HOSTLESS WLAN
HTTP API SPECIFICATION

SCA11H

Doc. No. 1424 Rev. 1

Table of Contents

1 Introduction

This document describes the HTTP API for the SCA11H BCG Sensor Node. The API enables client applications to communicate over HTTP, and the API can be used to query the device state and control the device operation through particular URLs.

The API enables multiple applications to coexist on a system, sharing the same TCP port (for example, port 80 for HTTP) and serving different parts of the URL namespace.

1.1 Data Format

The HTTP API is generated around a resource model represented by a URL for the device following RESTful web API.

- The HTTP method GET retrieves a representation of the addressed member of the resources, expressed in JSON format.
- The HTTP method POST updates existing resources and submits a JSON representation to the identified resource.
- HTTP 1.1 is supported except for features explicitly mentioned as not supported.
- HTTP Basic authentication is used for authorization with admin:admin as default username:password. Authorization is not used in configuration mode.

1.2 HTTP Response Status Code

The API uses HTTP status code to indicate the execution result. For any request to an access-controlled API, the server should first check the authentication, and return HTTP status code 401 for any invalid input. The description about server action below applies only when the session ID or serial number valid.
The status code used is described below.

Table 1 HTTP response status code

Code	Definition	Description
200	OK	Successful
400	Bad Request	The request has syntax error.
401	Unauthorized	In the case that a resource needs HTTP Basic authentication but it is missing in the request.
404	Not Found	Any request cannot be routed.
500	Internal Server Error	A generic error message, given when no more specific message is suitable.
505	HTTP Version Not Supported	The server does not support the HTTP protocol version used in the request.

1.3 Assumptions

The reader is assumed to be familiar with HTTP protocol.

1.4 Acronyms

BSN	BCG Sensor Node
REST	Representational State Transfer
SSID	Service Set Identifier

RSSI	Received Signal Strength Indication
BA	HTTP Basic authentication

1.5 Reference

Product Specification 1327 BCG binary protocol specification ENG

2 Success and Error Codes

2.1 Invalid URL

In response to invalid URL requests a 404 response code is returned.

2.2 Not Authorized

In response to not authorized requests a 401 response code is returned.

2.3 Common Scenarios

All the operation returns the following common success/error messages.

- For success
```
{
   “errno”:0
}
```
- For error
```
{
   “errno”:-1
}
```

Note: most likely cause, if some field is missing in the POST data or POST operation is not permitted in current working mode.

- For invalid URLs
```
{"error_msg":"Invalid HTTP API"}
```

3 Hostless WLAN System HTTP API

3.1 Introduction

The System HTTP API mainly includes one base resource system (/sys) which is comprised of several sub-resources that can be used to retrieve and modify the state of system.

3.2 Query Basic System Info

- URI:
  /sys
- HTTP Method:
  GET
- Return Parameter:

Table 2 Return data representation of /sys

Name	Type	Max. Length(bytes)	Description
UUID	string	12	Unique identifier for the device
name	string	16	Device name Default value is “BSN-XXXXXXXXXXXX”. (“XXXXXXXXXXXX” means device MAC address)
alias	string	32	Device alias Human readable name
fw_ver	string	12	Main firmware version Format is “1.0.0.0”
wlan_ver	string	12	WLAN firmware version Format is “1.0.0.0”
fs_ver	string	12	File system version Format is “1.0.0.0”

- Example Response body:

```
{
   "uuid": "5CF8A15D8714",
   "name": "BSN-5CF8A15D8714",
   "alias": "Bedsensor-1",
   "fs_ver": "1.0.4.0",
   "fw_ver": "2.3.0.0",
   "wlan_ver": "2.0.0.0"
}
```

3.3 Configure BSN Name

- URI:
  /sys

- HTTP Method:
  POST
- Request Parameter:

Table 3 Request data representation of /sys

Parameter	type	Max. Length(bytes)	Meaning
alias	string	32	Device alias Human readable name

- Example Request body:
  ```
  {
     "alias": "Bedsensor-2"
  }
  ```

3.4 Query Device MAC Address

- URI:
  /sys/mac
- HTTP Method:
  GET
- Return Parameter:

