



CoolMasterNet Universal Interface Adapter for HVAC Systems





Attention

1. During the HVAC system automatic operations the CoolMasterNet must be disconnected from the system. Examples of the automatic operations are:

- Test
- Reset
- Automatic charge

2. Do not change the original positions of the quadruple dip switch S inside CoolMasterNet when using it with VRV/VRF systems.

WEEE Directive & Product Disposal



At the end of its serviceable life, this product should not be treated as household or general waste. It should be handed over to the applicable collection point for the recycling of electrical and electronic equipment, or returned to the supplier for disposal.

Table of Contents

| | |
|---------------------------------------|-----------|
| 1 Document Revision History | 5 |
| 2 Acronyms | 6 |
| 3 Layout | 7 |
| 4 Connections | 8 |
| 4.1 HVAC Lines | 8 |
| 4.2 HomeAutomation/BMS Protocols..... | 8 |
| 4.3 RS232 Port | 9 |
| 4.4 Ethernet | 9 |
| ASCII I/F IP Server | 10 |
| ModBus IP | 10 |
| 4.5 RS485 | 10 |
| ModBus RTU | 10 |
| CoolGate 4 | 10 |
| CoolGate 5 | 11 |
| 4.6 GPIO | 12 |
| 4.7 USB | 13 |
| 4.8 Power | 13 |
| 5 DIP Switches | 14 |
| 6 ASCII I/F | 16 |
| 6.1 General Protocol Definitions..... | 16 |
| Messaging | 16 |
| Exit Code | 16 |
| UID | 17 |
| 6.2 Commands Reference..... | 18 |
| Configuration Commands | 18 |
| set | 18 |
| line | 19 |
| ifconfig | 21 |
| boot | 22 |
| sddp | 22 |
| knx | 23 |
| props | 25 |
| link | 26 |
| plug | 26 |
| hdl | 27 |
| simul | 28 |
| gpio | 28 |

| | |
|---|-----------|
| info | 29 |
| modbus | 30 |
| HVAC Status and Control Commands | 31 |
| on | 32 |
| allon | 32 |
| off | 32 |
| alloff | 33 |
| cool | 33 |
| heat | 33 |
| fan | 34 |
| dry | 34 |
| auto | 34 |
| haux | 35 |
| temp | 35 |
| fspeed | 36 |
| swing | 37 |
| filt | 37 |
| stat | 37 |
| ls | 38 |
| ls2 | 39 |
| query | 39 |
| feed | 41 |
| wh | 41 |
| main | 42 |
| vam | 43 |
| lock | 44 |
| group | 45 |
| CA | 45 |
| va | 46 |



1 Document Revision History

0.7

- Added Haier

0.6

- Added Chigo

0.5

- Added Midea, Kentatsu, Trane, Fujitsu

0.4 - Initial

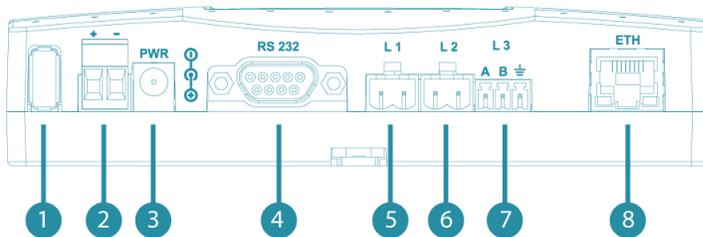
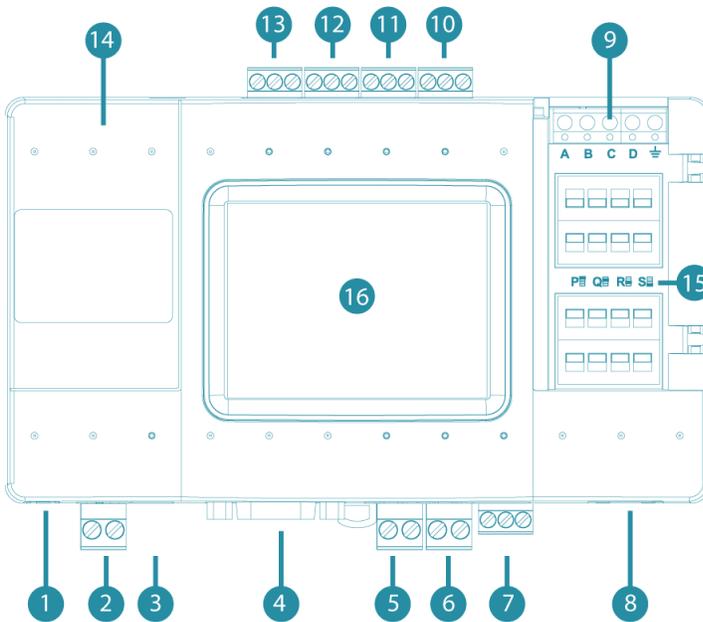
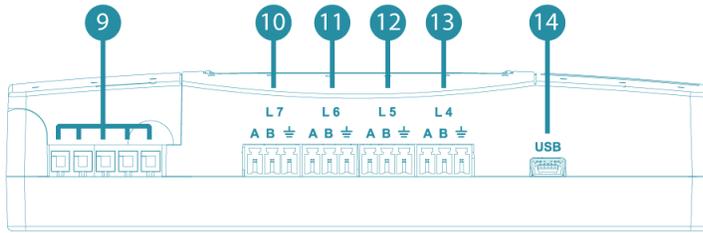
0.3 - Preliminary



2 Acronyms

| | |
|------|--|
| DTE | Data Terminal Equipment |
| ETH | Ethernet |
| GPIO | General Purpose Input/Output |
| HVAC | Heating Ventilation and Air Conditioning |
| TBD | To Be Defined |

3 Layout



1. USB Host, HVAC Line L8
2. Power
3. Power Plug
4. RS232 Port
5. HVAC Line L1
6. HVAC Line L2
7. HVAC Line L3 or RS485 Port
8. Ethernet Port
9. GPIO
10. HVAC Line L7
11. HVAC Line L6
12. HVAC Line L5
13. HVAC Line L4
14. USB Device Port
15. DIP Switches P,Q, R,S
16. LCD with Touch Screen

4 Connections

4.1 HVAC Lines

CoolMasterNet supports eight HVAC Lines labeled as **L1** ... **L8** (see [Layout](#)), intended for connection to various HVAC systems. All eight HVAC Lines can be used simultaneously (except **L1** with **L5** and **L2** with **L6**) and independently according to CoolMasterNet configuration. Most of the HVAC Lines capable to support a number of HVAC Types as specified in table below, but one at time, according to configuration of the specific Line.

| HVAC Manufacturer | Acronym | L1 | L2 | L3 | L4 | L5 | L6 | L7 | L8 |
|-----------------------------|---------|----|----|----|----|----|----|----|----|
| Daikin | DK | ✓ | ✓ | | | | | | |
| Mitsubishi Electric | ME | ✓ | ✓ | | | | | | |
| Sanyo | SA | ✓ | ✓ | | | | | | |
| Toshiba | TO | ✓ | ✓ | | | | | | |
| Panasonic | PN | ✓ | ✓ | | | | | | |
| Hitachi | HT | ✓ | ✓ | | | | | | |
| LG | LG | | | ✓ | ✓ | ✓ | ✓ | ✓ | |
| Mitsubishi Heavy Industries | MH | | | ✓ | ✓ | ✓ | ✓ | ✓ | |
| Gree | GR | | | ✓ | ✓ | ✓ | ✓ | ✓ | |
| Midea | MD | | | ✓ | ✓ | ✓ | ✓ | ✓ | |
| Kentatsu | KT | | | ✓ | ✓ | ✓ | ✓ | ✓ | |
| Trane | TR | | | ✓ | ✓ | ✓ | ✓ | ✓ | |
| Chigo | CG | | | ✓ | ✓ | ✓ | ✓ | ✓ | |
| Fujitsu | FJ | | | | | | | | ✓ |
| Samsung | SM | | | ✓ | ✓ | ✓ | ✓ | ✓ | |
| Tadiran Inverter | TI | | | ✓ | ✓ | ✓ | ✓ | ✓ | |
| Meitav | MT | | | ✓ | ✓ | ✓ | ✓ | ✓ | |
| Haier | HA | ✓ | ✓ | | | | | | |

Notes:

- [DIP Switches Q](#) and [R](#) are used to adjust lines **L1** and **L2** internal parameters to meet specific HVAC Type requirements.
- Lines **L1** and **L5** share the same internal resources of CoolMasterNet and can not be used simultaneously.
- Lines **L2** and **L6** share the same internal resources of CoolMasterNet and can not be used simultaneously. [DIP Switch P3](#) defines which line is enabled.
- Lines **L6** and **L7** can be linked into a single line **L7** with polarity auto-detection, by [DIP Switches P1](#) and [P2](#). Line **L2** can be enabled in this case.
- Line **L3** is by default used for ModBus RTU RS485 communication.

