX 68 65 6C 6C 6F
```

Parameters:
<port_id> = UART port number{1}
<format> = Data format - ASCII mode: Only printable ASCII characters (0x20–0x7E) are supported. Multibyte characters (e.g., Chinese) are not supported. - HEX mode: Hexadecimal mode: each byte is represented by two hex digits; case-insensitive. Bytes are separated by spaces. In HEX mode, <data> may be enclosed in double quotes or not; both forms are equivalent.
<data> = Data to send - ASCII mode: If the content contains spaces, double quotes, carriage return, or newline, it must be enclosed in double quotes. Supports four escape sequences: \" (double quote), \\ (backslash), \r (carriage return), \n (newline). No other escape sequences are supported. - HEX mode: <data> can be written as an unquoted sequence of hex bytes, or enclosed in double quotes; both forms are equivalent.

Additional Notes:

1.2 Set UART Properties

Set the UART (serial port) properties.

Command Structure
SET UART_CFG <port_id> <baud_rate> <data_bits> <parity> <stop_bits>[CR/LF]

Response Structure
UART_CFG <port_id> <baud_rate> <data_bits> <parity> <stop_bits>[CR/LF]

Command Example

```
SET UART_CFG 1 9600 8 NONE 1[CR/LF]
```

Response Example

```
UART_CFG 1 9600 8 NONE 1[CR/LF]
```

Parameters:
<port_id> = Serial port ID {1}
<baud_rate> = {9600, 19200, 38400, 57600, 115200}
<data_bits> = {7, 8}
<parity> = {NONE, ODD, EVEN}
<stop_bits> = {1, 2}

Additional Notes:

1.3 Enable or Disable UART Proxy Function

Enable or disable the proxy function on a specified UART (serial) port for transparent data passthrough control.

Command Structure **SET UART_PROXY_FN <port_id> <state>[CR/LF]**	Parameters: <port_id> = UART port number{1} <state> = {ON, OFF}
Response Structure **UART_PROXY_FN <port_id> <state>[CR/LF]**	
Command Example **SET UART_PROXY_FN 1 ON[CR/LF]**	
Response Example **UART_PROXY_FN 1 ON[CR/LF]**	
Additional Notes:	

1.4 Get UART Proxy Function Status

Query the current enabled status of the proxy function for the specified UART port.

Command Structure **GET UART_PROXY_FN <port_id>[CR/LF]**	Parameters: <port_id> = UART port number {1} <state> = {ON, OFF}
Response Structure **UART_PROXY_FN <port_id> <state>[CR/LF]**	
Command Example **GET UART_PROXY_FN 1[CR/LF]**	
Response Example **UART_PROXY_FN 1 ON[CR/LF]**	
Additional Notes:	

1.5 Set UART Proxy Mapping Port

Configure the logical mapping port used by the specified UART proxy function, to establish a passthrough binding with internal or external system ports.

Command Structure **SET UART_PROXY_PORT <port_id> <net_port>[CR/LF]**	Parameters: <port_id> = Serial port ID {1} <net_port> = Network port
Response Structure **UART_PROXY_PORT <port_id> <net_port>[CR/LF]**	
Command Example **SET UART_PROXY_PORT 1 5001[CR/LF]**	
Response Example **UART_PROXY_PORT 1 5001[CR/LF]**	
Additional Notes:	

1.6 Get UART Proxy Mapped Port

Query the target port number (mapped port) currently configured for the proxy function of the specified UART.

Command Structure **GET UART_PROXY_PORT <port_id>[CR/LF]**	Parameters: <port_id> = Serial port number {1} <net_port> = Network port
Response Structure **UART_PROXY_PORT <port_id> <net_port>[CR/LF]**	
Command Example **GET UART_PROXY_PORT 1[CR/LF]**	
Response Example **UART_PROXY_PORT 1 5001[CR/LF]**	
Additional Notes:	

2 System Commands

2.1 Factory Reset

Resets the device to factory default settings.

Command Structure **RESET[CR/LF]**	Parameters:
Response Structure **RESET[CR/LF]**	
Command Example **RESET[CR/LF]**	
Response Example **RESET[CR/LF]**	
Additional Notes:	

2.2 Device Reboot

Reboots the device.

