

---

# SPC Gateway User Manual

Revision 1.0

---



## History Record

Revision	Date	Author	Comment
1.0	25-Sep-2015	Göran Lundquist, Lundix IT	First edition

©2015 Lundix IT

Lundix IT  
 Renvägen 22  
 S-433 70 Sävedalen  
 Sweden  
 info@lundix.se

## Contents

1.1	SPC Gateway .....	5
1.2	Features .....	5
1.3	Hardware Specification.....	5
1.4	Document References.....	6
1.5	Terminology and Abbreviations .....	6
<b>2</b>	<b>INSTALLATION.....</b>	<b>7</b>
2.1	Installation Prerequisites.....	7
2.2	Installation Steps .....	7
<b>3</b>	<b>CONFIGURATION.....</b>	<b>8</b>
3.1	IP Address .....	8
3.2	Time Setting .....	8
3.3	Change Administration Password.....	8
3.4	SPC Communication (EDP).....	8
3.4.1	SPC Panel EDP Settings.....	8
3.4.2	SPC Gateway Settings.....	10
3.5	SPC Anywhere Server .....	10
3.5.1	Configuration of the SPC Anywhere Server.....	11
3.5.2	Configuration of the SPCanywhere App.....	12
3.6	Notifications .....	12
3.6.1	Notification Map.....	12
3.6.2	Configuration of email .....	12
3.6.3	Configuration of NMA .....	13
3.6.4	Configuration of Prowl .....	14
3.7	REST / Websockets API .....	15
3.8	Fibaro HC2 .....	16
3.9	Device I/O .....	18
3.9.1	I/O Description.....	19
3.9.2	Pin Layout.....	19
3.9.3	I/O Configuration.....	20
3.10	Test Tools.....	20
<b>4</b>	<b>ADVANCED USERS.....</b>	<b>21</b>
4.1	Backup of Configuration Settings .....	21
4.2	Upgrading software .....	21
4.3	SSH Access .....	21
4.4	MQTT Access.....	21
4.4.1	Enable access from other computers.....	21
4.4.2	MQTT topics.....	22
4.4.2.1	SPC - SIA Events .....	22
4.4.2.2	SPC – Alarm Zone Input Value .....	22
4.4.2.3	SPC – Alarm Zone Status Value .....	23
4.4.2.4	SPC – Alarm Area Status Value .....	24
4.4.2.5	I/O – Analog Input Value .....	24

---

4.4.2.6	I/O – Digital Input Value.....	25
4.4.2.7	I/O – Digital Output Command .....	25
4.4.2.8	I/O – Digital Output Value .....	26
<b>5</b>	<b>LICENSE AGREEMENTS .....</b>	<b>27</b>
5.1	End-User License Agreement for SPC Gateway (EULA).....	27
5.2	Open Source Software .....	28
5.3	Open Source Hardware .....	28

Introduction

## 1.1 SPC Gateway



SPC Gateway is a network device that adds extra features to a Siemens/Vanderbilt SPC Intrusion System. It is based on the Dragino platform ([www.dragino.com](http://www.dragino.com)) that provides both hardware and a Linux system that can rather easy be adapted for commercial products. The communication with the SPC Panel is based on Siemens/Vanderbilt proprietary EDP protocol. All software, except the EDP communication, is available as open source. If you are interested in the source code please contact Lundix IT.

## 1.2 Features

- SPC Panel Communication (EDP Protocol)
- Alarm and status notifications via mail, Notify My Android (NMA) and Prowl
- Status mirroring to Fibaro Home Center 2
- Status publishing on MQTT
- Generic REST and Websocket API
- Support for I/O hardwired to the device
- Support for Siemens App SPCanywhere, even after upgrading the SPC panel firmware to 3.6+
- Web based Admin GUI

## 1.3 Hardware Specification

Processor	400MHz, 24K MIPS
Flash	16MB
RAM	64MB
Power input	9 – 12V DC
Network	2 x 10M/100M RJ45 Connectors
WiFi	802.11 b/g/n
USB	1 x USB 2.0 host connector
I/O	4 x Digital Inputs (0 - 5V high impedance) 4 x Analog Inputs (0 – 5V high impedance, 10 bits resolution)

	2 x Digital Outputs (0 -5V low impedance, max 40mA)
Type Approval	FCC Part15 Subpart B, Subpart C, CE NB, C-Tick

## 1.4 Document References

Id	Description	Revision
[LUNDIX_SPC_WEB_GW_SPEC]	Lundix SPC Web Gateway Specification	1.1
[SPC_INST_CONF]	Siemens SPC42xx/43xx/52xx/53xx/63xx, Installation & Configuration Manual	3.2
[DRAGINO_INFO]	Dragino related info: <a href="http://wiki.dragino.com">http://wiki.dragino.com</a> and <a href="http://www.dragino.com">http://www.dragino.com</a>	
[OPENWRT_INFO]	OpenWRT related info: <a href="http://wiki.openwrt.org">http://wiki.openwrt.org</a>	

## 1.5 Terminology and Abbreviations

Term	Description
EDP	Siemens Enhanced Datagram Protocol
JSON	JavaScript Object Notation
SIA	Security Industry Association
SPC panel	Siemens/Vanderbilt SPC intrusion panel
URL	Uniform Resource Locator
WebSocket	Two-way TCP protocol RFC 6455