4.2 HomeAutomation/BMS Protocols

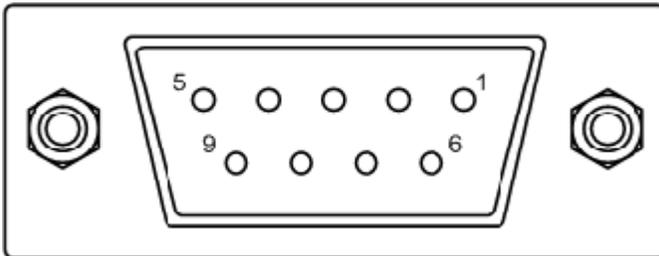
CoolMasterNet supports several Home Automation/BMS protocols.

| Home Automation/BMS Protocol | Acronym | Line(s)/Port |
|------------------------------|---------|----------------|
| KNX | KNX | L3 |
| PlugBus (CoolHub) | CH | L3,L4,L5,L6,L7 |
| HDL buspro 4-wire | HDL | L3,L4,L5,L6,L7 |
| HDL buspro IP | | Ethernet |

| | | |
|---------------------------------------|------------|------------------------|
| ModBus RTU CoolGate 4 | CG4 | L3,L4,L5,L6,L7 |
| ModBus RTU CoolGate 5 | CG5 | L3,L4,L5,L6,L7 |
| ModBus IP | | Ethernet |
| SDDP | | Ethernet |
| ASCII I/F | | RS232, Ethernet |
| CoolRemote | | Ethernet |

4.3 RS232 Port

RS232 Interface on CoolMasterNet is available from the RS232 DB9 female connector. Below is a DB9 connector front view and signals table.



| DB9 Pin | Signal Level | Description |
|-----------|--------------|--------------------------------------|
| 2 | ±12V | TxD (Data from CoolMasterNet to DTE) |
| 3 | ±12V | RxD (Data from DTE to CoolMasterNet) |
| 5 | GND | Ground |
| 1,4,6,7,8 | | Not Connected |

Gender and pinouts of the RS232 cable supplied with CoolMasterNet are suitable for connection to PC RS232 port directly or via standard RS232 to USB adapter. Maximal length of the RS232 Cable should not exceed 25m. By default RS232 Interface is dedicated for [ASCII I/F](#). The default CoolMasterNet RS232 Port settings are:

| | |
|----------------|------|
| Baud Rate | 9600 |
| Data Bits | 8 |
| Parity Control | None |
| Stop Bits | 1 |
| Flow Control | None |

4.4 Ethernet

CoolMasterNet incorporates an IEEE 802.3 compatible 10/100 Mb/s Ethernet port supported via RJ45 connector. Below are main port features.

| Parameter | Value | Notes |
|------------------------------|---------------------|-------------------------|
| Max Ethernet Cable Length | 137m | CAT5 twisted pair cable |
| Bit Rate | 10/100 Mb/s | |
| Supported Ethernet Protocols | 10BASE-T/100BASE-TX | |
| Protocol Auto-Negotiation | Enabled | Against Link Partner |

RJ45 connector comprises Link and Activity indication LEDs providing following statuses.

| LED | Color | Function |
|--------------|--------|-------------------------------------|
| Link Led | Green | ON for good link OFF for no link |
| Activity Led | Orange | BLINK for Tx/Rx Activity |

Ethernet port is used by a number of protocol modules available in CoolMasterNet

- ASCII I/F (via [ASCII I/F IP Server](#))
- [ModBus IP](#)
- HDL buspro IP
- SDDP
- CoolRemote

Network setting of the CoolMasterNet are controlled with [ifconfig](#) command.

4.4.1 ASCII I/F IP Server

ASCII I/F IP Server referenced as **aserver** is a classic row TCP/IP socket server. Aserver has following default characteristics:

| | |
|--|---------|
| Maximal number of simultaneous connections | 4 |
| Default TCP/IP port | 10102 |
| Prompt character > | enabled |

Aserver can be configured with [set](#) command.

4.4.2 ModBus IP

4.5 RS485

By default Line **L3** is used as an RS485 Interface line for **DTE** connection. CoolMasterNet supports the following RS485 based protocols:

- [ModBus RTU](#) (Slave mode)

4.5.1 ModBus RTU

CoolMasterNet can be used as a ModBus RTU slave device working in accordance with Modbus-IDA.ORG "MODBUS over serial line specification and implementation guide". RS485 default frame format is

| | |
|----------------|------|
| Baud Rate | 9600 |
| Data Bits | 8 |
| Parity Control | None |
| Stop Bits | 1 |
| Flow Control | None |

4.5.2 CoolGate 4

CoolGate 4 is based on ModBus RTU protocol. Its address map is similar to CoolGate device, previously manufactured by CoolAutomation (see CoolGate PRM for more information). CoolGate 4 protocol can be used for backward compatibility.

4.5.3 CoolGate 5

CoolMasterNet can be used as a ModBus RTU slave device working in accordance with Modbus-IDA.ORG "MODBUS over serial line specification and implementation guide". RS485 default frame format is (can be changed with [line baud](#) command)

| | |
|----------------|------|
| Baud Rate | 9600 |
| Data Bits | 8 |
| Parity Control | None |
| Stop Bits | 1 |
| Flow Control | None |

CoolGate 5 Indoors Address Map:

In order to simplify translation of the [UID](#) (like L1.100) into ModBus object address, CoolMasterNet uses [VA](#), associated with UID. Hex and Dec numbers printed after **va** command are Base Address used to access ModBus objects related to UID-VA association. Base Address can be calculated as $BaseAddress = VA * 16 + 1$. For example Indoor Unit L1.100 has associated [VA = 007](#), thus Base Address = $7 * 16 + 1 = 113$ (0x71).

Once the Base Address is determined, access to the specific Indoor Unit parameter(s) is done via Modbus objects listed below.

| Base Address | Discrete Inputs | Coils | Holding Registers | Input Registers | |
|--------------|-----------------|-----------------|-------------------|---------------------------|--|
| +0 | Reserved | On/Off | Operation Mode | UID | |
| +1 | | Filter Sign | Fan Speed | Room Temperature x10 | |
| +2 | | Reserved | Reserved | Set Temperature x10 | Failure Code String first two characters |
| +3 | | | | On/Off | Failure Code String last two characters |
| +4 | | | | Filter Sign | Reserved |
| +5 | | | | Swing | |
| +6 | | | | Room Temperature x10 (RO) | |
| +7 | | | | Failure Code (RO) | |
| +8 | | | | Lock Bits | |
| +9 | | | | Digital Input 1 | |
| +10 | | Digital Input 2 | Digital Output 2 | | |
| +11 | | Digital Input 3 | Digital Output 3 | | |
| +12 | | Digital Input 4 | Digital Output 4 | | |
| +13 | | Digital Input 5 | Digital Output 5 | | |
| +14 | | Digital Input 6 | Digital Output 6 | | |
| +15 | Reserved | Reserved | Reserved | Reserved | |

[UID](#) Encoding

| MSB | | LSB |
|-------------|------------|-----------|
| Bits 15..12 | Bits 11..8 | Bits 7..0 |
| Ln | X | YY |

Corresponds to Ln.XYY (like for example L1.100)

Operation Mode Encoding

| | | | |
|--------------|----------------|------------------------|-----------------|
| 0 - Cool | 1 - Heat | 2 - Auto | 3 - Dry |
| 4 - Haux | 5 - Fan | 6 - HH | |
| 8 - VAM Auto | 9 - VAM Bypass | 10 - VAM Heat Exchange | 11 - VAM Normal |

Fan Speed Encoding

| | | | |
|----------------------|-----------------------|----------|--------------------|
| 0 - Low | 1 - Medium | 2 - High | 3 - Auto |
| 4 - Top | 5 - Very Low (quiet) | | 7 - VAM Super High |
| 8 - VAM Low Fresh Up | 9 - VAM High Fresh Up | | |

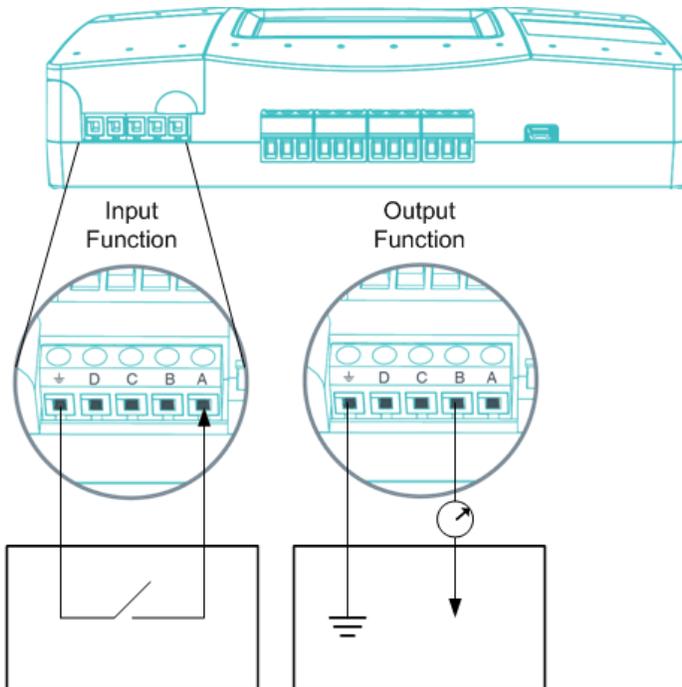
Swing Encoding

| | | | |
|----------------|------------|------------|------------|
| 0 - Vertical | 1 - 30 deg | 2 - 45 deg | 3 - 60 deg |
| 4 - Horizontal | 5 - Auto | 6 - OFF | |

There is a possibility to read/write holding registers of several indoors by one request. If the read request started from one of the readable registers (not reserved) the values of reserved registers will be 0xFFFF in the response. If the read request started from reserved register the response will be illegal data address. In a similar way it works with the write request. If the write request started from one of the writable registers (not reserved) the values that were intended for reserved registers will be ignored. If the write request started from one of the reserved registers the response will be illegal data address.

4.6 GPIO

CoolMasterNet supports four GPIO marked as **A B C D** (see [Layout](#)). GPIO can be in Input or Output mode depending on it's functionality



| Parameter | GPIO A | GPIO B | GPIO C | GPIO D |
|---------------------------------------|--------|--------|--------|--------|
| Input Low Voltage (V _{IL}) | <1V | <1V | <1V | ADC |
| Input High Voltage (V _{IH}) | >2.4V | >2.4V | >2.4V | ADC |

| | | | | |
|---|------|------|------|------|
| Output High Voltage (V_{OUT HI}) | 3.3V | 3.3V | 3.3V | 3.3V |
| Max Output Current (I_{OUT}) | 20mA | 20mA | 20mA | 20mA |
| Pull Up | 5.6K | 5.6K | 5.6K | N.A. |

Control of the GPIO functionality is done with [gpio](#) command.

4.7 USB

CoolMasterNet incorporate USB Device and USB Host ports. USB Device port is used for maintenance operations.

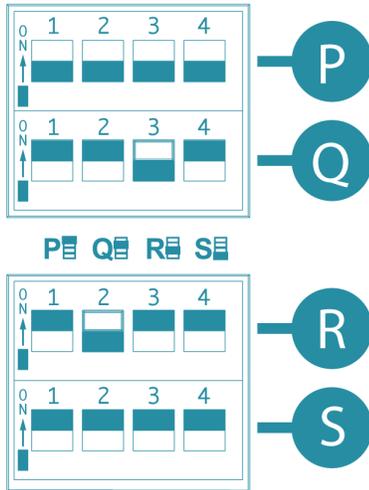
4.8 Power

CoolMasterNet can be powered from different power sources:

- AC/DC adapter supplied with CoolMasterNet
- 12-24V DC from HVAC or other equipment
- USB Device port

5 DIP Switches

DIP Switches are located behind the small access door at the upper right side of the CoolMasterNet.



