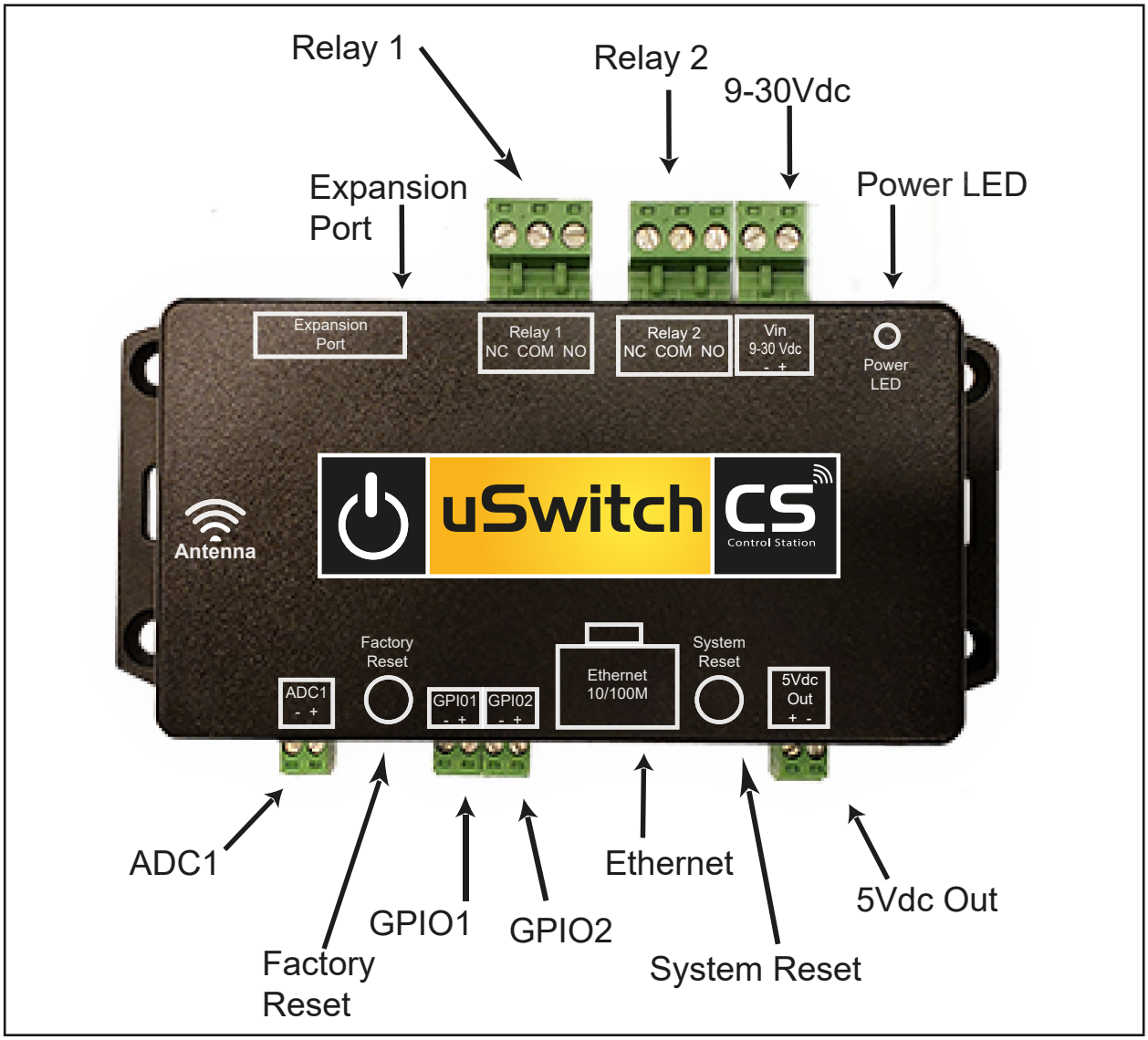
uSwitch CS™ API



uSwitch
by uHave Control

The uSwitch CS API Overview

The uSwitchCS provides a universal remote programmers API. Custom computer applications use this API to remotely Connect to, Control and Monitor our devices from within your custom application. The interface with the uSwitchCS API is an HTTP Client connection. Once an application establishes this connection, the uSwitch provides a comprehensive XML/HTTP command-based API.

An example of the XML API commands, described below, make up the remote programming API.

To acquire or control the uSwitchCS remotely from an application open a TCP/IP HTTP or HTTPS Client connection. There are two styles of XML commands for the uSwitchCS. Input commands and output commands. Input commands obtain status, control, and sensor information from the uSwitchCS. Output commands send either control, or configuration commands. XML commands to the uSwitch are sent packaged in an HTML standard GET request. There is a general command to acquire all data from the uSwitch, and selective commands which to acquire data unique to a specific feature or device. Once command XML data is received by the remote application it parses for the data it needs and may use it in whatever manner necessary. Output commands are used to change a setting or control output.

