



Okolab s.r.l.
P.IVA: IT04506341215
www.oko-lab.com



OKO-Control Server

UDP programming manual

Ver.02/2018 3.0.0

Administrative Headquarter
Via G. di Prisco 152 - 80040 Ottaviano (NA)
ITALY

Operative Headquarter
Via A. Olivetti 1 - 80078 Pozzuoli (NA)
Tel: +39 081 806 3470
Fax: +39 081 876 4410

1.	Introduction	4
2.	Commands	7
3.	Configuration Commands	10
3.1	List Products	10
3.2	List Devices	10
3.3	Get Product Info	10
3.4	List Connected Products	11
4.	Read Commands	12
4.1	Read COM Port	12
4.2	Read Name	12
4.3	Read Store Time	12
4.4	Read Simulation Mode	12
4.5	Read Start	13
4.6	Read Distance	13
4.7	Read Status	13
4.8	Read Bubbler	14
4.9	Read Bubbler Length	14
4.10	Read Data History Length	14
4.11	Read Value	14
4.12	Read Current Values	15
4.13	Read Set Point	16
4.14	Read Minimum Set Point	17
4.15	Read Maximum Set Point	17
4.16	Read Default Set Point	17
4.17	Read Time	18
4.18	Read Start Time	18
4.19	Read Find Status	19
4.20	Read Alarm	19
4.21	Read Working Mode	19
4.22	Read Cycles Data Length	20
4.23	Read Cycles Time	20
4.24	Read Cycles Set Point	20
4.25	Read Parameter	21
4.26	Read Available BL Parameters	22
4.27	Read BL Info	22
4.28	Read BL Parameter	23
5.	Write Commands	24
5.1	Reset Config	24
5.2	Close	24
5.3	Connect Product	24
5.4	Check and Connect Product	24
5.5	Find Product	24
5.6	Write Simulation Mode	25
5.7	Write COM Port	25
5.8	Write Start	25
5.9	Write Store Time	26
5.10	Clear Stored Data	26
5.11	Write Distance	26
5.12	Reset Bubbler	26
5.13	Write Bubbler Length	26
5.14	Write Set Point	27
5.15	Write Alarm	27
5.16	Write Working Mode	27
5.17	Write Parameter	28
5.18	Write BL Parameter	28
5.19	Add Impulse Cycle	28
5.20	Add Ramp Cycle	28



5.21	Add Square Wave Cycle	29
5.22	Delete Last Cycle	29
5.23	Delete All Cycles	29
6.	BL Parameters List	30
7.	Examples	34
8.	Change Log	47



1. Introduction

OKO-Control (OC): OKO-Control (OC) is a software that allows to communicate with the OKOlab incubators and digital gas mixers. The features present in OC are grouped in products, activated, case by case, by means of a software activation key. Each product is designed to handle a defined equipment, with all its devices, to achieve a specific need.

Table 1 reports the equipment managed by OC, the product ID, the description of the functionalities of each product, the corresponding hardware devices needed, the software key codes.

Table 1 was generated in August 2016, please contact software.support@oko-lab.com for an up to date version.

OKO-Control Server (OCS): OKO-Control Server (OCS) is an UDP server that allows to communicate with OC using commands sent by a third-party software (Client). OCS supports almost all the functions available in the graphical interface of OC, accepting any input and exposing any output.

OCS can operate in Local or Remote Mode: in Local Mode OCS will accept only read commands (see 4) in Remote Mode all commands will be accepted.

There are three ways to launch the OCS:

1. From OC, OCS can be launched from the user menu of OC (Tools / Server). In this way OC will minimize itself to the windows' system tray. By default, OCS will operate in Local Mode: user can start Remote Mode from the user menu (Functions / Remote Mode).
2. If OC is not running, any Client can directly launch OC from command line with the parameter "server" (`C:\>OKO-Control.exe server`), OC will be launched minimized to the windows' system tray and OCS will be automatically launched in Remote Mode. The user, can disable Remote Mode and shut down OCS by right-clicking the system tray icon. In this case OC can be started without license information using only simulated Products (see 3.3).
3. From version 2 of OCS, a stand-alone OCS executable is available. In this case, OCS can be run from Windows explorer or from command line without "server" parameter. Moreover, OCS does not need a license to work and it can be used in Remote Mode only.

In case 2 and 3, optional command line parameters are: UDP log (`log`, default: `n`) debug window (`debug`, default: `n`), UDP address (`address`, default: `localhost`), UDP port (`port`, default: `60000`) and automatic reconnection of devices already connected in previous sessions. (`autoconnect`, default: `n`). The UDP log is saved in Application Data /Okolab folder (eg. `C:\ProgramData\OKOlab\OKO-Control Server\UDP log.txt`)

Eg.:

```
C:\>OKO-Control.exe server debug=y address=localhost port=60000.
```

```
C:\>OKO-Control Server.exe debug=n port=60000 autoconnect=y.
```

When `debug=y`, a debug window will be opened logging every input and output server message.

Equipment	Pr. ID	Product description	Device controlled	Dev. ID
H101 series §	0	H101-Basic control software.	H101-T	0
			Water Bath *	1
	1	H101-Cryo control software.	H101-T	0
			Cryo Water Bath *	1
H201 §	2	Read Temperature Software. It allows imposing temperature set point values and storing the temperature profile during the experiment.	H201-T	0
H301 §	3	Read Temperature Software. It requires H301-TM. It allows impose temperature set point values and to store the temperature profile during the experiment.	Temperature 1	0
			Temperature 2	1
			Ref. Temp. (TM)	2
			Ref. Temp. (TC2)	3
			Humidity 1 *	4
			Humidity 2 *	5
CO2 Gas mixer §	4	Software for Digital CO2 Controllers Plus. It allows to impose CO2 set point values and to acquire and store CO2 concentration measurements in time.	DGT-CO2BX-PLUS	0
O2 Gas mixer §	5	Software for Digital O2 Controllers Plus. It allows to impose Oxygen set point values and to acquire and store O2 concentration measurements in time.	DGT-O2BX-PLUS	0
Fast Oscillating Gas Switcher §	6	Software for Fast Oscillating Gas Switcher. It allows imposing fast switches between two gas flows.	FOGS	0
UNO §	7	Software for all-in-one single specimen incubator with the size of a Multi Well plate.	Temperature 1	0
			Temperature 2*	1
			Humidity *	2
			Objective H. *	3
			Free Sensor *	4
H201 T Unit-BL	8	Software for Cage Incubator. It allow to control specimen temperature by controlling the temperature of air blown in the enclosure. It also allows to acquire and store temperature measurements in time.	H201-BL	0
H301 T Unit-BL	9	Software for Top Stage Incubator. It allow to control specimen temperature by controlling the temperature of the incubating chamber, heated glass cover and humidifier. It also allows to acquire and store temperature measurements in time.	H301-BL	0
CO2 Unit-BL	10	Software for Bold Line CO2 Controller. It allows to impose CO2 set point values and to acquire and store CO2 concentration measurements in time.	CO2-BL	0

Equipment	Pr. ID	Product description	Device controlled	Dev. ID
O2 Unit-BL [1-20]	11	Software for Bold Line Oxygen Controller. It allows to impose Oxygen set point values and to acquire and store O2 concentration measurements in time.	O2-BL	0
CO2-O2 Unit-BL [0-10;1-18]	12	Software for Bold Line combined CO2/Oxygen Controller. It allows to impose both CO2 and O2 set point values and to acquire and store concentration measurements in time for both gases.	CO2-BL	0
			O2-BL	1
CO2-O2 Unit-BL [0-20;1-95]	13	Software for Bold Line Gas Controller for Hypoxia applications. It allows to impose both CO2 and O2 set point values to a wider range of values and to acquire and store concentration measurements in time for both gases.	CO2-BL	0
			O2-BL	1
H101 T Unit-BL	14	Software for water jacket incubator controller (CRYO or BASIC).	H101-BL	0
H301-HM-ACTIVE	15	Virtual Product for active humidity controller, an optional module that can be installed to any Bold Line Gas Controller. It shares the COM port with the Gas Unit on which it is installed.	H301-HM-ACTIVE	0
H401-T-CONTROLLER	16	Software for H401-T-Controller: 2-Channel Temperature Controller compatible with all H401 heating devices. One controller can control independently up to two items.	Device 1	0
			Device 2	1
			T Sensor *	2
UNO-CONTROLLER	17	Software for UNO Stage Top Incubators: UNO-T, UNO-T-H-PREMIXED and UNO-T-H-CO2	Main	0
			Chamber *	1
			Lid *	2
			Humidifier *	3
			Objective H. *	4
			Heated Tube *	5
			T Sensor *	6

Table 1: Equipment managed by OC, with corresponding product ID, description, controlled devices and their IDs.