DIP Switch P

| Switch | ON | OFF |
|--------|---|-------------------------|
| P1,P2 | Link L6,L7 and enable polarity auto-detection on L7 | Separate L6,L7 |
| P3 | L6 Enabled, L2 Disabled | L2 Enabled, L6 Disabled |
| P4 | Production Mode | Normal Operation Mode |

Notes:

- Switches P1 and P2 should be in the same position. If they are both in ON position, HVAC Lines L6 and L7 are linked into one line with option to auto-detect line polarity. Otherwise, if both P1 and P2 are in OFF position HVAC Lines L6 and L7 are separate independent lines.
- If L6 and L7 are linked into L7, enabling of the L2 with P3 will not influence on L7 operation.
- Switch P4 must be in OFF position for normal operation of CoolMasterNet.

DIP Switches Q,R

DIP Switched Q and R are used to adjust HVAC Lines L1 and L2 internal parameters to meet specific HVAC type requirements.

| HVAC Type | DIP Switch Q - HVAC Line L1 | | | |
|-----------|-----------------------------|-----|-----|-----|
| | Q1 | Q2 | Q3 | Q4 |
| DK | ON | OFF | ON | OFF |
| ME | OFF | OFF | OFF | OFF |
| TO | OFF | ON | OFF | ON |
| SA | OFF | ON | OFF | ON |
| PN | OFF | ON | OFF | ON |
| HT | OFF | ON | OFF | ON |
| HA | OFF | ON | OFF | ON |

| HVAC Type | DIP Switch R - HVAC Line L2 | | | |
|-----------|-----------------------------|-----|-----|-----|
| | R1 | R2 | R3 | R4 |
| DK | ON | OFF | ON | OFF |
| ME | OFF | OFF | OFF | OFF |

| | | | | |
|-----------|-----|----|-----|----|
| TO | OFF | ON | OFF | ON |
| SA | OFF | ON | OFF | ON |
| PN | OFF | ON | OFF | ON |
| HT | OFF | ON | OFF | ON |
| HA | OFF | ON | OFF | ON |

Notes:

- If all four Dip Switches R1, R2, R3, R4 are in ON position during CoolMasterNet power reset, CoolMasterNet will be forced to enter BOOT Mode.

Dip Switch S

| Switch | ON | OFF |
|--------|----------------------------------|-----------------------------------|
| S1,S2 | Enable DC Output on HVAC Line L1 | Disable DC Output on HVAC Line L1 |
| S3,S4 | Enable DC Output on HVAC Line L2 | Disable DC Output on HVAC Line L2 |

Notes:

- Switches S1 and S2 should be in the same position.
- Switches S3 and S4 should be in the same position.
- DC Output on HVAC Line L1 or L2 is required only in case when the line is configured as DK or ME and non VRF equipment is connected to this line (via KRP, MAC or similar adapter). **And only if no other DC source is present on this line.**

6 ASCII I/F

CoolMasterNet provides a simple and comprehensive ASCII I/F Protocol, based on text (ASCII) strings, representing verbal commands and responses. ASCII I/F implemented in CoolMasterNet is fully backward compatible with previous versions of CoolAutomation products, but has a number of significant extensions and improvements mainly aimed to support additional CoolMasterNet functionality. ASCII I/F can be utilized via RS232 interface (see [RS232 Port](#)) or TCP/IP [Server](#).

6.1 General Protocol Definitions

6.1.1 Messaging

Communication between DTE and CoolLinkNet via ASCII I/F is based on text (ASCII) strings. Communication example is shown below

| | | |
|-------------------------------------|------------|--------------------|
| >ls L2 | ←command | DTE to CoolLinkNet |
| L2.102 OFF 20C 27C Auto Cool OK - 0 | ←response | |
| L2.103 OFF 20C 24C Low Auto OK - 0 | ←exit code | CoolLinkNet to DTE |
| OK | ←prompt | |
| > | | |

Command string sent to CoolLinkNet must be terminated with <CR> (carriage return 0x0D) <LF> (line feed 0x0A) sequence or a single <CR> character. Strings from CoolLinkNet (except prompt character) are always terminated with <CR> <LF>. Commands are case sensitive and should not contain leading or trailing spaces. The only separator between command name and command parameter(s) is space (0x20) character. Configuration parameter **echo** (see [set](#) command), defines if characters sent to CoolLinkNet via [RS232 interface](#) are echoed back or not. If **echo** is not zero - characters are echoed. In case of RS232 interface, prompt character '>' is unconditionally sent by CoolLinkNet. In case of [ASCII Server](#) prompt sending is configurable.

6.1.2 Exit Code

CoolMasterNet provides Exit Code in verbose or numeric form. Numeric form format is

ERROR : N

where N is a number in range 0...999. If verbose format is not specified in table below it means error has only numeric format.

| Numeric | Verbose | Description |
|---------|--------------------|--|
| 0 | OK | Command executed successfully |
| 1 | No UID | UID not found |
| 2 | Not Strict UID | UID must be precise |
| 3 | Bad Format | Command format is wrong |
| 4 | Failed | Command execution failed |
| 5 | Line Unused | Line is unused |
| 6 | Unknown Command | Command is unknown |
| 7 | Bad HVAC Line | Line number is wrong |
| 8 | Bad Function | Wrong function |
| 9 | Bad Line Type | Wrong line type definition |
| 10 | Bad Parameter | Command parameter is wrong |
| 11 | OK, Boot Required! | Command execution will be effective after reboot |
| 12 | Bad GPIO | Wrong GPIO |
| 13 | SDDP Disabled | SDDP module is disabled, enable it to proceed |

| | | |
|-----|-------------------------------------|---|
| 14 | Virtual Address In Use | Virtual address already in use |
| 15 | Bad Property | Wrong property |
| 16 | Number of lines exceeded | Can't define more line types |
| 17 | Warning! Dip Switch State Incorrect | Dip switch state is incorrect for defined line type |
| 18 | SDDP Not Initialized | SDDP is enabled, but Ethernet link is still down |
| 80 | ModBus Error:80 | No response from the addressee |
| 81 | ModBus Error:81 | Big timeout between bytes in received message |
| 82 | ModBus Error:82 | Small timeout between bytes in received message |
| 83 | ModBus Error:83 | Received message with internal timeout |
| 84 | ModBus Error:84 | Received message is too big |
| 85 | ModBus Error:85 | CRC error in received message |
| 86 | ModBus Error:86 | ModBus exception in response |
| 100 | Collision | Collision in sent command on HVAC line |
| 101 | Unsupported Feature | Unsupported command or command's parameter for this HVAC line |
| 102 | Incorrect Indoor Type | Chosen Indoor Unit doesn't support this command or command's parameter |
| 103 | No ACK From Indoor | Indoor Unit didn't acknowledged sent command |
| 104 | Time Out on Receive | No response from Indoor Unit |
| 105 | CS Error In Received Message | Check Sum error in received message |
| 106 | Line Init In Progress... | Can't show lines status due to initialization process |
| 107 | Line Error | Some error on the HVAC line |
| 108 | Feed Disabled | Indoor Unit can't receive measured ambient temperature due to its switch position |
| 150 | HDL Not Initialized | HDL line was not defined or ethernet cable is unplugged |
| 151 | HDL DB Overflow | HDL Data Base is full, can't add new configuration |
| 152 | HDL Eth Disabled | HDL over ethernet disabled |
| 200 | UID Not Found | Specified Indoor Unit not found in Data Base |
| 201 | Strict UID Not Found | Specified Indoor Unit by uid strict not found in Data Base |
| 202 | Indoor Removed | Indoor Unit removed from Data Base |
| 203 | DB Overflow | Indoor Units Data Base is full, can't add new one |
| 204 | Group DB Overflow | Group Data Base is full, can't add new group |
| 205 | VA DB Overflow | Virtual address Data Base is full, can't associate Indoor Unit with new virtual address |
| 206 | FDB5 Overflow | Properties Data Base is full, can't add new property |
| 250 | Link DB Overflow | Link Data Base is full, can't link new CoolPlug device with Indoor Unit |
| 251 | No CoolHub Line | CoolHub line not defined, define it to proceed |
| 252 | Auto Visibility Failed | During link creation there was an error on adding visibility props |
| 253 | Link already exists | CoolPlug device already linked, delete previous link before creating the new one |
| 307 | KNX DB Overflow | KNX Data Base is full, can't add new group |
| 309 | KNX Not Connected | No connection with KNX chip |
| 310 | KNX Line Not Started | KNX line not defined |

6.1.3 UID

UID is used to identify Indoor Unit or a set of Indoor Units. UID has the following format:

| Line | Dot | Indoor Number | |
|-----------|----------|---------------|-----------|
| Ln | . | X | YY |

- Line is a CoolLinkNet HVAC Line number in range L1..L4. **L*** means "any line". In some cases to provide backward compatibility Line can be omitted from the UID.
- Dot is a separator between Line and Indoor Number. If Line is omitted or has a **L*** form, dot must also be omitted.
- Indoor Number is an Indoor Unit number in HVAC system. Indoor Number can be '*' that means "any"

Examples:

| | |
|---------------|---|
| L1.102 | Indoor Unit 102 on line L1 |
| L2.003 | Indoor Unit 003 on line L2 |
| L*100 | Set of Indoor Units 100 on all lines |
| L3.1* | Set of Indoor Units 1xx on line L3 (L3.100, L3.101, ...) |
| L4 | All Indoors on line L4 |
| L* | All Indoors on all lines |
| 203 | Similar to L*203 (for backward compatibility only) |

UID_STRICT

In some cases only specific Indoor Unit should be referenced by UID. In this case it is required to use UID_STRICT in format **Ln.XYY** where '*' usage is prohibited.

6.2 Commands Reference

Synopsis and description of the commands listed below have the following notation:

- Parameters or parameters group in angle brackets < > are mandatory.
- Parameters or parameters group in square brackets [] are optional and can be omitted.
- The curly braces () are used to denote group inside braces
- The | character between parameters inside brackets means OR.

