

**Table of Contents**

Short description ..... 2

Comments ..... 2

Communication..... 2

ASCII commands..... 2

    Security ..... 3

        Login ..... 3

        Logout ..... 3

Device commands and Status read-outs..... 4

    Turn the device On or Off..... 4

    Dim or increase the light to a selected value ..... 4

    Read-out of the current status of the device..... 4

    Read-out of the Type of the device..... 4

    Read-out of the Name of the device..... 4

    Read-out of the Room name of the location of the device..... 4

    Read-out of the Floor name of the location of the device..... 4

    Read-out the current consumption of the device. .... 4

    Read-out of the total consumption ..... 5

    Read-out of total consumption for 24 hours of the previous 1-7 days..... 5

    Read-out of the 3 largest consumers in the last quarter. .... 5

    Read-out of the ID of registered (power-measuring) devices..... 5

    Read-out of the firmware version of the PC box ..... 5

    Read-out of cached mode on all devices..... 5

    Turn everything Off ..... 5

Scenarios ..... 6

    Create a Scenario ..... 6

    Add 'Set' action to Scenario ..... 6

    Add 'Fade' action to Scenario ..... 6

    Run Scenario ..... 6

    Read-out of the names of all 200 Scenarios..... 6

Home Simulation ..... 6

    Start Home Simulation..... 6

    Stop Home Simulation..... 6

    Read-out of the status of Home Simulation..... 6

# Zensehome PC box ASCII Interface

## Short description

This document describes, how to communicate with a Zensehome installation by the use of simple ASCII commands via the Zensehome PC box. It's possible to turn Zensehome units such as wall, lamp and DIN outlets on and off, and it's possible to dim the light on dimmable lamp and DIN outlets. Consumption in Watt can also be read on the individual device or for the total installation. Scenarios can be activated or even be programmed. Replay - also called Home Simulation - of the previous weeks activities and actions concerning wall, lamp and DIN outlets can be activated.

## Comments

The commands described in this document require that you run the PC box firmware and device firmware that supports the commands. In the remark, it will appear from which version that command is supported.

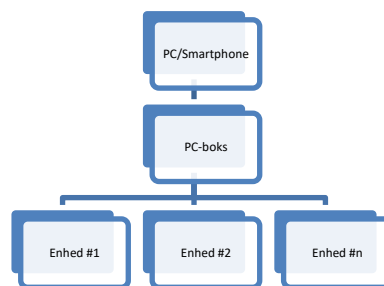
## Communication

Communication with the PC box is via the Internet or via the local area network, provided that the PC box is connected to the LAN. The communication takes place if the user has not changed in these settings under Communication Settings via TCP/IP at port 10001. By default, the PC box assigns an IP address via DHCP, but it is recommended that the user put the IP address on the PC box, and as high up in the IP address area as possible. Is the IP address area e.g. 192.168.0.1-250, the IP address of the PC box should be set to, for example, 192.168.1.250.

The PC box and all devices are physically connected to the 230V wiring and communicate via a built-in Powerline modem.

Each Zensehome device has a unique ID (e.g. 12345) that identifies the device and this ID can not be changed.

All external communication to a specific device is via the PC-box that interprets and relays the command and any responses as illustrated below.



## ASCII commands

Communication with the PC Box and LAN is via ASCII commands, which has the following format (*arguments are preceded and separated by space! Space placement is shown here but is omitted in subsequent tables*):

preamble	command	space	id	space	value	end mark
----------	---------	-------	----	-------	-------	----------

**Note that command is case sensitive!**

If the command is approved, the PC box returns with :

>>**[command]** OK<<

otherwise it returns:

>>**CMD** ?<<

If the command is known, but there are errors in the arguments, it returns:

>>**[command]** ?<<

## Zensehome PC box ASCII Interface

---

### Security

In order to make the PC box execute commands, one must be logged in with the Id of the PC Box. In three wrong trials, the PC box will be locked for new login attempts for 15 minutes.