Command Structure **REBOOT[CR/LF]**	Parameters:
Response Structure **REBOOT[CR/LF]**	
Command Example **REBOOT[CR/LF]**	
Response Example **REBOOT[CR/LF]**	
Additional Notes:	

2.3 Get Device Serial Number

Get the device serial number.

Command Structure **GET SYS_SN[CR/LF]**	Parameters: <sn> = Device serial number
Response Structure **SYS_SN <sn> [CR/LF]**	
Command Example **GET SYS_SN [CR/LF]**	
Response Example **SYS_SN 112233445566 [CR/LF]**	
Additional Notes:	

2.4 Get Device Firmware Version

Retrieve the device's firmware version.

Command Structure **GET VER[CR/LF]**	Parameters: <version> = VX.Y.Z; X <= 999, Y <= 999, Z <= 999
Response Structure **VER <version>[CR/LF]**	
Command Example **GET VER[CR/LF]**	
Response Example **VER V2.0.1[CR/LF]**	
Additional Notes:	

2.5 Get Device Model

Retrieve the device model.

Command Structure **GET DEVICE_MODEL [CR/LF]**	Parameters: <model> = Model number of the device
Response Structure **DEVICE_MODEL <model>[CR/LF]**	
Command Example **GET DEVICE_MODEL[CR/LF]**	
Response Example **DEVICE_MODEL SYN-CTL-HUB[CR/LF]**	
Additional Notes:	

2.6 Telnet Keepalive Heartbeat Mechanism

Used to maintain a Telnet channel connection and carry keepalive messages. Can be used for link state detection, latency analysis, or custom identification.

Command Structure **KEEPALIVE <msg>[CR/LF]**	Parameters: <msg> = any string (e.g., timestamp, ID, "ping", or other keepalive info), case-sensitive, maximum length 64 bytes
Response Structure **KEEPALIVE <msg>[CR/LF]**	
Command Example **KEEPALIVE 1[CR/LF]**	
Response Example **KEEPALIVE 1[CR/LF]**	
Additional Notes:	

2.7 Set Telnet Keepalive Timeout

Set the device's timeout threshold for receiving the KEEPALIVE command. If a KEEPALIVE is not received within this timeout, the connection is considered lost. A value of 0 disables the keepalive function.

Command Structure **SET KEEPALIVE_TIMEOUT <timeout>[CR/LF]**	Parameters: <timeout> = {0, 10 ~ 86400}, timeout value in seconds; 0 disables the keepalive timeout mechanism
Response Structure **KEEPALIVE_TIMEOUT <timeout>[CR/LF]**	
Command Example **SET KEEPALIVE_TIMEOUT 60[CR/LF]**	
Response Example **KEEPALIVE_TIMEOUT 60[CR/LF]**	
Additional Notes:	

2.8 Get Telnet Keepalive Timeout Setting

Query the current Telnet channel receive timeout (in seconds). This determines how long a lack of incoming messages will trigger a disconnect.

Command Structure **GET KEEPALIVE_TIMEOUT[CR/LF]**	Parameters: <timeout> = {0, 10 ~ 86400}, timeout in seconds; 0 disables the keepalive timeout mechanism
Response Structure **KEEPALIVE_TIMEOUT <timeout>[CR/LF]**	
Command Example **GET KEEPALIVE_TIMEOUT[CR/LF]**	
Response Example **KEEPALIVE_TIMEOUT 60[CR/LF]**	
Additional Notes:	

2.9 Set Device to Report Proactively

Configure the device to proactively report its parameters to all front ends.

Command Structure **SET REPORT_FN <state>[CR/LF]**	Parameters: <state> = {ON, OFF} — ON: enable proactive reporting; OFF: disable proactive reporting.
Response Structure **REPORT_FN <state>[CR/LF]**	
Command Example **SET REPORT_FN ON[CR/LF]**	
Response Example **REPORT_FN ON[CR/LF]**	
Additional Notes:	

2.10 Get Active Reporting Status

Get the active reporting status.