6.2.1 Configuration Commands

[set](#)
[line](#)
[ifconfig](#)
[boot](#)
[sddp](#)
[knx](#)
[props](#)
[link](#)
[plug](#)
[hdl](#)

[simul](#)
[gpio](#)
[info](#)
[modbus](#)

set

SYNOPSIS

```
set [<SETTING> <VALUE>]
set defaults
```

DESCRIPTION

Query or change CoolMasterNet setting(s). Without parameters **set** command will list all supported settings

and their values. To change setting use format with **<SETTING>** and **<VALUE>**. Some settings are read only (RO) and can not be changed.

set defaults will load default values to all settings

| <SETTING> | Mode | Printed as | Value* | Description |
|------------------------|-------------|-------------------|--------------------------|--|
| S/N | RO | S/N | | CoolMasterNet Serial Number |
| version | RO | version | X.Y.Z | CoolMasterNet Firmware Version |
| app | R/W | application | string | CoolMasterNet Application |
| baud | R/W | baud rate | 1200...115200 | RS232 Interface baud rate. Default: 9600 |
| echo | R/W | echo | 0 or 1 | RS232 Interface echo control. 0 - disabled, 1 - enabled |
| verbose | R/W | verbose | 0 or 1 | Exit Code format. 0 - numeric, 1 - verbose |
| aserver port | R/W | aserver port | integer | aserver TCP port. Default: 10102 |
| aserver prompt | R/W | aserver prompt | 0 or 1 | aserver prompt control |
| deg | R/W | deg C/F | C/c or F/f | Temperature scale Celsius or Fahrenheit |
| melody | R?W | melody | string | PowerUp melody. The value is printed as melody name, to change the melody its number should be typed. set melody will list supported melodies and their numbers. |
| filter visi | R/W | filter | visi_0 or visi_1 or - | Indoor Units visibility regarding props command. - props visibility applies only to LCD, 1 props visibility applies to LCD and status/control commands, 0 - props visibility logic is inverted and it applies to LCD and status/control commands |
| HVAC lines | RO | HVAC lines | 0...8 | Allowed number of HVAC lines. Can be changed through encoded activation string. |

* - Bold values are defaults.

EXAMPLE

Disable echo

```
>set echo 0
```

OK

Change aserver TCP port

```
>set aserver port 12345
```

OK

Load defaults

```
>set defaults
```

OK

Apply props visibility to status/control commands

```
>set filter visi 1
```

OK, Boot Required!

Set RS232 Interface to 19200, 8 data bits, even parity, 1 stop bit

```
>set baud 19200 8E1
```

OK

Set temperature scale to Fahrenheit

```
>set deg f
```

OK

Set PowerUp melody to 3

```
>set melody 3
```

OK

line

SYNOPSIS

```
line
line <PROPERTY> <Ln> <VAL>
```

DESCRIPTION

Query or change [HVAC Line\(s\)](#) status and configuration. In format without parameters **line** command will provide information about current Line statuses. Second format is used to change write enabled properties.

| <PROPERTY> | Mode | Value | Description | Notes |
|----------------|------|-----------------|---|--|
| master | R/W | 0 or 1 | CoolMasterNet acts as a Master device | DK, LG, SA, TO, PN only |
| simul | R/W | integer | Simulate given number of Indoor Units. To disable simulation use "0". Simulation is persistent (not disabled after power reset) By default the simulation model is: with failures, no ventilation units, no input/output units. To enable/disable special conditions use the following flags: +/-flr, +/-vam, +/-io (see example) | Input/Output units for ME only Ventilation units for DK, ME, LG only |
| myid | R/W | string | CoolMasterNet Own address on given HVAC Line. For ModBus RTU Line myID is a "Slave Device Address" | |
| baud | R/W | | Configure UART parameters for given Line. Value format is: <BAUD> <8 9><N E O><1 2> For example: 9600 8N1 | Where applicable |
| Tx | RO | integer counter | Transmitted messages | Printed as Last/Total Last means since previous print |
| Rx | RO | | Received messages | |
| TO | RO | | Timeout errors | |
| CS | RO | | Checksum errors | |
| Col | RO | | Collision errors | |
| NAK | RO | | NAK errors | |
| haux | R/W | heat or haux | Heat mode treated as Haux mode. This is only for situation when HomeAutomation controller doesn't have Haux support, but need to set Haux mode on ThermoPad(s) connected to CH line. | For CoolHub (CH) line only |
| type | R/W | string | Define line type by acronym . Number of HVAC lines cannot exceed the value of HVAC lines in set command. HomeAutomation/BMS lines don't count as HVAC Lines. To free a line use Unused instead of acronym. | See the full list of HVAC and HomeAutomation/BMS lines supported |
| scan | R/W | 0 or 1 | Implicitly detect Indoor Units | ME, HT, SA, TO, PN only |
| DCOUT | R/W | - or + | Control DC output on L1,L2 | DK,ME only |
| slink | R/W | O/o or N/n | Configure Mitsubishi Heavy SuperLink mode: O - old, N - new | MH only |
| | | | Configure Samsung model type: O - DVM, N - DVM-S | SM only |
| [A B] or [B A] | RO | | Current polarity | Line L7 only |

EXAMPLE

```
>line
```

```
L1: SA U00/G02 myID:E5
   Tx:732/732 Rx:47227/47227 TO:0/0 CS:0/0 Col:0/0 NAK:0/0
L2: DK Master U00/G07 myID:0B
   Tx:4579/4579 Rx:50633/50633 TO:3/3 CS:0/0 Col:10/10 NAK:0/0
L3: Unused
```

```

Tx:0/0 Rx:0/0 TO:0/0 CS:0/0 Col:0/0 NAK:0/0
L4: Unused
Tx:0/0 Rx:0/0 TO:0/0 CS:0/0 Col:0/0 NAK:0/0
L5: Unused
Tx:0/0 Rx:0/0 TO:0/0 CS:0/0 Col:0/0 NAK:0/0
L6: Unused
Tx:0/0 Rx:0/0 TO:0/0 CS:0/0 Col:0/0 NAK:0/0
L7: MH SLink:N U00/G02 myID:C900 [A|B]
Tx:20072/20072 Rx:26989/26989 TO:25/25 CS:0/0 Col:1/1 NAK:0/0
OK

```

Become Slave on Line L4

```
>line master L4 0
```

OK, Boot Required!

Set Line L3 baud rate to 19200, 8 data bits, even parity, 1 stop bit

```
>line baud L3 19200 8E1
```

OK, Boot Required!

Simulate 5 Units on line L2 without failures ventilation units and with input/output units

```
>line simul L2 5 -flr-vam+io
```

OK, Boot Required!

Heat mode will be treated as haux for ThermoPads connected to CH line

```
>line haux L3 heat
```

OK, Boot Required!

Free line L2 by defining it as Unused

```
>line type L2 Unused
```

OK, Boot Required!

Define Toshiba type on line L2

```
>line type L2 TO
```

OK, Boot Required!

Define HDL Bus Pro on line L3

```
>line type L3 HDL
```

OK, Boot Required!

ifconfig

SYNOPSIS

```

ifconfig
ifconfig <PROPERTY> <VALUE>
ifconfig          enable|disable

```

DESCRIPTION

Query or configure Ethernet network settings. Without parameters, **ifconfig** command will list current configuration. To change configuration use format with **<PROPERTY>** and **<VALUE>**. Parameter **IP** can be set to **DHCP** (DHCP client) or fixed IP number. In case of DHCP - Netmask and Gateway values are provided by DHCP server. By default CoolMasterNet is configured for DHCP client operation. CoolMasterNet Ethernet module can be enabled or disabled with corresponding command.

| <PROPERTY> | Mode | Value | Description | Notes |
|------------|------|------------|---|-------|
| MAC | RO | string | MAC (Media Access Control) address of CoolMasterNet | |
| Link | RO | Up or Down | Indicate whether ethernet link is active | |
| IP | R/W | string | Set fixed (static) IP address or set it to DHCP | |

| | | | | |
|---------|-----|--------|--|--|
| Netmask | R/W | string | Set Netmask address in case IP is static | |
| Gateway | R/W | string | Set Gateway address in case IP is static | |
| DNS1 | R/W | string | Set preferred DNS address | |
| DNS2 | R/W | string | Set alternate DNS address | |

EXAMPLE

Query

```

>ifconfig
MAC      : 28:3B:96:00:02:0F
Link     : Up
IP       : 192.168.16.108 (DHCP)
Netmask: 255.255.255.0
Gateway: 192.168.16.254
DNS1    : 8.8.8.8
DNS2    : 8.8.4.4
OK
Configure fixed IP, Gateway and Netmask
>ifconfig IP 192.168.1.102
OK, Boot Required!
>ifconfig Gateway 192.168.1.0
OK, Boot Required!
>ifconfig Netmask 255.255.0.0
OK, Boot Required!
Configure DHCP client operation
>ifconfig IP DHCP
OK, Boot Required!
Disable Ethernet
>ifconfig disable
OK, Boot Required!
Change DNS preferred address
>ifconfig DN1 208.67.222.222
OK, Boot Required!

```

boot**SYNOPSIS**

```

boot
boot [N]

```

DESCRIPTION

- <N> omitted - Enter Boot Mode
- <N> = 2 - Reset CoolMasterNet

sddp**SYNOPSIS**

```

sddp
sddp <enable|disable|identify|offline|alive>

```

DESCRIPTION

Configure Control4® Simple Device Discovery Protocol (SDDP) module. CoolMasterNet is compliant with Control4® SDDP v1.0 specification and can act as SDDP enabled device.

- Without parameters, if SDDP module was enabled and Ethernet cable connected, **sddp** command will list supported protocol version and packet counters.
- **sddp enable** - persistently enables SDDP module (effective after reset). Default state.
- **sddp disable** - persistently disables SDDP module (effective after reset)
- **sddp identify** - sends IDENTIFY message to Control4® Composer tool to add CoolMasterNet to the current project
- **sddp offline** - signals that CoolMasterNet is going offline
- **sddp alive** - signals that CoolMasterNet is online

EXAMPLE

Enable SDDP module

```
>sddp enable
```

```
OK, Boot Required!
```

Send IDENTIFY message

```
>sddp identify
```

```
OK
```

knx

SYNOPSIS

```
knx
knx  addr  <area/line/device>
knx  ram  <R>
knx  funcs
knx  group
knx  group <GA>[ <func> <direction> <UID> ]
knx  group dellall
knx  group -<G>
```

DESCRIPTION

Query or change KNX status and configuration. In format without parameters **knx** command will provide information about current connection status and number of used groups.