### Login

preamble	command	id	end mark	Remark
>>	Login	{Id}	<<	5.10.3+

Returns: >>**Login OK**<<

In case of incorrect Id, it returns: >>**Login failed**<<

### Logout

The connection will be interrupted automatically by the PC-box after 45 seconds of inactivity.

preamble	command	end mark	Remark
>>	Logout	<<	5.8.5+

Returns: >>**Logout OK**<<

## Zensehome PC box ASCII Interface

### Device commands and Status read-outs

#### Turn the device On or Off

preamble	command	Id	status value	end mark	Remark
>>	Set	{id}	0 or 1	<<	5.7.2+, 1.37.6+

**status value** is indicated by 0 (Off) or 1 (On). It returns: >>Set OK<<

#### Dim or increase the light to a selected value

preamble	command	Id	value	end mark	Remark
>>	Fade	{id}	0 to 100	<<	5.7.2+, 1.37.6+

Returns: >>Fade OK<<. (value is an integer)

#### Read-out of the current status of the device

preamble	command	Id	end mark	Remark
>>	Get	{id}	<<	5.8.0+, 1.37.6+

Returns: >>Get {status}<<. (status: integer between 0-100. 0/100 is Off/On, and faded in between)

#### Read-out of the Type of the device

preamble	command	Id	end mark	Remark
>>	Get Type	{id}	<<	5.8.4+, 1.37.6+

Returns: >>Get Type {type number}<<. (type number: an integer between 0 and 19)

Type number is interpreted as follows (FUGA® is a Danish wall cavity/ceiling/wall mounting standard):

0	Switch (FUGA®)	1	Lamp Outlet, Dimmable (FUGA®)
2	Universal Outlet, Relay	3	Universal Outlet, Dimmable (Doesn't exist!)
4	Wall Outlet (FUGA®)	5	PC Boks
6	Lamp Outlet, Dimmable/PIR (FUGA®)	7	Lamp Outlet, Relay (FUGA®)
8	Lamp Outlet, Relay/PIR (FUGA®)	9	DIN-module, Relay
10	DIN-module, Dimmable	11	DIN-module, 8 port input
12	DIN Poly-phase, Metering	13	DIN Poly-phase, Relay
14	Switch EU	15	Wall Outlet EU
16	Lamp Outlet, Dimmable EU	17	Lamp Outlet, Dimmable/PIR EU
18	Lamp Outlet, Relay EU	19	Lamp Outlet, Relay/PIR EU

#### Read-out of the Name of the device

preamble	command	Id	end mark	Remark
>>	Get Name	{id}	<<	5.8.4+, 1.37.6+

Returns: >>Get Name '{name}'<<<sup>1</sup>. name is a string.

#### Read-out of the Room name of the location of the device

preamble	command	Id	end mark	Remark
>>	Get Room	{id}	<<	5.8.4+, 1.37.6+

Returns: >>Get Room '{name}'<<<sup>1</sup>. name is a string.

#### Read-out of the Floor name of the location of the device

preamble	command	Id	end mark	Remark
>>	Get Floor	{id}	<<	5.8.4+, 1.37.6+

Returns: >>Get Floor '{name}'<<<sup>1</sup>. name is a string.

#### Read-out the current consumption of the device.

preamble	command	Id	end mark	Remark
>>	WNow	{id}	<<	5.7.9+, 1.37.6+

Returns: >>WNow {Watt}<<<sup>2</sup>. Watt is a number in the range 0 to  $2^{32}-1$ .

<sup>1</sup> Returns only the first 15 characters of the name

## Zensehome PC box ASCII Interface

### Read-out of the total consumption

The reading is averaged over the last 30 minutes - last quarter reading.

preamble	command	end mark	Remark
>>	WTotal15M	<<	5.7.6+, 1.37.6+

Returns: >>WTotal15M {Watt-quarter}<<<sup>3</sup>

### Read-out of total consumption for 24 hours of the previous 1-7 days.

preamble	command	end mark	Remark
>>	WTotal{x}D	<<	5.7.6+, 1.37.6+

{x} days since - is a number between 1 and 7. Returns: >>WTotal{x}D {Watt-hours}<<<sup>3</sup>

### Read-out of the 3 largest consumers in the last quarter.

preamble	command	end mark	Remark
>>	WMax15M	<<	5.7.6+, 1.37.6+

Returns: >>WMax15M {id} {Watt-quarter} {id} {Watt-quarter} {id} {Watt-quarter}<<<sup>3</sup>

### Read-out of the ID of registered (power-measuring) devices.

preamble	command	end mark	Remark
>>	Get Devices	<<	5.8.4+, 1.37.6+

Returns: >>Get Devices {id1}{, id2}{, id3}....<<

### Read-out of the firmware version of the PC box

preamble	command	end mark	Remark
>>	Version	<<	5.10.3

Returns: >>Version {version}<<. version returns **5167** if the installed firmware version on the PC-box is 5.16.7.