* Device without its own set point

§ Legacy device

2. Commands

Any message from the Client to the Server has the following general syntax:

`<Command><SPACE><Parameter 1><SPACE><Parameter 2><CR>`

Each command referring to a specific functionality requires ID of the Product as first parameter: `<Product ID>`. See Table 1 for a list of Product ID.

Any message from the server to the client has the following general syntax:

`<OC Status Code>_<command echo>_<answer><CR>`

Any message ends with the carriage return character `<CR>`.

OC Status Code:

- 0 → everything is OK
- 1 → some problems within OC, not with the Product the current command refers to.
- 2 → there is a problem with the Product the current command refers to. The command will be however executed correctly.
- 3 → there is a problem with the Product the current command refers to. The command will not be executed. Usually, this will be generated by commands that need to write on the device, e.g. Write set point (see 5.14).

There are 3 families of commands:

1. Configuration

Are used to get configuration information about available products and device.

2. Read

Are used to query OCS current data values available in OC (e.g. Current temperature value). They are always actives. Are typically equivalent to read a data value on OC graphical interface.

`<answer>`: requested data.

3. Write

Are used to set OC input parameter (e.g. Change Temperature Set Point).

If Remote Mode is not active these commands will not be executed.

`<answer>`: Remote Mode.

Remote Mode:

- 0 → not active

- 1 → active

In case of invalid command, the `answer` can be one of the following error codes:

e-####	Generic or internal error (negative number)
e10	Invalid Product ID: selected Product does not exist
e11	Invalid Parameter: one of the parameters in the command is not valid
e12	Product is connected: the command cannot be executed while selected Product is connected
e13	Unconfigured Product: the command cannot be executed until all the Devices of the Product are correctly configured
e14	Invalid command: command is not valid
e15	Invalid Device ID: selected Device does not exist
e16	Invalid connection: you can have only one Incubator product connected
e17	Product is locked
e18	Slope is too high or invalid: selected cycle will not be added
e19	Product is not connected: the command can be executed only while selected Product is connected
e20	Communication Error
e21	Invalid parameter ID
e22	Parameter value not available
e23	Parameter value out of range
e24	Device not ready
e25	Slave Device: USB cable need to be connected to the Master Unit (the one connected to the touch screen)

Table 2: Error answer

Note: These errors do not indicate a malfunctioning of OC, being only related to the command execution and do not affect the OC Status Code.

Example 1: Read last measured value of H101-BASIC (in this case, value is the Sample Temperature). See 4.11 for a detailed description of the command.

Client	<code>readval 0 1 -1</code>
Server	<code>0_readval 0 1 -1_37.3</code>



Example 2: Set current temperature set point of H101-CRYO. See 5.14 for a detailed description of the command.

Client	writesetpoint 1 37.2
Server	0_writesetpoint 1 37.2_1

Example 3: Read with a single instruction current Temperature of an H201, CO2 from a DGT-CO2BX-PLUS, O2 from an DGT-O2BX-PLUS See 4.12 for a detailed description of the command.

Client	readcurval 2 4 5
Server	0_readcurval 2 4 5_37.0 5.00 20.0

3. Configuration Commands

3.1 List Products

Function: Lists all Products

Command: `listprod`

Answer: `Product_1 Product_n`
ID of all Products (e.g. 0 1 3 7).

3.2 List Devices

Function: Lists Devices relative to a specific Product

Command: `listdev Product_ID`

Answer: `Device_1 Device_n`
ID of Devices (e.g. 0 1 2).

Possible errors: e10 (see Table 2)

The device ID is progressive for any product, starting from 0, following the list reported in Table 1. E.g. for an H101 series product the H101-T control box device will always be the ID 0, bath device will always be the ID 1.

3.3 Get Product Info

Function: List main information of a selected Product

Command: `getprodinfo Product_ID`

Answer: `Available Connected Simulated`

Possible errors: e10 (see Table 2)

Answer	Description	Value
Available	Product's Activation key presence	0: No, 1: licensed
Connected	Product is connected to OKO-Control	0: No, 1: connected
Simulated	Product is in Simulation Mode	0: Hardware, 1: simulated

NOTE:

A simulated product is always available.



3.4 List Connected Products

Function: List IDs of products that are connected

Command: `listconprod [optional param]`

Answer: `Product_ID Product_ID ... Product_ID`

Optional Parameters:

- None or 0
→ Read connected products in hardware mode
- 1
→ Read connected products in simulation mode
- 2
→ Read connected products in hardware mode that are working correctly

Possible errors: e11 (see Table 2)

4. Read Commands

4.1 Read COM Port

Function: Reads a Device COM Port of the selected Product

Command: `comport Product_ID Device_ID`

Answer: `COM port ID`

Device COM Port (eg. 1)

Possible errors: e10, e15 (see Table 2)

4.2 Read Name

Function: Read the name of the specified Product

Command: `name Product_ID`

Answer: Name

Catalog name (eg. *CO2 Unit-BL*)

Possible errors: e11 (see Table 2)

4.3 Read Store Time

Function: Reads data store time in milliseconds

Store time is the frequency values and set point are stored.

Command: `storetime`

Answer: 5000 (data are stored each 5 seconds)

NOTE:

It is recommended to use a store time higher than 500ms: shorter store times could result in a loss of accuracy, depending on the number of connected products.

4.4 Read Simulation Mode

Function: Uses a Product in Simulated Mode (without hardware physically available)

Command: `mode Product_ID`

Answer:

0 → Hardware Mode

1 → Simulated Mode

Possible errors: e10 (see Table 2)

4.5 Read Start

Function: Reads if OC is started

Command: `start`

Answer:

0 → Not started

1 → Started

4.6 Read Distance

Function: Reads actual distance mode

Command: `distance`

Answer:

0 → Long Distance

1 → Oil Immersion

4.7 Read Status

Function: Reads devices status

Command: `status`

Answer: `Product_ID_Device_ID_Errorcode`

error codes for each device space delimited (e.g. 1_0_22 1_1_25 4_0_21)

If no error occurs, answer is empty: only the OC Status Code and command echo will be sent.

20	Generic device error
21	Device serial error
22	Temperature sensor error
23	Bath Refrigerator error
24	Bath Over temperature
25	Bath water low level
26	Bath pump blocked
27	Bath internal fault 1
28	Bath internal fault 2
29	Bath external sensor
30	Device not ready
31	Slave Device: USB cable need to be connected to the Master Unit

Table 3: Error Codes List



NOTE:

These errors indicate a possible malfunctioning of OC. As a warning, if any of these errors occurs OC Status Code will be 1.

4.8 Read Bubbler

Function: Reads Bubbler value

Command: `getbubbl`

Answer: ##

bubbler percent value (e.g. 68 = 68%)

4.9 Read Bubbler Length

Function: Reads Bubbler duration in hours

Command: `getbubbldur`

Answer: ##.#

bubbler duration (e.g. 48.5 = 48h 30m)

4.10 Read Data History Length

Function: Reads number of available data

Command: `datalen`

Answer: ###

number of available data (e.g. 123 or 4).

4.11 Read Value

Function: OC reads and stores one or more measurement for each Product. This function allows to read the stored values.

Command: `readval Product_ID Device_ID [optional parameters]`

Optional Parameters:

- None
→ Read all stored values
- -1
→ Read current measured value. Current measured value can be slightly different from the last stored one



- `index`
→ Read stored values from index to the end. If `index ≥ datalen` (see 4.10), no data will be answered
- `index length`
→ Read length stored values starting from index
Note: first element is `<index>=0`

Device_ID:

Refer to Table 1 for the list of the devices available for each Product. The special `-1` parameter can be used to always get the most significant value, strictly related to the sample measure.

Answer: `##.# ##.# ... ##.#`

Values Sequence (e.g. `36.8 34.9 ... 37.0`)

Values in the sequence typically have a single fractional digit (the number of digits following the decimal point); only DGT Gas Controllers (CO₂ Gas mixer, O₂ Gas mixer) have two fractional digits and humidity controller has no fractional digit.

In case a value is not available NaN (not a number) is added to the sequence. (e.g. `36.8 NaN NaN 37.1 ... 37.0`).

Possible errors: `e10`, `e15`, `e19` (see Table 2).

NOTE:

UDP maximum datagram packet size is approximately 64kbit, depending on your OS settings. It is recommended to split large list of data in small packages.

4.12 Read Current Values

Function: Reads current value of one or more devices. It is equivalent to call several times `readval Product_ID Device_ID -1` for different Product IDs (see 4.11).

