

KNX Module User Guidelines



KNX Module 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.

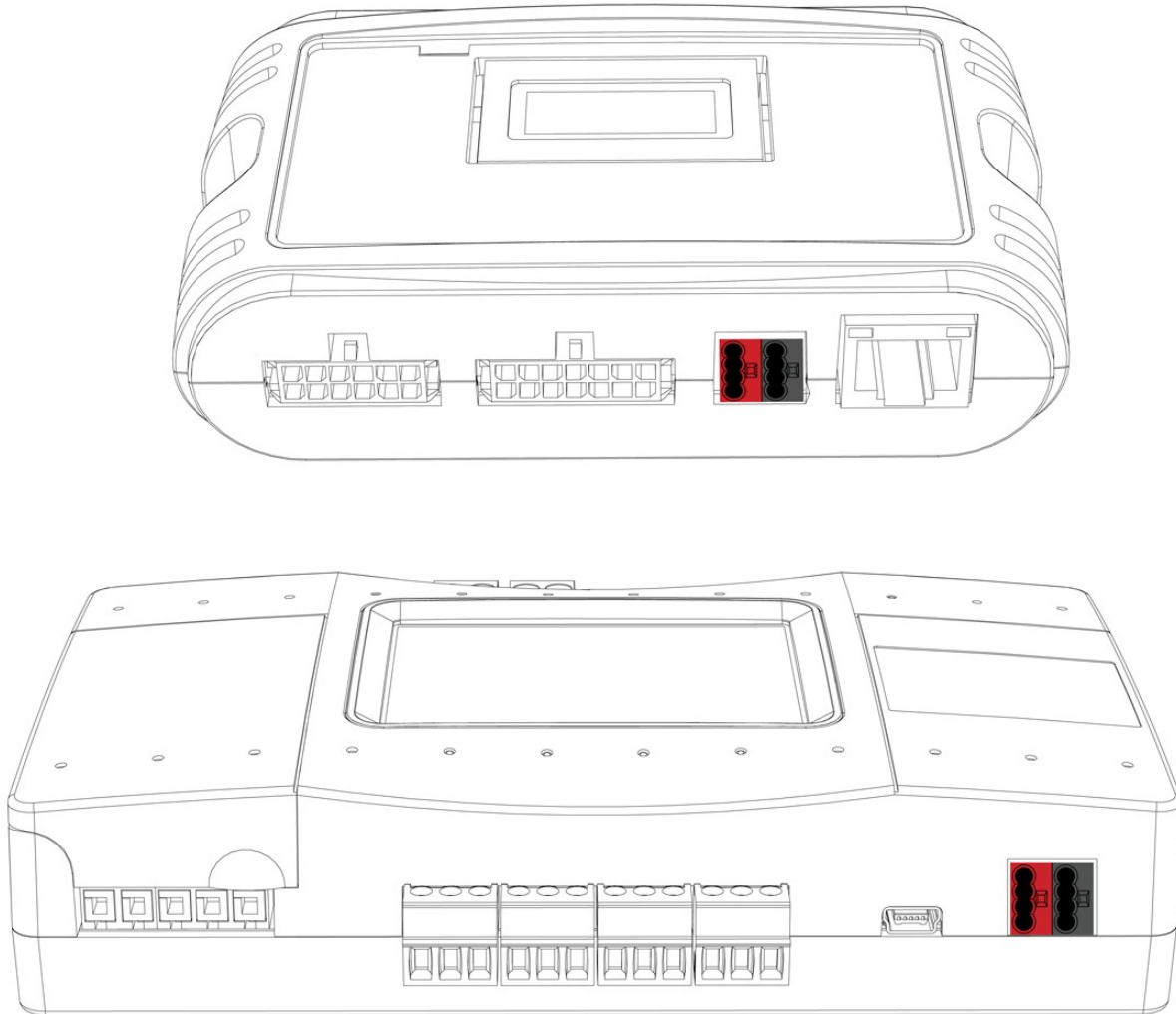


Table of Contents

1 Overview	4
2 Configuration	5
2.1 KNX Module activation.....	5
2.2 KNX Module settings.....	5
3 KNX Project	7
3.1 Group Addresses.....	7
3.2 KNX group command.....	7
3.3 KNX Module functions.....	8
3.4 Examples of grouping.....	9
4 KNX Module commands reference	10