To send a command to set a relay or configure a device an XML output command is sent. Such as “<url>/UHCapi.xml?cmd=relay; Relay1=0”, where <url> is the devices url such as “https://192.168.0.1/. This command de-energizes “Relay1”. Similarly, “<url>/UHCapi.xml?cmd=relay;Relay1=1”, energizes the relay named “Relay1”. For any relay commands their assigned name must be used to program the relay. For instance, if relay1 had been assigned the name, “FrontGate” then “<url>/UHCapi.xml?cmd=relay;FrontGate=0”, will open or close the relay attached to the Front Gate access control device. Valid Relay output values for this command are any of the following 0, 1.

More complex commands may require additional fields. An example is the Pulse command which requires a time parameter. The following is an example of the pulse command: “control.xml?pulse01=1&time=5”, as stated above this command would operate identical to “control.xml?doorLock=0&time=5”.

Note correct case is required. If names used or commands are not in the correct upper or lower case or the commands themselves have incorrect case the result is the command will be ignored or may generate unpredictable results.ave incorrect case the result is the command will be ignored or may generate unpredictable results.

To set specific Relays Value(s): 192.168.0.190/UHCapi.xml?cmd=relay&<relay Name>=<0/1>&<relay_na-me>=<0/1>; …

To read all Relay Settings: http://192.168.4.1/UHCapi.xml?cmd=relay&GetAll

<Relay1-i>On</Relay1-i>
<Relay2-i>Off</Relay2-i>
<Relay3-e>Off</Relay3-e>
<Relay4-e>Off</Relay4-e>

Read a specific ADC's Inputs Value: http://192.168.4.1/UHCapi.xml?cmd=adc&Vin2&Vin3&Temp7...

Returns:

<Vin2>3.32 Vdc </Vin2>
<Vin3>5.5 Vdc </Vin3>
<Temp7>72 F </Temp7>

Read all ADC Input Values: http://192.168.4.1/UHCapi.xml?cmd=adc&GetAll

Returns:

<vIn1-i>0.63 Vdc </vIn1-i>
<Vin2-e>3.32 Vdc </Vin2-e>
<Vin3-e>-0.00 Vdc </Vin3-e>
<Vin4-e>3.32 Vdc </Vin4-e>
<Vin5-e>0.00 Vdc </Vin5-e>
<Temp1-e>70.42 F </Temp1-e>
<Humidity1-e>52.03 RH </Humidity1-e>
<Pressure1-e>30.10 inHg </Pressure1-e>
<Temp2-e>70.42 F </Temp2-e>
<Humidity2-e>58.43 RH </Humidity2-e>
<Lux1-e>235.93 B </Lux1-e>

Read a specific GPIO Inputs Values: http://192.168.4.1/UHCapi.xml?cmd=gpio&GPIO1-i&-GPIO2-i;...;

Returns:

```
<i>
<GPIO1-i>On</GPIO1-i>
<GPIO2-i>Start</GPIO2-i>
</i>
```

Read all GPIO Values: http://192.168.4.1/UHCapi.xml?cmd=gpio&GetAll

Returns:

```
<i>
<GPIO1-i>On</GPIO1-i>
<GPIO2-i>Start</GPIO2-i>
<FactoryReset>Clear</FactoryReset>
<GPIO3-e>Start</GPIO3-e>
<GPIO4-e>Start</GPIO4-e>
<GPIO5-e>Start</GPIO5-e>
<GPIO6-e>Start</GPIO6-e>
<GPIO7-e>Stop</GPIO7-e>
<GPIO8-e>Stop</GPIO8-e>
<GPIO9-e>Stop</GPIO9-e>
<GPIO10-e>Stop</GPIO10-e>
</i>
```

Creating Custom Interfaces and Applications

Every feature of the uSwitchCS is configurable through XML via an HTTP library or interface such as Curl. The following Tables show HTTP GET requests that are used to program a uSwitchCS. All GET Requests use the parm=value protocol. Multiple parm/value strings can be sent in a single Get Request by separating each parm=value pair with an ampersand (&). Sent requests follow the standard HTTP Get format

GET Request?parm=value HTTP/1.1
(Bracketed fields are user specified, other fields are fixed)

Request	-	Specific Configuration command
parm	-	parameter or variable to program
value	-	a data value of the types shown below

The following types represent the different value fields by type. Fields within square brackets represent a type of value within a range. Fields within braces represent a specific set of values that are allowed. Items in braces represent a list of acceptable values, and only a single value from the list may be chosen for a given parameter.