## 2 Installation

### 2.1 Installation Prerequisites

- Siemens SPC panel with firmware > 3.2
- Network router with DHCP enabled
- Network connection between the SPC Gateway and the SPC panel
- Internet access (to be able to use the notification functions and ntp time sync)
- Fibaro HC2 with firmware >= 4.X. (If you would like to use the Fibaro HC2 binding)

### 2.2 Installation Steps

1. Read carefully **End-User License Agreement for SPC Gateway (EULA)** in chapter 5 in this document. If you do not agree to the terms of the EULA, do not install or use the SPC Gateway.
2. Connect SPC Gateway LAN port, with a regular network cable, to your network switch or router.
3. Power up the device by connecting the included power adapter to a wall socket and then to the SPC Gateway device.
4. Wait (~2 minutes) until the SYS LED (globe symbol) has stopped blinking and the LAN LED has turned on.
5. Login to your router and look for the IP address assigned to the SPC Gateway.
6. Open a web browser and enter the IP address of the SPC Gateway.
7. In the login window, enter Username (root) and Password (default is dragino) and you will see the main menu and a status overview of the device.
8. You can now start with the configuration.

## 3 Configuration

As the SPC Gateway device is based on a standard Linux (OpenWrt) platform, normally used for routers, you will find a lot of configuration options in the Web GUI. In this manual only configuration specific for the SPC Gateway functions is described. Most of these functions are available in the main menu – *SPC Gateway*. For configuration of generic OpenWrt functions please see [OPENWRT\_INFO] and for Dragino specific info [DRAGINO\_INFO].

### 3.1 IP Address

Default will SPC Gateway use DHCP to get an IP Address. To set a static IP Address, go to *Network -> Interfaces* and:

1. Select *LAN – Edit*.
2. In *Interfaces – LAN*, select Static address in the *Protocol* option menu
3. Push button *Switch Protocol*
4. In *Common Configuration – General Setup* fill in; IPv4 address, netmask, gateway and custom DNS servers.
5. In *DHCP Server – General Setup*; check the checkbox Ignore interface.
6. Push button *Save & Apply*.
7. Redirect your browser to the new IP Address.

### 3.2 Time Setting

To set correct Timezone, go to *System -> System* and select *Timezone* in section *System Properties – General Settings*.

The device is using NTP to synchronize time. This setting is in *System -> System* section *Time Synchronization*.

**NOTE!** The device has no RTC clock. During boot the device can have incorrect time. Some events in the system log can therefore have incorrect timestamps.

### 3.3 Change Administration Password

The default administration password for accessing the device is **dragino**. Of security reasons it is highly recommended to change the password as soon as possible. Go to *System -> Administration* section *Router Password* to change the password. The same password is used in both Web GUI and for ssh access to the device.

### 3.4 SPC Communication (EDP)

To setup the communication link between SPC Panel and SPC Gateway you have to configure the link in both SPC Panel and SPC Gateway ends.

#### 3.4.1 SPC Panel EDP Settings

Use SPC Pro or SPC Web interface to define an EDP communication link according to following figure. (Example from SPC FW 3.6.6 WebGUI)



Communications	FlexC®	Reporting	PC Tools
Analog ARC	EDP	C/EI-ABI	

**Edit Receiver**

<b>Description</b>	<input type="text" value="SPC Gateway"/>	Description of receiver.
<b>Receiver Id</b>	<input type="text" value="1100"/>	Unique identification number of EDP receiver used by this panel. (1 - 999997)
<b>Protocol version</b>	<input type="text" value="Version 2"/>	Select version of EDP protocol to use with this receiver

**Security**

<b>Commands Enable</b>	<input checked="" type="checkbox"/>	Check if incoming commands are allowed from this receiver.
<b>Change user PINs</b>	<input type="checkbox"/>	Check if changing user PINs is allowed from this EDP receiver.
<b>Virtual Keypad</b>	<input type="checkbox"/>	Check to allow virtual keypad access from this EDP receiver.
<b>Live streaming</b>	<input type="text" value="Always available"/>	Select Live Streaming privacy options
<b>Encryption Enabled</b>	<input checked="" type="checkbox"/>	Check if data to and from this receiver is encrypted.
<b>Encryption Key</b>	<input type="text" value="*****"/>	32 Hexadecimal Digits

**Network**

<b>Network Enable</b>	<input checked="" type="checkbox"/>	Check if events can be reported through Network
<b>Network Protocol</b>	<input type="text" value="TCP/IP"/>	Select transport layer protocol over Ethernet.
<b>Receiver IP Address</b>	<input type="text" value="X.X.X.X"/>	IP address of receiver.
<b>Receiver IP Port</b>	<input type="text" value="16000"/>	IP port of receiver.
<b>Always Connected</b>	<input checked="" type="checkbox"/>	Check if panel should keep a permanent connection to the receiver. If n event.
<b>Panel Master</b>	<input checked="" type="checkbox"/>	Check this to make the panel master of polling messages.
<b>Polling Interval</b>	<input type="text" value="10"/>	Seconds between polls
<b>Generate a Network Fault</b>	<input type="checkbox"/>	A polling failure will generate a network fault

**Dial-up**

<b>Dial-up Enable</b>	<input type="checkbox"/>	Check if events can be reported through dial-up
-----------------------	--------------------------	---