1 Overview

Physical connection of CoolAutomation devices: CoolMasterNet or CoolLinkNet to the KNX bus with TP physical layer is made with "CoolAutomation KNX expansion board", further referenced as "KNX Module".



KNX Module is pre-installed inside CoolMasterNet or CoolLinkNet body by CoolAutomation if KNX expansion option was ordered together with a CoolAutomation device.

2 Configuration

2.1 KNX Module activation

KNX Module has to be activated by assigning appropriate communication Line of the CoolAutomation device for the KNX functionality. In CoolMasterNet it has to be Line **L3** and in CoolLinkNet it has to be Line **L5**. To check if KNX Module is already activated, **line** command should be used:

CoolMasterNet

```
>line
L1: DK Master U00/G00 myid:0B
Tx:2/2 Rx:2/2 TO:0/0 CS:0/0 Col:0/0 NAK:0/0
L2: Unused
Tx:0/0 Rx:0/0 TO:0/0 CS:0/0 Col:0/0 NAK:0/0
L3: KNX Physical Address:Not Assigned
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: Unused
Tx:0/0 Rx:0/0 TO:0/0 CS:0/0 Col:0/0 NAK:0/0
L8: Unused
Tx:0/0 Rx:0/0 TO:0/0 CS:0/0 Col:0/0 NAK:0/0
OK
```

CoolLinkNet

```
>line
L1: Unused
Tx:0/0 Rx:0/0 TO:0/0 CS:0/0 Col:0/0 NAK:0/0
L2: Unused
Tx:0/0 Rx:0/0 TO:0/0 CS:0/0 Col:0/0 NAK:0/0
L3: Unused
Tx:0/0 Rx:0/0 TO:0/0 CS:0/0 Col:0/0 NAK:0/0
L4: MLM2 Slave U00/G00 Not Connected
Tx:0/0 Rx:0/0 TO:0/0 CS:0/0 Col:0/0 NAK:0/0
L5: KNX Physical Address:Not Assigned
Tx:0/0 Rx:0/0 TO:0/0 CS:0/0 Col:0/0 NAK:0/0
OK
```

If for some reason KNX Module is not activated, it can be done with below command:

CoolMasterNet

```
>line type L3 KNX
OK, Boot Required!
```

CoolLinkNet

```
>line type L5 KNX
OK, Boot Required!
```

2.2 KNX Module settings

- Setting a number of supported group addresses.

By default KNX Module supports up to 128 group addresses. If that amount is not sufficient it can be increased (or decreased) with below command:

```
>knx ram 200
OK, Boot Required!
>
```

This will increase a maximal number of group addresses supported by KNX Module to 200. The absolute maximal number of supported group addresses is 1024.

- Setting a physical address of the KNX Module.

Physical address (also referenced as Individual Address in KNX specifications) has a following format:

Area: 4bit				Line: 4bit				Bus device: 8bit							
A	A	A	A	L	L	L	L	D	D	D	D	D	D	D	D

The physical address has no significance during normal operation of the installation. For KNX Module, physical



address can be left unassigned or, if required, it can be set with command like:

```
>knx addr 10/2/45  
OK  
>
```

An individual address must be unique within a KNX installation.

3 KNX Project

3.1 Group Addresses

KNX bus is implemented as a decentralized system. Decentralized management is implemented within the devices, they are communicating directly to each other without recourse to hierarchy or network supervisory. The communication between KNX communication objects (sensors and actuators) is made with telegrams sent to group addresses. The group addressing is based on the exchange of data coded with common rules between communication objects. Actuators can listen to several group addresses. Sensors however can only send one group address per telegram.