Table 4 Return data representation of /sys/mac

Name	Type	Max. Length(bytes)	Description
mac	string	12	Device MAC address

- Example Response body:
  ```
  {
     "mac": "5CF8A15D8714"
  }
  ```

3.5 Scan Network Results

- URI:
  /sys/scan
- HTTP Method:
  GET
- Return Parameter:

Table 5 Return data representation of /sys/scan

Name	Type	Max. Length(bytes)	Description
networks	array	1024	The list of found networks. Empty list [] if no network is found
SSID	string	32	Wireless network SSID
BSSID	string	18	AP Mac address

security	string	12	Security type Possible values “Open, WEP, WPA2 PSK, WPA/WPA2 PSK or unknown”
channel	integer	1	WLAN channel. Depending on the country code setting, the channel range is 1-11, 1-13 or 1-14.
RSSI	integer	4	Received signal strength indication in dBm

- Example Response body:

```
{
   "networks": [
      [
         "CMCC",
         "00:11:b5:1f:1d:21",
         "Open",
         11,
         -76
      ],
      [
         "dlink-yqi",
         "c8:d3:a3:18:48:88",
         "Open",
         1,
         -17
      ]
   ]
}
```

3.6 Query Network Settings

- URI:
  /sys/network
- HTTP Method:
  GET
- Return Parameter:

Table 6 Return data representation of /sys/network

Name	Type	Max. Length(bytes)	Description
sta	string	3	Indicate a station composite object. Contains details of the network BSN connects to in local/cloud mode.
ap	string	2	Indicate a soft AP composite object. Contains details of the configuration AP network.

ssid	string	32	Wireless network SSID
security	string	12	Security type Possible values “Open, WEP, WPA2 PSK, WPA/WPA2 PSK”
passphrase	string	64	The passphrase of the secure wireless network
channel	integer	1	AP channel, Fix to channel 11.
dhcp	integer	1	Enable DHCP client.
dhcpd	integer	1	Enable DHCP server.
ip	string	16	Static IP address
netmask	string	16	Static Subnet mask
gateway	string	16	Static Gateway address
ipdns1	string	16	Static Primary DNS server address
ipdns2	string	16	Static Secondary DNS server address

- Example Response body:

```
{
   "ap": {
      "ssid": "muRata_CFG_5D8714",
      "security": "WPA2 PSK",
      "passphrase": "24681012",
      "channel": 11,
      "dhcpd": 1
   },
   "sta": {
      "ssid": "testNetwork",
      "security": " WPA2 PSK ",
      "passphrase": "12345678",
      "dhcp": 1
   }
}
```

3.7 Configure Network Settings

- URI:
  /sys/network
- HTTP Method:
  POST
- Request Parameter:

Table 7 Request data representation of /sys/network

Name	Type	Length(bytes)	Description
sta	string	3	Indicate a station composite object.

			Contains details of the network BSN connects to in local/cloud mode.
ssid	string	32	Wireless network SSID
security	string	12	Security type Possible values “Open, WEP, WPA2 PSK, WPA/WPA2 PSK”
passphrase	string	64	The passphrase of the secure wireless network
dhcp	integer	1	Enable DHCP client.
ip	string	16	Static IP address
netmask	string	16	Static Subnet mask
gateway	string	16	Static Gateway address
ipdns1	string	16	Static Primary DNS server address
ipdns2	string	16	Static Secondary DNS server address

- Example Request body:

```
{
   "sta": {
      "ssid": "testNetwork",
      "security": "WPA2 PSK",
      "passphrase": "12345678",
      "dhcp": 1
   }
}
```

3.8 Query Current Network Info

- URI:
  /sys/netinfo
- HTTP Method:
  GET
- Return Parameter:

Table 8 Return data representation of /sys/netinfo

Name	Type	Max. Length(bytes)	Description
sta	string	3	Indicate a station composite object. Contains details of the network BSN connects to in local/cloud mode. Returned only if in use.
ap	string	2	Indicate a soft AP composite object. Contains details of the configuration AP network. Returned only if in use.