**Events**

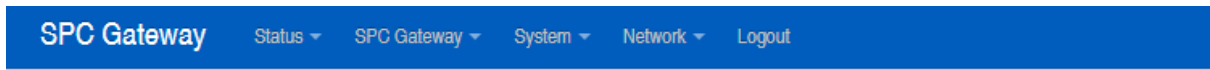
<b>Primary Receiver</b>	<input checked="" type="checkbox"/>	Check if primary, clear for backup
<b>Re-queue Events</b>	<input type="checkbox"/>	Check if events that fail to report are to be requeued for transmission.
<b>Verification</b>	<input checked="" type="checkbox"/>	Check if Audio/Video verification should be sent to this receiver.
<b>Event Filter</b>	<input type="text" value="Filter"/>	Configure which events are reported to this receiver

**Note:**

- *Receiver IP Address* is the IP Address of SPC Gateway
- *Receiver ID, Encryption Key and Receiver IP Port* must match corresponding values in SPC Gateway Communication Settings.
- In the *filter* settings window you should enable all event types and instead filter the events in SPC gateway Notifications Settings.

### 3.4.2 SPC Gateway Settings

Go to *SPC Gateway* -> *SPC Communication* and fill in the form according to:




## SPC Communication

Configuration of Communication with Siemens/Vanderbilt SPC Panel. It is very important that this settings corresponds with the settings in the SPC Panel.

### EDP Receiver

Enable EDP Encryption

Encryption Key    
32 hex numbers (0-9, a-f, A-F)

EDP Panel ID

EDP Receiver ID

EDP Receiver IP Port

### Time Zone Settings

Time difference   
Time difference, in hours, between SPC panel and Gateway

Daylight Saving Time  Set DST if enabled on SPC panel

**Note:**

- *EDP Panel ID* is the ID of the SPC Panel found in SPC Panel EDP Settings
- *Encryption Key*, *EDP Receiver ID* and *IP Port* must match corresponding values in SPC Panel EDP link settings.
- *Time difference* is the time difference in hours between time in SPC Panel and Gateway
- Check *Daylight Saving Time* if Automatic Daylight Saving time is enabled in SPC Panel Clock Settings.

## 3.5 SPC Anywhere Server

From SPC firmware version 3.6+, the app **SPCanywhere** is no longer supported by Siemens/Vanderbilt. SPCanywhere had support for both local and remote network connection to your panel. The new app, **SPC Connect**, has no local network support and is entirely dependent of the cloud service [www.spconnect.com](http://www.spconnect.com) for remote access to your panel. To be able to continue using

SPCanywhere even after upgrading to 3.6+, you can redirect your connection to SPC Gateway embedded proxy server. This way you can use SPCanywhere for local access via your private network and SPC Connect for remote access via internet. The functions Doors, Access log and WPA log are not supported.

### 3.5.1 Configuration of the SPC Anywhere Server

Go to *SPC Gateway* -> *SPC Anywhere Server* and fill in the form according to:

**Note:**

- The User and Password must match corresponding values in the App
- The values of Encrypted communication and IP Address are only for information and are not editable.
- IP Address is the IP address of the SPC Gateway. The port is 8090.

### 3.5.2 Configuration of the SPCanywhere App

**Note:**

- IP Address format; XXX.XXX.XXX.XXX:8090. Example: 192.168.0.101:8090
- No support for Doors, Access log and WPA log

## 3.6 Notifications

SPC Gateway has support for three types of notifications; email, Prowl and NMA. Prowl is for iPhone/iPad users and NMA is for Android users. For email notifications, Gmail is recommended. Up to three recipients can be defined for each notification type.

### 3.6.1 Notification Map

In *SPC Gateway -> Notification Map* configure the notification recipients for each type of SPC SIA event. The SIA Events are categorized in different Event groups. In the tab *SIA Events* you can see the SIA events for each group.

### 3.6.2 Configuration of email

Go to *SPC Gateway -> Notifications -> Email*. Configure email accounts according:

SPC Gateway   Status ▾   SPC Gateway ▾   System ▾   Network ▾   Logout

- Notification Map   SIA Events   **Email**   NMA   Prowl

## Email Properties

Configuration of Email SMTP Server and Email Recipients of notifications.

Enable Email Notifications

SMTP Server

SMTP Port

SMTP User Name

SMTP Password

Enable SSL

Customize Script

This script can be used to customize notification subject and message

From

From email address

To Email1

First email notification recipient

To Email2

Second email notification recipient

To Email3

Third email notification recipient

For advanced users it is possible to customize the email subject and body in the Lua script `/usr/lib/luajit/luajit/bin/luajit -l/usr/lib/luaspc-gateway/email-custom.lua`. This file is only accessible via ssh login.

### 3.6.3 Configuration of NMA

Go to *SPC Gateway* -> *Notifications* -> *NMA*. To be able to use NMA you need to be registered on <http://www.notifymyandroid.com/>. Configure NMA settings according to:

**SPC Gateway**    Status ▾    SPC Gateway ▾    System ▾    Network ▾    Logout

Notification Map    SIA Events    Email    **NMA**    Prowl

## NMA Properties

Configuration of Notify My Android notification services. To be able to use this service you need to register at least one NMA account

Enable NMA Notifications

Customize Script `/usr/lib/lu/spc-gateway/nma-custom.lua`  
 ⓘ This script can be used to customize notification subject and message

NMA1 - API Key  ⓘ  
 ⓘ API Key to first NMA recipient

NMA2 - API Key  ⓘ  
 ⓘ API Key to second NMA recipient

NMA3 - API Key  ⓘ  
 ⓘ API Key to third NMA recipient

For advanced users it is possible to customize the notification subject and body in the Lua script `/usr/lib/lu/spc-gateway/nma-custom.lua`. This file is only accessible via ssh login.