- **knx addr <area/line/device>** - Set KNX physical address in the form of area/line/device
- **knx ram <R>** - Create group Data Base for R groups. The operation will take effect after reboot. Use this operation only if default group Data Base is too small. The current Data Base usage can be seen in **knx** command.
- **knx funcs** - Print all available group functions.
- **knx group** - List existing KNX groups (linkage status)
- **knx group <GA>** - List existing KNX groups (linkage status) for the given group address
- **knx group <GA> <func> <direction> <UID>** - Create new KNX group, i.e. link KNX group address <GA> with CoolMasterNet function and UID. <GA> can be in the form of Main/Mid/Sub or Main/Sub, direction: < - CoolMasterNet input, > - CoolMasterNet output.
- **knx group dellall** - Delete all KNX groups
- **knx group -<G>** - Delete KNX group number <G>. Group numbers displayed in the list of existing groups.

SUPPORTED FUNCTIONS

| Name used in command | Full name | Values | Data type | Can be sent to CoolMasterNet | Can be sent from CoolMasterNet |
|----------------------|-----------------|-------------------------------|-----------|------------------------------|--------------------------------|
| onoff | On/Off | 0 - off, 1 - on | 1.001 | ✓ | ✓ |
| ST | Set temperature | float value according to data | 9.001 | ✓ | ✓ |

| | | | | | |
|-------|-------------------|--|-------|---|---|
| | | type | | | |
| RT | Room Temperature | float value according to data type | 9.001 | ✓ | ✓ |
| Mode | Mode value | 0 - cool, 1 - heat, 2 - auto, 3 - dry, 5 - fan | 5.010 | ✓ | ✓ |
| M | Mode | 0 - cool, 1 - heat | 1.001 | ✓ | ✓ |
| Fstep | Fan speed step | 0 - previous fan speed, 1 - next fan speed (according to fspeed properties*) | 1.001 | ✓ | |
| F8 | Fan speed count | value is a sequential fan speed regarding allowed fan speeds in fspeed properties* | 5.010 | ✓ | ✓ |
| F% | Fan speed scale % | the percent value is converted to fan speed count (previous function) and then to actual fan speed, the steps are inversed when fan speed is sent from CoolMasterNet | 5.001 | ✓ | ✓ |

Notes:

* - If properties record doesn't exist default fan speeds are used: low, medium, high, auto

EXAMPLE

Print KNX status

```
>knx
```

```
KNX Line   : Not Detected
```

```
KNX RAM    : 7/128
```

```
OK
```

Create KNX group: link 10/0/1 with L1.100 as CoolMasterNet Mode input

```
>knx group 10/0/1 M < L1.100
```

```
OK
```

Create KNX group: link 10/0/1 with L1.100 as CoolMasterNet Room Temperature output

```
>knx group 10/0/1 RT > L1.100
```

```
OK
```

List KNX group for group address 10/0/1

```
>knx group 10/0/1
```

```
G000:10/0/1 [ On/Off           ] < L1.100
```

```
G001:10/0/1 [ Mode             ] < L1.100
```

```
G002:10/0/1 [ Room temperature ] > L1.100
```

```
OK
```

Print all available group functions

```
>knx group 10/0/1
```

```
onoff -> On/Off           DT:{1.001} [<|>]
```

```
ST    -> Set temperature  DT:{9.001} [<|>]
```

```
RT    -> Room temperature  DT:{9.001} [<|>]
```

```
M     -> Mode              DT:{1.001} [<|>]
```

```
Fstep -> Fan speed step    DT:{1.001} [<| ]
```

```
F8    -> Fan speed count  DT:{5.010} [<|>]
```

```
F%    -> Fan speed scale % DT:{5.001} [<|>]
```

```
OK
```

Delete group number 2

```
>knx group -2
OK
```

props

SYNOPSIS

```
props
props <UID_STRICT> <PROPERTY> <VAL>
```

DESCRIPTION

Query or change Indoor Unit(s) properties. In format without parameters **props** command will list all stored properties in comfortable table.

| <PROPERTY> | Mode | Value | Description | Notes |
|------------|------|-----------------------------|---|--|
| visible | R/W | 0 or 1 | Define Indoor Unit visibility on LCD. 0 - invisible, 1 - visible. Also depends on filter setting. | The property take effect after reboot |
| name | R/W | string | Define Indoor Unit name to display on LCD. Default value is no name. | When Indoor Unit doesn't have name property the default name in format Ln. XYY is displayed |
| fspeed | R/W | <<+ ->v, l, m, h, t, a | Define fan speed previous/next switch options on LCD: + allow, - forbid. Works also with fan speed commands to/from KNX bus. v - very low/quiet, l - low, m - medium, h - high, t - top, a - auto | Different fan speed properties can be combined in one command (see example below) |
| mode | R/W | <<+ ->c, d, f, h, hx, hh, a | Mode properties logic not implemented yet. c - cool, d - dry, f - fan, h - heat, hx - haux, hh - heat & haux, a - auto | Different mode properties can be combined in one command (see example below) |

Notes:

Indoor Unit properties record created when defining any of the listed properties, while the rest properties receive default values that can be overwritten later.

Bold values are defaults.

EXAMPLE

Add name property to Indoor Unit 204 on line L6

```
>props L6.204 name Kitchen
OK
```

Make Indoor Unit 102 on line L1 invisible on LCD

```
>props L1.102 visible 0
OK, Boot Required!
```

Forbid medium and auto fan speeds to Indoor Unit 107 on line L2

```
>props L1.102 fspeed -m-a
OK
```

Define cool, fan, haux modes to Indoor Unit 101 on line L1

```
>props L1.101 mode +c+f+hx-h-d-hh-a
OK
```

List all stored properties

```
>props
```

| UID | Name | Visi | Modes | Fspeeds |
|--------|---------|------|-----------|---------|
| L6.204 | Kitchen | 1 | c d f h a | l m h a |

```
L1.102 | - | 0 | c d f h | a | l m h a |
L2.107 | - | 1 | c d f h | a | l h |
L1.101 | - | 1 | c f hx | | l m h a |
OK
```

link**SYNOPSIS**

```
link
link delall
link -L
link ram R
link <UID1 STRICT><=|~><UID2 STRICT>
```

DESCRIPTION

Control link operation in CoolMasterNet. Linking ThermoPad on [CoolHub](#) line with Indoor Unit will grant the ThermoPad full control over the Unit.

- In format without parameters **link** will list existing links
- **link delall** - delete all links
- **link -L** - delete link number L (link numbers displayed in the list of existing links)
- **link ram R** - create link Data Base for R groups. The operation will take effect after reboot. Use this operation only if default link Data Base is too small.
- **link <UID1 STRICT><=|~><UID2 STRICT>** - create new link, where UID1 is ThermoPad on [CoolHub](#) line and UID2 is Indoor Unit. The command generate invisible [property](#) for one of the UID's. <=|~> defines link type: = regular link, the ThermoPad will become invisible; ~ link with haux mode usage, the Indoor Unit will become invisible. Regular link will allow to control the Indoor Unit through UID2, while haux link through UID1.

EXAMPLE

List existing links

```
>link L3.082=L1.101
L000: L3.081 <--> L7.096
L001: L3.083 <--> L1.100
OK
```

Delete link number 0

```
>link -0
```

OK

Link ThermoPad 082 on line L3 with Indoor Unit 101 on line L1 (regular link)

```
>link L3.082=L1.101
```

OK, Boot Required!

Link ThermoPad 080 on line L3 with Indoor Unit 100 on line L1 (haux link)

```
>link L3.080~L1.100
```

OK, Boot Required!

Delete all links

```
>link delall
```

OK

Create link Data Base for 60 links

```
>link ram 60
```

OK, Boot Required!

plug**SYNOPSIS**

```
plug <UID STRICT> <ascii if command>
```

DESCRIPTION

Forward compatible [HVAC status and control](#) or [configuration](#) command to CoolPlug device connected to [CoolHub](#) line. CoolPlug device will be recognized by CoolMasterNet as Indoor Unit with [UID](#). Refer to CoolPlug PRM for list of compatible commands.

EXAMPLE

Send [set](#) command to CoolPlug device 080 on CoolMasterNet line L3

```
>plug L3.080 set
S/N          : 00E60100
version      : 0.0.2
baud rate    : 9600
echo         : 1
verbose      : 1
deg C/F      : C
HVAC lines   : 1
OK
OK
```

Query current [line](#) statuses of CoolPlug device 080 on CoolMasterNet line L3

```
>plug L3.080 line
L1: P1P2 A:Master U00/G10 Simul(10) Not Main RC
Tx:1154/1695 Rx:1153/1694 TO:0/0 CS:0/0 Col:0/0 NAK:0/0
L2: PB Address:0x50(80) 9600_8N1
Tx:752/1038 Rx:1497/2319 TO:0/0 CS:0/0 Col:0/0 NAK:0/0
OK
OK
```

hdl**SYNOPSIS**

```
hdl
hdl delall
hdl - <UID STRICT>
hdl + <UID STRICT> <channel> <AC_No> <enable>
<m0m1m2m3m4f0f1f2f3>
hdl eth
hdl eth <enable|disable>
hdl eth myid <ID>
```

DESCRIPTION

Query or change HDL status and configuration. In format without parameters **hdl** command will list existing AC (Air Condition) configurations in similar way it is displayed in HDL Buspro Setup tool.

- **hdl delall** - delete all AC configurations
- **hdl - <UID STRICT>** - delete AC configuration for specific Indoor Unit
- **hdl + <UID STRICT> <channel> <AC_No> <enable> <m0m1m2m3m4f0f1f2f3>** - create AC configuration for Indoor Unit. All parameters are taken from AC configurations table in HDL Buspro Setup tool: channel - line number in AC configurations table, AC_No - integer number to bind to Indoor Unit, enable - 0 (invalid) or 1 (valid), modes - m0=cool m1=heat m2=fan m3=auto m4=dry, fan speeds - f0=auto f1=high f2=medium f3=low: +=allowed -=not allowed (see example).
- **hdl eth** - print HDL ethernet status and packet counters
- **hdl eth <enable|disable>** - enable/disable HDL over ethernet
- **hdl eth myid <ID>** - change Subnet and Device ID: the ID is entered in hexadecimal format, MS byte is Subnet ID, LS byte is Device ID (see example)

Notes:

Subnet ID and Device ID of HDL RS485 line can be changed with [line myid](#) command.