Command: `readcurval Product_ID(Device_ID) Product_ID(Device_ID) ... Product_ID(Device_ID)`

Device_ID:

Refer to Table 1 for the list of the devices available for each Product. The special `-1` parameter can be used to always get the most significant value, strictly related to the sample measure

Answer: `##.# ##.# ... ##.#`

Values Sequence (e.g. `36.8 5.8 ... 28.7`)

Possible errors: `e10`, `e15`, `e19` (see Table 2).

For some Products with a single device, Device_ID is a Unit_ID instead, as for Read Status



command (4.7).

NOTE:

Device_ID is an optional parameter; if no round brackets are present after Product_ID, the measure related to the main device (-1) of the specified product is retrieved.

4.13 Read Set Point

Function: OC stores the set point history for each product. Use this function to reads past Set Points.

Command: `readsetpoint Product_ID Device_ID [optional parameters]`

Optional Parameters:

- None
→ Read all stored Set Points
- -1 → Read current Set Point of selected product
- index
→ Read Set Points from index to the end of selected product. If $\text{index} \geq \text{datalen}$ (see 4.10), no data will be answered
- index length
→ Read length Set Points starting from index of selected product

Note: First element is index=0

Answer: `##.# ##.# ... ##.#`

Set Points Sequence of selected device (e.g. 20.0 30.0 ... 37.0)

Values in the sequence typically have a single fractional digit (the number of digits following the decimal point); only DGT Gas Controllers (CO2 Gas mixer, O2 Gas mixer) have two fractional digits and humidity controller has no fractional digit.

In case a value is not available NaN (not a number) is added to the sequence. (eg. 36.8 NaN NaN 37.1 ... 37.0).

Possible errors: e10, e15, e19 (see Table 2).

NOTE:

UDP maximum datagram packet size is approximately 64kbit, depending on your OS settings. It is recommended to split large list of data in small packages.

4.14 Read Minimum Set Point

Function: reads the minimum set point value allowed for the specified Product.

Command: `readmin Product_ID [Device_ID]`

Answer: `##. #`

Optional Parameter `Device_ID`:

- None or 0
→ Read minimum set point for the first device of the specified product.
- `Device_ID`
→ Read minimum set point for the specified device.

Decimal values typically have a single fractional digit (the number of digits following the decimal point); only DGT Gas Controllers (CO₂ Gas mixer, O₂ Gas mixer) have two fractional digits and humidity controller has no fractional digit.

Possible errors: e10, e15, e19 (see Table 2)

4.15 Read Maximum Set Point

Function: reads the maximum set point value allowed for the specified Product.

Command: `readmax Product_ID [Device_ID]`

Answer: `##. #`

Optional Parameter `Device_ID`:

- None or 0
→ Read minimum set point for the first device of the specified product.
- `Device_ID`
→ Read minimum set point for the specified device.

Decimal values typically have a single fractional digits (the number of digits following the decimal point); only DGT Gas Controllers (CO₂ Gas mixer, O₂ Gas mixer) have two fractional digits and humidity controller has no fractional digit.

Possible errors: e10, e15, e19 (see Table 2)

4.16 Read Default Set Point

Function: reads the default set point of the specified Product.

Command: `readdef Product_ID [Device_ID]`

Answer: `##. #`

Optional Parameter `Device_ID`:



- None or 0
→ Read minimum set point for the first device of the specified product.
- Device_ID
→ Read minimum set point for the specified device.

Decimal values typically have a single fractional digit (the number of digits following the decimal point); only DGT Gas Controllers (CO2 Gas mixer, O2 Gas mixer) have two fractional digits and humidity controller has no fractional digit.

Possible errors: e10, e15, e19 (see Table 2)

4.17 Read Time

Function: Reads Time in milliseconds

Command: `readtime [optional parameters]`

Optional Parameters:

- None
→ Read all Times values
- -1
→ Read current Time. Current measured time can be slightly different from the last stored one.
- index
→ Read Time from index to the end. If `index ≥ datalen` (see 4.10), no data will be answered
- index length
→ Read length Times starting from index

Note: first element is index=0

Answer: ### ### ... ###

Time Sequence (e.g. 368 1473 ... 23635)

NOTE:

UDP maximum datagram packet size is approximately 64kbit, depending on your OS settings. It is recommended to split large list of data in small packages.

4.18 Read Start Time

Function: Reads Start Control Time in Absolute format: %M-%D-%Y %H.%M.%S

Hours are in 24 hours format.

Command: `readstarttime`

Answer: mm-dd-yyyy hh.mm.ss

Start time (e.g. 01-20-2010 18.15.02)



Note: if never started, answer is: 01-01-1904 01.00.00

4.19 Read Find Status

Function: Reads Start Control Time in Absolute format: %M-%D-%Y %H.%M.%S

Hours are in 24 hours format.

Command: `findstat`

Answer: #

Find procedure percentage from 0 to 100 (e.g. 65)

Note: if find procedure is not running, answer is -1

4.20 Read Alarm

Function: This function is relative to a color code used to signal the value is within a specific interval = set point \pm range.

If the read value goes outside the interval, the alarm will be on (color code = amber). Inside, color code is green. Color code affects the system tray icon, OC widget and full GUI, control units numerical display.

This function reads the alarm range for a specified product. If alarm range is 0.0, alarm is disabled.

Command: `alarm Product_ID`

Answer: #.#

Alarm range (e.g. 1.2)

Possible errors: e10, e19 (see Table 2)

4.21 Read Working Mode

Function: Reads working mode of the specified product. Only some products have a valid working mode.

Command: `workmode Product_ID`

Answer: #

Working Mode (e.g. 1)

Possible errors: e10 (see Table 2)

Equipment	Working Mode description	Value
-----------	--------------------------	-------

H101 series ¹	Metal	0
	Sample (default)	1
H301 TC2 mode ²	TC2 (default)	2
	TM	3

Table 4: Working modes.

4.22 Read Cycles Data Length

Function: Reads number of available cycles data of a specified Product

Command: `cyclelen Product_ID`

Answer: ###

number of available cycles data (e.g. 18 or 4).

Possible errors: e10 (see Table 2)

4.23 Read Cycles Time

Function: Read cycle times of a specified Product in seconds

Command: `cycletime Product_ID [optional parameters]`

Optional Parameters:

- None
→ Read all Times values
- `index`
→ Read Time from index to the end
- `index length`
→ Read length Times starting from index

Note: first element is index=0

Answer: ### ### ... ###

Time Sequence (e.g. 368 1473 ... 23635)

Possible errors: e10 (see Table 2)

4.24 Read Cycles Set Point

Function: Read cycle times of a specified Product in seconds

¹ Control Metal or Sample Temperature: see user manual for further details.

² H301 is available in different modes. Temperature control mode TC2 can be software-wise downgraded to TM, the downgrade requires also different cable connection. See user manual for further details.

Command: `cyclesp Product_ID [optional parameters]`

Optional Parameters:

- None
→ Read all Set Point values
- `index`
→ Read Set Points from index to the end
- `index length`
→ Read length Set Points starting from index

Note: first element is index=0

Answer: `##.# ##.# ... ##.#`

Set Point Sequence (e.g. 5.0 5.0 2.0 2.0 ... 2.0)

Values in the sequence typically have a single fractional digits (the number of digits following the decimal point); only DGT Gas Controllers (CO2 Gas mixer, O2 Gas mixer) have two fractional digits and humidity controller has no fractional digit.

Possible errors: e10, e19 (see Table 2)

4.25 Read Parameter

Function: Each Product can have some specific parameters, which can be of interest in addition to the main value. This command allow to read a parameter value.

Command: `param Product_ID Parameter [input_1] [input_2]...[input_n]`

Different parameters can require a different number of inputs_# in the command, according to Table 5.

Answer: `parameter_value parameter_value ... parameter_value`

Possible errors: e10, e11 (see Table 2)

Equipment	Parameter	Input_#	Description
H101 series	tmax		Maximum temperature of water bath device
	tmin		Minimum temperature of water bath device
	incubator		Incubator state: 0 → closed 1 → opened
H301	caliboffT1		Calibration offset of "Temperature 1"
	caliboffT2		Calibration offset of "Temperature 2"
	caliboffRT		Calibration offset of "Reference Temperature"
	caliboffHM1		Calibration offset of "Humidity Module 1"

	caliboffHM2		Calibration offset of "Humidity Module 2"
	config		Name of the active configuration
	confignum		Number of available configuration names
	configlist		Available configuration names
	configinfo	config_name	chamber_name, base_name, lid_name, Troom, Tmin, Tmax
H101 series H301	offset		Temperature offset

Table 5: Parameters: For each equipment (1st column), it is reported the parameter supported (2nd column), the input if required (3rd column) and a description (4th column).

4.26 Read Available BL Parameters

Function: Each Bold Line product typically has a rather long list of parameters, which can be monitored in addition to the main value. This function allow to get a sequence of IDs corresponding to the parameters available for a specific product.