[string]	-	alpha-numeric text string
[int]	-	any numeric string
[a..b]	-	numeric string in the range of ‘a’ to ‘b’
[float]	-	numeric string including a decimal point
{port}	-	integer value from {1..65535}
{bool}	-	on/off
{color}	-	Yellow/Green/Red/Black/LightGray/Cyan/Purple/Tan/Gray/Magenta/Blue/Brown
{POS}	-	Off/On/Last
{relayMode}	-	Manual/Resettable/Clearable/Momentary/Scheduled/Event/Watchdog/Slave
{vRMode}	-	None/Manua/Resettable/Clearable/Momentary/Relay/Event
{gpioMode}	-	Binary/Counter/Rate
{i/o name}	-	Text name of an existing I/O (GPIO or ADC)
{NetworkType}	-	Station/AccessPoint/APStation/Ethernet
{URL}	-	an IP address such as “192.168.0.29” or a URL such as “https://cdn.shopify.-com”
{WifiSecurity}	-	Open/WEP/WPA-PSK/WPA2-PSK/WPA-WPA2-PSK/WPA2-ENTERPRISE
{mail trigger}	-	Off/On/Change/None
{input trigger}	-	Low/High/Any
{ADC Max}	-	11/15/22/30 (max dc input voltage for internal ADCs)
{load}	-	Pull-Up/Pull-Down/Float

{LTimeout} - 0/15/30/60/120 (login timeout)
{emailMode} - Off/Standard/Detailed
{measures} - English/Metric
{weekday} - Sunday/Monday/Tuesday/Wednesday/Thursday/Friday/Saturday
{month} - January/February/March/April/May/June/July/August/September/October/
November/December
{cellCarrier} - None/AT&T/BoostMobile/C-Spire/Consumer/Cricket/Google-Fi/Metro-PCS/MintMobile/ PagePlus/RedPocket/Republic/Rogers/SimpleMobile/Sprint/Suncom/Qwest/T-Mobile/
Ting/TracPhone/Telus/U.S.Cellular/Verizon/Virgin/Xfinity
{region} - UnitedStates/Austria/Australia/Belgium/Brazil/Canada/Switzerland/China/Cyprus/Czecheslovakia/
Germany/Denmark/Estonia/Spain/Finland/France/Britain/Greece/HongKong/Hungary/Indonesia/
Ireland/Israel/India/Iceland/Italy/Japan/SouthKorea/Lithuania/Luxembourg/Latvia/Malaysia/Netherlands/
Norway/NewZealand/Philippines/Poland/Portugal/Sweden/Singapore/Slovenia/Slovakia/Thailand/Taiwan/
SouthAfrica/NonCountry

XML Relay Settings Commands

(per Relay Settings Menu)

GET **ConfigRelays**?RELAY_PARM[x]=[value] HTTP/1.1
(Bracketed fields user specified, other fields are fixed)

Description	parm/value	Example usage/Comments
Assign a name to a relay	relN[i] =[string]	; configRelays?relN2=Entrance
Assign Text String to De-energized State	relOff[i]=[string]	; configRelays?relOff1=Open
Set Relay to Operational Mode	relMode[i]={rMode}	; configRelays?relMode1=Manual
Enable Relay Latching (event Mode Only)	relLat[i]={bool}	; configRelays?relLat1=off
Assign Text String to Energized State	relOn[i]=[string]	; configRelays?relOn1=Locked
Assign Color for Relay De-energized State	relLoClr[i]={color}	; configRelays?relLoClr1=Yellow
Assign Color for Relay Energized State	relHiClr[i]={color}	;
Assign a momentary time for relay	relMom[i]=[int]	; configRelays?relMom=500
Set Relay to Power On State	relPOS[i]={POS};	
Synchronize Relay to an Input	relSync[i]={i/o name}	;
Display Relay on Control Page	relShow[i]={bool}	;
Add Relay Pulse, Watchdog Override	relP[i]={bool}	;
Send Email on Relay Event	relEm[i]=[mail trigg]	;
Set Relay Opposite Event	relOp[i]={bool}	;

Multiple parameters can be sent in a single GET request type by separating each parameter/value string with an ampersand (&) such as:

XML Digital I/O Settings Commands

(per Digital I/O Settings Menu)