Command Structure **GET REPORT_FN[CR/LF]**	Parameters:
Response Structure **REPORT_FN <state>[CR/LF]**	
Command Example **GET REPORT_FN[CR/LF]**	
Response Example **REPORT_FN ON[CR/LF]**	
Additional Notes:	

3 Network Configuration

3.1 Set Static IP

Set static IP address

Command Structure **SET IPADDR STATIC IP4ADDR <xx.xx.xx.xx> NETMASK <xx.xx.xx.xx> GATEWAY <xx.xx.xx.xx>[CR/LF]**	Parameters:
Response Structure **IPADDR STATIC IP4ADDR <xx.xx.xx.xx> NETMASK <xx.xx.xx.xx> GATEWAY <xx.xx.xx.xx>[CR/LF]**	
Command Example **SET IPADDR STATIC IP4ADDR 192.168.11.243 NETMASK 255.255.255.0 GATEWAY 192.168.11.1[CR/LF]**	
Response Example **IPADDR STATIC IP4ADDR 192.168.11.243 NETMASK 255.255.255.0 GATEWAY 192.168.11.1[CR/LF]**	
Additional Notes:	

3.2 Set DHCP IP Address

Configure the device to obtain its IP address via DHCP.

Command Structure **SET IPADDR DHCP[CR/LF]**	Parameters:
Response Structure **IPADDR DHCP[CR/LF]**	
Command Example **SET IPADDR DHCP[CR/LF]**	
Response Example **IPADDR DHCP[CR/LF]**	
Additional Notes:	

3.3 Get IP Address

Retrieve the device IP address and its configuration (DHCP or static).

Command Structure **GET IPADDR[CR/LF]**	Parameters:
Response Structure **IPADDR DHCP IP4ADDR <ipv4> NETMASK <netmask> GATEWAY <gateway>[CR/LF]** **"OR"** **IPADDR STATIC IP4ADDR <ipv4> NETMASK <netmask> GATEWAY <gateway>[CR/LF]**	
Command Example **GET IPADDR[CR/LF]**	
Response Example **IPADDR DHCP IP4ADDR 192.168.11.243 NETMASK 255.255.255.0 GATEWAY 192.168.11.1[CR/LF]** **"OR"** **IPADDR STATIC IP4ADDR 192.168.11.243 NETMASK 255.255.255.0 GATEWAY 192.168.2.1[CR/LF]**	
Additional Notes:	

4 Services Configuration

4.1 Set Web Login Password

Set the web login password.

Command Structure **SET WEBPWD <name> <oldpwd> <newpwd>[CR/LF]**	Parameters: <name> = username <oldpwd> = old password <newpwd> = new password
Response Structure **WEBPWD <name> <oldpwd> <newpwd>[CR/LF]**	
Command Example **SET WEBPWD admin theCurrPassword theNewPassword[CR/LF]**	
Response Example **WEBPWD admin theCurrPassword theNewPassword[CR/LF]**	
Additional Notes:	

4.2 Set Telnet Service Port

Set the listening port number for the Telnet service.

Command Structure **SET TELNET_PORT <net_port>[CR/LF]**	Parameters: <net_port> = port number (1–65535)
Response Structure **TELNET_PORT <net_port>[CR/LF]**	
Command Example **SET TELNET_PORT 2323[CR/LF]**	
Response Example **TELNET_PORT 2323[CR/LF]**	
Additional Notes:	

4.3 Get Telnet Service Port

Get the port number the Telnet service is currently listening on.

Command Structure **GET TELNET_PORT[CR/LF]**	Parameters: <net_port> = {1 ~ 65535} (valid port range: 1 to 65535)
Response Structure **TELNET_PORT <net_port>[CR/LF]**	
Command Example **GET TELNET_PORT[CR/LF]**	
Response Example **TELNET_PORT 2323[CR/LF]**	
Additional Notes:	

5 Infrared Commands

5.1 Send Infrared Data

Send infrared data.

