IntesisBox®
MH-RC-KNX-1i v1.1

User's Manual
Issue Date: 08/2013
r2.0 eng
Information in this document is subject to change without notice. The software described in this document is furnished under a license agreement or nondisclosure agreement. The software may be used only in accordance with the terms of those agreements. No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or any means electronic or mechanical, including photocopying and recording for any purpose other than the purchaser’s personal use without the written permission of Intesis Software S.L.

Intesis Software S.L.
Milà i Fontanals, 1 bis
08700 Igualada
Spain

TRADEMARKS
All trademarks and trade names used in this document are acknowledged to be the copyright of their respective holders.
Gateway for integration of Mitsubishi Heavy Industries (MHI) air conditioners into KNX TP-1 (EIB) control systems. Compatible with RAC* Series, FD Series, KX6 and KXR6 (VRF) Series air conditioners commercialized by Mitsubishi Heavy Industries.

*RAC Series require optional SC-BIKN-E from MHI.

Application’s Program Version: 1.1

Order Code: **MH-RC-KNX-1i**
### INDEX

1. Presentation .................................................................................................................. 6
2. Connection ................................................................................................................... 7
   2.1 MH-RC-KNX-i without MHI Remote Controller ......................................................... 7
   2.2 MH-RC-KNX-i with MHI Remote Controller .............................................................. 7
3. Configuration and setup ................................................................................................. 9
4. ETS Parameters ............................................................................................................ 10
   4.1 General dialog ........................................................................................................... 11
      4.1.1 MH-RC-KNX-i is master in X Y bus ...................................................................... 11
      4.1.2 Send READs for Control_objects on bus recovery .............................................. 11
      4.1.3 Scene to load on bus recovery / startup ............................................................. 11
      4.1.4 Disallow control from remote controller ......................................................... 12
      4.1.5 Enable func “Control_Lock Control Obj” ........................................................ 12
      4.1.6 Enable use of objects for Filter .......................................................................... 13
      4.1.7 Enable func “Operating Hours Counter” ............................................................ 13
      4.1.8 Enable object “Error Code [2byte]” ................................................................. 13
      4.1.9 Enable object “Error Text Code [14byte]” ........................................................ 14
   4.2 Mode Configuration dialog ...................................................................................... 14
      4.2.1 Indoor unit has AUTO mode .............................................................................. 15
      4.2.2 Enable use of Heat / Cool bit-type obj ............................................................... 15
      4.2.3 Enable PID-Compat. Scaling Mode Objects ....................................................... 15
      4.2.4 Enable use of +/- object for Mode .................................................................... 16
      4.2.5 Enable use of bit-type Mode objects (for control) .............................................. 17
      4.2.6 Enable use of bit-type Mode objects (for status) ................................................ 17
      4.2.7 Enable use of Text object for Mode ................................................................... 17
   4.3 Special Modes Configuration dialog ......................................................................... 18
      4.3.1 Enable use of POWER mode ............................................................................. 18
      4.3.2 Enable use of ECONOMY mode ....................................................................... 19
      4.3.3 Enable use of ADDITIONAL HEATING mode .................................................. 20
      4.3.4 Enable use of ADDITIONAL COOLING mode ................................................ 21
   4.4 Fan Speed Configuration dialog .............................................................................. 21
      4.4.1 Available fanspeeds in Indoor Unit .................................................................... 22
      4.4.2 DPT object type for fanspeed ........................................................................... 22
      4.4.3 Enable use of +/- object for Fan Speed .............................................................. 23
      4.4.4 Enable use of bit-type Fan Speed objects (for Control) ....................................... 24
      4.4.5 Enable use of bit-type Fan Speed objects (for Status) ......................................... 24
      4.4.6 Enable use of Text object for Fan Speed ........................................................... 25
   4.5 Vanes Up-Down Configuration dialog .................................................................... 25
      4.5.1 Indoor unit has U-D Vanes ................................................................................ 26
      4.5.2 DPT object type for Vanes Up-Down ................................................................. 26
      4.5.3 Enable use of +/- object for Vanes U-D ............................................................. 27
      4.5.4 Enable use of bit-type Vane U-D objects (for Control) ......................................... 28
      4.5.5 Enable use of bit-type Vane U-D objects (for Status) .......................................... 28
      4.5.6 Enable “Vanes U-D Swing” objects (for Control and Status) .............................. 29
      4.5.7 Enable use of Text object for Vane U-D ............................................................ 29
   4.6 Temperature Configuration dialog ............................................................................ 30
      4.6.1 Periodic sending of “Status_ AC Setp” ............................................................... 30
      4.6.2 Transmission of “Status_ AC Reference Temp” ................................................ 30
      4.6.3 Enable use of +/- object for Setpoint Temp ...................................................... 31
      4.6.4 Enable limits on Control_Setpoint obj ............................................................. 31
      4.6.5 Ambient temp. ref. is provided from KNX .......................................................... 32
   4.7 Scene Configuration dialog ...................................................................................... 33
      4.7.1 Enable use of scenes .......................................................................................... 33
      4.7.2 Scenes can be stored from KNX bus ................................................................. 34
      4.7.3 Enable use of bit objects for scene execution .................................................... 34

© Intesis Software S.L. - All rights reserved
This information is subject to change without notice
IntesisBox® is a registered trademark of Intesis Software SL
4.7.4 Scene “x” preset .............................................................. 35
4.8 Switch-Off Timeouts Configuration dialog .................................................. 36
  4.8.1 Enable use of Open Window / Switch off timeout function ......................... 37
  4.8.2 Enable use of Occupancy function ....................................................... 38
  4.8.3 Enable use of SLEEP timeout ............................................................ 40
4.9 Binary Input “x” Configuration dialog ......................................................... 41
  4.9.1 Enable use of Input “x” ................................................................. 41
  4.9.2 Contact type ................................................................................. 41
  4.9.3 Debounce time ............................................................................... 42
  4.9.4 Disabling function ........................................................................... 42
  4.9.5 Function ......................................................................................... 42
5. Specifications .................................................................................................. 50
6. AC Unit Types compatibility. ........................................................................... 51
7. Error Codes .................................................................................................... 52
Appendix A – Communication Objects Table ....................................................... 53
1. Presentation

MH-RC-KNX-1i allows a complete and natural integration of MITSUBISHI HEAVY INDUSTRIES air conditioners with KNX control systems.

Compatible with RAC* Series, FD Series, KX6 and KXR6 (VRF) Series air conditioners commercialized by MITSUBISHI HEAVY INDUSTRIES.

Main features:

- Reduced dimensions, quick installation.
- Multiple objects for control and status (bit, byte, characters...) with KNX standard datapoint types.
- Status objects for every control available.
- Timeout for Open Window and Occupancy. Sleep function also available.
- Control of the AC unit based in the ambient temperature read by the own AC unit, or in the ambient temperature read by any KNX thermostat.
- AC unit can be controlled simultaneously by the wired remote control of the AC unit and by KNX.
- Direct connection to the AC indoor units. Up to 16 AC indoor units can be connected to MH-RC-MBS-1, controlling them as one (not individually).
- Total Control and Monitoring of the AC unit from KNX, including monitoring of AC unit’s state of internal variables, running hours counter (for filter maintenance control), and error indication and error code.
- Up to 5 scenes can be saved and executed from KNX, fixing the desired combination of Operation Mode, Set Temperature, Fan Speed, Vane Position and Remote Controller Lock in any moment by using a simple switching.
- Four potential-free binary inputs provide the possibility to integrate many types of external devices. Also configurable from ETS, they can be used for switching, dimming, shutter/blind control, and more
2. Connection

Connection of the MH-RC-KNX-1i to the AC indoor unit

2.1 MH-RC-KNX-1i without MHI Remote Controller

The MH-RC-KNX-1i can be connected directly to the X/Y bus of the indoor unit (no MHI remote controller -RC from now on- also connected in the X/Y bus). If this is the case, MH-RC-KNX-1i must be configured as master (using the ETS software), see connection diagram below.