### 3.6.4 Configuration of Prowl

Go to *SPC Gateway -> Notifications -> Prowl*. To be able to use Prowl you need to be registered on <http://www.prowlapp.com/>. Configure Prowl settings according to:

## Prowl Properties

Configuration of Prowl notification service. To be able to use this service you need to register at least one Prowl account

Enable Prowl Notifications

Customize Script `/usr/lib/lu/spc-gateway/prowl-custom.lua`  
 ⓘ This script can be used to customize notification subject and message

Prowl1 - API Key    
 ⓘ API Key to first Prowl recipient

Prowl2 - API Key    
 ⓘ API Key to second Prowl recipient

Prowl3 - API Key    
 ⓘ API Key to third Prowl recipient

For advanced users it is possible to customize the notification subject and body in the Lua script `/usr/lib/lu/spc-gateway/prowl-custom.lua`. This file is only accessible via ssh login.

### 3.7 REST / Websockets API

The SPC Gateway provides a generic web interface to the SPC panel. The main purpose of this interface is to simplify SPC integration with third party applications and products such as Home and Building Automation Systems, Smartphone App's and Web applications. The Web API is using HTTP and REST principles (RESTful) for requests to SPC panel and WebSockets to reporting events from the SPC panel.

The API is described in [LUNDIX\_SPC\_WEB\_GW\_SPEC].

To configure the interface, go to *SPC Gateway -> REST / Websockets API* and fill in the form. For security reasons it is highly recommended that you enable SSL encryption (default) and set own passwords for GET, PUT and Websockets access. GET methods are used for getting information from the SPC panel, PUT methods for commands to the SPC panel and Websockets to get SIA Events from the SPC Panel.

**NOTE!** The security settings above are only used for external access of the API. Internal processes in the SPC Gateway that are using this interface are not affected of the settings.

If you would like to test the API you can use the command utility curl on another computer in your network:

**Example GET method:** Get the status of zone 1:

```
curl -X GET https://IP_ADDRESS_SPC_GATEWAY:8088/spc/zone/1 -u \
get_user:get_pwd -k --digest

{"status":"success","data":{"zone":[{"id":"1","type":"0","zone_name":
"Entrance","area":"1","area_name":"Area 1","input":"0","status":"0"}]}}
```

**Example PUT method: Isolate zone 1:**

```
curl -X PUT https://IP_ADDRESS_SPC_GATEWAY:8088/spc/zone/1/isolate \
-u put_user:put_pwd -k --digest

{"status":"success","data":"null"}
```

### 3.8 Fibaro HC2

SPC Gateway has support for mirroring status values to a Fibaro Home Center 2 device. The status will be available in the Fibaro device as Global Variables, and can be used in Virtual Devices to e.g triggering scenes.

**NOTE!** This feature is only supported from Fibaro HC2 version 4.X.

Following values are mirrored:

Description	HC2 Global Variable Name	X	Possible values
SPC - Zone Input Value	G_SPC_ZONE_INPUT_X	X is SPC Zone ID, X = 1 to max number of zones.	“CLOSED” “OPEN” “SHORT” “DISCONNECTED” “PIR_MASKED” “DC_SUBSTITUTION” “SENSOR_MISSING” “OFFLINE” “UNKNOWN”
SPC - Zone Status Value	G_SPC_ZONE_STATUS_X	X is SPC Zone ID, X = 1 to max number of zones.	“OK” “INHIBIT” “ISOLATE” “SOAK” “TAMPER” “ALARM” “TROUBLE” “UNKNOWN”
SPC - Area Status Value	G_SPC_AREA_STATUS_X	X is SPC Area ID, X = 1 to max number of areas.	“UNSET” “PARTSET_A” “PARTSET_B” “SET” “UNKNOWN”
Device I/O - Analog Input Value	G_IO_AI_STATUS_X	X is Analog Input Number. X = 1 - 4	Scaled value; value = ad_value/1023 *



			(max – min) + min
Device I/O - Digital Input Value	G_IO_DI_STATUS_X	X is Digital Input Number. X = 1 - 4	"0" "1"
Device I/O - Digital Output Value	G_IO_DO_STATUS_X	X is Digital Output Number. X = 1 - 2	"0" "1"

**NOTE!** In this version of SPC Gateway the only way to set the digital outputs is via MQTT.

The communication is configured in *SPC Gateway -> Fibaro HC2*.

SPC Gateway
Status ▾
SPC Gateway ▾
System ▾
Network ▾
Logout

### Fibaro Home Center 2 Binding

Configuration of binding to Fibaro HC2. SPC status are mirrored to global variables in the HC2 device.