**The firmware version on the PC-box must be at least 5.16.7 to work with this API!**

### Read-out of cached mode on all devices.

preamble	command	end mark	Remark
>>	Get Status	<<	5.11.6+

Returns: >>Get Status {0-31}{,32-63}....<<

If **Get Devices** retrieved 5 IDs, **Get Status** could return 0x0000001A corresponding to {0b0000 0000 0000 0000 0000 0001 1010}, which means ID number 5, 4, 2 is On and 3, 1 is Off. Since there are only 5 IDs, the other values after bit #5 are irrelevant.

If **Get Devices** returned 64 IDs, then **Get Status** might return 0x00000000 0xF0000001, giving {0b0000 ... 0000 0000} {0b1111 0000 0000 0000 0000 0000 0000 0001} corresponding to all the first 32 devices being off and that device number 33, 61, 62, 63 and 64 are turned On.

### Turn everything Off

preamble	command	end mark	Remark
>>	Sluk Alt	<<	5.7.9+, 1.37.6+

In Danish 'Sluk alt' means 'Turn everything Off'. Units in the 'Sluk Alt'/'Turn everything Off' group configured in Zense HomeControl will be turned Off.

<sup>2</sup> Consumption measurements are interpreted as such: The returned Watt value is in the range 0 to  $2^{32}-1$  and represents either watt-quarter or watt-hours depending on the ASCII-command. In practice, the values will never reach near  $2^{32}-1$ .

<sup>3</sup> Energy measurement in the units is done in Joule (J), from which is calculated the Watt consumption per quarter or per hour ( $W = J/s$ ). Depending on the ASCII-command, it returns {watt-quarter} or {watt-hours}. A 40Watt light bulb, powered on for 15 minutes, has spent 36,000 Joules in 900 seconds. So a 40Watt light bulb consumes 40Watt-quarter in one quarter. Turned on for 1 hour, it uses 40Watt-hours in one hour. Finding the hourly consumption based upon Watt-quarter read-outs by adding four {watt quarter} read-outs, gives a consumption of 160Watt, which is incorrect, so this sum must be divided by 4, if one want the correct consumption for one hour based upon four quarter read-outs.

## Zensehome PC box ASCII Interface

### Scenarios

It is possible to create different scenarios, so that different units with a single command can respectively turn On, turn Off or fade up or down. Under each scenario, a number of actions can be created, that are executed when the scenario is run.

#### Important!

When creating scenarios with the ASCII API, each scenario is save onto the PC box, but these scenarios will be overwritten as soon as a transfer is made via Zense HomeControl. So when any change is made to the Zensehome installation via Zense HomeControl, all scenarios created by the ASCII API are lost. Remember not to overwrite existing scenarios created in Zense HomeControl.

#### Create a Scenario

preamble	command	scenario #	end mark	Remark
>>	Scene Init	{number}	<<	5.8.4+, 1.37.6+

Returns: >>Scene Init Ok<<. {number} is a number between 0 and 200.

#### Add 'Set' action to Scenario

preamble	command	scenario #	action	Id	value	end mark	Remark
>>	Scene Add	{number}	Set	{id}	1 or 0	<<	5.8.4+, 1.37.6+

Returns: >>Scene Add Ok<<.

#### Add 'Fade' action to Scenario

preamble	command	scenario #	action	Id	value	end mark	Remark
>>	Scene Add	{number}	Fade	{id}	0 to 100	<<	5.8.4+, 1.37.6+

Returns: >>Scene Add Ok<<

#### Run Scenario

preamble	command	scenario #	end mark	Remark
>>	Scene Run	{number}	<<	5.8.4+, 1.37.6+

Returns: >>Scene Run Ok<<

#### Read-out of the names of all 200 Scenarios

preamble	command	end mark	Info	Remark
>>	Get Scene Names	<<	'Scene Names' are separated with \r\n (cr, lf)	5.16.7+, 1.37.6+

Returns: >>Get Scene Names Scenario 1 Scenario 2 ... Scenario 200<< (*Active as well as inactive*)

### Home Simulation

#### Start Home Simulation

preamble	command	end mark	Remark
>>	Sim On	<<	5.8.4+, 1.37.6+

Returns: >>Sim On Ok<<

#### Stop Home Simulation

preamble	command	end mark	Remark
>>	Sim Off	<<	5.8.4+, 1.37.6+

Returns: >>Sim Off Ok<<

#### Read-out of the status of Home Simulation

preamble	command	end mark	Remark
>>	Sim ?	<<	5.8.4+, 1.37.6+

Returns: >>Sim ? 0<< for Home Simulation is Off

Returns: >>Sim ? 1<< for Home Simulation is On.