2.2 MH-RC-KNX-1i with MHI Remote Controller

If a MHI remote controller (RC) is present and connected to the X/Y bus, there are two configuration options:

- **Wired remote control available.** Connect the gateway as Slave in parallel with the wired remote controllers (controller acts as Master).

- **Infrared remote control available.** Connect the gateway as Master in parallel with the infrared remote controller (Infrared receiver acts as Slave).

![Figure 2.1 MHI RC PCB backside, Master/Slave switch](image)

Disconnect mains power from the AC unit and use a 2 wire cable with a diameter of 0.75mm² to 1.25mm² for the connection of MH-RC-KNX-1i, Mitsubishi Heavy Industries remote controller and its corresponding indoor unit. Screw the suitably peeled cable ends in the corresponding X/Y terminals of each device, as summarized in the Figure 2.2.

Maximum X/Y bus length is 600 meter. MHI RC and MHI-RC-KNX-1i are no polarity dependent.

Connection of the MH-RC-KNX-1i to the KNX bus:

Disconnect power of the KNX bus. Connect the MH-RC-KNX-1i to the KNX TP-1 (EIB) bus using the KNX standard connector (red/grey) of the MH-RC-KNX-1i, respect polarity.

Reconnect power of the KNX bus, and mains power of the AC unit.
Connection diagrams:

MH-RC-KNX-1i without MHI RC

MH-RC-KNX-1i with MHI RC

Figure 2.2 MH-RC-KNX-1i connection diagrams
3. Configuration and setup

This is a fully compatible KNX device which must be configured and setup using standard KNX tool ETS.

ETS database for this device can be downloaded from:

http://www.intesis.com/down/eib/MH-RC-KNX-1i.zip

Please consult the README.txt file, located inside the downloaded zip file, to find instructions on how to install the database.

⚠️ Important: Do not forget to select the correct settings of AC indoor unit being connected to the MH-RC-KNX-1i. This is in "Parameters" of the device in ETS.
4. ETS Parameters

When imported to the ETS software for the first time, the gateway shows the following default parameter configuration:

![Default parameter configuration](image)

**Figure 4.1 Default parameter configuration**

With this configuration it’s possible to send On/Off (Control_On/Off), change the AC Mode (Control_Mode), the Fan Speed (Control_Fan Speed) and also the Setpoint Temperature (Control_Setpoint Temperature). The Status_objects, for the mentioned Control_objects, are also available to use if needed. Also objects Status_AC Reference Temp and Status_Error/Alarm are shown.

![Default communication objects](image)

**Figure 4.2 Default communication objects**
4.1 General dialog

Inside this parameter’s dialog it is possible to activate or change the parameters shown in the Figure 4.1.

The first field shows the URL where to download the database and the user manual for the product.

4.1.1 MH-RC-KNX-1i is master in X Y bus

This parameter changes the gateway’s behavior, being able to program it as master or slave in X Y bus.

- If set to “no”, the gateway will work as a slave and it will be necessary to have a BRC remote controller configured as a master.
- If set to “yes” the gateway will be master of the bus. It is not necessary to have any BRC remote controller in this case but, if there are, they must be configured as slaves.

The next parameter is also shown when selecting MH-RC-KNX-1i as master in X Y bus:

![Figure 4.3 Parameter detail](image)

4.1.2 Send READs for Control_ objects on bus recovery

When this parameter is enabled, MH-RC-KNX-1i will send READ telegrams for the group addresses associated on its Control_ objects on bus recovery or application reset/start-up.

- If set to “no” the gateway will not perform any action.
- If set to “yes” all Control_ objects with both Transmit (T) and Update (U) flags enabled will send READs and their values will be updated with the response when received.

![Figure 4.4 Parameter detail](image)

- **Delay before sending READs (sec):**
  
  With this parameter, a delay can be configured between 0 and 30 seconds for the READs sent by the Control_ objects. This is to give time enough to other KNX devices on the bus to start-up before sending the READs.

4.1.3 Scene to load on bus recovery / startup

This parameter executes a selected scene on bus recovery or startup, only if the selected scene has an enabled preset or values previously saved from KNX bus (see Scene Configuration dialog).
If the gateway is disconnected from the indoor unit the scene will not be applied, even when connecting to the indoor unit again.

4.1.4 Disallow control from remote controller

This parameter allows:

1- Having the remote controller always locked, or
2- Decide through a new communication object if the RC is locked or not.

- If set to “yes” all the actions performed through the remote controller will be disabled.
- If set to “no” the remote controller will work as usually. It also appears a new parameter and the communication object Control_ Lock Remote Control.

![Figure 4.5 Parameter detail](image)

---

32 Control_ Lock Remote Control [DPT_1.002 - 1bit] - 0-Unlocked; 1-Locked

Disallow control from remote controller

> Enable comm obj “Ctrl_ Remote Lock”

![Figure 4.6 Communication object and parameter detail](image)

- **Enable comm obj “Ctrl_ Remote Lock”:**
  - If set to “no” the object will not be shown.
  - If set to “yes” the Control_ Lock Remote Control object will appear.
    - When a “1” value is sent to this communication object, the remote controller is locked. To be unlocked a “0” value must be sent. The gateway remembers the last value received even if a KNX bus reset/failure happens.

⚠️ **Important:** If an initial scene is enabled and it has as Value for Remote Lock (unchanged) or unlocked, this would unlock the remote controller because the initial scene has priority over the Control_ Lock Remote Control communication object.

4.1.5 Enable func “Control_ Lock Control Obj”

This parameter shows/hide the Control_ Lock Control Obj communication object which, depending on the sent value, locks or unlocks ALL the Control_ communication objects except itself.

33 Control_ Lock Control Objects [DPT_1.002 - 1bit] - 0-Unlocked; 1-Locked

- If set to “no” the object will not be shown.
- If set to “yes” the Control_ Lock Control Objects object will appear.
4.1.6 Enable use of objects for Filter

This parameter shows/hides Control_ Reset Filter and Status_ Filter Status that lets reset the filter status and also monitor if there is a filter alarm.

- When a "1" value is sent to this communication object, all the Control_ objects will be locked. To unlock a "0" value must be sent, as the gateway remembers the last value received even if a KNX bus reset/failure happens.

4.1.7 Enable func “Operating Hours Counter”

This parameter shows/hides the Status_ Operation Hour Counter communication object which counts the number of operating hours for the MH-RC-KNX-1i.

- When a "0" value is shown in the Status_ object, it indicates no filter alarm. When a "1" value is shown in the Status_ object, it indicates that the filter is full. Once the filter has been cleaned, alarm should be reset sending a "1" value to the Control_ Reset Filter object.

4.1.8 Enable object “Error Code [2byte]”
This parameter shows/hides the Status_ Error Code communication object which shows the indoor unit errors, if occurred, in numeric format.