ssid	string	32	Wireless network SSID
security	string	12	Security type Possible values “Open, WEP, WPA2 PSK, WPA/WPA2 PSK”
passphrase	string	64	The passphrase of the secure wireless network
channel	integer	1	WLAN channel Depending on the country code setting, the channel range is 1-11, 1-13 or 1-14.
dhcp	integer	1	Enable DHCP client.
dhcpd	integer	1	Enable DHCP server.
ip	string	16	IP address
netmask	string	16	Subnet mask
gateway	string	16	Gateway address
ipdns1	string	16	Primary DNS server address
ipdns2	string	16	Secondary DNS server address

- Example Response body:

```
{
   "sta": {
      "ssid": "testNetwork",
      "security": "WPA2 PSK",
      "passphrase": "12345678",
      "dhcp": 1,
      "ip": "10.3.1.131",
      "netmask": "255.255.0.0",
      "gateway": "10.3.1.12"
   }
}
```

3.9 Query Network RSSI

- URI:
  /sys/rssi
- HTTP Method:
  GET
- Return Parameter:

Table 9 Return data representation of /sys/rssi

Name	Type	Max. Length(bytes)	Description
rssi	integer	4	Received signal strength indication in dBm

- Example Response body:
  ```
  {
     "rssi": -54
  }
  ```

3.10 Query Authentication Account

- URI:
  /sys/account
- HTTP Method:
  GET
- Return Parameter:

Table 10 Return data representation of /sys/account

Name	Type	Max. Length(bytes)	Description
username	string	16	Username to access HTTP API

- Example Response body:
  ```
  {
     "username": "admin",
  }
  ```

3.11 Configure Authentication Account

- URI:
  /sys/account
- HTTP Method:
  POST
- Request Parameter:

Table 11 Request data representation of /sys/account

Name	Type	Length(bytes)	Description
password	string	16	Password to access HTTP API
new_password	string	16	New password

- Example Request body:
  ```
  {
     “password”: ”admin”,
     "new_password": "12345678"
  }
  ```

3.12 System Command

- URI:
  /sys/cmd
- HTTP Method:
  POST
- Request Parameter:

Table 12 Request data representation of /sys/cmd

Name	Type	Length(bytes)	Description
cmd	string	8	Possible values: “reboot and “restore” “reboot”, reboots the system. “restore”, restore system’s factory settings and reboot the system.

- Example Request body:
```
{
    "cmd": "reboot"
}
{
    "cmd": "restore"
}
```

3.13 Query Communication Settings

- URI:
  /sys/comm
- HTTP Method:
  GET
- Return Parameter:

Table 13 Return data representation of /sys/comm

Name	Type	Max.Length(bytes)	Description
mode	integer	1	Communication mode 1-cloud mode, 0-local mode
https_enable	integer	1	Enable HTTPS communication.
url	string	128	Cloud server URL
username	string	16	Username to access cloud server
report_interval	integer	4	Number of samples in one XML-message (5…90)
network_id	string	64	Identification of network group
node_id	string	64	Identification of BCG Sensor Node
reset_interval	integer	4	Seconds between two consecutive time sync/BCG timestamp reset

- Example Response body:
```
{
   "mode": 1,
   "https_enable": 1,
   "url": "your.cloud.server",
   "username": "user",
   "report_interval": 30,
   "network_id": "network",
   "node_id": "testSensor",
   "reset_interval": 900
}
```

3.14 Configure Communication Settings

- URI:
  /sys/comm
- HTTP Method:
  POST
- Request Parameter:

Table 14 Request data representation of /sys/comm

Name	Type	Length(bytes)	Description
mode	integer	1	Communication mode 1 - cloud mode, 0 - local mode The parameters below are used in “1 - cloud mode” only
https_enable	integer	1	Enable HTTPS communication
url	string	128	Cloud server URL
username	string	16	Username to access cloud server
new_password	string	16	Password to access cloud server
report_interval	integer	4	Number of samples in one XML-message (5…90).
network_id	string	64	Identification of network group
node_id	string	64	Identification of BCG Sensor Node
reset_interval	integer	4	Seconds between two consecutive time sync/BCG timestamp reset. Must be divisible by “report_interval”.