Command: `bl-param-list Product_ID [Type]`

Answer: ### ### ### ###

Param_ID Sequence (e.g. 48 73 74 75 82 ... 144)

Type:

- None or -1
→ List all the parameters
- 0
→ List all the read requests
- 1
→ Read all the write requests
- 2
→ Read all the RPC requests

Possible errors: e10, e19 (see Table 2)

4.27 Read BL Info

Function: Each Product can have some specific parameter other than the value. Use this function to read parameters value.

Command: `bl-param-info Product_ID Param_ID`

Answer: name_measure-unit_description_type



Parameter details (e.g. Heating Unit Power_W_Power Consumption of the Heating Unit_0)

Type can be 0 (read), 1 (write), or 2 (RPC)

Possible errors: e10, e21 (see Table 2)

4.28 Read BL Parameter

Function: Each Product can have some specific parameter other than the value. Use this function to read parameters value.

Command: `bl-param-read Product_ID Param_ID`

Answer: `parameter-value`

(e.g. 150)

Possible errors: e10, e19, e20, e21, e22 (see Table 2)

5. Write Commands

5.1 Reset Config

Function: Cancels actual configuration and close OC. On the next start, OC will use default configuration (no product connected, default settings)

Command: `resetconfig`

5.2 Close

Function: Closes OC

Command: `close`

5.3 Connect Product

Function: Connect selected Product

Command: `connect Product_ID Operation`

Operation:

0 → Disconnect

1 → Connect

Possible errors: e10, e11, e12, e13, e16, e17, e19 (see Table 2)

5.4 Check and Connect Product

Function: Connect selected Product only if it is found and working.

Command: `checkcon Product_ID`

Possible errors: e10, e11, e12, e13, e16, e17, e19, e20, e24, e25 (see Table 2)

5.5 Find Product

Function: automatically search and connect a specified Product. The find procedure is asynchronous as it can take some seconds to complete. So the OCS client should:

- call the "Find Product" command.
- check every 0.5 or 1 second if the procedure has been completed with "Read Find Status" command (see 4.19).
- When the procedure is completed, check which products has been found and connected using "Get Product Info" command (see 3.3) or "List Connected Products" command (see 3.4).

Command: `find Product_ID`

Possible errors: e10, e12 (see Table 2)

NOTE:

If the specified product is found, settings are saved so when OCS is restarted it is automatically connected.

It is not possible to run a find procedure if it is already in progress. No error will be reported in this case, so OCS client should check it before using “Read Find Status” command (see 4.19).

5.6 Write Simulation Mode

Function: use a Product in Simulated Mode (without hardware device)

Command: `mode Product_ID Mode`

Mode:

0 → Hardware Mode

1 → Simulated Mode

Possible errors: e10, e11 (see Table 2)

NOTE:

After a simulation mode change, OCS need to be restarted.

5.7 Write COM Port

Function: Sets a Device COM Port for the selected Product.

Command: `comport Product_ID Device_ID COM_Port_Number`

E.g.: To set the com port of a water bath (Device_ID 1) for an H101-BASIC (Product_ID 0) to COM3: `comport 0 1 3`

Possible errors: e10, e12, e15 (see Table 2)

If two or more Bold Line (BL) units are stacked, related Products share the same COM port. Indeed, only the master unit (the one connected to the touch screen) must be connected to the PC.

5.8 Write Start

Function: Starts or stops data storage and control

Command: `start Status`



Status:

0 → Stop

1 → Start

Possible errors: e11 (see Table 2)

If the control is stopped and then started again the new data will be appended to the stored ones. In order to start a control with refreshed data, see 5.10.

5.9 Write Store Time

Function: Set data store frequency in milliseconds

Command: `storetime Store_Time`

Store time is the frequency values and set point are stored.

Possible errors: e11 (see Table 2)

5.10 Clear Stored Data

Function: clear all stored data and restart all times

Command: `clear`

5.11 Write Distance

Function: sets actual distance mode

Command: `distance type`

Type:

0 → Long Distance

1 → Oil Immersion

Possible errors: e11 (see Table 2)

5.12 Reset Bubbler

Function: Resets bubbler status

Command: `resetbubbl`

5.13 Write Bubbler Length

Function: Writes bubbler duration in hours



Command: `setbubldur ##. #`

Possible errors: e11 (see Table 2)

5.14 Write Set Point

Function: Writes the Set Point of selected product

Command:

`writesetpoint Product_ID Device_ID #. #`

Parameter: `##. #`

New Set Point (e.g. `02.3` = 2.3 °C)

Possible errors: e10, e11, e23 (see Table 2)

5.15 Write Alarm

Function: This function is relative to a color code used to signal the value is within a specific interval = set point ± range.

If the read value goes outside the interval, the alarm will be on (color code = amber). Inside, color code is green. Color code affects the system tray icon, OC widget and full GUI, control units numerical display.

This function writes the alarm range for a specified product. If alarm range is 0, alarm is disabled.

Command:

`alarm Product_ID #. #`

Parameter: `##. #`

New alarm range (eg. `1.4`)

Possible errors: e10, e11, e19 (see Table 2)

5.16 Write Working Mode

Function: Changes working mode of the specified product. Only some products have a valid working mode (see Table 4)

Command: `workmode Product_ID working_mode`

Possible errors: e10, e11 (see Table 2)

NOTE:

You need to restart OC to use the new working mode.

5.17 Write Parameter

Function: Each Product can have some specific parameter, which can be modified in addition to the set point (see Table 5). Use this function to set parameters value.

Command: `param Product_ID Parameter parameter_value`

Possible errors: e10, e11, e19 (see Table 2)

5.18 Write BL Parameter

Function: Each Bold Line product typically has several parameters, which can be modified in addition to the set point. This function is used to set a parameter value.

Command: `bl-param-write Product_ID Parameter_ID parameter_value`

Possible errors: e10, e19, e20, e21, e22, e23 (see Table 2)

5.19 Add Impulse Cycle

Function: Adds an impulse cycle

Command:

`impulse Product_ID Time Value`

Time: ##

Start Time in seconds (e.g. `563.5` = 563.5 seconds)

Value: ##. #

New Set Point (e.g. `02.3` = 2.3 °C)

Possible errors: e10, e11, e19 (see Table 2)

5.20 Add Ramp Cycle

Function: Add a ramp cycle is used to create a linear increase/decrease of the product set point from the old value to a new value, the time required for the increase/decrease will be length.

Command:

`ramp Product_ID Time Value Length`

Time: ##. #

Start Time in seconds (eg. `563.5` = 563.5 seconds)

Value: ##. #

New Set Point (eg. `02.3` = 2.3 °C)



Length: ##

Duration of the ramp in seconds (eg. 47 = 47 seconds)

Possible errors: e10, e11, e18, e19 (see Table 2)

5.21 Add Square Wave Cycle

Function: Add a square wave cycle

Command:

```
squarew Product_ID Time Value SPtime SPnum
```

Time: ##

Start Time in seconds (eg. 563.5 = 563.5 seconds)

Value: ##. #

New Set Point (eg. 02.3 = 2.3 °C)

SPtime: ##

Duration of each semi-period time in seconds (eg. 22.5 = 22.5 seconds)

SPnum: ##

number of semi-periods (eg. 5 = 5 semi-periods)

Possible errors: e10, e11, e18, e19 (see Table 2)

5.22 Delete Last Cycle

Function: Deletes the last cycle programmed for future times. Past cycles cannot be deleted.

Command:

```
delcycle Product_ID
```

Possible errors: e10, e19 (see Table 2)

5.23 Delete All Cycles

Function: Deletes all cycles programmed for future times. Past cycles cannot be deleted.