- If set to “no” the object will not be shown.
  - This object can be read and also sends the indoor unit error, if occurred, in numeric format. If a “0” value is shown that means no error.

4.1.9 Enable object “Error Text Code [14byte]”

This parameter shows/hides the Status_ Error Text Code communication object which shows the indoor unit errors, if occurred, in text format.

- If set to “no” the object will not be shown.
- If set to “yes” the Status_ Error Text Code object will appear.
  - This object can be read and also sends the indoor unit error, if occurred, in text format. The errors shown have the same format as at the remote controller and at the error list from the indoor unit manufacturer. If the object’s value is empty that means no error.

4.2 Mode Configuration dialog

All the parameters in this section are related with the different mode properties and communication objects.

- Control_Mode [DPT_20.105 - 1byte] - 0-Aut;1-Hea;2-Cool;3-Fan;4-Dry
The byte-type communication object for Mode works with the DTP_20.105. Auto mode will be enabled with a "0" value, Heat mode with a "1" value, Cool mode with a "3" value, Fan mode with a "9" value and Dry mode with a "14" value.

4.2.1 Indoor unit has AUTO mode

This parameter has to be used to indicate if the indoor unit has the auto mode available.

- If set to "no", the indoor unit doesn’t have the auto mode available.
- If set to "yes", the indoor unit has the auto mode available.

⚠️ Important: Read the documentation of your indoor unit to check if it has AUTO mode available.

4.2.2 Enable use of Heat / Cool bit-type obj

This parameter shows/hides the Control_ and Status_ Mode Cool/Heat communication objects.

- If set to "no" the objects will not be shown.
- If set to "yes" the Control_ and Status_ Mode Cool/Heat objects will appear.
  - When a "1" value is sent to the Control_ communication object, Heat mode will be enabled in the indoor unit, and the Status_ object will return this value.
  - When a "0" value is sent to the Control_ communication object, Cool mode will be enabled in the indoor unit, and the Status_ object will return this value.

4.2.3 Enable PID-Compat. Scaling Mode Objects

This parameter shows/hides the Control_ Mode Cool & On and Control_ Mode Heat & On communication objects.

- If set to "no" the objects will not be shown.
- If set to "yes" the Control_ Mode Cool & On and Control_ Mode Heat & On objects will appear.
  - These objects provide compatibility with those KNX thermostats that control the demand of heating or cooling by using scaling (percentage) objects. In
these thermostats, the percentage demand is meant to be applied on a fluid valve of the heating / cooling system.

- MH-RC-KNX-1i device does not provide individual control on the internal parts of the indoor unit (as can be its compressor, refrigerant valves, etc). Rather, it provides the same level of control as a (user) remote controller.

- Objects “Control_Mode Cool & On” and “Control_Mode Heat & On” intend to bring compatibility between thermostats oriented to the control of custom heating / cooling systems and ready-made AC indoor units, by applying the following logic:
  - Whenever a non-zero value (>0%) is received at “Control_Mode Cool & On”, indoor unit will switch On in COOL mode.
  - Whenever a non-zero value (>0%) is received at “Control_Mode Heat & On”, indoor unit will switch On in HEAT mode.
  - Lastest updated object will define the operating mode
  - Indoor unit will switch off only when both objects become zero (0%) or when an OFF is requested at object “0. On/Off [DPT_1.001 - 1bit]”

⚠️ Important: These objects function is only to send On/Off and Cool/Heat to the indoor unit. The PID (Inverter system) is calculated by the indoor unit itself. Please consider introducing an appropriate PID configuration to the external KNX thermostat to not interfere the indoor unit PID.

4.2.4 Enable use of +/- object for Mode

This parameter shows/hides the Control_Mode +/- communication object which lets change the indoor unit mode by using two different datapoint types.

- If set to “no” the object will not be shown.
- If set to “yes” the Control_Mode +/- object and a new parameter will appear.

![Figure 4.8 Parameter detail](image)

- **DPT type for +/- Mode Object**

  This parameter lets choose between the datapoints 0-Up / 1-Down [DPT_1.008] and 0-Decrease / 1-Increase [DPT_1.007] for the Control_Mode +/- object.

  The sequence followed when using this object is shown below:
4.2.5 Enable use of bit-type Mode objects (for control)

This parameter shows/hides the bit-type Control_ Mode objects.

- If set to “no” the objects will not be shown.
- If set to “yes” the Control_ Mode objects for Auto, Heat, Cool, Fan and Dry will appear. To activate a mode by using these objects a “1” value has to be sent.

4.2.6 Enable use of bit-type Mode objects (for status)

This parameter shows/hides the bit-type Status_ Mode objects.

- If set to “no” the objects will not be shown.
- If set to “yes” the Status_ Mode objects for Auto, Heat, Cool, Fan and Dry will appear. When enabled, a mode will return a “1” through its bit-type object.

4.2.7 Enable use of Text object for Mode

This parameter shows/hides the Status_ Mode Text communication object.

- If set to “no” the object will not be shown.
If set to "yes" the Status_Mode Text object will appear. Also, in the parameters, will be shown five text fields, one for each mode, that will let modify the text string displayed by the Status_Mode Text when changing mode.

| String when mode is AUTO (if available) | AUTO |
| String when mode is HEAT               | HEAT |
| String when mode is COOL               | COOL |
| String when mode is FAN                | FAN  |
| String when mode is DRY                | DRY  |

**Figure 4.9 Parameter detail**

### 4.3 Special Modes Configuration dialog

The Special Modes can be parameterized through the ETS parameters dialog, and they can be used to give extra functionality.

⚠️ **Important:** When executing any of the Special Modes the real state of the indoor unit will NOT be shown in KNX.

⚠️ **Important:** When the pre-defined time for the Special Mode is finished or a "0" value is sent to stop it, the previous state will be recovered.

⚠️ **Important:** If a value concerning On/Off, Mode, Fan Speed or Setpoint Temperature is received from KNX while any Special Mode is running ("1"), the Special Mode will stop and the previous state will be recovered. The value received will be also applied then.

⚠️ **Important:** If a value concerning On/Off, Mode, Fan Speed or Setpoint Temperature is modified through the remote controller, the Special Mode will stop WITHOUT recovering the previous state. Then the real indoor unit state will be shown in KNX including the new value received through the remote controller.

#### 4.3.1 Enable use of POWER mode
This parameter shows/hides the Control_ Power Mode and Status_ Power Mode communication objects. The Power Mode lets change the Setpoint Temperature and the Fan Speed within a given period of time.

- If set to “no” the objects will not be shown.
- If set to “yes” the Control_ Power Mode and Status_ Power Mode objects and new parameters will appear.

![Table](4.11 Parameter detail)

- When a “1” value is sent to the Control_ communication object Power Mode will be enabled, and the Status_ object will return this value.
- When a “0” value is sent to the Control_ communication object, Power Mode will be disabled, and the Status_ object will return this value.

⚠️ **Important:** This mode will ONLY work if the indoor unit is both turned on and in a Heat, Cool, Auto-Heat or Auto-Cool Mode.

- **Action time for this mode (minutes):**
  Duration of Power Mode, in minutes, once started.

- **Setpoint delta increase (HEAT) or decrease (COOL) – in Celsius:**
  Number of degrees Celsius that will increase in Heat Mode, or decrease in Cool Mode, while in Power Mode.