- Example Request body:

```
{
    "mode": 1,
    "https_enable": 1,
    "url": "your.cloud.server",
    "username": "user",
    "new_password": "12345678",
    "report_interval": 30,
    "network_id": "network",
    "node_id": "testSensor",
    "reset_interval": 900
}
```

3.15 Query All Firmware Info

- URI:

/sys/fwinfo

- HTTP Method:
  GET
- Return Parameter:

Table 15 Request data representation of /sys/fwinfo

Name	Type	MAX. Length(bytes)	Description
fs_ver	string	12	File system version Format is “1.0.0.0”
bfs_ver	string	12	Backup file system version Format is “1.0.0.0”
fw_ver	string	12	Main firmware version Format is “1.0.0.0”
bfw_ver	string	12	Backup main firmware version Format is “1.0.0.0”
dct_ver	string	12	Device configuration table version Format is “1.0.0.0”
bdct_ver	string	12	Backup device configuration table version Format is “1.0.0.0”
wlan_ver	string	12	WLAN firmware version Format is “1.0.0.0”
boot_ver	string	12	Bootloader firmware version Format is “1.0.0.0”
ota_ver	string	12	OTA recovery firmware version Format is “1.0.0.0”

- Example Response body:

```
{
   "fs_ver": "1.0.4.0",
   "bfs_ver": "1.0.4.0",
   "fw_ver": "2.2.0.0",
   "bfw_ver": "2.2.0.0",
   "dct_ver": "2.2.0.0",
   "bdct_ver": "2.2.0.0",
   "wlan_ver": "2.0.0.0",
   "boot_ver": "2.0.0.0",
   "ota_ver": "1.0.0.0"
}
```

Note: If a firmware doesn't exist in flash, the returned field is an empty value.

3.16 Query OTA Settings

- URI:
  /sys/ota
- HTTP Method:
  GET
- Return Parameter:

Table 16 Return data representation of /sys/ota

Name	Type	MAX. Length(bytes)	Description
autoupd	integer	1	Enable firmware auto-upgrade from server.
url	string	128	OTA server URL
username	string	16	Username to access OTA server

- Example Response body:

```
{
   "autoupd": 1,
   "url": "your.cloud.server",
   "username": "user",
}
```

3.17 Configure OTA Settings

- URI:
  /sys/ota
- HTTP Method:
  POST
- Request Parameter:

Table 17 Request data representation of /sys/ota

Name	Type	Length(bytes)	Description
autoupd	integer	1	Enable auto-upgrade firmware from server.
url	string	128	OTA server URL
username	string	16	Username to access OTA server
new_password	string	16	New password to access OTA server

- Example Request body:

```
{
   "autoupd": 1,
   "url": "your.cloud.server",
   "username": "user",
   "new_password": "12345678"
}
```

3.18 System Upgrade Firmware

- URI:
  /sys/fwupd
- HTTP Method:
  POST
- Request Parameter:

Table 18 Request data representation of /sys/fwupd

Name	Type	Length(bytes)	Description
image	Text/plain	1024K	Possible firmware: “Main FW”, “DCT FW”, “FS FW” and “ALL FW” Only “Main FW” and “FS FW” are verified after upload.

- Example Response body:
  success

Note: Returns “text/plain” instead of “application/json” due to browser compatibility. Possible return values: “success” and “error -1”.

Note: After ”System Upgrade Firmware” command is completed, a separate reboot command is required to activate the new firmware. BSN will run the old firmware until reboot.

3.19 Query Webpage Language

- URI:
  /sys/language
- HTTP Method:
  GET
- Return Parameter:

Table 19 Return data representation of /sys/language

Name	Type	Length(bytes)	Description
language	string	2	Webpage language

- Example Response body:

```
{
   “language": “en”
}
```

3.20 Configure Webpage Language

- URI:
  /sys/language
- HTTP Method:
  POST
- Request Parameter:

Table 20 Request data representation of /sys/language

Name	Type	Length(bytes)	Description
language	string	2	Configure webpage language. Possible values: “zh” (Chinese) and “en” (English)