GET **configInputs**?GPIO_PARM[x]=[value] HTTP/1.1

Description	**parm/value**	**Example usage/Comments**
Assign a name to a GPIO	gpN[i]=[string]	; configInputs?gpN1=Door1
Assign Operating Mode to GPIO	gpMode[i]={gpioMode}	; configInputs?pgMode1=Rate
Assign a Measurement unit to GPIO	gpU[i]=[string]	;
Assign Text String to Low State of GPIO	gpOff[i]=[string]	;
Assign Text String to High State of GPIO	gpOn[i]=[string]	;
Assign Color for GPIO Low State	gpLoClr[i]={color}	;
Assign Color for GPIO Target State	gpHysClr[i]={color}	; Rate Mode Only
Assign Color for GPIO High State	gpHiClr[i]={color}	;
Assign load State to GPIO	gpPD[i]={load}	; configInputs?gpPD1=Pull-Up
GPIO Triggering Event	gpTR[i]={gpio trigger}	;
Assign Debounce Value to GPIO	gpDB[i]=[int]	; value in ms
Assigne Total Cycle Time for GPIO Rate	gpTI[i]=[i]	; Rate Cycle Time (rate mode only)
Assign Cuttoff or Rollover count value	gpRCtr[i]=[float]	;
GPIO Target Value	gpHV[i]=[float]	;
GPIO Calibration Slope	gpSlope[i]=[float]	;
GPIO Calibration Offset	gpOffset[i]=[float]	;
GPIO Tolerance Value	gpDMZ[i]=[float]	;
GPIO Decimal places	gpDP[i]=[0..3]	; max 2
Show GPIO on Control Page	gpShow[i]={bool}	;
GPIO email on	gpEM[i]={bool}	;

XML Analog I/O Settings Commands

(per Analog I/O Settings Menu)

GET **configADCs**?ADC_PARM[x]=[value] HTTP/1.1

Description	**parm/value**	**Example usage/Comments**
Assign name to adc	adcN[i]=[string]	;
Assign measurement units tag	adcU[i]=[string]	; measurement Units
ADC color Low Target Range	adcLoClr[i]={color}	;
ADC colorTarget Range	adcMiClr[i]={color}	;
ADC color in Above Target Range	adcHiClr[i]={color}	;
ADC slope Calibration	adcSlope[i]=[float]	;
ADC Offset Calibration	adcOffset[i]=[float]	;
ADC Decimal Points	adcDP[i]=[0..3]	;
ADC Event Trigger	adcT[i]={input trigger}	;
ADC Target Value	adcTarg[i]=[float]	;
ADC Hysteresis Value	adcDMZ[i]=[float]	;
Settling Time for Alarm Event	adcSet[i]=[int]	;
Internal ADC Attenuation Value	adcAtn[i]={ADC Max}	;
Internal ADC hardware Samples	adcSamp[i]=[int]	; Sets number of Samples for Internal ADC
Show ADC on Control Page	adcShow[i]={bool}	;
Enable Notification for Email	adcEM[i]={bool}	;
Enable Internal ADC	adcEn[i]={bool}	;

XML Virtual Relay Settings Commands

(per Virtual Relay Settings Menu)

GET **configVRelays?**VRelay_PARM[x]=[value] HTTP/1.1

Description	**parm/value**	**Example usage/Comments**
Assign Virtual Relay [i] Name	vRel[i]=[string]	; configVRelays?vRel1=Master1
Specify Virtual Relay [i] Mode	vRMode[i]={vRMode}	;configVRelays?vRMode1=Resettable
Assign Text String to Low State VR [i]	vROn[i]=[string]	;
Assign Text String to Low State VR [i]	vROff[i]=[string]	;
Assign Low State color to VR [i]	vRLoClr[i]={color}	;
Assign High State color to VR [i]	vRHiClr[i]={color}	;
Momentary Second Timer VR[i]	vRMom[i]=[int]	;
Set Power on State for VR [i]	vRPOS[i]={POS}	; configVRelays?vRPOS1=Last
IP address to First Slave Relay	vRIP[i]=[string]	;
Consecutive Slaves for VR [i]	vRCnt[i]=[int]	;
TCP Command port for VR [i] slaves	vRPort[i]={port}	; configVRelays?vRPort1=9673
Slave Relays to Control by VR [i]	vRID[i]=[int]	;
Enable Sub Controls for Slave Relays	vRelP[i]={bool}	;
Notifications on Virtual Relays	vREmail[i]={bool}	;

XML Watchdog Settings Commands

(per Relay Watchdogs Menu)