- **Fan Speed for this mode:**
  Fan Speed that will be set in the unit while in Power Mode.

4.3.2 Enable use of ECONOMY mode

This parameter shows/hides the Control_ Econo Mode and Status_ Econo Mode communication objects. The Econo Mode lets change the Setpoint Temperature and the Fan Speed within a given period of time.
4 Control_ Econo Mode [DPT_1.010 - 1bit] - 0-Stop;1-Start
83 Status_ Econo Mode [DPT_1.001 - 1bit] - 0-Off;1-On

- If set to "no" the objects will not be shown.
- If set to "yes" the Control_ Econo Mode and Status_ Econo Mode objects and new parameters will appear.
  - When a "1" value is sent to the Control_ communication object, EconoMode will be enabled, and the Status_ object will return this value.
  - When a "0" value is sent to the Control_ communication object, EconoMode will be disabled, and the Status_ object will return this value.

⚠️ Important: This mode will ONLY work if the indoor unit is both turned on and in a Heat, Cool, Auto-Heat or Auto-Cool Mode.

- Action time for this mode (minutes):
  Duration of EconoMode, in minutes, once started.

- Setpoint delta increase (HEAT) or decrease (COOL) – in Celsius:
  Number of degrees Celsius that will increase in Heat Mode, or decrease in Cool Mode, while in EconoMode.

- Fan Speed for this mode:
  Fan Speed that will be set in the unit while in EconoMode.

4.3.3 Enable use of ADDITIONAL HEATING mode

This parameter shows/hides the Control_ Additional Heat Mode and Status_ Additional Heat Mode communication objects. The Additional Heating Mode lets change the Setpoint Temperature and the Fan Speed within a given period of time.

- 36 Control_ Additional Heat [DPT_1.010 - 1bit] - 0-Stop;1-Start
- 84 Status_ Additional Heat [DPT_1.001 - 1bit] - 0-Off;1-On

- If set to "no" the objects will not be shown.
- If set to "yes" the Control_ Start Additional Heat Mode and Status_ Additional Heat Mode objects and new parameters will appear.
  - When a "1" value is sent to the Control_ communication object, Additional Heating Mode will be enabled, and the Status_ object will return this value.
  - When a "0" value is sent to the Control_ communication object, Additional Heating Mode will be disabled, and the Status_ object will return this value.
Important: This mode will ALWAYS turn on the indoor unit in Heat mode.

- **Action time for this mode (minutes):**
  Duration of Additional Heating Mode, in minutes, once started.

- **Setpoint temp for this mode (°C):**
  Setpoint temperature that will be applied while in Additional Heating Mode.

- **Fan Speed for this mode:**
  Fan Speed that will be set in the unit while in Additional Heating Mode.

### 4.3.4 Enable use of ADDITIONAL COOLING mode

This parameter shows/hides the Control__Additional Cool Mode and Status__Additional Cool Mode communication objects. The Additional Heating Mode lets change the Setpoint Temperature and the Fan Speed within a given period of time.

- If set to “no” the objects will not be shown.
- If set to “yes” the Control__Start Additional Cool Mode and Status__Additional Cool Mode objects and new parameters will appear.
  - When a “1” value is sent to the Control__ communication object, Additional Cooling Mode will be enabled, and the Status__ object will return this value.
  - When a “0” value is sent to the Control__ communication object, Additional Cooling Mode will be disabled, and the Status__ object will return this value.

Important: This mode will ALWAYS turn on the indoor unit in Cool mode.

- **Action time for this mode (minutes):**
  Duration of Additional Cooling Mode, in minutes, once started.

- **Setpoint temp for this mode (°C):**
  Setpoint temperature that will be applied while in Additional Cooling Mode.

- **Fan Speed for this mode:**
  Fan Speed that will be set in the unit while in Additional Cooling Mode.

### 4.4 Fan Speed Configuration dialog
Figure 4.12 Default Fan Speed Configuration dialog

All the parameters in this section are related with the Fan Speed properties and communication objects.

4.4.1 Available fanspeeds in Indoor Unit

This parameter lets choose how many fan speeds are available in the indoor unit.

![Figure 4.13 Parameter detail](image)

⚠️ **Important:** Read the documentation of your indoor unit to check how many fan speeds are available.

⚠️ **Important:** If “1” fan speed is selected, no Fan Speed communication object will appear in the ETS software.

4.4.2 DPT object type for fanspeed

With this parameter is possible to change the DPT for the Control_ Fan Speed and Status_ Fan Speed byte-type communication objects. Datapoints Scaling (DPT_5.001) and Enumerated (DPT_5.010) can be selected.

⚠️ **Important:** The communication objects shown in this section may be different depending on the number of fan speeds available, although they all share the same communication object number.

- When “Enumerated [DPT 5.010]” is selected, Control_ Fan Speed and Status_ Fan Speed communication objects for this DPT will appear.

  - 11 Control_ Fan Speed / 2 Speeds [DPT_5.010 - 1byte] - Speed values: 1,2
  - 62 Status_ Fan Speed / 2 Speeds [DPT_5.010 - 1byte] - Speed Values: 1,2
The first fan speed will be selected if a “1” is sent to the `Control_` object. The second one will be selected sending a “2”; the third one (if available) will be selected sending a “3”; the fourth one (if available) will be selected sending a “4”.

The `Status_` object will always return the value for the fan speed selected.

⚠️ **Important:** If a “0” value is sent to the `Control_` object, the minimum fan speed will be selected. If a value bigger than “4” is sent to the `Control_` object, then the maximum fan speed will be selected.

- When “Scaling [DPT 5.001]” is selected, `Control_ Fan Speed` and `Status_ Fan Speed` communication objects for this DPT will appear.

  - 11 Control_Fan Speed / 2 Speeds [DPT_5.001 - 1byte] - Threshold: 75%
  - 62 Status_Fan Speed / 2 Speeds [DPT_5.001 - 1byte] - 50% and 100%

The next table shows the range of values that can be sent through the `Control_` object and the value returned by the `Status_` object.

<table>
<thead>
<tr>
<th>Fan Speed 1</th>
<th>Fan Speed 2</th>
<th>Fan Speed 3</th>
<th>Fan Speed 4</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Control_</strong></td>
<td>0% - 74%</td>
<td>75% - 100%</td>
<td></td>
</tr>
<tr>
<td><strong>Status_</strong></td>
<td>50%</td>
<td>100%</td>
<td></td>
</tr>
<tr>
<td><strong>Control_</strong></td>
<td>0% - 49%</td>
<td>50% - 82%</td>
<td>83% - 100%</td>
</tr>
<tr>
<td><strong>Status_</strong></td>
<td>33%</td>
<td>67%</td>
<td>100%</td>
</tr>
<tr>
<td><strong>Control_</strong></td>
<td>0% - 37%</td>
<td>38% - 62%</td>
<td>63% - 87%</td>
</tr>
<tr>
<td><strong>Status_</strong></td>
<td>25%</td>
<td>50%</td>
<td>75%</td>
</tr>
</tbody>
</table>

⚠️ **Important:** Read the documentation of your indoor unit to check how many fan speeds are available.

4.4.3 Enable use of +/- object for Fan Speed

This parameter shows/hides the `Control_ Fan Speed` +/- communication object which lets increase/decrease the indoor unit fan speed by using two different datapoint types.