- Example Request body:
  ```
  {
     “language": “zh”
  }
  ```

3.21 Query WiFi Country Code

- URI:
  /sys/country
- HTTP Method:
  GET
- Return Parameter:

Table 21 Return data representation of /sys/country

Name	Type	Length(bytes)	Description
country	string	2	WiFi country code Possible values: “US”, “JP” or “EU/CN”

- Example Response body:
  ```
  {
     “country": “US”
  }
  ```

3.22 Configure WiFi Country Code

- URI:
  /sys/country
- HTTP Method:
  POST
- Request Parameter

Table 22 Request data representation of /sys/ country

Name	Type	Length(bytes)	Description
country	string	2	Configure WiFi country code Possible values: “US”, “JP” or “EU/CN”

- Example Request body:
  ```
  {
      “country": “JP”
  }
  ```

4 Hostless WLAN BCG HTTP API

4.1 Introduction

The BCG HTTP API mainly includes one base resource system (/bcg) which is comprised of several sub-resources that can be used to retrieve and modify the state of BCG module.

4.2 Query Basic BCG Info

- URI:
  /bcg
- HTTP Method:
  GET
- Return Parameter:

Table 23 Return data representation of /bcg

Name	Type	MAX. Length(bytes)	Description
version	string	32	BCG firmware version Format is “BCG Sensor_1.5.0.0”
mode	integer	1	BCG running mode 0-BCG (The module measures acceleration with 1kHz interval and runs the result through the algorithm. Processed output data is sent @1Hz rate.) 1-Data logger (1-axis, AC. The module measures and sends raw acceleration data with 1kHz.)
pars	string	64	BCG calibration parameters
dir	integer	1	BCG measurement direction 0- inverted, 1-normal

- Example Response body:

```
{
   "version": "BCG Sensor_1.5.0.0",
   "mode": 0,
   "pars": "7000,270,5000,0,1500,7",
   "dir": 0
}
```

4.3 Query BCG Firmware Version

- URI:
  /bcg/version
- HTTP Method:
  GET
- Return Parameter:

Table 24 Return data representation of /bcg/version

Name	Type	MAX. Length(bytes)	Description
version	string	32	BCG firmware version Format is “BCG Sensor_1.5.0.0”

- Example Response body:
  ```
  {
     "version": "BCG Sensor_1.5.0.0",
  }
  ```

4.4 Query BCG Serial Number

- URI:
  /bcg/sn
- HTTP Method:
  GET
- Return Parameter:

Table 25 Return data representation of /bcg/sn

Name	Type	MAX. Length(bytes)	Description
sn	string	13	Alphanumeric serial number (XXX XX XXXXX -XX)

- Example Response body:
  ```
  {
     "sn": "1234567890-22"
  }
  ```

4.5 Query BCG Running Mode

- URI:
  /bcg/mode
- HTTP Method:
  GET
- Return Parameter:

Table 26 Return data representation of /bcg/mode

Name	Type	MAX. Length(bytes)	Description
mode	integer	1	BCG running mode 0-BCG (The module measures acceleration with 1kHz interval and runs the result through the algorithm. Processed output data is sent @1Hz rate.) 1-Data logger (1-axis, AC. The module measures and sends raw acceleration data with 1kHz.)

- Example Response body:
```
{
   "mode": 0
}
```

4.6 Configure BCG Running Mode

- URI:
  /bcg/mode
- HTTP Method:
  POST
- Return Parameter:

Table 27 Request data representation of /bcg/mode

Name	Type	Length(bytes)	Description
mode	integer	1	BCG running mode 0-BCG (The module measures acceleration with 1kHz interval and runs the result through the algorithm. Processed output data is sent @1Hz rate.) 1-Data logger (1-axis, AC. The module measures and sends raw acceleration data with 1kHz.)