Enable Fibaro HC2 Binding

Customize Script `/usr/lib/lua/spc-gateway/fibaro-custom.lua`  
 ⓘ This script can be used to customize events send to Fibaro HC2

HC2 IP Address

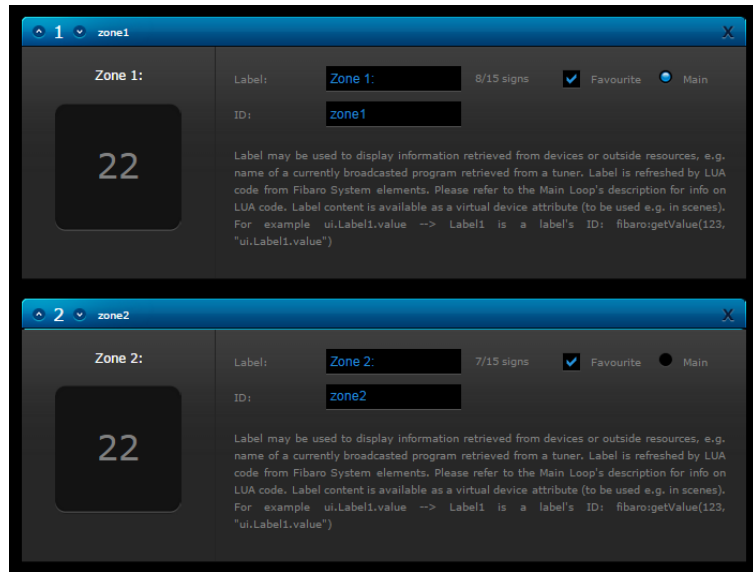
HC2 User

HC2 Password

For advanced users it is possible to customize the event, before it is sent to Fibaro HC2, in the Lua-script `/usr/lib/lua/spc-gateway/fibaro-custom.lua`. The file is only accessible via ssh login

Simple example of usage in Fibaro HC2 (Firmware 4.X):

SPC Virtuell enhet	
Zone 1:	CLOSED
Zone 2:	CLOSED
Zone 3:	CLOSED
Zone 4:	CLOSED
Zone 5:	CLOSED
Zone 9:	OPEN
Zone 10:	OPEN
Zone 11:	CLOSED
Zone 12:	CLOSED
Zone 13:	CLOSED
Zone 14:	CLOSED



### Fibaro HC2 Lua script:

```

thisId = fibaro:getSelfId();

fibaro:call(thisId, "setProperty", "ui.zone1.value",
fibaro:getGlobalValue("G_SPC_ZONE_INPUT_1"));
fibaro:call(thisId, "setProperty", "ui.zone2.value",
fibaro:getGlobalValue("G_SPC_ZONE_INPUT_2"));
fibaro:call(thisId, "setProperty", "ui.zone3.value",
fibaro:getGlobalValue("G_SPC_ZONE_INPUT_3"));
fibaro:call(thisId, "setProperty", "ui.zone4.value",
fibaro:getGlobalValue("G_SPC_ZONE_INPUT_4"));
fibaro:call(thisId, "setProperty", "ui.zone5.value",
fibaro:getGlobalValue("G_SPC_ZONE_INPUT_5"));
fibaro:call(thisId, "setProperty", "ui.zone9.value",
fibaro:getGlobalValue("G_SPC_ZONE_INPUT_9"));
fibaro:call(thisId, "setProperty", "ui.zone10.value",
fibaro:getGlobalValue("G_SPC_ZONE_INPUT_10"));
fibaro:call(thisId, "setProperty", "ui.zone11.value",
fibaro:getGlobalValue("G_SPC_ZONE_INPUT_11"));
fibaro:call(thisId, "setProperty", "ui.zone12.value",
fibaro:getGlobalValue("G_SPC_ZONE_INPUT_12"));
fibaro:call(thisId, "setProperty", "ui.zone13.value",
fibaro:getGlobalValue("G_SPC_ZONE_INPUT_13"));
fibaro:call(thisId, "setProperty", "ui.zone14.value",
fibaro:getGlobalValue("G_SPC_ZONE_INPUT_14"));

```

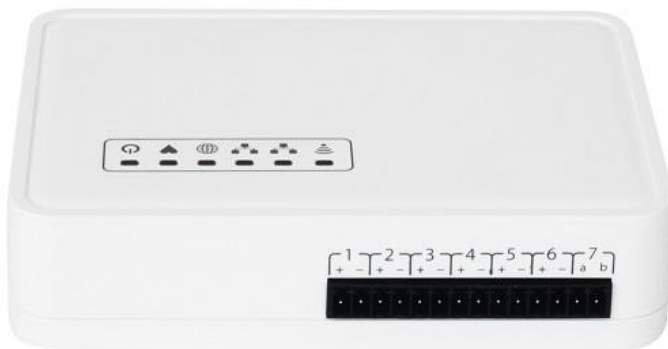
## 3.9 Device I/O

The SPC Gateway device has 4 digital inputs, 2 digital outputs and 4 analog inputs. At the moment the values of the I/O are only passed to the Fibaro HC2 device and MQTT. The only way to set the digital outputs is with MQTT. Better support (Notification etc) for the I/O is planned in coming releases.