- 16 Control_Fan Speed +/- [DPT_1.007 - 1bit] - 0=Decrease; 1=Increase

- If set to “no” the object will not be shown.

- If set to “yes” the `Control_ Fan Speed` +/- object and a new parameter will appear.
### 4.4.4 Enable use of bit-type Fan Speed objects (for Control)

This parameter shows/hides the bit-type Control_ Fan Speed objects.

- 12 Control_Fan Speed 1 [DPT_1.002 - 1bit] - 1-Set Fan Speed 1
- 13 Control_Fan Speed 2 [DPT_1.002 - 1bit] - 1-Set Fan Speed 2
- 14 Control_Fan Speed 3 [DPT_1.002 - 1bit] - 1-Set Fan Speed 3
- 15 Control_Fan Speed 4 [DPT_1.002 - 1bit] - 1-Set Fan Speed 4

- If set to “no” the objects will not be shown.
- If set to “yes” the Control_ Fan Speed objects for Speed 1, Speed 2 (if available), Speed 3 (if available), and Speed 4 (if available) will appear. To activate a Fan Speed by using these objects a “1” value has to be sent.

### 4.4.5 Enable use of bit-type Fan Speed objects (for Status)

This parameter shows/hides the bit-type Status_ Fan Speed objects.
If set to "no" the objects will not be shown.

If set to "yes" the Status_Fan Speed objects for Speed 1, Speed 2 (if available), Speed 3 (if available), and Speed 4 (if available) will appear. When a Fan Speed is enabled, a "1" value is returned through its bit-type object.

4.4.6 Enable use of Text object for Fan Speed

This parameter shows/hides the Status_Fan Speed Text communication object.

If set to "no" the object will not be shown.

If set to "yes" the Status_Fan Speed Text object will appear. Also, in the parameters, will be shown four text fields, one for each Fan Speed, that will let modify the text string displayed by the Status_Fan Speed Text when changing a fan speed.

| String when fan speed is 1               | SPEED 1 |
| String when fan speed is 2 (if available) | SPEED 2 |
| String when fan speed is 3 (if available) | SPEED 3 |
| String when fan speed is 4 (if available) | SPEED 4 |

**Figure 4.15** Parameter detail

4.5 Vanes Up-Down Configuration dialog
All the parameters in this section are related with the Vane Up-Down properties and communication objects.

### 4.5.1 Indoor unit has U-D Vanes

This parameter lets choose if the unit has Up-Down Vanes available or not.

- If set to “no” all the parameters and communication objects for the Up-Down Vanes will not be shown.
- If set to “yes” all the parameters and communication objects (if enabled in the parameters dialog) for the Up-Down Vanes will be shown.

⚠️ **Important:** Read the documentation of your indoor unit to check if Up-Down Vanes are available.

### 4.5.2 DPT object type for Vane Up-Down

With this parameter is possible to change the DPT for the Control_Vanes U-D and Status_Vanes U-D byte-type communication objects. Datapoints Scaling (DPT_5.001) and Enumerated (DPT_5.010) can be selected.

⚠️ **Important:** The communication objects shown in this section may be different depending on the number of vanes position available, although they all share the same communication object number.
When "Enumerated [DPT 5.010]" is selected, Control_ Vanes U-D and Status_ Vanes U-D communication objects for this DPT will appear.

- 17 Control_ Vanes U-D / 4 Pos [DPT 5.010 - 1byte] - Position values: 1, 2, 3, 4
- 68 Status_ Vanes U-D / 4 Pos [DPT 5.010 - 1byte] - Position values: 1, 2, 3, 4

To choose a vanes position, values from "1" to "4" can be sent to the Control_ object. Each value will correspond to the position (i.e. Value "3" = Position 3).

The Status_ object will always return the value for the vane position selected.

⚠️ Important: If a "0" value is sent to the Control_ object, the Position 1 will be selected. If a value bigger than "4" is sent to the Control_ object, then the higher Position will be selected.

When "Scaling [DPT 5.001]" is selected, Control_ Vanes U-D and Status_ Vanes U-D communication objects for this DPT will appear.

- 17 Control_ Vanes U-D / 4 Pcs [DPT 5.001 - 1byte] - Thresholds: 38%, 63% and 88%
- 68 Status_ Vanes U-D / 4 Pcs [DPT 5.001 - 1byte] - 25%, 50%, 75% and 100%

The next table shows the range of values that can be sent through the Control_ object and the value returned by the Status_ object.

<table>
<thead>
<tr>
<th>Vanes Pos.1</th>
<th>Vanes Pos.2</th>
<th>Vanes Pos.3</th>
<th>Vanes Pos.4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Control_</td>
<td>0% - 37%</td>
<td>38% - 62%</td>
<td>63% - 87%</td>
</tr>
<tr>
<td>Status_</td>
<td>25%</td>
<td>50%</td>
<td>75%</td>
</tr>
</tbody>
</table>

4.5.3 Enable use of +/- object for Vanes U-D

This parameter shows/hides the Control_ Vanes U-D -/+ communication object which lets change the indoor unit vane position by using two different datapoint types.

- 23 Control_ Vanes U-D -/+ [DPT 1.007 - 1bit] - 0-Decrease; 1-Increase

- If set to "no" the object will not be shown.
- If set to "yes" the Control_ Vanes U-D -/+ object and a new parameter will appear.

<table>
<thead>
<tr>
<th>Enable use of +/- object for Vanes U-D</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>&gt; DPT type for +/- Vanes U-D object</td>
<td>No-Decrease / 1-Increase [DPT_1.007]</td>
</tr>
<tr>
<td>&gt; Does +/- sequence include SWING vanes Up-Down?</td>
<td>No</td>
</tr>
<tr>
<td>&gt; Roll over Vanes at upper/lower limit (when controlling with +/- obj)</td>
<td>No</td>
</tr>
</tbody>
</table>

Figure 4.18 Parameter detail
- **DPT type for +/- Vanes U-D object**

  This parameter lets choose between the datapoints **0-Up / 1-Down [DPT_1.008]** and **0-Decrease / 1-Increase [DPT_1.007]** for the **Control_ Vanes U-D -/+** object.

- **Does +/- sequence include SWING vanes Up-Down?**

  This parameter lets choose if SWING function is included (**“yes”**) or not (**“no”**) in the sequence when using **Control_ Vanes U-D -/+** object as shown in the discontinuous segment at the picture below.

- **Roll over Vanes at upper/lower limit**

  This parameter lets choose if roll-over will be enabled (**“yes”**) or disabled (**“no”**) for the **Control_ Vanes U-D -/+** object.

  ![Diagram of roll over Vanes at upper/lower limit](image)

  **Only if Roll-over is enabled**

  - Up / Increase
  - Down / Decrease
  - * If Available

4.5.4 **Enable use of bit-type Vane U-D objects (for Control)**

This parameter shows/hides the bit-type **Control_ Vanes U-D** objects.

- 18 Control_Vanes U-D Pos 1 [DPT_1.002 - 1bit] - 1-Set Position 1
- 19 Control_Vanes U-D Pos 2 [DPT_1.002 - 1bit] - 1-Set Position 2
- 20 Control_Vanes U-D Pos 3 [DPT_1.002 - 1bit] - 1-Set Position 3
- 21 Control_Vanes U-D Pos 4 [DPT_1.002 - 1bit] - 1-Set Position 4

  - If set to **“no”** the objects will not be shown.
  - If set to **“yes”** the **Control_ Vanes U-D** objects for each Position will appear. To activate a Vanes Position by using these objects, a “1” value has to be sent.