Command:

```
delcycles Product_ID
```

Possible errors: e10, e19 (see Table 2)

6. BL Parameters List

ID	Name	Unit	Type	Description	Product IDs
75	Air Heater flow rate	m ³ /h	Read	The current Air Heater flow rate.	8
89	Air Heater flow rate Set Point	m ³ /h	Read	Desired Air Heater flow rate set by the user.	8
90	Air Heater flow rate Set Point	m ³ /h	Write	Desired Air Heater flow rate set by the user.	8
97	Air Heater power	W	Read	The thermal power generated by the Air Heater.	8
73	Air Heater relay percentage	%	Read	The percentage on time in which the Air Heater relay is active.	8
74	Air Heater temperature	°C	Read	The current Air Heater temperature.	8
28	Air flow rate	ml/min	Read	The current air flow rate.	10, 11, 12, 13
237	Air mode status		Read	Air mode status (also called CO2 only mode). (0: Disabled, 1: Enabled)	11, 12, 13
238	Air mode status		Write	Air mode status (also called CO2 only mode). (0: Disabled, 1: Enabled)	11, 12, 13
18	Air valve current	mA	Read	The current measured by air valve.	10, 11, 12
13	Air valve duty	%	Read	The duty cycle of the air valve.	10, 11, 12, 13
23	Air valve opening percentage	%	Read	The actual opening percentage of air valve.	10, 11, 12
115	Base offset	°C	Read	Current temperature offset applied to heating chamber.	9, 14
103	Base relay percentage	%	Read	The percentage on time in which the Base relay is active.	9
107	Base temperature	°C	Read	The temperature currently measured by the heating chamber.	9, 14
30	CO2 flow rate	ml/min	Read	The current CO2 flow rate.	10, 12, 13
129	CO2 status		Read	Status code of the CO2. (0: OK, 1: Transient, 2: Alarm, 3: Error)	10, 12, 13
20	CO2 valve current	mA	Read	The current measured by CO2 valve.	10, 12
15	CO2 valve duty	%	Read	The duty cycle of the CO2 valve.	10, 12, 13
25	CO2 valve opening percentage	%	Read	The actual opening percentage of CO2 valve.	10, 12
87	Cage temperature	°C	Read	The temperature currently measured inside the cage	8
83	Control mode		Read	Control mode currently in use. (0: Sample, 1: Cage)	8
84	Control mode		Write	Control mode currently in use. (0: Sample, 1: Cage)	8

123	Control mode		Read	Control mode currently in use. (0: Sample, 1: Chamber)	9, 14
124	Control mode		Write	Control mode currently in use. (0: Sample, 1: Chamber)	9, 14
98	Fan speed	rpm	Read	The current speed of the Air Heater fan.	8
186	Filter replacement alert		Read	A number used to signal the need to replace a filter. (0: Filters ok, 1: HEPA filter alarm, 2: Prefilter alarm)	8
119	Free sensor temperature	°C	Read	Temperature currently measured by the free sensor.	9, 14
375	Gas control paused		Read	Pause status of gas control. (0: Running, 1: Paused)	10, 11, 12
376	Gas control paused		Write	Pause status of gas control. (0: Running, 1: Paused)	10, 11, 12
31	Gas1 product code		Read	The product name of the first gas unit.	10, 11, 12, 13
32	Gas1 production date		Read	The production date of the first gas unit.	10, 11, 12, 13
34	Gas1 serial number		Read	The serial number of the first gas controller.	10, 11, 12, 13
33	Gas1 service date		Read	The date of the last update for the first gas unit.	10, 11, 12, 13
35	Gas1 software version		Read	The software version of the first gas unit.	10, 11, 12, 13
58	Gas2 product code		Read	The product name of the secondary gas unit.	10
59	Gas2 production date		Read	The production date of the secondary gas unit.	10
61	Gas2 serial number		Read	The serial number of the secondary gas unit.	10
60	Gas2 service date		Read	The date of the last update for the secondary gas unit.	10
62	Gas2 software version		Read	The software version of the secondary gas unit.	10
185	HEPA filter usage time	day	Read	The time elapsed since last HEPA filter change.	8
214	Humidifier relay percentage	%	Read	The percentage on time in which the humidifier relay is active	12, 13, 15
206	Humidifier temperature	°C	Read	The current Temperature of the Humidifier.	10, 12, 13, 15
117	Humidity module 1 offset	°C	Read	Current temperature offset applied to humidity module n.1	9
105	Humidity module 1 relay percentage	%	Read	The percentage on time in which the humidity module n.1 relay is active.	9
109	Humidity module 1 temperature	°C	Read	The temperature currently measured at the humidity module n.1	9
106	Humidity module 2 relay percentage	%	Read	The percentage on time in which the humidity module n.2 relay is active.	9
110	Humidity module 2 temperature	°C	Read	The temperature currently measured at the humidity module n.2	9
362	Humidity product code		Read	The product name of the humidity unit.	15

363	Humidity production date		Read	The production date of the humidity unit.	15
365	Humidity serial number		Read	The serial number of the humidity unit.	15
364	Humidity service date		Read	The date of the last update for the humidity unit.	15
366	Humidity software version		Read	The software version of the humidity unit.	15
233	Humidity status		Read	Status code of the Humidity. (0: OK, 1: Transient, 2: Alarm, 3: Error)	10, 12, 13, 15
116	Lid Offset	°C	Read	Current temperature offset applied to heating chamber cover.	9
104	Lid Relay percentage	%	Read	The percentage on time in which the Lid relay is active.	9
108	Lid temperature	°C	Read	The temperature currently measured by the heating chamber cover.	9
29	N2 flow rate	ml/min	Read	The current flow rate of nitrogen.	11, 12, 13
19	N2 valve current	mA	Read	The current measured by nitrogen valve.	11, 12
14	N2 valve duty	%	Read	The duty cycle of the nitrogen valve.	11, 12, 13
24	N2 valve opening percentage	%	Read	The actual opening percentage of nitrogen valve.	11, 12
150	O2 flow rate	l/min	Read	The current oxygen flow rate.	13
130	O2 status		Read	Status code of the O2. (0: OK, 1: Transient, 2: Alarm, 3: Error)	11, 12, 13
151	O2 valve duty	%	Read	The duty cycle of the oxygen valve.	13
192	Objective Heater offset	°C	Read	Current temperature offset applied to objective heater.	9, 14
190	Objective Heater relay percentage	%	Read	The percentage on time in which the Objective Heater relay is active.	9, 14
189	Objective Heater temperature	°C	Read	The temperature currently measured by the sensor inside the objective heater.	9, 14
254	Open incubator		Read	Open Incubator mode status. (0: Closed, 1: Opened)	9, 14
383	Open incubator		Write	Open Incubator mode status. (0: Closed, 1: Opened)	9, 14
184	Prefilter usage time	day	Read	The time elapsed since last prefilter change.	8
132	Room temperature	°C	Read	The temperature currently measured by the room sensor.	9, 14
88	Sample temperature	°C	Read	The temperature currently measured by the free sensor.	8
215	Sensing cell relay percentage	%	Read	The percentage on time in which the sensing cell relay is active (Humidifier).	10, 12, 13, 15
209	Sensing cell temperature	°C	Read	The current temperature of the sensing cell (Humidifier).	10, 12, 13, 15
64	Temperature product code		Read	The product name of the temperature unit.	8, 9, 14
65	Temperature production date		Read	The production date of the temperature unit.	8, 9, 14

67	Temperature serial number		Read	The serial number of the temperature unit.	8, 9, 14
66	Temperature service date		Read	The date of the last update for the temperature unit.	8, 9, 14
68	Temperature software version		Read	The software version of the temperature unit	8, 9, 14
128	Temperature status		Read	Status code of the Temperature. (0: OK, 1: Transient, 2: Alarm, 3: Error)	8, 9, 14
138	Total flow rate	l/min	Read	The flow rate of the output gas.	10, 11, 12, 13, 15
153	Total flow rate setpoint	l/min	Read	The total flow rate desired value currently set by the user.	10, 11, 12, 15
154	Total flow rate setpoint	l/min	Write	The total flow rate desired value currently set by the user.	10, 11, 12, 15
367	Total flow valve duty	%	Read	The duty cycle of the total flow valve.	15
259	Water Bath reference temperature	°C	Read	The temperature reference of the water thermostat	14

7. Examples

Example 1

Following is an example of an UDP session.

A client configures a H101-basic and a DGT-O2BX-PLUS-S, the measurements is started and the client acquires temperature and O2 concentration values. Some errors have been added as example.