- Example Request body:
```
{
  “mode”:1
}
```

4.7 Query BCG Calibration Parameters

- URI:
  /bcg/pars
- HTTP Method:
  GET
- Return Parameter:

Table 28 Return data representation of /bcg/pars

Name	Type	MAX. Length(bytes)	Description
pars	string	64	BCG calibration parameters

- Example Response body:
```
{
   "pars": "7000,270,5000,0,1500,7"
}
```

4.8 Configure BCG Calibration Parameters

- URI:
  /bcg/pars
- HTTP Method:
  POST
- Request Parameter:

Table 29 Request data representation of /bcg/pars

Name	Type	Length(bytes)	Description
pars	string	64	BCG calibration parameters

- Example Request body:

```
{
  "pars": "7000,270,5000,0,1500,8"
}
```

4.9 Query BCG Measurement Direction

- URI:
  /bcg/dir
- HTTP Method:
  GET
- Return Parameter:

Table 30 Return data representation of /bcg/dir

Name	Type	MAX. Length(bytes)	Description
dir	integer	1	BCG measurement direction 0-inverted, 1-normal

- Example Response body:

```
{
    "dir": 1
}
```

4.10 Configure BCG Measurement Direction

- URI:
  /bcg/dir
- HTTP Method:
  POST
- Request Parameter:

Table 31 Request data representation of /bcg/dir

Name	Type	Length(bytes)	Description
dir	integer	1	BCG measurement direction 0-inverted, 1-normal

- Example Request body:
  {
  "dir": 0
  }

4.11 Query BCG Calibration Status

- URI:
  /bcg/cali
- HTTP Method:
  GET
- Return Parameter:

Table 32 Return data representation of /bcg/cali

Name	Type	Max. Length(bytes)	Description
status	integer	1	Calibration warning definition 0 - Success 1 - Please calibrate empty bed first. 2 - Risk for inaccurate calibration. External vibration detected. 3 - Risk for inaccurate calibration. Weak signal detected. 7 - Risk for inaccurate calibration. External vibration and weak signal detected. 8 - Failed. Bed sensor not responding -1 - Calibration is in progress. -2 - Calibration has not been done after boot.
phase	integer	1	Calibration phase 1 - empty bed calibration 2 - occupied bed calibration Note: Phase is only returned if calibration has been done after boot.
step	integer	1	Calibration step. 0x00: start of calibration 0x01-0x3B: time since the start of calibration in seconds 0xFF: end of calibration Note: Step is only returned if calibration has been done after boot.

- Example Response body:
  {
  "status": -1,
  "phase": 2,

```
    "step": 0
}
or
{
    "status": -2
}
```

4.12 Start BCG Calibration

- URI:
  /bcg/cali
- HTTP Method:
  POST
- Return Parameter:

Table 33 Request data representation of /bcg/cali

Name	Type	Length(bytes)	Description
phase	integer	1	Calibration phase 1-empty bed calibration 2-occupied bed calibration Return message: {“errno”:0} BCG calibration started successfully. {“errno”:-1} BCG calibration start failed. {“errno”:-2} Calibration is in progress. {“errno”:-3} Calibration is disabled in cloud mode because URL is blank.

- Example Request body:

```
{
    "phase": 1
}
```

4.13 BCG Command

- URI:
  /bcg/cmd
- HTTP Method:
  POST
- Request Parameter:

Table 34 Request data representation of /bcg/cmd

Name	Type	Length(bytes)	Description
cmd	string	16	Possible values: “restore” and “set_default_pars”

			“restore”: Restore BCG factory settings. (BCG parameters, direction and running mode) “set_default_pars”: Restore default BCG parameters.

- Example Request body:
  ```
  {
     "cmd": "restore"
  }
  {
     "cmd": "set_default_pars"
  }
  ```

4.14 BCG Upgrade Firmware

- URI:
  /bcg/fwupd
- HTTP Method:
  POST
- Request Parameter:

Table 35 Request data representation of /bcg/fwupd

Name	Type	Length(bytes)	Description
image	Text/plain	64K	

- Example Response body:
  success

Note: Returns “text/plain” instead of “application/json” due to browser compatibility. Possible return values: “success” or “error -1”.

Note: After ”BCG Upgrade Firmware” command is completed, a separate BSN reboot command is required.

5 Document Change Control

Rev.	Date	Change Description
1	07-Sep-15	Document moved to new control system and template, major improvement of the document readability. Removed section 4.5. Updated section 2.1, 2.2 and table 14.