4.5.5 **Enable use of bit-type Vane U-D objects (for Status)**

This parameter shows/hides the bit-type **Status_ Vanes U-D** objects.
4.5.6 Enable “Vanes U-D Swing” objects (for Control and Status)

This parameter shows/hides the Control_ Vanes U-D Swing and Status_ Vanes U-D Swing communication objects.

- If set to “no” the objects will not be shown.
- If set to “yes” the Control_ Vanes U-D Swing and Status_ Vanes U-D Swing objects will appear.
  - When a “1” value is sent to the Control_ communication object, Vanes Up-Down will be in Swing mode, and the Status_ object will return this value.
  - When a “0” value is sent to the Control_ communication object, Vanes Up-Down will stop Swing mode. The Status_ object will return this value.

4.5.7 Enable use of Text object for Vane U-D

This parameter shows/hides the Status_ Vanes U-D Text communication object.

- If set to “no” the object will not be shown.
- If set to “yes” the Status_ Vanes U-D Text object will appear. Also, in the parameters will be shown five text fields, four for the Vane Position and one for the Swing function that will let modify the text string displayed by the Status_ Vanes U-D Text when changing a vane position.
4.6 Temperature Configuration dialog

All the parameters in this section are related with the Temperature properties and communication objects.

4.6.1 Periodic sending of “Status_ AC Setp”

This parameter lets change the interval of time (in seconds, from 0 to 255) at the end of which the AC setpoint temperature is sent to the KNX bus. For a “0” value, the AC setpoint temperature will ONLY be sent on change. The AC setpoint temperature is sent through the communication object Status_ AC Setpoint Temp.

Figure 4.21 Parameter detail

⚠️ Important: In case the ambient temperature is provided from KNX, the setpoint temperature returned from this object, will be the one resulting from the formula shown in the section “4.6.5 Ambient temp. ref. is provided from KNX”.

4.6.2 Transmission of “Status_ AC Reference Temp”

This parameter lets to you choose if the AC return temperature will be sent “only cyclically”, “only on change” or “cyclically and on change”. The AC reference temperature is sent through the communication object Status_ AC Reference Temp.
4.6.3 Enable use of +/- object for Setpoint Temp

This parameter shows/hides the Control_Setpoint Temp -/+ communication object which lets change the indoor unit setpoint temperature by using two different datapoint types.

- If set to “no” the object will not be shown.
- If set to “yes” the Control_Setpoint Temp -/+ object and a new parameter will appear.

4.6.4 Enable limits on Control_Setpoint obj

This parameter enables to define temperature limits for the Control_Setpoint Temperature object.

- DPT type for +/- Setp Temp object
  - This parameter lets choose between the datapoints 0-Up / 1-Down [DPT_1.008] and 0-Decrease / 1-Increase [DPT_1.007] for the Control_Setpoint Temp -/+ object.

  (Lower limit) 20ºC  21ºC  ...  29ºC  30ºC (Upper limit)

  ▪ Up / Increase
  ▪ Down / Decrease
If set to “no” the setpoint temperature limits for the Control_ Setpoint Temperature object will be the default: 18°C for the lower limit and 30°C for the upper limit.

If set to “yes” it is possible to define temperature limits for the Control_ Setpoint Temperature object.

- **Lower limit (°C)**
  
  This parameter lets to define the lower limit for the setpoint temperature.

- **Upper limit (°C)**
  
  This parameter lets to define the upper limit for the setpoint temperature.

⚠️ **Important:** If a setpoint temperature above the upper defined limit (or below the lower defined limit) is sent through the Control_ Setpoint Temperature object, it will be ALWAYS applied the limit defined.

⚠️ **Important:** When limits are enabled, any setpoint temperature sent to the AC (even through scenes, special modes, etc.) will be limited.

### 4.6.5 Ambient temp. ref. is provided from KNX

This parameter shows/hides the Control_ Ambient Temperature communication object which lets use an ambient temperature reference provided by a KNX device.

![Control_Ambient Temperature](DPT_9.001 - 2byte - °C)

- If set to “no” the object will not be shown.

- If set to “yes” the Control_ Ambient Temperature object will appear. Meant to be enabled when you want the temperature provided by a KNX sensor to be the reference ambient temperature for the air conditioner. Then, the following formula applies for calculation of real Control_ Setpoint Temperature sent to the AC unit:

  \[
  \text{“AC Setp. Temp”} = \text{“AC Ret. Temp”} - (\text{“KNX Amb. Temp.”} - \text{“KNX Setp. Temp”})
  \]

  - AC Setp. Temp: AC indoor unit setpoint temperature
  - AC Ret. Temp: AC indoor unit return temperature
  - KNX Amb. Temp.: Ambient temperature provided from KNX
  - KNX Setp. Temp: Setpoint temperature provided from KNX

As an example, consider the following situation:

**User wants:** **19°C** (“KNX Setp. Temp.”)

**User sensor (a KNX sensor) reads:** **21°C** (“KNX Amb Temp.”)

**Ambient temp. read by MHI system is:** **24°C** (“AC Ret. Temp”)

---

© Intesis Software S.L. - All rights reserved
This information is subject to change without notice
IntesisBox® is a registered trademark of Intesis Software SL

URL | http://www.intesis.com
---

Email | info@intesis.com
---

tel | +34 938047134
In this example, the final setpoint temperature that MH-RC-KNX-1i will send out to the indoor unit (shown in “Setp. Temp.”) will become 24ºC – (21ºC - 19ºC) = 22ºC. This is the setpoint that will actually be requested to MHI unit.

This formula will be applied as soon as the Control_Setpoint Temperature and Control_Ambient Temperature objects are written at least once from the KNX installation. After that, they are kept always consistent.

Note that this formula will always drive the AC indoor unit demand in the right direction, regardless of the operation mode (Heat, Cool or Auto).

### 4.7 Scene Configuration dialog

All the parameters in this section are related with the Scene properties and communication objects. A scene contains values of: On/Off, Mode, Fan speed, Vane position, Setpoint Temperature and Remote Controller Disablement.

#### 4.7.1 Enable use of scenes

This parameter shows/hides the scene configuration parameters and communication objects.

- If set to “no” the scene parameters and communication objects will not be shown.
- If set to “yes” the scene parameters and communication objects will be shown. To execute a scene through the byte-type object, a value from “0” to “4” has to be sent, corresponding each one to a different scene (i.e. “0” = Scene 1; ... “4” = Scene 5).
4.7.2 Scenes can be stored from KNX bus

This parameter shows/hides the Control_Save/Exec Scene and all the Control_Store Scene (if enabled) communication objects.

- [ ] 38 Control_Save/Exec Scene [DPT_18.001 - 1byte] - 0.4-Exec1-5:128..132-Save1-5
- [ ] 39 Control_Store Scene 1 [DPT_1002 - 1bit] - 1-Store Scene 1
- [ ] 40 Control_Store Scene 2 [DPT_1002 - 1bit] - 1-Store Scene 2
- [ ] 41 Control_Store Scene 3 [DPT_1002 - 1bit] - 1-Store Scene 3
- [ ] 42 Control_Store Scene 4 [DPT_1002 - 1bit] - 1-Store Scene 4
- [ ] 43 Control_Store Scene 5 [DPT_1002 - 1bit] - 1-Store Scene 5

- If set to "no" the communication objects will not be shown.
- If set to "yes" the communication objects and a new parameter will appear. To store a scene through the byte-type object, a value from "128" to "132" has to be sent to the object, corresponding each one to a different scene (i.e. "128" = Scene 1;... "132" = Scene 5).