GET **configWatch**?PARM=[value] HTTP/1.1

Description	parm/value	Example usage/Comments
Watchdog Relay[i] URL1 for Relay[i]	wdURL0[i]={URL}	; configWatch?wdURL00=google.com
Watchdog Relay[i] URL2 for Relay[i]	wdURL1[i]={URL}	;
Watchdog Relay[i] URL3 for Relay[i]	wdURL2[i]={URL}	;
Watchdog Relay[i] Input Heartbeat	wdIO[i]={i/o name}	; Only Specify if using input as Heartbeat
Watchdog Notifications	wdEmail[i]={bool}	;
Watchdog Relay[i] Ping All URL(s)	wdAll={bool}	;
Watchdog Relay[i] Max Reboot Attempts	wdMRA[i]=[int]	;
Watchdog Relay[i] Max Ping Failures	wdMPF[i]=[int]	;
Watchdog Relay[i] pulse delay minutes	wdPDMin[i]=[int]	;
Watchdog Relay[i] pulse delay seconds	wdPDSec[i]=[int]	;
Watchdog Relay[i] startup delay minutes	wdSDMin[i]=[int]	;
Watchdog Relay[i] startup delay seconds	wdSDSec[i]=[int]	;
Watchdog Relay[i] reboot cycle minutes	wdRMin[i]=[int]	;
Watchdog Relay[i] reboot cycle seconds	wdRSec[i]=[int]	;
Watchdog Relay[i] fault mode hours	wdFHr[i]=[int]	;
Watchdog Relay[i] fault mode seconds	wdFMin[i]=[int]	;
Watchdog Relay[i] fault mode seconds	wdFSec[i]=[int]	;

XML Security & Port Settings Commands

(per Security & Port Menu)

GET **configAuth?**PARM=[value] HTTP/1.1

Description	**parm/value**	**Example usage/Comments**
User Password	pass=[string]	; configAuth?pass=password
2nd Entry User Password	repass=[string]	; Must submit User Password twice for acceptance
Admin Password	apass=[string]	; Admin Password String
2nd Entry Admin Password	srepass=[string]	; Must submit Admin Password twice for acceptance
Hide Passwords from User	brwPassHide={bool}	;
Require Passwords to Login	brwPassChk={bool}	;
Send Notification on Login	brwLoginEM={bool}	;
Use HTTP	brwHTTP={bool}	;
Web Server's HTTPS Port	brwSport={port}	; configAuth?brwSport=465
Web Server's HTTP Port	brwHport={port}	; configAuth?brwHpot=80
TCP Command Port	brwTport={port} ;	
Brower Timeout after N minutes (0=None)	brwLTS={LTimeout}	;
Show User Entered Passwords	brwPassHide={bool}	;
Software Watchdog Enable	brwSoftWDG={bool}	;
Network Watchdog Enable	brwNetWDG={bool}	;

XML Notification/Email Settings Commands

(per Notification Menu)

GET **configSMTP**?PARM=[value] HTTP/1.1

Description	parm/value	Example usage/Comments
Specify Notification Mode	smtpMode={emailMode}	;configSMTP?smptMode=Standard
Specify Mail Server	smtpServer=[string]	; configSMTP?smtpServer =smtp.gmail.com
Specify Mail Server Port	smtpPort={port}	; configSMTP?smtpPort=465
Set if Mail Server Requires Authentication	smtpAuth={bool}	; configSMTP?smtpAuth=on
Specify Mail Account User Name	smtpUser=[string]	; configSMTP?smtpUser= joe.smith
Specify Mail Account User Password	smtpPwd=[string]	; configSMTP?smtpPwd=admin
Specify Email of mail Account Sender	smtpSender=[string]	; configSMTP?smtpSender= send@gmail.com
Name of First Mail Recipient	smtpRcv0=[string]	; configSMTP?smtpRcv0= joseph@gmail.com
Name of First Mail Recipient	smtpRcv1=[string]	;
Name of First Mail Recipient	smtpRcv2=[string]	;
Name of First Mail Recipient	smtpRcv3=[string]	;configSMTP?smtpRcv3= 6177809000
Name of First Mail Recipient	smtpRcv4=[string]	;
Specify Recipient 4's Cell Phone Carrier	smtpCell3={cellCarrier}	; configSMTP?smtpCell3= Verizon
Specify Recipient 5's Cell Phone Carrier	smtpCell4={cellCarrier}	;
Send Notification on system Reboot	smtpBoot={bool}	;
Send Notification on Configuration Update	smtpCfg={bool}	;

XML Log Settings Commands

(per Log Settings Menu)

GET **configLog**?PARM=[value] HTTP/1.1

Description	parm/value	Example usage/Comments
Enable Logging	logEn={bool}	;configLog?logEn=on
Log Configuration Changes	logCC={bool}	;
Log Relay Changes	logRC={bool}	;
Log all Reboots	logBoot={bool}	;configLog?logBoot=on
Log any Watchdog Reboots	logWR={bool}	;
Log any Watchdog Faults	logWF={bool}	;configLog?logWF=on
Log any Page Changes	logPC={bool}	;
Log any User/Admin Logins	logLI={bool}	;
Log any Flash Updates	logFSW={bool}	;
Add User Name to Logged Events	logUse={bool}	;