### 3.9.1 I/O Description

I/O	Description	Number	Update Interval
Digital Input	5V high-impedance digital inputs. To steer an input to a known state, if no input is present, use a pullup resistor (to +5V) or a pulldown resistor (to GND) on the input. A 10K resistor is a good value for a pullup or pulldown resistor.	4	5 seconds
Digital Output	5V low-impedance digital outputs. Can source (provide positive current) or sink (provide negative current) up to 40 mA to other devices/circuits. Short circuits on pins, or attempting to run high current devices from them, can damage or destroy the output transistors in the pin, or damage the I/O module or the entire device. For this reason it is a good idea to connect output pins to other devices with 470Ohm or 1 kOhm resistors, unless maximum current draw from the pins is required for a particular application.	2	Event-triggered.
Analog Input	0-5V high-impedance analog inputs. 10 bits resolution (i.e 0 – 1023)	4	60 seconds

### 3.9.2 Pin Layout



Pin	Usage
1+	+5V Out
1-	GND
2+	Digital Input 1 (DI1)
2-	Digital Input 2 (DI2)
3+	Digital Input 3 (DI3)
3-	Digital Input 4 (DI4)
4+	Analog Input 1 (AI1)
4-	Analog Input 2 (AI2)

5+	Analog Input 3 (AI3)
5-	Analog Input 4 (AI4)
6+	+5V Out
6-	GND
7a	Digital Output 1 (DO1)
7b	Digital Output 2 (DO2)

### 3.9.3 I/O Configuration

Go to *SPC Gateway* -> *Device I/O* to configure the in/outputs.

### 3.10 Test Tools

In *SPC Gateway* -> *Test Tools* you will find a tool to test SPC Panel Communication. You can select between:

- GET SPC Area Status
- GET SPC Zone Status
- GET SPC System Log (last 10 events)

## 4 Advanced users

### 4.1 Backup of Configuration Settings

In *System -> Backup / Flash Firmware*, section *Backup / Restore*, you can save a backup copy of your settings on your PC and later use it to restore the settings.

### 4.2 Upgrading software

In *System -> Backup / Flash Firmware*, section *Flash new firmware image*, you can upgrade the device firmware to a new version. The firmware file should have the name **dragino2-spc-gateway-vX.X.X-squashfs-sysupgrade.bin**. For minor upgrades you can keep your current settings by selecting Keep settings. For major upgrades it is preferable to not keep the settings, because they can be incompatible with the new firmware.

### 4.3 SSH Access

The device has as default ssh access enabled. Login as user root and with same password as in the Web GUI (default dragino). The settings for the SSH access can be altered in the Web GUI (*System -> Administration, section SSH Access*).

### 4.4 MQTT Access

The device has as an embedded MQTT message broker (Mosquitto) that you can use to subscribe for alarm and I/O-events.

#### 4.4.1 Enable access from other computers

Of security reasons the access to the MQTT broker is restricted for internal usage (127.0.0.1). To enable access from other computers on your network you have to change the configuration of the broker.

Login via ssh and comment out the **bind\_address** entry (127.0.0.1) in the configuration file **/etc/mosquitto/mosquitto.conf**:

```
# IP address/hostname to bind the default listener to. If not
# given, the default listener will not be bound to a specific
# address and so will be accessible to all network interfaces.
# bind_address ip-address/host name
#bind_address
#bind_address 127.0.0.1
```

You also need the **username** and **password** of the broker. You find these in the file **/etc/config/spc-gw-mqtt**.

After the modification, **restart the SPC Gateway** to have the changes to take effect.

You can then test the communication from another computer that you have mosquitto clients on, by entering following command:

```
mosquitto_sub -h SPC_GATEWAY_IP -u MQTT_USER -P MQTT_PASSWORD -v -t /#
```

## 4.4.2 MQTT topics

### 4.4.2.1 SPC - SIA Events

Description	MQTT topic	
SIA Event from SPC PANEL	/SPCGW/SIA_EVENT	

The **MQTT message** is embedded in a JSON frame. The format of the message is:

```
{
  "status": "success",
  "data": {
    "sia": {
      "device_id": "<device_id>",
      "timestamp": "<timestamp>",
      "sia_code": "<sia_code>",
      "sia_address": "<sia_address>",
      "description": "<description>",
      "flags": "<flags>",
      "verification_id": "<verification_id>"
    }
  }
}
```

Element	Type	Description
device_id	Integer	ID of the device
timestamp	Timestamp	Event timestamp in POSIX format (seconds since 1 Jan 1970)
sia_code	String	Two character code specifying the event. Please see [SPC_INST_CONF] section 21.7 for valid SIA codes.
sia_address	Integer	Number that depend on the SIA code.
description	String	Textual description of the event
flags	String	Each flag is a single character, and more than one flag can be concatenated to form a string. If there are no flags the flags string will be empty. A – Area Fullset D – Area Unset I – Zone Isolated or Unisolated U – Door Unlocked V – Event has video verification data W – Event has audio verification data 1 – Event is being sent by primary modem 2 – Event is being sent by backup modem
verification_id	Integer	Zero means the panel has no audio or video information for the event, otherwise it is the Verification Zone ID.

### 4.4.2.2 SPC – Alarm Zone Input Value

Description	MQTT topic	X
SPC - Zone Input Value	/SPCGW/ALARM_ZONE_X/INPUT	X is SPC Zone ID, X = 1 to max number of zones.