Figure 4.27 Parameter detail

- Enable use of bit objects for storing scenes (from bus)

  If set to "no" the objects will not be shown.
  If set to "yes" the Control_Store Scene objects for storing scenes will appear. To store a scene by using these objects, a "1" value has to be sent to the scene’s object we want to store (i.e. to store scene 4, a "1" has to be sent to the Control_Store Scene 4 object).

4.7.3 Enable use of bit objects for scene execution

This parameter shows/hides the Control_Execute Scene bit-type communication objects.

- [ ] 44 Control_Execute Scene 1 [DPT_1002 - 1bit] - 1-Execute Scene 1
- [ ] 45 Control_Execute Scene 2 [DPT_1002 - 1bit] - 1-Execute Scene 2
- [ ] 46 Control_Execute Scene 3 [DPT_1002 - 1bit] - 1-Execute Scene 3
- [ ] 47 Control_Execute Scene 4 [DPT_1002 - 1bit] - 1-Execute Scene 4
- [ ] 48 Control_Execute Scene 5 [DPT_1002 - 1bit] - 1-Execute Scene 5

Figure 4.28 Parameter detail

- Enable use of bit objects for scene execution

  If set to "no" the communication objects will not be shown.
4.7.4 Scene “x” preset

This parameter lets define a preset for a scene (the following description is valid for all the scenes).

- If set to “yes” the communication objects will appear. To execute a scene by using these objects, a “1” value has to be sent to the scene’s object we want to execute (i.e. to execute scene 4, a “1” has to be sent to the Control_Execute Scene 4 object).

- If set to “no” the preset for the scene “x” will be disabled.

- If set to “yes” the preset will be enabled. When a scene is executed the values configured in the preset will be applied.

⚠️ **Important:** If a scene’s preset is enabled, will not be possible to modify (store) the scene from the KNX bus.

**Figure 4.29 Parameter detail**

<table>
<thead>
<tr>
<th>Scene 1 preset</th>
<th>yes (scene will NOT be modifiable from KNX bus)</th>
</tr>
</thead>
</table>

**Figure 4.30 Parameter detail**

- **Scene “x” / Value for On-Off**

  This parameter lets choose the power of the indoor unit when the scene is executed. The following options are available: “ON”, “OFF” or “(unchanged)”.

- **Scene “x” / Value for Mode**

  This parameter lets choose the mode of the indoor unit when the scene is executed. The following options are available: “AUTO (if available)”, “HEAT”, “COOL”, “FAN”, “DRY”, or “(unchanged)”.

- **Scene “x” / Value for Fan Speed (if available)**

  This parameter lets choose the fan speed of the indoor unit when the scene is executed. The following options are available: “FAN SPEED 1”, “FAN SPEED 2 (if available)”, “FAN SPEED 3 (if available)”, “FAN SPEED 4 (if available)”, or “(unchanged)”.

- **Scene “x” / Value for Vane U-D (if available)**

  This parameter lets choose the vane U-D direction of the indoor unit when the scene is executed. The following options are available: “VANE U-D SWING” or “(unchanged)”.

- **Scene “x” / Value for Setpoint Temp**

  This parameter lets choose the setpoint temperature of the indoor unit when the scene is executed. The following options are available: “22.0 °C” (if available) or “(unchanged)”.

- **Scene “x” / Value for Remote Lock**

  This parameter lets choose the remote lock status of the indoor unit when the scene is executed. The following options are available: “(unchanged)”.
This parameter lets choose the vane position of the indoor unit when the scene is executed. The following options are available: "VANES U-D POS 1", "VANES U-D POS 2", "VANES U-D POS 3", "VANES U-D POS 4", "VANES U-D SWING" or "(unchanged)".

- **Scene “x” / Value for Setp Temp (ºC)**

This parameter lets choose the setpoint temperature of the indoor unit when the scene is executed. The following options are available: from "18ºC" to "30ºC" (both included), or "(unchanged)".

- **Scene “x” / Value for Remote Lock**

This parameter lets choose the remote controller status of the indoor unit when the scene is executed. The following options are available: "locked", "unlocked", or "(unchanged)".

**Important**: If any preset value is configured as "(unchanged)", the execution of this scene will not change current status of this feature in the AC unit.

**Important**: When a scene is executed, Status_Current Scene object shows the number of this scene. Any change in previous items does Status_Current Scene show "No Scene". Only changes on items marked as "(unchanged)" will not disable current scene.

### 4.8 Switch-Off Timeouts Configuration dialog

![Figure 4.31 Default Switch-Off Timeouts Configuration dialog](image)

All the parameters in this section are related with the timeout properties and communication objects.
4.8.1 Enable use of Open Window / Switch off timeout function

This parameter shows/hides the Control_ Switch Off Timeout communication object which lets Start/Stop a timeout to switch off the indoor unit.

- If set to “no” the object will not be shown.
- If set to “yes” the Control_ Switch Off Timeout object and new parameters will appear. If a “1” value is sent to this object, and the indoor unit is already turned on, the switch-off timeout will begin. If a “0” value is sent to this object, the switch-off timeout will stop.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable use of Open Window / Switch off timeout function</td>
<td>Yes/No</td>
</tr>
<tr>
<td>AC switch-off timeout (min)</td>
<td></td>
</tr>
<tr>
<td>DPT for Window / Switch-off timeout</td>
<td></td>
</tr>
<tr>
<td>Reload last On/Off val once window is closed</td>
<td></td>
</tr>
<tr>
<td>Disallow On/Off operation while window is Open</td>
<td>Yes/No</td>
</tr>
</tbody>
</table>

*Figure 4.32 Parameter detail*

- **AC switch-off timeout (min)**
  This parameter lets select how much time (in minutes) to wait before switching off the indoor unit.

- **DPT for Window / Switch-off timeout**
  This parameter lets choose between the datapoints 0-Open / 1-Closed Window [DPT_1.009] and 0-Stop / 1-Start Timeout [DPT_1.010] for the Control_ Switch Off Timeout.

- **Disallow On/Off operation while window is Open**
  If set to “no”, On/Off commands while the window is open will be accepted.
    - If a “1” value is sent to the Control_ Switch Off Timeout object the switch-off timeout period will begin again.
    - If a “0” value is sent to the Control_ Switch Off Timeout object, no action will be performed.
  If set to “yes”, On/Off commands, while the window is open, will be saved (but not applied). These commands will be used in the next parameter if set to “yes”.
➢ **Reload last On/Off value once window is closed?**

If set to “no”, once the switch-off timeout is stopped, any value will be reloaded.

If set to “yes”, once the switch-off timeout is stopped, the last On/Off value sent will be reloaded.

- If a “1” value is sent to the *Control_ Switch Off Timeout* object after the timeout period, the indoor unit will **turn on**.
- If a “0” value is sent to the *Control_ Switch Off Timeout* after the timeout period, no action will be performed.

### 4.8.2 Enable use of Occupancy function

This parameter shows/hides the *Control_ Occupancy* communication object which lets apply different parameters to the indoor unit depending on the presence/no presence in the room.