EXAMPLE

List existing AC configurations

```
>hdl
```

| Ch | AC | En | Remark | CA | CHFAD | AHML |
|----|----|----|--------|--------|-------|------|
| 1 | 1 | 1 | L1.100 | L1.100 | ++--- | -+++ |
| 2 | 2 | 1 | L1.101 | L1.101 | ++++- | ++++ |

```
OK
```

Delete AC configuration for Indoor Unit L1.101

```
>hdl - L1.101
```

```
OK
```

Create AC configuration for Indoor Unit L2.301: channel=2, AC number=2, valid, modes=cool heat, fan speeds=high medium

```
>hdl + L2.301 2 2 1 ++-----++
```

```
OK
```

HDL ethernet status

```
>hdl eth
```

```
status      : enabled
ID          : 0x01 0x63
send cntr   : 0
recv cntr   : 0
crc cntr    : 0
```

```
OK
```

Enable HDL over ethernet

```
>hdl eth enable
```

```
OK, Boot Required!
```

Change HDL ID of CoolMasterNet: Subnet ID=0x01=1, Device ID=0x63=99

```
>hdl eth myid 0163
```

```
OK, Boot Required!
```

simul**SYNOPSIS**

```
simul [Ln] <CNT>
```

DESCRIPTION

Simulate <CNT> Indoor Units on HVAC Line <Ln>. If <Ln> is omitted the first not "Unused" HVAC Line will be taken. Simulation is not persistent and CoolMasterNet reset will terminate it.

EXAMPLE

Simulate 5 Indoor Units on HVAC Line L2

```
>simul L2 5
```

```
OK
```

gpio**SYNOPSIS**

```
gpio
gpio func <A|B|C|D> <GPIO_FUNCTION>
```

gpio norm <A|B|C|D> <c|C|o|O>**DESCRIPTION**

Query or configure [GPIO](#) functionality. GPIO configurations are persistent (over power reset) and take effect only after reboot.

- Without parameters **gpio** command provides information about current GPIO configuration in the below format:

```
>gpio
A: ALL OFF (HI), N.O.
B: ALL ON (LO), N.C.
C: Unused (HI)
D: Unused (HI)
OK
```

| Field | Value | Description |
|------------------------------|---------------|--|
| GPIO name | A,B,C,D | |
| GPIO function | Unused | GPIO has no functionality and stays in HiZ input state |
| | ALL OFF Input | Turn all Indoor Units OFF in transition from Normal to Active state |
| | ALL ON Input | Turn all Indoor Units ON in transition from Normal to Active state |
| | OOS Output | On/Off Status. GPIO is in Normal state if all Indoor Units are OFF. GPIO is in Active state if all Indoor Units are ON |
| Current GPIO Signal Level | (HI) | Signal Level on GPIO is HI |
| | (LO) | Signal Level on GPIO is LO |
| Signal Level in Normal State | N.O. | Normal state of GPIO is "Normally Open" (HI) |
| | N.C. | Normal state of GPIO is "Normally Closed" (LO) |

- gpio func <A|B|C|D> <GPIO_FUNCTION>**
Configure GPIO function. Supported GPIO functions are listed in table above.

- gpio norm <A|B|C|D> <c|C|o|O>**
Configure Normal state of GPIO.

EXAMPLE

Disable GPIO A functionality

```
>gpio func A Unused
OK, Boot Required!
```

Set GPIO C function to "ALL OFF"

```
>gpio func C ALL OFF
OK, Boot Required!
```

Set Normal state of GPIO A to N.C.

```
>gpio norm A C
OK
```

info**SYNOPSIS**

```
info
```

DESCRIPTION

Query CollMasterNet dip switches position and DC output on lines L1 and L2.

```
>info
DIP P: | X | OFF | X | X |
DIP Q: | ON | X | ON | X |
DIP R: | X | X | X | X |
DIP S: | X | ON | ON | ON |
L1 DC- OFF 0V
L2 DC+ ON 16V
OK
```

| Parameter | Description | |
|------------|-------------|---|
| DIP switch | ON | DIP switch is ON |
| | OFF | DIP switch is OFF |
| | X | DIP switch can't be read |
| Line DC | + | DCOUT enable command was given to CoolMasterNet |
| | - | DCOUT disable or no command at all was given to CoolMasterNet |
| | 0v | No voltage detected on the line |
| | 16v | 16 volt detected on the line (own or external) |
| | ON | CoolMasterNet enabled DC output on the line On reboot, CoolMasterNet will enable DC output only if DCOUT enable command was given and DIP switch S is in correct position, otherwise it will disable DC output |
| | OFF | CoolMasterNet disabled DC output on the line |

modbus**SYNOPSIS**

```
modbus [<SETTING> <VALUE>]
modbus cg4
```

DESCRIPTION

Query or change CoolMasterNet ModBus configurations. Without parameters **modbus** command will list current ModBus configurations.

| <SETTING> | Mode | Printed as | Value | Description |
|-------------|------|-------------|-------------------|---|
| IP | R/W | ModBus IP | enable or disable | Enable/Disable ModBus IP server |
| server port | R/W | server port | integer | Change Modbus IP server port |
| ignore | R/W | CG4 ignore* | r | The setting is toggable, first command will enable the setting, second will disable (see example) |
| cg4 | RO | | | List CoolGate 4 ModBus addresses of existing (visible with ls command) Indoor Units |

Notes:

* - This setting is applicable only for [CoolGate 4](#). When the setting is enabled it will allow to read/write registers of several indoors by one request. Pay attention that there are reserved registers for every indoor, that will return an error when read/write requested without this setting. If the read request started from one of the readable registers (not reserved) the values of reserved registers will be 0xFFFF in the response. If the read request started from reserved register the response will be illegal data address. In a similar way it works with the write request. If the write request started from one of the writable registers (not reserved) the values that were intended for reserved registers will be ignored. If the write request started from one of the reserved registers the

response will be illegal data address.

EXAMPLE

List current ModBus configuration

```
>modbus
ModBus IP      : disabled
server port    : 502
CG4 ignore     : none
OK
```

Enable ignore setting

```
>modbus ignore r
OK
```

List current ModBus configuration

```
>modbus
ModBus IP      : disabled
server port    : 502
CG4 ignore     : regs
OK
```

Disable ignore setting

```
>modbus ignore r
OK
```

Enable ModBus IP

```
>modbus IP enable
OK, Boot Required!
```

Change ModBus IP server port

```
>modbus server port 500
OK, Boot Required!
```

List CoolGate 4 ModBus addresses of existing Indoor Units

```
>modbus cg4
L1.100 --> Hex: 0x1001, Dec: 4097
L1.101 --> Hex: 0x1011, Dec: 4113
OK
```

6.2.2 HVAC Status and Control Commands

[on](#)
[allon](#)
[off](#)
[alloff](#)
[cool](#)
[heat](#)
[fan](#)
[dry](#)
[auto](#)
[haux](#)
[temp](#)
[fspeed](#)
[swing](#)
[filt](#)
[stat](#)
[ls](#)
[ls2](#)
[query](#)
[feed](#)

[wh](#)
[main](#)
[vam](#)
[lock](#)
[group](#)
[CA](#)
[va](#)

on

SYNOPSIS

on **[UID]**

DESCRIPTION

Turn on Indoor Unit(s).

EXAMPLE

Turn on Indoor Unit 102 on line L1

```
>on L1.102
```

```
OK
```

Turn on all Indoor Units on Line L2

```
>on L2*
```

```
OK
```

Turn on all Indoor Units

```
>on
```

```
OK
```

allon

SYNOPSIS

allon

DESCRIPTION

Turn on all Indoor Units.

off

SYNOPSIS

off **[UID]**

DESCRIPTION

Turn off Indoor Unit(s).

EXAMPLE

Turn off Indoor Unit 102 on line L1

```
>off L1.102
```

```
OK
```

Turn off all Indoor Units on line L2

```
>off L2*
```

```
OK
```

Turn off all Indoor Units

```
>off
```

OK

aloff

SYNOPSIS

aloff

DESCRIPTION

Turn off all Indoor Units.

cool

SYNOPSIS

cool [\[UID\]](#)

DESCRIPTION

Set Indoor Unit(s) operation mode to cool.

EXAMPLE

Set Indoor Unit 102 on line L1 to cool mode

```
>cool L1.102
```

OK

Set all Indoor Units on line L2 to cool mode

```
>cool L2*
```

OK

Set all Indoor Units to cool mode

```
>cool
```

OK

heat

SYNOPSIS

heat [\[UID\]](#)

DESCRIPTION

Set Indoor Unit(s) operation mode to heat.

EXAMPLE

Set Indoor Unit 102 on line L1 to heat mode

```
>heat L1.102
```

OK

Set all Indoor Units on line L2 to heat mode

```
>heat L2*
```

OK

Set all Indoor Units to heat mode

```
>heat
```

OK

fan**SYNOPSIS****fan** [UID]**DESCRIPTION**

Set Indoor Unit(s) operation mode to fan.

EXAMPLE

Set Indoor Unit 102 on line L1 to fan mode

>fan L1.102

OK

Set all Indoor Units on line L2 to fan mode

>fan L2*

OK

Set all Indoor Units to fan mode

>fan

OK

dry**SYNOPSIS****dry** [UID]**DESCRIPTION**

Set Indoor Unit(s) operation mode to dry.

EXAMPLE

Set Indoor Unit 102 on line L1 to dry mode

>dry L1.102

OK

Set all Indoor Units on line L2 to dry mode

>dry L2*

OK

Set all Indoor Units to dry mode

>dry

OK

auto**SYNOPSIS****auto** [UID]**DESCRIPTION**

Set Indoor Unit(s) operation mode to auto.

EXAMPLE

Set Indoor Unit 102 on line L1 to auto mode

>auto L1.102

OK

Set all Indoor Units on line L2 to auto mode

```
>auto L2*
OK
Set all Indoor Units to auto mode
>auto
OK
```

haux

SYNOPSIS

haux [UID]

DESCRIPTION

Set ThermoPad(s), connected to CoolHub line, operation mode to haux.

EXAMPLE

```
Set ThermoPad 083 on line L3 to haux mode
>haux L3.083
OK
Set all ThermoPads on line L3 to haux mode
>haux L3
OK
Set all ThermoPads to haux mode
>haux
OK
```

temp

SYNOPSIS

temp [UID] [±]<TEMP>
temp [UID] <TEMP.d>

DESCRIPTION

Change Indoor Unit(s) Set Temperature.