XML Time & Date Settings Commands

(per Time/Date Settings Menu)

GET **configTD**?PARM=[value] HTTP/1.1

Description	parm/value	Example usage/Comments
Enable NTP	tmNTP={bool}	;
Primary NTP Server address	tmNTP0=[url]	;configTD?tmNTP0=north-america.pool.ntp.org
Secondary NTP Server address	tmNTP1=[url]	;
Days Between FTP update requests	tmUF=[int 1..20]	;
Set Time Format 12 or 24 Hour	tmTF=[12/24]	;configTD?tmTF=24
Time Zone	tmTZ=[int -12..14]	;configTD?tmTZ=-5
Enable Daylight Savings	tmDL={bool}	;
Daylight Savings Start Month	tmDLM0={month}	;
Daylight Savings Start Week	tmDLW0=[1..5]	;
Daylight Savings Start Weekday	tmDLWD0={weekday}	;configTD?tmDLWD0=Sunday
Daylight Savings Start Hour	tmDLHr0=[0..23]	;
Daylight Settings End Month	tmDLM1={month}	;configTD?tmDLM1=March
Daylight Savings End Week	tmDLW1=[1..5]	;
Daylight Savings End Weekday	tmDLWD1={weekday}	;
Daylight Savings End Hour	tmDLHr1=[0..23]	;

XML User Interface Settings Commands

(per User Interface Menu)

GET **configSystem**?PARM=[value] HTTP/1.1

Description	**parm/value**	**Example usage/Comments**
Measurement Standard	sysMetric={measures}	;configSystem?sysMetric=English
Color for non-I/O buttons	sysBClr={color}	;
Max Internal Temperature	sysTemp=[int]	;configSystem?sysTemp=140
Blink Triggered Relays when displayed	sysBlnk={bool}	;
Show Context Sensitive Help	sysHelp={bool}	;
Display OEM Image	sysImg={bool}	;
Device Name	sysName=[string]	;configSystem?sysName=uSwitch
Company Name	sysOEM=[string]	;
String/URL on Each Page's footer	sysFoot=[string]	;
Physical Location of device	sysLoc=[string]	;
Emergency Contact Number	sysPhone=[string]	;
Pointer to OEM's Manual	sysMan={URL}	;sysMan=https://OEMurl.com/s/files/Manual-2.pdf?v=1613674918

XML Network Settings Commands

(per Network Settings Menu)