ETS tool should be used to assign group addresses for communication objects provided by KNX devices used in project. CoolMasterNet and CoolLinkNet KNX Modules do not require ETS interaction. CoolAutomation devices are configured by internal set of commands described later in this document.

When setting the group address via ETS, a “2-level” (main group/ subgroup) or “3-level” structure (main group/ middle group/subgroup) can be selected. The level structure can be changed in the project properties of each individual project. The group address 0/0/0 is reserved for so-called broadcast messages (telegrams to all available bus devices).

- Group Address 2 level Main/Sub



- Group Address 3 level Main/Middle/Sub



3.2 KNX group command

To link KNX group objects, defined with ETS tool, with CoolMasterNet or CoolLinkNet functions, 'knx group' command should be used.

```
knx group [<GA> <func> <direction> <UID>]
```

- <GA> - KNX group address. M/S or M/Mi/S structure formats are supported
- <func> - Required CoolMasterNet or CoolLinkNet function (see [KNX functions](#)).
- <direction> - Data direction, encoded as:
 - < - Into KNX Module from Group Object with W - Write direction flag
 - > - From KNX Module to Group Object with R -Read direction flag
- <UID> - Indoor Unit, in same format as reported by **Is** command

KNX group command binds between KNX Group Object with given group address - <GA> and specific function of CoolMasterNet or CoolLinkNet that will be applied to the Indoor Unit(s) with given <UID>. Groups are stored in CoolMasterNet or CoolLinkNet non volatile memory. The maximal number of groups is configurable (see [KNX Module settings](#)).

Other formats of the 'knx group' command invocations provide options to list already defined groups or delete specific or all groups.

List KNX groups:

```
>knx group
G000:10/0/1 [ On/Off ] < L1.002
G001:10/0/2 [ On/Off ] > L1.002
```



```
G002:10/0/3 [ Set temperature ] > L1.002
G003:10/0/4 [ Set temperature ] < L1.002
G004:10/0/5 [ Mode           ] > L1.002
G005:10/0/6 [ Mode           ] < L1.002
OK
```

Dele group G002:

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

Delete all groups:

```
>knx group delall
OK
```

3.3 KNX Module functions

The **<func>** parameter of the 'knx group' command defines both: function associated with a group and a datapoint type of the information that will be transferred in corresponding KNX telegram. Functions supported by KNX Module are listed below.

<func>	Function description	Datapoint Type	Dir.	Min. FW Version
onoff	ON/OFF control or status. 0-OFF, 1-ON	{1.001} DPT_Switch B1	< >	Any
ST	Set point temperature °C	{9.001} DPT_Value_Temp F16	< >	
RT	Room temperature °C	{9.001} DPT_Value_Temp F16	>	
M	Mode: Cool/Heat. 0-Cool, 1-Heat	{1.001} DPT_Switch B1	< >	
Mode	Mode: 0-Cool, 1-Heat, 2-Auto, 3-Dry, 5-Fan	{5.010} DPT_Value_1_Ucount U8	< >	0.2.8
Fstep	Skip to next fan speed	{1.001} DPT_Switch B1	<	0.2.1
F8	Fan speed count	{5.010} DPT_Value_1_Ucount U8	< >	0.2.1
F%	Fan speed scaled	{5.001} DPT_Scale	< >	0.2.3
E16	HVAC Error (failure) code (0 - no failure)	{7,1} DPT_Value_2_Ucount U16	>	0.4.9
E	HVAC Error (failure) indication: 0-no failure, 1-failure exists	{1,1} DPT_Switch B1	>	0.4.9
inhb	Inhibit (see notes below)	{1,1} DPT_Switch B1	<	0.4.9
filter	Filter cleaning indication	{1,1} DPT_Switch B1	< >	0.4.9

Notes:

- **Fstep** function takes current fan speed and increases it in one step (regardless to the value sent in KNX telegram). If current fan speed is maximal supported by related Indoor Unit, lowest possible fan speed will be taken. For example if Indoor Unit supports Low, Medium and High fan speeds **Fstep** will change fan speed from Low to Medium, from Medium to High or from High to Low. This function has only one direction - from KNX device to the KNX Module.
- **F8** function takes all supported fan speeds of the related Indoor Unit and gives them a numbers starting from 1 (one). In < direction (to the KNX Module), upon reception of the corresponding KNX telegram datapoint value will be translated into fan speed. In > direction (from the KNX Module) current fan speed number will be sent in KNX telegram.
- **F%** function is similar to **F8** function except of the fan speed encoding. In case of F%, encoding is scaled in % according to the below table.

Supported Fan Speeds				
Auto	Low	Medium	High	Top



0%	50%		100%	
0%	33%	66%	100%	
0%	25%	50%	75%	100%

- **inhb** function activates or deactivates (according to datapoint value 1 or 0) inhibit of the indoor unit ON operation. Upon inhibit activation current ON/OFF status of the indoor unit is stored and indoor unit is unconditionally turned OFF after about 8s delay. Indoor unit will be forced OFF until inhibit deactivation. On inhibit deactivation indoor unit will be turned ON if ON status was previously stored at inhibit activation. This function is oriented to work with window sensor to prevent HVAC operation while window is open.

3.4 Examples of grouping

```
>knx group 10/0/1 onoff < L1.101  
OK
```

Indoor Unit L1.101 will be switched ON or OFF, once KNX telegram with destination group address 10/0/1 and 1-bit datapoint type {1.001} is sent by KNX device like thermostat or push-button. Value of "0" will switch the Indoor Unit OFF and value of "1" - ON.

```
>knx group 10/0/2 onoff > L1.101  
OK
```

Upon ON/OFF status change of the Indoor Unit L1.101, KNX telegram with destination group address 10/0/2 will be sent by KNX Module. Telegram will contain 1-bit datapoint type with value "0" - if Indoor Unit status is OFF and "1" - if ON. Note how the direction characters '>' and '<' work.

```
>knx group 10/0/3 ST < L1.101  
OK
```

Set point temperature of the Indoor Unit L1.101 will be changed by KNX Module upon reception of the KNX telegram to destination group address 10/0/3. Temperature value is encoded as {9.001} datapoint type.

```
>knx group 10/0/4 ST > L1.101  
OK
```

In this case change of the set point temperature of the Indoor Unit L1.101 will be reported by KNX Module that will send KNX telegram to group address 10/0/4 with {9.001} datapoint type.

4 KNX Module commands reference

- **knx** - Query KNX Module status

```
>knx
KNX Line   : Connected
KNX RAM    : 10/128
OK
>
```

KNX Line status can be "Connected" or "Disconnected". RAM usage reflects a number of defined group addresses and a total supported number of group addresses.

- **knx addr <A/L/D>** - Set KNX Module physical address.

```
>knx addr 10/2/45
OK
```

- **knx ram <N>** - Set number of supported group addresses

```
>knx ram 200
OK, Boot Required!
```

- **knx group** - List defined KNX groups

```
>knx group
G000:10/0/1 [ On/Off           ] < L1.002
G001:10/0/2 [ On/Off           ] > L1.002
G002:10/0/3 [ Set temperature  ] > L1.002
G003:10/0/4 [ Set temperature  ] < L1.002
G004:10/0/5 [ Mode             ] > L1.002
G005:10/0/6 [ Mode             ] < L1.002
OK
```

- **knx group <GA> <func> <direction> <UID>** - Define KNX group

```
>knx group 10/0/3 ST < L1.101
OK
```

- **knx group -<N>** - Delete KNX group

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

- **knx group dellall** - Delete all KNX groups



```
>knx group delall  
OK
```

- **knx funcs** - List supported functions

```
>knx funcs  
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
```