- In form **temp [UID] [±]<TEMP>**
<TEMP> parameter must be decimal natural number. Command can work in relative or absolute manner. If plus '+' or minus '-' sign precedes <TEMP> parameter it's value will be added to or substituted from current Set Temperature value. Otherwise Set Temperature will be set to the given <TEMP> value.
- In form **temp [UID] <TEMP.d>**
Set Temperature parameter <TEMP.d> is a fractal number with 0.1 precision. (In this case preceding +/- are not allowed). If HVAC System does not support 0.1 precision for the Set Temperature (see table below), the final value will be nearest supported value.

| AC Type | Set Temperature Precision |
|---------|---------------------------|
| DK | 0.1°C |
| ME | 0.1°C |
| SM | 0.1°C |
| FJ | 0.3°C |
| SA | 0.5°C |
| TO | 0.5°C |
| PN | 0.5°C |

| | |
|-------------|-------|
| MH | 0.5°C |
| LG | 0.5°C |
| HT | 1°C |
| GR | 1°C |
| MD,CG,KT,TR | 1°C |
| TI | 1°C |
| MT | 1°C |

The **deg** setting (see [set](#) command) defines which temperature scale Celsius or Fahrenheit is used for **<TEMP>** and **<TEMP.d>** parameters value.

EXAMPLE

Set Indoor Unit 102 on line L1 Set Temperature to 23°

```
>temp L1.102 23
```

```
OK
```

Decrease all Indoor Units on line L2 Set Temperature by 2°

```
>temp L2* -2
```

```
OK
```

Set all Indoor Units on line L2 Set Temperature to 24.5°

```
>temp L2* 24.5
```

```
OK
```

Set all Indoor Units Set Temperature to 22°

```
>temp 22
```

```
OK
```

fspeed

SYNOPSIS

```
fspeed [UID] <v,V|l,L|m,M|h,H|t,T|a,A>
```

DESCRIPTION

Set Indoor Unit(s) Fan Speed to:

- v, V - very low or quiet
- l, L - low
- m, M - medium
- h, H - high
- t, T - top or very high
- a, A - auto

Not all Indoor Units support Fan Speed options listed above. Specific Fan Speeds support depend on specific Indoor Unit capabilities. If requested Fan Speed is not supported by Indoor Unit(s) **fspeed** command will have no effect.

EXAMPLE

Set Indoor Unit 102 on line L1 Fan Speed to low

```
>fspeed L1.102 l
```

```
OK
```

Set all Indoor Units on Line L2 Fan Speed to high

```
>fspeed L2* h
```

```
OK
```

Set all Indoor Units Fan Speed to medium

```
>fspeed m
```

```
OK
```

swing**SYNOPSIS**

swing [UID] <h|v|a|3|4|6|x>

DESCRIPTION

Set Indoor Unit(s) louver position to:

- h - horizontal
- v - vertical
- a - auto (swing)
- 3 - 30°
- 4 - 45°
- 6 - 60°

Not all Indoor Units support louver position options listed about or have louver position control at all. Louver control is capability of the specific Indoor Unit type. If requested louver position is not supported by Indoor Unit(s), **swing** command will have no effect or will return an error.

EXAMPLE

Set Indoor Unit 102 on line L1 louver to horizontal position

```
>swing L1.102 h
```

```
OK
```

Set all Indoor Units louver to 30° position

```
>swing 3
```

```
OK
```

filt**SYNOPSIS**

filt [UID]

DESCRIPTION

Reset Filter Sign.

EXAMPLE

Reset Filter Sign on Indoor Unit 102 On line L1

```
>filt L1.102
```

```
OK
```

Reset Filter Sign on all Indoor Units on Line L2

```
>filt L2*
```

```
OK
```

Reset Filter Sign on all Indoor Units

```
>filt
```

```
OK
```

stat**SYNOPSIS**

stat [UID]
stat2 [UID]
stat3 [UID]

stat4 [UID]**DESCRIPTION**

Get Indoor Unit(s) status list.

These commands are deprecated and are not recommended for use in CoolMasterNet. They are implemented for backward compatibility only. Detailed description of these commands can be found in the PRM of previous CoolMaster versions.

ls**SYNOPSIS**

ls [UID]

DESCRIPTION

Get Indoor Unit(s) status list. If UID is omitted all Indoor Units connected to CoolMasterNet will be listed. Indoor Unit status line has strict format, so that every status field is printed in fixed position.

- Indoor Unit status line with Celsius temperature scale

```
0123456789012345678901234567890123456
L2.102 OFF 20C 27C High Cool OK - 0
```

- Indoor Unit status line with Fahrenheit temperature scale

```
012345678901234567890123456789012345678
L2.102 OFF 120F 127F High Cool OK - 0
```

| Field | Position in string | | Values |
|---------------------|--------------------|------------|---|
| | Celsius | Fahrenheit | |
| UID | 0-5 | | LN.XYY |
| On/Off | 7-9 | | ON,OFF |
| Set Temperature | 11-12 | 11-13 | nnC or nnnF |
| Room Temperature | 15-16 | 16-18 | nnC or nnnF |
| Fan Speed | 19-22 | 21-24 | VLow, Low, Med, High, Top, Auto |
| Operation Mode | 24-27 | 26-29 | Cool, Heat, Fan, Dry, Auto |
| Indoor Failure Code | 29-32 | 31-34 | OK - no failure, else Indoor Failure Code |
| Filter Sign | 34 | 36 | - or # (Filter Sign) |
| Demand | 36 | 38 | 0 or 1 |

EXAMPLE

```
>ls L2
L2.101 ON 25C 27C Low Cool OK - 1
L2.102 OFF 20C 27C High Cool OK - 0
OK
>ls L2.101
L2.101 ON 25C 27C Low Cool OK - 1
OK
>ls
L1.101 ON 25C 24C Low Cool OK - 1
L1.102 ON 22C 23C Med Cool OK - 0
L2.101 ON 25C 27C Low Cool OK - 1
L2.102 OFF 20C 27C High Cool OK - 0
OK
```

Is2**SYNOPSIS****Is2** [[UID](#)]**DESCRIPTION**

Same as Is, but with decimal precision in temperatures.

Get Indoor Unit(s) status list. If UID is omitted all Indoor Units connected to CoolMasterNet will be listed.

Indoor Unit status line has strict format, so that every status field is printed in fixed position.

- Indoor Unit status line with Celsius temperature scale

```
01234567890123456789012345678901234567890
```

```
L1.102 ON 16.9C 27.0C High Cool OK - 0
```

- Indoor Unit status line with Fahrenheit temperature scale

```
0123456789012345678901234567890123456789012
```

```
L1.102 ON 062.4F 080.6F High Cool OK - 0
```

| Field | Position in string | | Values |
|---------------------|--------------------|------------|---|
| | Celsius | Fahrenheit | |
| UID | 0-5 | | LN.XYY |
| On/Off | 7-9 | | ON,OFF |
| Set Temperature | 11-14 | 11-15 | nn.nC or nnn.nF |
| Room Temperature | 17-20 | 18-22 | nn.nC or nnn.nF |
| Fan Speed | 23-26 | 25-28 | VLow, Low, Med, High, Top, Auto |
| Operation Mode | 28-31 | 30-33 | Cool, Heat, Fan, Dry, Auto |
| Indoor Failure Code | 33-36 | 35-38 | OK - no failure, else Indoor Failure Code |
| Filter Sign | 38 | 40 | - or # (Filter Sign) |
| Demand | 40 | 42 | 0 or 1 |

EXAMPLE

```
>ls L2
L2.101 ON 25.7C 27.2C Low Cool OK - 1
L2.102 OFF 20.0C 27.3C High Cool OK - 0
OK
>ls L2.101
L2.101 ON 25.7C 27.2C Low Cool OK - 1
OK
>ls
L1.101 ON 25.4C 24.1C Low Cool OK - 1
L1.102 ON 22.1C 23.4C Med Cool OK - 0
L2.101 ON 25.7C 27.2C Low Cool OK - 1
L2.102 OFF 20.0C 27.3C High Cool OK - 0
OK
```

query**SYNOPSIS**

query [<UID STRICT>](#) [<o|m|f|t|h|e|a|s>](#)

DESCRIPTION

Query one of the operation conditions of given Indoor Unit. [<UID STRICT>](#) parameter must define single Indoor

Unit in form **Ln.XYY** or **XYX**. Resulting value is printed as alpha-numeric value according to the table below.

| Query | Operation Condition | Value |
|-------|---------------------|--|
| o | On/Off | 0 - Off, 1 - On |
| m | Operation Mode | 0 - Cool 1 - Heat 2 - Auto 3 - Dry 4 - Haux 5 - Fan |
| f | Fan Speed | 0 - Low 1 - Medium 2 - High 3 - Auto 4 - Top |
| t | Set Temperature | Natural |
| e | Failure Code | 0 - No failure, otherwise failure code same as in ls command |
| a | Ambient Temperature | Natural |
| h | Set Temperature | 0.01° Precision |
| s | Louver Position | 0 - No Louver Control a - auto (swing) h - horizontal 3 - 30° 4 - 45° 6 - 60° v - vertical x - stop (swing) |

EXAMPLE

```
>query L1.100 o
1
OK
>query L1.100 m
0
OK
>query L1.100 t
25
OK
>query L1.100 a
27
OK
>query L1.100 f
2
OK
>query L1.100 e
U4
OK
>query L1.100 e
0
OK
>query L1.100 h
20.50
```

OK

feed**SYNOPSIS**

```
feed [UID] [±]<TEMP[.d]>
```

DESCRIPTION

This command defines CoolMasterNet Own Ambient Temperature (as if it were measured by CoolMasterNet itself). Command can work in relative or absolute manner. If plus '+' or minus '-' sign precedes <TEMP> parameter it's value will be added to or substituted from current value. Otherwise Own Ambient Temperature will be set to the given <TEMP> value. <TEMP> can be provided with 0.1 precision. If <TEMP> is zero CoolMasterNet will not use Own Ambient Temperature for it's operation. CoolMasterNet keeps separate Own Ambient Temperature for each Indoor Unit.

Own Ambient Temp usage depends on HVAC Line type.

| HVAC Line | Own Ambient Temp usage |
|-----------|---|
| ME | If not zero, sent to Indoor Unit as Smart Controller temperature sensor value |
| M1M2 | |
| FUS | |
| P1P2 | |
| H1H2 | If not zero, sent to Indoor Unit as Wired Thermostat temperature sensor value |
| TAD | Temperature sent to Indoor Unit in i-feel message |

EXAMPLE

Define Own Ambient Temperature to all Indoor Units on line L2

```
>feed L2 24.5
```

OK

Define Own Ambient Temperature to Indoor Unit 101 on line L1

```
>feed L1.101 27
```

OK

wh**SYNOPSIS**

```
wh <UID STRICT> <h|e|w|a|t<+|-|temp>|b<+|->>
```

DESCRIPTION

Control Water Heater Unit. When letter is omitted special water heater status is displayed.

| Letter | Operation | Compatibility | |
|-------------|------------------------------------|---------------|-----------------|
| | | ME | P1P2 (Altherma) |
| h | Heat Mode | ✓ | |
| e | Eco Mode | ✓ | |
| w | Hot Mode | ✓ | |
| a | Anti-freeze Mode | ✓ | |
| t | Tank (+=on, -=off, temp=set point) | | ✓ |
| b | Booster (+=on, -=off) | | ✓ |
| "no letter" | Print status | | ✓ |

ON/OFF control of the Water Heater Unit is performed with regular **on** and **off** commands.