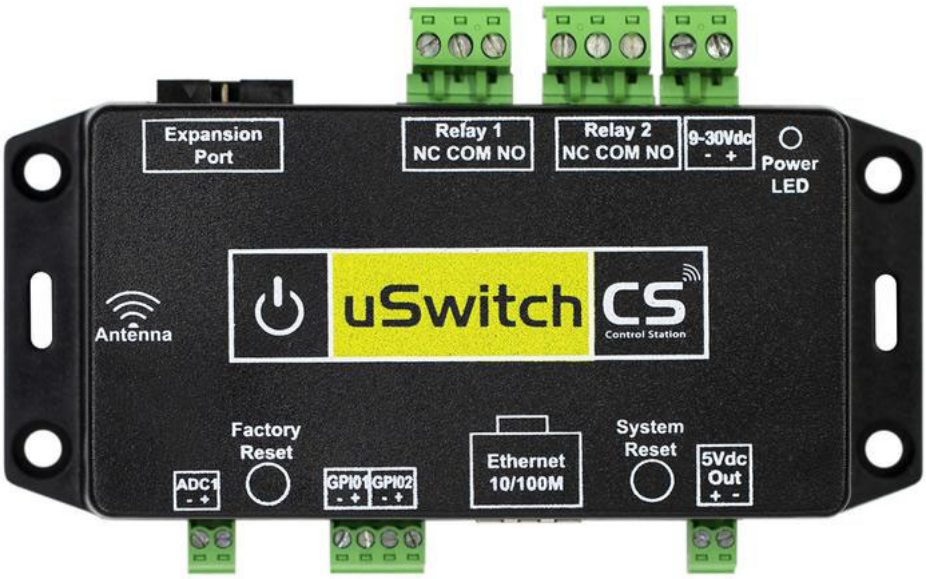
GET **configNet**?PARM=[value] HTTP/1.1

Description	parm/value	Example usage
Set Network Type	NtType={NetworkType}	;configNet?NtType=APStation
Set SSID of uSwitch AP	NtAPSSID=[string]	;
Set Security Type for AP	NtAPAuth={WifiSecurity}	;configNet?NtAPAuth=WPA-WPA2-PSK
Set Passphrase for AP	NtAPKey=[string]	;
Set IP Address of AP	NtIPAddr={URL}	;
Set DNS for AP	NtAPDNS={URL}	;
Enable SP SSID broadcast	NtAPBCast={bool}	;
Enable AP Repeater/NAT mode	NtAPNat={bool}	;
Enable DHCP Server on AP	NtAPDHCPS={bool}	;
Set WiFi Access Point for Station	NtStaSSID=[string]	;
Set WiFi Security Mode for Remote AP	NtStaAuth={WifiSecurity}	;configNet?NtStaAuth=WPA2-PSK
Set WiFi Passphrase for Remote AP	NtStaKey=[string]	;
Set IP address for Station	NtStaAddr={URL} ;	
Station is using DHCP	NtDHCP={bool}	;
Set Channel Number for Station	NtStaChn=[int 1..11 or Auto]	;
Set Region Code for WiFi	NtRegion={region}	;
Set WiFi Power Level	NtTxPwr=[int 5.68]	;
Set IP Address of Network Gateway	NtGate={URL}	;
Set Subnet Mask for uSwitch	NtSubN={URL}	;
Set uSwitch Host Name	NtHost=[string]	;
Set Primary DNS URL	NtDNS0={URL}	;
Set Secondary DNS URL	NtDNS1={URL}	;
Set Tertiary DNS URL	NtDNS2={URL}	;
Set MTU value	NtMTUs=[int 128..1492]	;

uSwitch CS

Control. Monitor. Reboot. Notify

The ultimate uSwitch with Infinite Possibilities



- Built-In Web Server/WiFi Access Point/Repeater/Range Extender
- Text and Email Alerts
- Scheduling
- Logging
- Full Encryption including SSL/TLS for HTTPS & SMTP
- Peer-to-Peer (P2P/M2M) Communications
- Voltage/Current Monitoring
- Relay, I/O, Sensor Expansion Port
- Up to 110/220V 5 Amp relay contacts
- No programming required
- Easy to install
- Compatible with iPhone, Android, PC and Mac as well as Most Browsers
- Voltage 9-30VDC
- OTA uploads are secure using HTTPS

Networking

- Ethernet: 10/100 Base-T, Full/Half Duplex*
- WiFi 802.11 b/g/n
- WiFi Adapter Supports: Station, Access Point, Station and Access Point/Repeater with up to 10 WiFi

Client Connections

• WiFi: Security Modes: WEP, WPA-PSK, WPA2-PSK, WPA-WP2-PSK/WPA2-Enterprise modes (WEP not supported in AP Mode)
• Dynamic switching all WiFi Modes and to Ethernet Mode*
• Static IP or DHCP* all communication modes
• All WiFi Settings User Configurable for Max Customization Including: Tx Power (to optimize radio Performance), Region, Radio Channel.
• Auto Scan for SSID, all Station Modes
• All WiFi modes support configurable SSIDs and PassPhrases
• AP Modes allow multiple WiFi Station Connections (repeater)*
• Reset Button supports WiFi AP mode or full factory Reset.
WiFi/Ethernet Protocols and Encryption Options
• HTTP*, HTTPS, SSL/TLS, XML, TCP/IP, SMTP, NTP, MDNS

Timing

• Real-Time Clock supports multiple NTP servers or manual time with backup.
• Options: Adjustable NTP Sync Time, Time Zones, Daylight Savings & configurable time formats
• RTC has Super-Cap for time backup.

Notifications

• Secure Email with SSL/TLS
• Secure SMS Messaging
• Fully Configurable

Logging

• Customizable, Resettable and Downloadable and User Log

Passwords

• Multi-Level Passwords protection (staff and admin), with auto reset and user configurable password timeouts

Watchdogs

• Two High Power Relays can function as dedicated remote Device Ping Watchdogs
• Each Watchdog supports up to 3 External Pingable addresses and can also be triggered from external pulse
• Multiple, user configurable and redundant Internal Watchdogs: Hardware, OS, Firmware and Network.

Interfaces

• XML, HTTP (optional), HTTPS tested on Chrome, Safari, Brave, Internet Explorer, Edge, Samsung, Opera, Firefox, Puffin, Phoenix, Dolphin,
User Interface:
• Intuitive and Simple User Interface
• Fully OEM customizable interface works on computers, Phones , I-Pads or Standalone.
• Menus are user Customizable and Dynamically created
• Icons are Downloadable and may be Deactivated/Activated
• Documentation, Full support Online as well as Comprehensive In Application Help Screens and Menus
• Documentation Fully User Customizable with Dynamic Links to OEM/Integrator Manuals, Support Details and Website directly from Interface.

Control Screen

• User Control Screen with customizable Input/Output Names, auto-ranging colors, state text, setpoints and limits.