1	Client	listprod	List available products.
2	Server	0_listprod_0 1 2 3 4 5 6	
3	Client	listdev 74	List devices.
4	Server	0_listdev 74_e10	<u>Error:</u> The product ID doesn't exist.
5	Client	listdev 0	List H101-Basic devices.
6	Server	0_listdev 0_0 1	
7	Client	listdev 5	List DGT-O2BX-PLUS-S devices.
8	Server	0_listdev 5_0	
9	Client	getprodinfo 0	Get H101-Basic information:
10	Server	0_getprodinfo 0_1 0 0	product is available, not connected and in hardware mode.
11	Client	getprodinfo 5	Get DGT-O2BX-PLUS-S information:
12	Server	0_getprodinfo 5_1 0 0	product is available, not connected and in hardware mode.
13	Client	connect 0 1	Connect H101-Basic Product.
14	Server	0_connect 0 1_e13	<u>Error:</u> The device is not configured (COM is 0).
15	Client	mode 0 1	Change working mode of H101-Basic to Simulated.
16	Server	0_mode 0 1_1	
17	Client	connect 0 1	Connect H101-Basic Product.
18	Server	0_connect 0 1_1	
19	Client	getprodinfo 0	Get H101-basic information:
20	Server	0_getprodinfo 0_1 1 1	product is available, connected and simulated.
21	Client	mode 5 1	Change working mode of DGT-O2BX-PLUS-S to simulated.
22	Server	0_mode 5 1_1	
23	Client	connect 5 1	Connect DGT-O2BX-PLUS-S

24	Server	0_connect 5 1_1	Product.
25	Client	getprodinfo 5	Get DGT-O2BX-PLUS-S information: product is available, connected and simulated.
26	Server	0_getprodinfo 5_1 1 1	
27	Client	storetime 10000	Set store time to 10 seconds.
28	Server	0_storetime 10000_1	
29	Client	storetime	Read store time value to check if it is correctly changed.
30	Server	0_storetime_10000	
31	Client	status	Read status: no errors.
32	Server	0_status_	
33	Client	start 1	Start control: H101 will start the Temperature control, and Temperature and O2 concentration will be stored once every storetime.
34	Server	0_start 1_1	
35	Client	datalen	Software has stored 76 data.
36	Server	0_datalen_76	
37	Client	readval 0 0 65	Read all stored values of H101-basic product from index 65.
38	Server	0_readval 0 0 65_35.1 36.5 37.5 35.0 37.8 35.8 35.7 37.8 35.8 35.0 36.8	
39	Client	readval 0 0 50 4	Read 4 stored values of H101-basic product from index 50.
40	Server	0_readval 0 0 50 4_35.5 36.3 35.6 36.4	
41	Client	readval 0 0 -1	Read current measured value of H101-basic.
42	Server	0_readval 0 0 -1_36.3	
43	Client	readval 5 0 -1	Read current measured value of DGT-O2BX-PLUS-S.
44	Server	0_readval 5 0 -1_5.22	
45	Client	readval 0 0 70	Read 6 stored values of H101-basic product from index 70.
46	Server	0_readval 0 0 70_6.30.5 30.6 30.0 30.1 29.9 30.0	
47	Client	readsetpoint 0 0 -1	Read current Set Point value of H101-basic.
48	Server	0_readsetpoint 0 0 -1_37.0	
49	Client	readsetpoint 0 0 6 4	Read 4 stored Set Point value of H101-basic from index 4.
50	Server	0_readsetpoint 0 0 6 4_37.0 37.0 37.0 37.0	
51	Client	readsetpoint 5 0 75 1	Read last stored Set Point value of

52	Server	0_readsetpoint 5 0 75 1_5.00	DGT-O2BX-PLUS-S.
53	Client	writesetpoint 0 0 37.2	Change current Set Point value of H101-basic.
54	Server	0_writesetpoint 0 0 37.2_1	
55	Client	readsetpoint 0 0 -1	Read current Set Point value of H101-basic.
56	Server	0_readsetpoint 0 0 -1_37.2	
57	Client	datalen	Software has stored 78 data.
58	Server	0_datalen_78	
59	Client	readsetpoint 0 0 75	Read stored Set Point values of H101-basic from index 75.
60	Server	0_readsetpoint 0 0 75_37.0 37.0 37.2 37.2	

Example 2

A client configures an H101-Cryo Product in hardware mode and connect it.

1	Client	getprodinfo 1	Get H101-Cryo information: product is available, not connected and in hardware mode.
2	Server	0_getprodinfo 1_1 0 0	
3	Client	connect 1 1	Try to connect H101-Cryo, command is refused with e13 (unconfigured Product).
4	Server	0_connect 1 1_e13	
5	Client	comport 1 0 1	Configure H101-T1 Device of H101-Cryo Product (ID 0) to use COM 1.
6	Server	0_comport 1 0 1_1	
7	Client	comport 1 1 2	Configure Cryo Water Bath Device of H101-Cryo Product (ID 1) to use COM 2.
8	Server	0_comport 1 1 2_1	
9	Client	connect 1 1	Connect H101-Cryo Product.
10	Server	0_connect 1 1_1	
11	Client	status	Check OC status: H101-T1 has a serial problem.
12	Server	1_status_1_1_21	
13			Water Bath is off! Turn on Water Bath.
14	Client	status	Check OC status: now everything is working fine.
15	Server	0_status_	
16	Client	start 1	Start data storage and temperature control.
17	Server	0_start 1_1	

Example 3

Manage license: current license is not valid, the license is changed and checked.

1	Client	licensestat	Check if current license is valid: it is not valid.
2	Server	0_licensestat_0	
3	Client	license ## xxxxxxxxxxxxxxxxx	Change license with a new one: serial number = ##, activation code = xxxxxxxxxxxxxxxxx.
4	Server	0_license ## xxxxxxxxxxxxxxxxx_1	
5	Client	close	Shut down OC.
6	Server	0_close_1	
7	cmd	OKO-Control.exe server	OC is launched from command line, OCS is in remote mode.
5	Client	licensestat	Check if current license is valid: now it is valid.
6	Server	0_licensestat_1	

Example 4

A client configured and connected an H201-T Product. Distance mode is changed.

1	Client	distance	Distance mode is "Long Distance".
2	Server	0_distance_0	
3	Client	distance 1	Change distance mode to "Oil Immersion".
4	Server	0_distance 1_1	
5	Client	distance	Distance Mode is "Oil Immersion".
6	Server	0_distance_1	

Example 5

A client configured a H101-Cryo Product, and started the control. Some information are read, the incubator is opened and closed.

1	Client	datalen	OC has collected 1612 measures and set points for each Product.
2	Server	0_datalen_1612	
3	Client	readsetpoint 1 0 -1	Read current Set Point value of H101-Cryo.
4	Server	0_readsetpoint 1 0 -1_37.0	
5	Client	readval 1 0 1600 5	Read 5 stored values of H101-Cryo

6	Server	0_readval 1 0 1600 5_37.0 37.0 37.0 37.0 37.0	from index 1600
7	Client	param 1 incubator	Incubator is closed.
8	Server	0_param 1 incubator_0	
9	Client	param 1 incubator 1	Open incubator.
10	Server	0_param 1 incubator 1_1	
11	Client	param 1 incubator	Incubator is opened
12	Server	0_param 1 incubator_1	
13	Client	datalen	OC has collected 1957 measures and set points for each Product.
14	Server	0_datalen_1957	
15	Client	readval 1 0 1955 5	Read 5 stored values of H101-Cryo from index 1940.
16	Server	0_readval 1 0 1940 5_33.1 33.0 32.9 32.8 32.7	
17	Client	param 1 incubator 0	Close incubator.
18	Server	0_param 1 incubator 0_1	
19	Client	param 1 incubator	Incubator is closed.
20	Server	0_param 1 incubator_0	

Example 6

A client configured a H301-TC1. After, checks if calibration offsets are right, according to H301-TC Calibration Document. Finally, it checks and changes current configuration.

1	Client	param 3 caliboffset1	Read the current offset of "Temperature 1" module.
2	Server	0_param 3 caliboffset1_-0.8	T1 Offset reported in H301-TC calibration document is -0.8 °C so it is correct.
3	Client	param 3 caliboffset2	Read the calibration offset of "Temperature 2" module
4	Server	0_param 3 caliboffset2_-0.7	T2 Offset reported in H301-TC calibration document is -0.3 °C so it is <u>not</u> correct.
5	Client	param 3 caliboffset2 -0.3	Change the calibration offset of "Temperature 2" module
6	Server	0_param 3 caliboffset2 -0.3_1	The new calibration offset of "Temperature 2" module has been correctly received by OCS
7	Client	param 3 caliboffset2	Read the calibration offset of "Temperature 2" module
8	Server	0_param 3 caliboffset2_-0.3	Now the calibration offset of "Temperature 2" module is correct
9	Client	param 3 config	Check the current configuration
10	Server	0_param 3 config_EC_35BASE_35LID@23C	The current configuration is EC_35BASE_35LID@23C
11	Client	param 3 configinfo EC_35BASE_35LID@23C	Check information of current configuration
12	Server	0_param 3 configinfo ec_35base_35lid@23c_H301-EC H301-EC-35BASE H301-EC-35LID 23 29.2 40.3	The configuration EC_35BASE_35LID@23C has the following information: Chamber: H301-EC Base: H301-EC-35BASE Lid: H301-EC-35LID Room Temperature: 23°C Minimum Temperature: 29.2 °C Maximum Temperature: 40.3 °C
13	Client	param 3 confignum	Ask for number of available configurations

14	Server	0_param 3 confignum_9	Nine configurations are available
15	Client	param 3 configlist	Ask for all the available configurations
16	Server	0_param 3 configlist EC_2x35BASE_HG-LID@24C EC_3560BASE_3560LID@24C EC_35BASE_35LID@23C EC_35BASE_HG-LID@24C EC_6MWBASE_HG-LID@25C EC_GS35BASE_GS35LID@23C EC_GS35BASE_HG-LID@23C EC_LG-35BASE_LG-LID@25C EC_PZ100_GS35PA@23C	The names of all available configurations are listed previously were .
17	Client	param 3 config EC_6MWBASE_HG-LID@25C	Change the current configuration to EC_6MWBASE_HG-LID@25C
18	Server	0_param 3 config ec_6mwbase_hg-lid@25c_1	The new configuration has been correctly received by OCS
19	Client	param 3 config	Check if configuration has been changed
20	Server	0_param 3 config EC_6MWBASE_HG-LID@25C	Configuration has been correctly changed

Example 7

A client configured an H201 and a DGT-CO2BX-PLUS Products, but DGT-CO2BX-PLUS has a communication problem.