EXAMPLE

Set Hot Mode on Water Heater Unit 101 on Line L1

```
>wh L1.101 w
```

OK

Set Altherma Unit 000 on Line L4 Tank Set Temperature to 40°

```
>wh L4.000 t40
```

OK

Turn on Tank of Altherma Unit 000 on Line L4

```
>wh L4.000 t+
```

OK

Turn on Tank Booster of Altherma Unit 000 on Line L4

```
>wh L4.000 b+
```

OK

Print Tank status of Altherma Unit 000 on Line L4

```
>wh L4.000
```

```
L4.000 ON 40C 35C +
```

OK

main

SYNOPSIS

```
main [Ln]
```

```
main <UID_STRICT> <0|1>
```

DESCRIPTION

Change or query Daikin Indoor Units main RC setting. Without parameters **main** command will list main RC setting of all Daikin Indoor Units on all lines. Line number **Ln** can be indicated to list Daikin Indoor Units on specific line.

| Main RC setting | Description |
|-----------------|----------------------------|
| + | Indoor Unit is main RC |
| - | Indoor Unit is not main RC |
| ? | No main RC in the system |

Notes:

To set Indoor Unit as main RC the previous main RC have to be unset first.

EXAMPLE

List main RC setting of Indoor Units on line L2

```
>main L2
```

```
L2.200 -
```

```
L2.201 -
```

```
L2.202 -
```

```
L2.203 -
```

```
L2.204 -
```

```
L2.205 -
```

```
L2.206 +
```

OK

List main RC setting of all Indoor Units

```
>main
```

```

L1.300 -
L1.301 +
L1.302 -
L2.100 ?
L2.101 ?
L2.102 ?
L2.103 ?
OK
Unset main RC: Indoor Unit 206 on line L2
>main L2.206 0
OK
Set new main RC: Indoor Unit 201 on line L2
>main L2.201 1
OK

```

vam**SYNOPSIS**

```

vam <UID STRICT>          <a|b|x|n|l|h|H|s|t|A|+|->
vam
vam <UID STRICT>

```

DESCRIPTION

Control Ventilation Unit.

- **vam <UID_STRICT>** - get Ventilation Unit status.
- **vam** - get all Ventilation Units status.

| Letter | Operation | Compatibility | | | |
|--------|----------------------------|---------------|----|----|----|
| | | DK | ME | LG | SM |
| a | Auto Mode | ✓ | ✓ | ✓ | ✓ |
| b | Bypass (Bps) Mode | ✓ | ✓ | | ✓ |
| x | Heat Exchange (HExc) Mode | ✓ | ✓ | ✓ | ✓ |
| n | Normal Mode | | | ✓ | |
| S | Sleep Mode | | | | ✓ |
| l | Low Fan | ✓ | ✓ | ✓ | ✓ |
| L | Low Fan with Fresh-Up | ✓ | | | |
| h | High Fan | ✓ | ✓ | ✓ | ✓ |
| H | High Fan with Fresh-Up | ✓ | | | |
| s | Super High Fan | | | ✓ | ✓ |
| t | Top Fan | | ✓ | | |
| A | Auto Fan | | ✓ | ✓ | |
| + | Turn on ventilation unit* | | | | ✓ |
| - | Turn off ventilation unit* | | | | ✓ |

ON/OFF control of the Ventilation Unit is performed with regular **on** and **off** commands.

There are, though, dual units, that consist of Air Condition Unit and Ventilation unit. In this case **on** and **off** commands will control the Air Condition Unit, while **vam <UID STRICT> +/vam <UID STRICT> -** will control the Ventilation Unit.

EXAMPLE

Set Heat Exchange mode on Ventilation Unit 101 on Line L1

```
>vam L1.101 x
```

OK

lock**SYNOPSIS**

```
lock <UID STRICT>
lock [UID] <-|+>[o|m|t|n]
```

DESCRIPTION

Most of the HVAC systems have an prohibit/lock/inhibit functionality to prevent user from changing Indoor Unit settings via wired or remote Local Controller. Same functionality is provided by CoolMasterNet with **lock** command.

- **lock <UID STRICT>** - Query locks for specific Indoor Unit. [<UID STRICT>](#) defines single Indoor Unit in form Ln.XYY

| Lock | Operation | Compatibility | | | |
|------|--|---------------|------------|----|----------------|
| | | DK | SA, TO, PN | HT | MD, KT, TR, CG |
| + | Full lock*: On/Off, Mode, Set Temperature | ✓ | ✓ | ✓ | ✓ |
| - | Full unlock**: On/Off, Mode, Set Temperature | ✓ | ✓ | | |
| +o | Lock On/Off | ✓ | ✓ | | |
| -o | Unlock On/Off | ✓ | ✓ | | |
| +m | Lock Mode | ✓ | ✓ | | |
| -m | Unlock Mode | ✓ | ✓ | | |
| +t | Lock Set Temperature | ✓ | ✓ | | |
| -t | Unlock Set Temperature | ✓ | ✓ | | |
| +n | Lock On | ✓ | | | |
| -n | Unlock On | ✓ | | | |

Notes:

Different invocations can be combined in one command (see example below)

Lock* - means operation(s) is(are) prohibited

Unlock** - means operation(s) is(are) enabled

EXAMPLE

Query locks for Indoor Unit L1.103 (SA/TO/PN)

```
>lock L1.103
```

```
-o -m -t
```

```
OK
```

Query locks for Indoor Unit L2.101 (MD/KT/TR/CG/HT)

```
>lock L2.101
```

```
+
```

```
OK
```

Query locks for Indoor Unit L1.100 (DK)

```
>lock L1.100
```

```
-o -m -t +n
```

```
OK
```

Lock Mode change and Unlock Set Point change

```
>lock L1.102 +m-t
```

```
OK
```

Full Lock

```
>lock L5.002 +
OK
```

group

SYNOPSIS

```
group
group <UID1 STRICT> <UID2 STRICT>
group delall
group -G
group ram R
```

DESCRIPTION

Control group operation in CoolMasterNet. Grouping of two Indoor Units means that second Indoor Unit will follow ON/OFF, Mode, Fan Speed, Set Temperature and Swing settings of the first Indoor Unit.

- In format without parameters **group** will list existing groups
- **group delall** - delete all groups
- **group -G** - delete group number G (group numbers displayed in the list of existing groups)
- **group <UID1 STRICT> <UID2 STRICT>** - create new group, where UID2 will follow UID1
- **group ram R** - create group Data Base for R groups. The operation will take effect after reboot. Use this operation only if default group Data Base is too small.

EXAMPLE

Group Indoor unit L5.001 with L5.002

```
>group L5.001 L5.002
OK
```

List existing groups

```
>group
G000: L5.001 --> L5.002
G001: L5.001 --> L5.003
OK
```

Delete group number 0

```
>group -0
OK
```

Delete all groups

```
>group delall
OK
```

Create group Data Base for 50 groups

```
>group ram 50
OK, Boot Required!
```

CA

SYNOPSIS

```
CA <Ln> [adr]
CA <Ln> ?
```

DESCRIPTION

In format **CA <Ln> ?** query centralize address of Indoor Units on line **Ln** (only for SA/TO/PN). In format **CA <Ln> [adr]** distribute new group addresses on line **Ln** starting from **adr**, without **adr** the default value of 100 is taken (only for DK).

EXAMPLE

Query centralize address of Indoor Units on line L2 (SA/TO/PN)

```
>CA L2 ?
L2.101 --> Z56
L2.102 --> Z62
L2.103 --> Z23
L2.104 --> Z33
OK
```

Distribute new group addresses on line L1 starting from 300 (DK)

```
>CA L1 300
OK, Boot Required!
```

va**SYNOPSIS**

```
va
va auto
va delall
va ram R
va - <UID STRICT|VA>
va + <UID STRICT> <VA>
```

DESCRIPTION

In order to simplify translation of the [UID](#) into ModBus object address, CoolMasterNet uses VA (Virtual Address), associated with [UID](#). Each [UID](#) can have none, one or a number of associated VA's, but each VA can be associated only with one [UID](#). VA's are plain numbers starting from 001 (decimal). Virtual Address concept is applicable to [CoolGate 5](#) and [ModBus IP](#).

- In format without parameters **va** command will list all VA associations and their ModBus addresses in hexadecimal and decimal format.
- **va auto** - CoolMasterNet will automatically associate VA's with existing (visible with [ls](#) command) [UID](#)'s. This will delete previous associations
- **va delall** - delete all VA associations
- **va ram R** - create VA Data Base for R associations. The operation will take effect after reboot. Use this operation only if default VA Data Base is too small.
- **va - <UID STRICT|VA>** - delete all VA associations with specific Indoor Unit or delete specific VA association
- **va + <UID STRICT> <VA>** - add new VA association

EXAMPLE

Auto associate all existing Indoor Units

```
>va auto
OK
```

Associate VA 007 with Indoor Unit 100 on line L1

```
>va + L1.100 7
OK
```

List all VA associations

```
>va
L1.100 --> 001 [Hex: 0x0011 | Dec: 0017]
      +-> 007 [Hex: 0x0071 | Dec: 0113]
L1.101 --> 002 [Hex: 0x0021 | Dec: 0033]
L2.101 --> 003 [Hex: 0x0031 | Dec: 0049]
L2.102 --> 004 [Hex: 0x0041 | Dec: 0065]
L2.103 --> 005 [Hex: 0x0051 | Dec: 0081]
```



```
L2.104 --> 006 [Hex: 0x0061 | Dec: 0097]
OK
Delete all VA associations of Indoor Unit 100 on line L1
>va - L1.100
OK
Delete VA association number 4
>va - 4
OK
Delete all VA associations
>va delall
OK
Create VA Data Base for 200 associations
>va ram 200
OK, Boot Required!
```