Hardware Interfaces

• 5 Vdc Out (up to XX ma) 2-Pin Terminal
• 3.3 Vdc Out (up to XX ma) Expansion Bus
• 2-High Voltage dry contract relays each with Independent removable terminal connectors supporting both N/O and N/C options.
• Relays can be manual or momentary operations triggered by user input, or programmable events.
• Event driven relays are triggered by schedule, other GPIOs or sensor inputs either from the main unit or any remote unit that is connected via the LAN/WAN, WiFi, or Ethernet.
• When driven externally relay outputs can be latched forcing manual user intervention to reset from an alarm, or hazardous condition for added security
• Power-Up state is user configurable for the digital outputs.
• 2 user configurable digital inputs with debounce and cutoff filters and
• Digital I/Os have removable terminal connection limiting crosstalk and simplifying connections.
• Calibratable digital Inputs trigger On/Off and High/Low/Range events (hysteresis supported), and support full numerical scaling for calibration.
• Digital Inputs have user configurable pull-ups and pull-downs.
• Internal Temperature monitor with user configurable alerts
• Internal (minimal profile) WiFi Radio Antenna with options for External Antennas

Expansion Port

• 10 Pin Expansion port supporting I2C, SPI and RS232 devices, and 3.3Vdc output.
• Expansion options include, temperature, pressure, humidity, GPS, CO2, light, proximity, motion, biofeedback(heartrate,ECG,EKG,Pulseoxmeter), relays, analog/digital inputs, LCD displays and touchscreens all with built-in auto-discovery

Scheduling

• Event driven schedule with two independent schedules and up to 20 events can be configured to trigger relays.

Reset Options

• The Interfaces supports a remote board reset option and factory reset options to reset individual devices remotely.
• An on-board pushbutton can reset the device to all initial factory settings, or only network settings to an initial configuration in instance that network change has caused a disconnect
Point to Point
• uSwitch supports M2M communication (via TCP Sockets) so that an entire network of units can operate as a system or completely independently or a mixture of both.

OTA Firmware

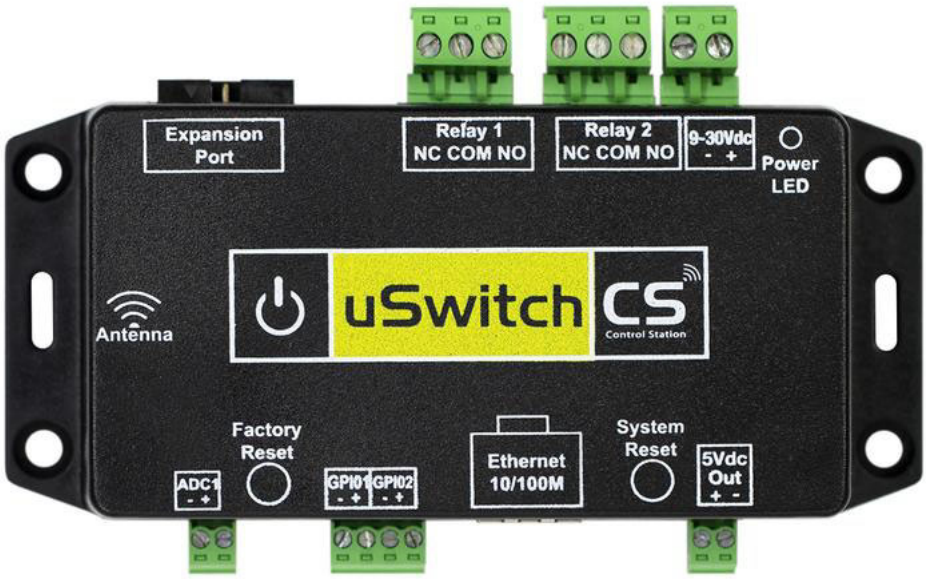
• Over The Air software supports remote. Update SSL keys, Certificates, OEM Icons, and device settings from anywhere in the world.
Operating Temp.
• -XXF to 1xx F, (C to C)
• Dimensions
• Weight

Terminal Connections

• Relays 2- 3 Pin removable NO/NC/COM terminal connectors (for xx-XX AWG wire)
• GPIOs 1-4 Pin Removable 2 Gnd, 2V+ (for xx-XX AWG wire)
• ADC 1-2 Pin Removable (for xx-XX AWG wire) max current
• 5VDC out 1-2 Pin Removable +5, Gnd (for xx-xx AWG wire) max current
• Power in 1-2 Pin Removable (for xx-XX AWG wire) max current
• Network 1-8 Pin RJ-45

uSwitch

by uHave Control



241 Perkins Street, A501 ● Jamaica Plain, MA 02130

Tel: 617-933-9983 ● www.uHaveControl.com