1	Client	<code>readval 2 0 -1</code>	Read the current sample Temperature from an H201.
2	Server	<code>1_readval 2 0 -1_37.1</code>	There is some error, not relative to the product 2, so the actual temperature is correctly read.
3	Client	<code>readval 4 0 -1</code>	Read the current CO2 concentration from a DGT-CO2BX-PLUS.
4	Server	<code>2_readval 4 0 -1_NaN</code>	There is an error related to the product 4 (DGT-CO2BX-PLUS), it is possible to read the CO2 concentration, querying values stored by OC, but the value given is NaN (Not a Number) since OC was not able to read it.
5	Client	<code>readsetpoint 4 0 -1</code>	Read the current CO2 setpoint from a DGT-CO2BX-PLUS.
6	Server	<code>2_readsetpoint 4 0 -1_5.00</code>	There is an error related to the product 4 (DGT-CO2BX-PLUS) , but it is still possible to read the CO2 setpoint, since the value has been previously memorized by OC, the value given is 5.00%.
7	Client	<code>writesetpoint 4 0 5.5</code>	Write the current CO2 setpoint to a DGT-CO2BX-PLUS.
8	Server	<code>3_writesetpoint 4 0 5.5_1</code>	There is an error related to the product 4 (DGT-CO2BX-PLUS) and it is not possible to change product setpoint (status code is 3).
9	Client	<code>readsetpoint 4 0 -1</code>	Read the current CO2 setpoint from a DGT-CO2BX-PLUS.
10	Server	<code>2_readsetpoint 4 0 -1_5.00</code>	Set Point has not been changed (see row 6).
11	Client	<code>status</code>	Check the status.

12	Server	1_status_4_0_21	Product 4 gives an error on the device 0 (DGT-CO2BX-PLUS). Error code 21: device serial error. Probably the device is off or unplugged.
13	Client	connect 4 0	Disconnect the DGT-CO2BX-PLUS
14	Server	2_connect 4 0_1	The product is disconnected.
15	Client	readval 2 0 -1	Read the current sample Temperature from an H201
16	Server	0_readval 2 0 -1_37.0	There is no error, the actual temperature is correctly read
17	Client	readval 4 0 -1	Read the current CO2 concentration from a DGT-CO2BX-PLUS
14	Server	0_readval 4 0 -1_e19	The answer is an error e19: Product is not connected, the command cannot be executed while selected Product is not connected. Status code is 0: no error among connected products.

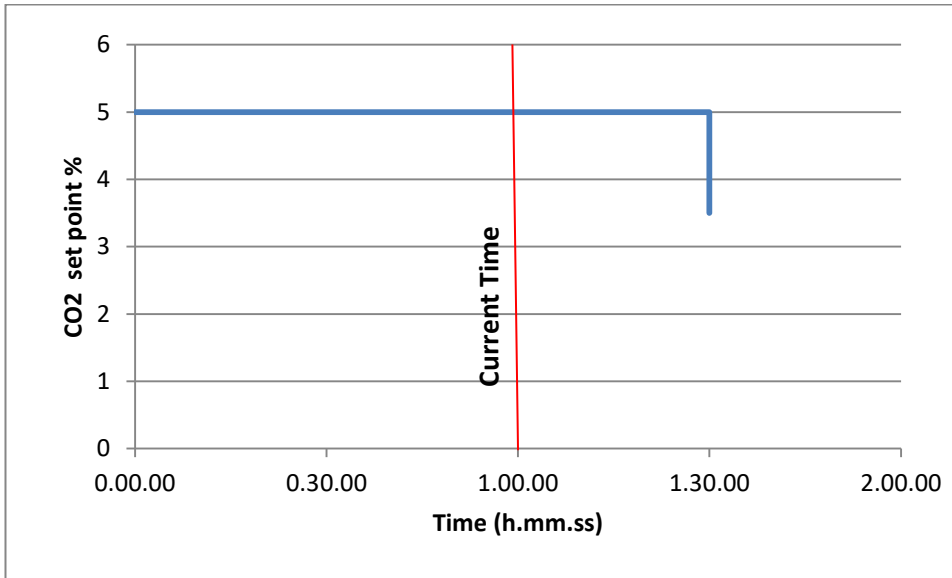
Example 8

Bold Line example: a client configured a CO2-O2 Unit-BL [0-10;1-18] Products; Product is working fine and some BL parameters are read/set.

1	Client	readval 12 0 -1	Read the current CO2 concentration from CO2-BL device.
2	Server	0_readsetpoint 12 0 -1_5.0	
3	Client	readval 12 1 -1	Read the current O2 concentration from O2-BL device.
4	Server	0_readval 12 1 -1_4.5	
5	Client	readcurval 12(0) 12(1)	Read current value of CO2 and O2 concentrations using a single command.
6	Server	0_readcurval 12(0) 12(1)_5.0 4.5	
7	Client	bl-param-list 12	List all available BL parameter IDs.
8	Server	0_bl-param-list 12_13 14 15 18 19 20 23 24 25 28 29 30 34 35 37 41 100 101 129 130 138 153 154 158 159 161	
9	Client	bl-param-info 12 14	Read information about parameter 14: Name, unit, description and type (read).
10	Server	0_bl-param-info 12 14_N2 valve duty_%_The duty cycle of the Nytrogen valve._0	
11	Client	bl-param-read 12 14	Read parameter 14 value.
12	Server	0_bl-param-read 12 14_74.0	
13	Client	bl-param-write 12 14	Try to write a read-only parameter: error e21 occurs.
14	Server	0_bl-param-write 12 14_e21	
15	Client	bl-param-list 12 1	List only write BL parameter IDs.
16	Server	0_bl-param-list 12 1_154	
17	Client	bl-param-info 12 154	Check information about parameter 154.
18	Server	0_bl-param-info 12 154_Total flow_rate Set Point_l/min_New desired total flow rate._1	
19	Client	bl-param-write 12 154 0.7	Try to change value of a write parameter: it works fine.
20	Server	0_bl-param-write 12 154 0.7_1	

Example 9

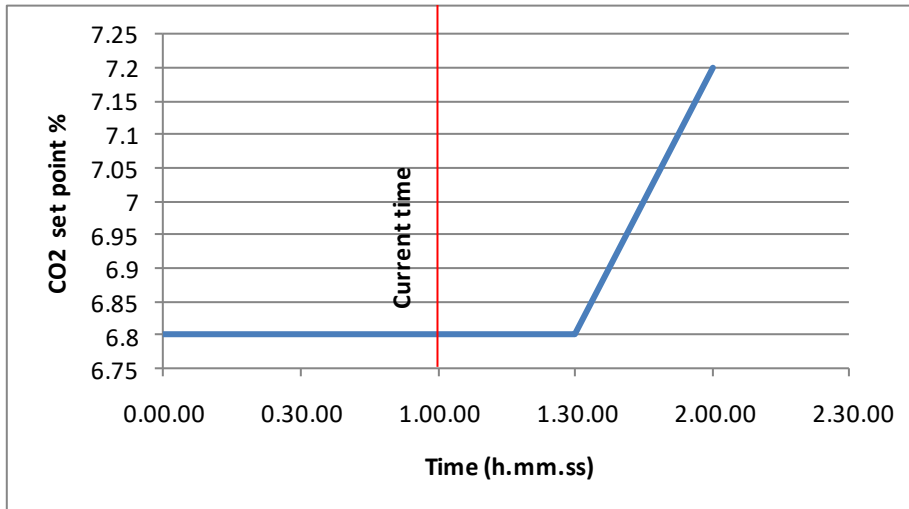
In case the current CO2 set point is 5%, current time is 1 hour, we want to change the CO2 set point to 3.5% when time is = 1 h 30 min. Set Point time profile is reported in the graph below.