The **MQTT message** is embedded in a JSON frame. The format of the message is:

```
{
  "type": "ALARM_ZONE_INPUT",
  "update_time": "<timestamp>",
  "name": "<zone_name>",
  "value": "<value>"
}
```

Element	Type	Description
type	String	ALARM_ZONE_INPUT
update_time	Timestamp	Event timestamp in POSIX format (seconds since 1 Jan 1970)
zone_name	String	Zone Name
value	String	Input Value: <ul style="list-style-type: none"> <li>- CLOSED</li> <li>- OPEN</li> <li>- SHORT</li> <li>- DISCONNECTED</li> <li>- PIR_MASKED</li> <li>- DC_SUBSTITUTION</li> <li>- SENSOR_MISSING</li> <li>- OFFLINE</li> <li>- UNKNOWN</li> </ul>

#### 4.4.2.3 SPC – Alarm Zone Status Value

Description	MQTT topic	X
SPC - Zone Status Value	/SPCGW/ALARM_ZONE_X/STATUS	X is SPC Zone ID, X = 1 to max number of zones.

The **MQTT message** is embedded in a JSON frame. The format of the message is:

```
{
  "type": "ALARM_ZONE_STATUS",
  "update_time": "<timestamp>",
  "name": "<zone_name>",
  "value": "<value>"
}
```

Element	Type	Description
type	String	ALARM_ZONE_STATUS
update_time	Timestamp	Event timestamp in POSIX format (seconds since 1 Jan 1970)
zone_name	String	Zone Name
value	String	Status Value: <ul style="list-style-type: none"> <li>- OK</li> <li>- INHIBIT</li> <li>- ISOLATE</li> <li>- SOAK</li> <li>- TAMPER</li> <li>- ALARM</li> <li>- TROUBLE</li> </ul>

		- UNKNOWN
--	--	-----------

#### 4.4.2.4 SPC – Alarm Area Status Value

Description	MQTT topic	X
SPC - Area Status Value	/SPCGW/ALARM_AREA_X/STATUS	X is SPC Area ID, X = 1 to max number of areas.

The **MQTT message** is embedded in a JSON frame. The format of the message is:

```
{
  "type": "ALARM_AREA_STATUS",
  "update_time": "<timestamp>",
  "name": "<area_name>",
  "value": "<value>"
}
```

Element	Type	Description
type	String	ALARM_AREA_STATUS
update_time	Timestamp	Event timestamp in POSIX format (seconds since 1 Jan 1970)
area_name	String	Area Name
value	String	Area Status Value: <ul style="list-style-type: none"> <li>- UNSET</li> <li>- PARTSET_A</li> <li>- PARTSET_B</li> <li>- SET</li> <li>- UNKNOWN</li> </ul>

#### 4.4.2.5 I/O – Analog Input Value

Description	MQTT topic	X
Device I/O - Analog Input Value	/SPCGW/IO_ANA_IN_X/STATUS	X is Analog Input Number. X = 1 - 4

The **MQTT message** is embedded in a JSON frame. The format of the message is:

```
{
  "type": "IO_ANA_IN_STATUS",
  "update_time": "<timestamp>",
  "name": "<input_name>",
  "value": "<value>",
  "unit": "<unit>",
  "error": "<error>"
}
```

Element	Type	Description
type	String	IO_ANA_IN_STATUS
update_time	Timestamp	Event timestamp in POSIX format (seconds since 1 Jan 1970)
input_name	String	The name of the input



value	Number	Input scaled value value = ad_value/1023 * (max - min) + min
unit	String	Unit
error	Number	Error 0 – OK 1 – Invalid value

#### 4.4.2.6 I/O - Digital Input Value

Description	MQTT topic	X
Device I/O - Digital Input Value	/SPCGW/IO_DIG_IN_X/STATUS	X is Digital Input Number. X = 1 - 4

The **MQTT message** is embedded in a JSON frame. The format of the message is:

```
{
  "type": "IO_DIG_IN_STATUS",
  "update_time": "<timestamp>",
  "name": "<input_name>",
  "value": "<value>",
  "error": "<error>"
}
```

Element	Type	Description
type	String	IO_DIG_IN_STATUS
update_time	Timestamp	Event timestamp in POSIX format (seconds since 1 Jan 1970)
input_name	String	The name of the input
value	Number	0 or 1
error	Number	Error 0 – OK 1 – Invalid value

#### 4.4.2.7 I/O - Digital Output Command

The digital outputs can be set by **publishing** following topic and message:

Description	MQTT topic	Message
Device I/O - Digital Output Command	/SPCGW/IO_DIG_OUT_X/COMMAND  X is Digital Output Number. X = 1 - 2	0 or 1

**NOTE!** Don't use the retain flag.

**Example.** Set output 1 using mosquitto\_pub

```
mosquitto_pub -h SPC_GATEWAY_IP -u MQTT_USER -P MQTT_PASSWORD \
-t /SPCGW/IO_DIG_OUT_1/COMMAND -m 1
```

#### 4.4.2.8 I/O - Digital Output Value

Description	MQTT topic	X
Device I/O - Digital Output Value	/SPCGW/IO_DIG_OUT_X/STATUS	X is Digital Output Number. X = 1 - 2

The **MQTT message** is embedded in a JSON frame. The format of the message is:

```
{
  "type": "IO_DIG_OUT_STATUS",
  "update_time": "<timestamp>",
  "name": "<output_name>",
  "value": "<value>",
  "error": "<error>"
}
```