Command Structure **ACT IR_SEND <port_id> <format> <ir_code>[CR/LF]**	Parameters: <port_id> = port number {1} <format> = {0, 1}; 0 = custom pulse format 1 = Pronto hex format <ir_code> = IR code string in the specified format
Response Structure **IR_SEND <port_id> <format> <ir_code>[CR/LF]**	
Command Example **ACT IR_SEND 1 1 0000 0067 0000 0015 0060 0018 0018 0018 0030 0018 0030 0018 0030 0018 0018 0018 0018 0018 0018 0018 0018 0030 0018 0018 0018 0030 0018 0030 0018 0030 0018 0030 0018 0030 0018 0030 0018 0018 0018 0030 0018 0030 0018 0018 01d9[CR/LF]**	
Response Example **IR_SEND 1 1 0000 0067 0000 0015 0060 0018 0018 0018 0030 0018 0030 0018 0030 0018 0018 0018 0018 0018 0018 0018 0018 0018 0030 0018 0018 0018 0030 0018 0030 0018 0030 0018 0030 0018 0030 0018 0030 0018 0018 0018 0030 0018 0030 0018 0018 01d9[CR/LF]**	
Additional Notes:	

6 GPIO Commands

6.1 GPIO Input Status Report

Reports the status of a GPIO input.

Command Structure	Parameters: <port_id> = port number {1} <state> = {HIGH, LOW}
Response Structure **GPIO_IN_STATUS <port_id> <state>[CR/LF]**	
Command Example	
Response Example **GPIO_IN_STATUS 1 HIGH[CR/LF]**	
Additional Notes:	

6.2 GPIO Input Status Query

Query GPIO input status.

Command Structure **GET GPIO_IN_STATUS <port_id>[CR/LF]**	Parameters: <port_id> = Port number {1} <state> = {HIGH, LOW}
Response Structure **GPIO_IN_STATUS <port_id> <state>[CR/LF]**	
Command Example ****	
Response Example **GPIO_IN_STATUS 1 HIGH[CR/LF]**	
Additional Notes:	

7 Relay Commands

7.1 Relay Control

Controls the relay state for a specified port. Can be set to OPEN or CLOSE.

Command Structure **SET RELAY_CTL <port_id> <mode>[CR/LF]**	Parameters: <port_id>: Relay port number, {1, 2, 3, 4} <mode>: Relay control mode: OPEN = causes the relay to open (disconnect) CLOSE = causes the relay to close (connect)
Response Structure **RELAY_CTL <port_id> <mode>[CR/LF]**	
Command Example **SET RELAY_CTL 1 OPEN[CR/LF]**	
Response Example **RELAY_CTL 1 OPEN[CR/LF]**	
Additional Notes:	

7.2 Get Relay Status

Get the current operating state (OPEN or CLOSE) of the relay at the specified port.

Command Structure **GET RELAY_CTL <port_id>[CR/LF]**	Parameters: <port_id>: Relay port number, {1, 2, 3, 4} <mode>: Relay control mode: OPEN = open the relay (disconnect) CLOSE = close the relay (connect)
Response Structure **RELAY_CTL <port_id> <mode>[CR/LF]**	
Command Example **GET RELAY_CTL 1[CR/LF]**	
Response Example **RELAY_CTL 1 OPEN[CR/LF]**	
Additional Notes:	

Publication Disclaimer:

The material contained in this document consists of information that is the sole property of WyreStorm. This document is intended to provide information to allow interfacing to the relevant WyreStorm equipment by third party products

WYRESTORM IS NOT RESPONSIBLE FOR MALFUNCTIONS AND/OR THE IN-OPERABILITY WHICH MAY BE CAUSED BY THE APPLICATION OF THIS INFORMATION, WHETER EXPECTED OR NOT.

WyreStorm reserves the right to change software, control codes and specifications without notice.

WyreStorm will not be liable for any use of this information or any changes it may make to these products. The use of this information constitutes an agreement by the user to these limitations and exclusions.

WyreStorm Offices
North America: 27 Wood Road, Round Lake NY
12151 Tel: +1 518-289-1293

EMEA: Unit 22, Ergo Business Park, Swindon, Wiltshire, SN3
3JW, UK Tel: +44 (0) 1793230343

WyreStorm Technologies reserves the right to change physical appearance or technical specification of this product at any time. Visit wyrestorm.com for the latest product information.