1	Client	<code>readsetpoint 4 0 -1</code>	current CO2 set point is 5%
2	Server	<code>0_readsetpoint 4 0 -1_5.00</code>	
3	Client	<code>readtime -1</code>	current time is 3 600 000 millisec = 1 hour
4	Server	<code>0_readtime -1_3600000</code>	
5	Client	<code>impulse 4 5400 3.5</code>	change the CO2 set point to 3.5% when time is 5400 sec = 1 h 30 min
6	Server	<code>0_impulse 4 5400 3.5_1</code>	

Example 10

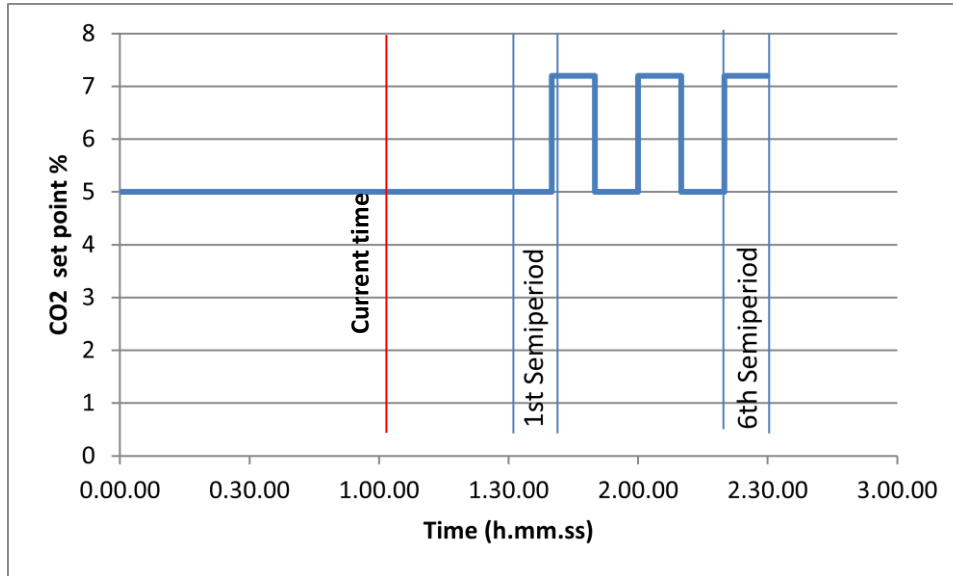
In case the current CO2 set point is 6.8%, current time is 1 hour, we want to change the CO2 set point to 7.2% with a ramp at constant growth rate, starting when time is = 1 h 30 min, and ending when time is 2h (when time is 2h the set point value will be 7.2). Set Point time profile is reported in the graph below.



1	Client	<code>readsetpoint 4 0 -1</code>	current CO2 set point is 6.8%
2	Server	<code>0_readsetpoint 4 0 -1_6.80</code>	
3	Client	<code>readtime -1</code>	current time is 3 600 000 millisec = 1 hour
4	Server	<code>0_readtime -1_3600000</code>	
5	Client	<code>ramp 4 5400 7.2 1800</code>	change the CO2 set point beginning when time is 5400 sec = 1 h 30 min and ending at 7200 sec = 2h. Length is 7200-5400=1800 sec.
6	Server	<code>0_ramp 4 5400 7.2 1800_1</code>	

Example 11

In case the current CO2 set point is 5%, current time is 1 hour, we want to start a square wave oscillation between 5% and 7.2% in the set point, the amplitude of each semiperiod will be 10 minutes, the wave will start when time is = 1 h 30 min, and will have 5 semiperiods.



As you can see from the graph, the first semiperiod will be at 5%, from 1h30min to 1h40min.

The 6th (last) semiperiod will be at 7.2% from 2h20 min to 2h30 min. After the cycle the set point will remain = 7.2% unless a further cycle is programmed.

1	Client	<code>readsetpoint 4 0 -1</code>	Current CO2 set point is 5.0%
2	Server	<code>0_readsetpoint 4 0 -1_5.00</code>	
3	Client	<code>readtime -1</code>	current time is 3 600 000 millisec = 1 hour
4	Server	<code>0_readtime -1_3600000</code>	
5	Client	<code>squarew 4 5400 7.2 600 6</code>	Start square wave oscillation between current set point and 7.2%, starting when time is 5400 sec = 1h 30 min. The amplitude of each wave is 600 sec. The number of semiperiods will be 6
6	Server	<code>0_squarew 4 5400 7.2 600 6_1</code>	



8. Change Log

Version 2.11.3

1. Fixed: no connection for product 16 if only COM Port of device 0 was set
2. Fixed: device 3 of product 17 is now called Humidifier

Version 2.11.2

3. Fixed: for products 16 and 17 minimum and maximum set point values are read from the device

Version 2.11.1

1. Fixed: listdev command for UNO-CONTROLLER product returned only device 0
2. Fixed: T Sensor device for UNO-CONTROLLER was always 0.0 in Chamber mode
3. Fixed: readmin and readmax when heating module is not plugged

Version 2.11.0

1. Fixed: UNO product connection worked even with UNO-CONTROLLER device
2. Removed "UNIT ID" concept: now there are devices without set point
3. Added additional Device IDs for H401-T-CONTROLLER and UNO-CONTROLLER

Version 2.10.1

1. Fixed bug setting simulation mode.

Version 2.10.0

1. Added H401-T-CONTROLLER and UNO-CONTROLLER products

Version 2.9.4

1. Fixed maximum allowed set point for UNO

Version 2.9.3

1. Fixed bug writing some parameters with bl-param-write.
2. Added: if a unit is not available, its value is "0.0", for example Objective Heater in UNO.

Version 2.9.2

1. Updated BL-param database.



2. Added BL-param list in this manual.
3. Fixed bug reading additional values with UNO. They can be read using a Device_ID greater than 0 (see Table 5).

Version 2.9.1

1. Improved connection speed.
2. Added compatibility with new UNO firmware. Supports backward compatibility.
3. Fixed bug in changing set point with new H301-HM-ACTIVE firmware. Supports backward compatibility.
4. Auto-connection is disabled by default (`autoconnect=n`).

Version 2.9.0

1. Added compatibility with USB board version 2 (mini-USB connector).

Version 2.8.0

1. Added H301-HM-ACTIVE product.
2. Improved `find speed` and `findstat` accuracy.

Version 2.7.0

1. Added H101 T Unit-BL product.

Version 2.6.1.3

1. Improved USB driver installation: now command line window is not showed.

Version 2.6.1

1. Fixed the number of devices for UNO Product (List Devices command).
2. Unit IDs for UNO Product are listed in the Read Status command.
3. Improved USB plugging connection errors checking.

Version 2.6.0

1. Fixed UNO COM port bug.
2. Fixed connection error due to USB cable plugging.
3. Added new command line option: `log`.
4. Added Read Name command.



Version 3.0.0

1. Fixed compatibility with Bold Line 2 devices.

Version 2.5.2

2. Fixed shutdown bug: OCS gracefully quits if the user shuts down Windows.

Version 2.5

1. Fixed `readval` and `writesetpoint` bug when the system decimal symbol is different than point.
2. Added new command line option: `autoconnect`.

Version 2.4

1. Added errors `e24` and `e25` to `checkcon` command (see Table 2).
2. Added codes `30` and `31` to `status` command (see Table 3).
3. Improved `find` speed.
4. Managed out of range error in `writesetpoint` command.
5. For Bold Line Products, `readsetpoint` returns a refreshed value even if the set point is changed using OKO-Touch.

Version 2.3

1. Fixed `checkcon` bug.
2. Fixed error codes for Bold Line Products.

Version 2.2

1. Generic errors (negative numbers) are now in the same format of the other error codes. Moreover, they are terminated by the carriage return character, like any other answer.
2. Read COM Port command (`comport`): improved confusing behaviour with multi-device products (es `CO2-O2 BL [0-20;1-95]`).
3. Read Minimum / Maximum / Default Set Point commands (`readmin`, `readmax`, `readdef`): fixed bug device “-1”, now the default is the first device.
4. OCS Setup installs Smartbox drivers too.
5. Added Check and Connect command (`checkcon`)
6. Improved `Device_ID` description
7. Improved debug window: added elapsed time and time with milliseconds.



Version 2.0

1. Managed Bold Line products.
2. Added BL parameters commands.
3. OCS stand-alone setup and executable with no activation code required.
4. Added Find Product and Read Find Status commands.
5. Changed reserved input parameter (Measure ID or Set Point ID) used in some commands (Read Value, Read Set Point, Write Set Point, ...) to Device ID. These changes are not compatible with the previous version of OCS, since the reserved input parameter was always set to "0".
6. Added a new optional parameter (Device_ID) to commands "Read Minimum Set Point", "Read Maximum Set Point" and "Read Default Set Point".
7. Removed license commands (Check License Status, Read License Information, Write License Information).
8. Fixed some examples.