# REST API

*Applies to: SocketPlug, WallSwitchOne, WallSwitchDual*

Formera devices contains a web server responding to web request.

* Requests/response content: application/json

## Overview

Endpoint overview

GET http://{{url}}/api/device

GET http://{{url}}/api/settings  
PUT http://{{url}}/api/settings

GET http://{{url}}/api/switches  
GET http://{{url}}/api/switches/{switchId}  
PUT http://{{url}}/api/switches/{switchId}

GET http://{{url}}/api/wifi/networks  
POST http://{{url}}/api/requestReset  
POST http://{{url}}/api/requestFactoryReset

## Status

Information about the current device state

### Device status endpoint

GET http://{{url}}/api/device

Example JSON response:

{

"chip\_id": 8911740,

"company": "Formera",

"type": "WallSwitchDual",

"fw\_version": "0.12.0b",

"uptime": 868,

"wifi\_connects": 8,

"free\_memory": 26400,

"name": "",

"mac": "5C:CF:7F:87:FB:7C",

"ssid": "WiFi Network",

"signal\_strength": "-68dBm",

"hostname": "Formera8911740",

"ip\_address": "192.168.0.115",

"gateway": "192.168.0.1",

"subnet": "255.255.255.0",

"primary\_dns": "83.255.255.2",

"secondary\_dns": "83.255.255.1",

"mqtt\_status": "Disabled",

"apMode": false

}

|  |  |
| --- | --- |
| Property | Description |
| chip\_id | ESP chip id |
| company | Formera |
| type | SocketPlug | WallSwitchOne | WallSwitchDual |
| fw\_version | Current firmware version |
| uptime | Uptime in seconds |
| wifi\_connects | WiFi connects since startup |
| free\_memory | Device free memory in kB |
| name | Current device name |
| mac | Device MAC address |
| ssid | Current wifi connection SSID |
| signal\_strength | WiFi signal strength |
| host\_name | Current host name |
| ip\_address | Current IP address |
| gateway | Current gateway |
| subnet | Current subnet |
| primary\_dns | Current primary DNS |
| secondary\_dns | Current secondary DNS |
| mqtt\_status | MQTT connection status: Disabled | Disconnected | Connected |
| apMode | False in normal operation, True in Access Point mode |

## Settings

Get and update device settings

### Get settings

Read current settings

GET http://{{url}}/api/settings

Example JSON response:

{

"silent\_mode": false,

"use\_dhcp": true,

"ssid": "Team Succe HQ",

"password": "\*",

"name": "",

"hostname": "Formera8911740",

"static\_ip": "0.0.0.0",

"gateway": "0.0.0.0",

"subnet": "0.0.0.0",

"mqtt\_enabled": false,

"mqtt\_secure": false,

"mqtt\_host": "",

"mqtt\_port": 33333,

"mqtt\_user": "",

"mqtt\_token": "\*"

}

### Update settings

Write new settings to device. Only included settings are updated (underposting).

PUT http://{{url}}/api/settings

Result:   
200 OK

Example request body to update WiFi settings

{

"ssid": "MyWiFi",

"password": "itsasecret"

}

Available settings

|  |  |  |
| --- | --- | --- |
| Property | Type | Description |
| silent\_mode | bool | Use signal LEDs (SocketPlug) |
| use\_dhcp | bool | Use DHCP |
| ssid | string | SSID of WiFi network to connect to |
| password | string | WiFi password (always returns ‘\*’) |
| name | string | Device name |
| hostname | string | Device host name |
| static\_ip | string | IP address to use if not DHCP |
| gateway | string | Gateway |
| subnet | string | Subnet |
| mqtt\_enabled | bool | Use MQTT client |
| mqtt\_secure | bool | Use SSL for mqtt clienet |
| mqtt\_host | string | Url to MQTT server |
| mqtt\_port | int | MQTT server port |
| mqtt\_user | string | MQTT user name |
| mqtt\_token | string | MQTT password (Always returns ‘\*’) |
| base\_devicetime | string | ? |

## Switch endpoints

Read switch status and control switch states.

### Get all switches

Returns all switches on the device in a JSON array

GET http://{{url}}/api/switches

Example JSON response (WallSwitchDual):

[

{

"id": 0,

"switch\_type": "relay",

"state": 0

},

{

"id": 1,

"switch\_type": "relay",

"state": 0

}

]

Switch object

|  |  |  |
| --- | --- | --- |
| Property | Type | Description |
| id | int | The zero based index of the switch. [0..n] |
| switch\_type | string | The type of the switch: one of "relay", "dimmer", … |
| state | int | 0 or 1 for "relay", 0…255 for "dimmer" |

### Get switch

Return a specific switch using zero-based index **switchId**

GET http://{{url}}/api/switches/{switchId}

Example JSON response:

{

"id": 0,

"switch\_type": "relay",

"state": 0

}

### Set switch state

Update a switch using zero-based index **switchId**

PUT http://{{url}}/api/switches/{switchId}

Response:

200 OK

Example request body to update switch 0 to state on

{

"state": 0

}

## System endpoints

### Get WiFi networks

Scan for networks

GET http://{{url}}/api/wifi/networks

Example JSON response:

{

{

"ssid": "ComHem9A873B",

"encryption": 4,

"rssi": -91

},

{

"ssid": "TN\_private\_VPCFU3",

"encryption": 2,

"rssi": -94

}

}

### Reset device

Do a soft reboot of the device

POST http://{{url}}/api/requestReset

Response:  
200 OK

### Factory reset device

Clear settings. Device is not rebooted.

POST http://{{url}}/api/requestFactoryReset

Response:  
200 OK