Element	Type	Description
type	String	IO_DIG_OUT_STATUS
update_time	Timestamp	Event timestamp in POSIX format (seconds since 1 Jan 1970)
output_name	String	The name of the output
value	Number	0 or 1
error	Number	Error 0 – OK 1 – Invalid value

## 5 License Agreements

### 5.1 End-User License Agreement for SPC Gateway (EULA)

#### **IMPORTANT PLEASE READ THE TERMS AND CONDITIONS OF THIS LICENSE AGREEMENT CAREFULLY BEFORE USING THE SPC GATEWAY SOFTWARE OR HARDWARE.**

SPC Gateway End-User License Agreement ("EULA") is a legal agreement between you (either an individual or a single entity) and Lundix IT, Sweden, for the SPC Gateway software and hardware product(s) (referred to as the "PRODUCT") which may also include associated software components, media, printed materials, and "online" or electronic documentation. By installing, copying, or otherwise using the PRODUCT, you agree to be bound by the terms of this EULA. This license agreement represents the entire agreement concerning the PRODUCT between you and Lundix IT (referred to as "licenser"), and it supersedes any prior proposal, representation, or understanding between the parties. If you do not agree to the terms of this EULA, do not install or use the PRODUCT.

The PRODUCT is protected by copyright laws and international copyright treaties, as well as other intellectual property laws and treaties. The software is licensed, not sold.

#### **GRANT OF LICENSE.**

The PRODUCT is licensed as follows:

- The program **SPC Web Gateway** is based on Siemens/Vanderbilt propriety protocol EDP and therefore **NOT open source**.
- **All other software in SPC Gateway** is licensed under many different open source licenses.
- **Backup Copies.** You may make copies of the software as may be necessary for backup and archival purposes.

#### **DESCRIPTION OF OTHER RIGHTS AND LIMITATIONS.**

- **Maintenance of Copyright Notices.**  
You must not remove or alter any copyright notices on any and all copies of the PRODUCT.
- **Prohibition on Reverse Engineering, Decompilation, and Disassembly.**  
You may not reverse engineer, decompile, or disassemble the program SPC Web Gateway.
- **Support Services.**  
Lundix IT may provide you with support services related to the PRODUCT ("Support Services"). Any supplemental software code provided to you as part of the Support Services shall be considered part of the PRODUCT and subject to the terms and conditions of this EULA.
- **Compliance with Applicable Laws.**  
You must comply with all applicable laws regarding use of the PRODUCT.

#### **COPYRIGHT**

All title, including but not limited to copyrights, in and to the PRODUCT and any copies thereof are owned by Lundix IT or its suppliers. All title and intellectual property rights in and to the content which may be accessed through use of the PRODUCT is the property of the respective content owner and may be protected by applicable copyright or other intellectual property laws and treaties. This EULA grants you no rights to use such content. All rights not expressly granted are reserved by Lundix IT.

#### **NO WARRANTIES**

Lundix IT expressly disclaims any warranty for the PRODUCT. The PRODUCT is provided 'As Is' without any express or implied warranty of any kind, including but not limited to any warranties of merchantability, noninfringement, or fitness of a particular purpose. Lundix IT does not warrant or

assume responsibility for the accuracy or completeness of any information, text, graphics, links or other items contained within the PRODUCT. Lundix IT makes no warranties respecting any harm that may be caused by the transmission of a computer virus, worm, time bomb, logic bomb, or other such computer program. Lundix IT further expressly disclaims any warranty or representation to Authorized Users or to any third party.

#### LIMITATION OF LIABILITY

In no event shall Lundix IT be liable for any damages (including, without limitation, lost profits, business interruption, or lost information) rising out of 'Authorized Users' use of or inability to use the PRODUCT, even if Lundix IT has been advised of the possibility of such damages. In no event will Lundix IT be liable for loss of data or for indirect, special, incidental, consequential (including lost profit), or other damages based in contract, tort or otherwise. Lundix IT shall have no liability with respect to the content of the PRODUCT or any part thereof, including but not limited to errors or omissions contained therein, libel, infringements of rights of publicity, privacy, trademark rights, business interruption, personal injury, loss of privacy, moral rights or the disclosure of confidential information.

## 5.2 Open Source Software

The SPC gateway software is based on **OpenWrt**, a Linux distribution that bundles lots of third party software, under many different licenses. Source code for OpenWrt is available on <http://dev.openwrt.org>.

The most frequently used licenses are:

**GNU General Public License (GPL) and GNU Lesser General Public License (LGPL) version 2.** These firmware images contain software licensed under the GPLv2. A copy of that license can be found at <http://www.gnu.org/licenses/gpl-2.0.txt>.

**Apache License version 2.0.** These firmware images contain software licensed under the APLv2. You may obtain a copy of the License at <http://www.apache.org/licenses/LICENSE-2.0>. Modified files carry prominent notices stating who made the changes.

**MIT License.** Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions: The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software. THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

## 5.3 Open Source Hardware

The SPC Gateway hardware is based on the open source hardware Dragino, <http://www.dragino.com>. The Dragino hardware design is released under the **Creative Commons**

**License**, <https://creativecommons.org/licenses/>. The modular Dragino design enables rapid development cycles for commercial products.

END OF DOCUMENT