- If set to “no” the object will not be shown.
- If set to “yes” the *Control_ Occupancy* object and new parameters will appear. If a “1” value is sent to this object (no room occupancy), the timeout will begin. If a “0” value is sent to this object, the timeout will stop.

```
30 Control_Occupancy [DPT 1.018 - 1bit] - 0-Not Occupied; 1-Occupied
```

**Figure 4.33 Parameter detail**

➢ **Timeout to apply action (minutes)**

This parameter lets choose how much time to wait (in minutes) before executing the action specified in the next parameter (“Action after timeout elapsed”).

➢ **Action after timeout elapsed**

When **Switch-Off** is selected, once the timeout has elapsed, the indoor unit will be turned off.

When **Apply Preset Delta** is selected, once the timeout has elapsed, a delta temperature will be applied in order to save energy (decreasing the setpoint when in Heat mode, or increasing the setpoint when in Cool mode). Also new parameters will appear.

```
> Temp delta decrease (HEAT) or increase (COOL) (°C)
  2.0°C

> Enable secondary timeout
  yes
```

**Figure 4.34 Parameter detail**
➢ **Temp delta decrease (HEAT) or increase (COOL) (°C)**

This parameter lets configure the delta temperature (increase or decrease) that will be applied when the timeout has elapsed.

**Important:** When there is occupancy again after the application of a delta, the same delta will be applied inversely. (i.e. In a room with AC in cool mode and 25°C setpoint temperature, a +2°C delta is applied after the occupancy timeout, setting the setpoint at 27°C because there is no occupancy in the room. If the setpoint is raised to 29°C during that period, when the room is occupied again, a -2°C delta will be applied and the final setpoint temperature will then be 27°C).

➢ **Enable secondary timeout**

If set to **“no”** nothing will be applied.

If set to **“yes”**, a new timeout will be enabled and two new parameters will appear.

![Figure 4.35 Parameter detail](image)

✈ **Timeout to apply action (minutes)**

This parameter lets choose how much time to wait (in minutes) before executing the action specified in the next parameter ("Action after timeout elapsed").

✈ **Action after timeout elapsed**

When **Switch-Off** is selected, once the timeout has elapsed, the indoor unit will turn off.

When **Apply Preset Delta** is selected, once the timeout configured is extinguished, a delta temperature will be applied (decreasing the setpoint when in Heat mode, or increasing the setpoint when in Cool mode). Also new parameters will appear.

✈ **Temp delta decrease (HEAT) or increase (COOL) (°C)**

This parameter lets configure the delta temperature that will be applied when the timeout is extinguished.
**Important:** When there is occupancy again after the application of a delta, the same delta will be applied inversely as explained above.

- **Disallow On/Off operation while not Occupied**

  If set to “no”, On/Off commands while the window is open will be accepted.

  - If a “1” value is sent to the Control_ Occupancy object the switch-off timeout period will begin again.
  - If a “0” value is sent to the Control_ Occupancy object, no action will be performed.

  If set to “yes”, On/Off commands while not occupied will be saved (but not applied). These commands will be used in the next parameter if set to “Yes”.

  ![Parameter detail](image)

- **Reload last On/Off value when Occupied**

  If set to “no”, once the switch-off timeout has elapsed, any value will be reloaded.

  If set to “yes”, once the switch-off timeout has elapsed, the last On/Off value will be reloaded.

  - If a “1” value is sent to the Control_ Occupancy object after the timeout period, the indoor unit will turn on.
  - If a “0” value is sent to the Control_ Occupancy after the timeout period no action will be performed.

4.8.3 Enable use of SLEEP timeout

This parameter shows/hides the Control_ Sleep Timeout communication object which lets start a timeout to automatically turn off the indoor unit.

![Credit](image)

- If set to “no” the object will not be shown.

- If set to “yes” the Control_ Sleep Timeout object and a new parameter will appear. If a “1” value is sent to this object the switch-off timeout will begin. If a “0” value is sent to this object, the switch-off timeout will stop.
4.9 Binary Input “x” Configuration dialog

All the parameters in this section are related with the binary inputs properties and communication objects.

4.9.1 Enable use of Input “x”

This parameter enables the use of the Input “x” and shows/hides the Status_Inx communication object(s) which will act as configured in the “Function” parameter.

- If set to “no” the objects will not be shown.
- If set to “yes” the Status_Inx object(s) and new parameters will appear.

4.9.2 Contact type

This parameter lets choose the behavior that will have the binary input depending on if the contact is normally open or normally closed.
There are two possible options to configure the contact type: "NO: Normally Open" and "NC: Normally Closed".

4.9.3 Debounce time

This parameter lets choose a debounce time (in milliseconds) that will be applied to the contact.

4.9.4 Disabling function

This parameter shows/hides the Control Disable Input x communication object which will let disable/enable the input x.

- If set to "no" any object will be shown.
- When "DPT 1.003: 0-Disable; 1-Enable" is selected, the input can be disabled using the value "0" and enabled using the value "1".
- When "DPT 1.002: 1-True (Disable); 0-False (Enable)" is selected, the input can be disabled using the value "1" and enabled using the value "0".

4.9.5 Function

This parameter lets choose the function that will have the binary input. There are 7 different functions available: Switching, Dimming, Shutter/Blind, Value, Execute Scene (internal), Occupancy (internal) and Window Contact (internal).

- When "Switching" is selected the communication object and new parameters for the Input "x" will appear as shown below.

  ![Figure 4.39 Parameter detail](image)

  ➢ Send telegram after bus recovery
This parameter lets select if the Binary Input “x” will send a telegram, or not, after a bus recovery, and the type of telegram sent (if enabled).

- When “No action” is selected, no telegram will be sent after a bus recovery.
- When “Current status” is selected, the binary input will send a telegram with its current status after a bus recovery. Also a new parameter will appear (see below).
- When “On” is selected, the binary input will send a telegram with a “1” value after a bus recovery. Also a new parameter will appear (see below).
- When “Off” is selected, the binary input will send a telegram with a “0” value after a bus recovery. Also a new parameter will appear (see below).

![Sending delay after bus recovery (seconds)](image)

**Figure 4.40 Parameter detail**

- **Sending delay after a bus recovery (seconds)**
  This parameter lets configure a delay (in seconds) that will be applied after a bus recovery and, after which, a telegram will be sent.

- **Value on rising edge**
  This parameter lets select the value that the Binary Input “x” will send on a rising edge (contact activated).
  - When “On” is selected, the binary input will always send telegrams with a “1” value.
  - When “Off” is selected, the binary input will always send telegrams with a “0” value.
  - When “Toggle (On/Off)” is selected, the binary input will send a “1” value after a “0” value and vice versa.
  - When “No action” is selected, the binary input will not perform any action.

- **Value on falling edge**
  This parameter lets select the value that the Binary Input “x” will send on a falling edge (contact deactivated).
  - When “On” is selected, the binary input will always send telegrams with a “1” value.
  - When “Off” is selected, the binary input will always send telegrams with a “0” value.
• When “Toggle (On/Off)” is selected, the binary input will send a “1” value after a “0” value and viceversa.

• When “No action” is selected, the binary input will not perform any action.

➤ Cyclic sending

This parameter lets enable/disable cyclical sending when a determined condition is met.

• When “When output value is On” is selected, everytime a “1” value is sent, it will be sent cyclically. Also a new parameter will appear (see below).

• When “When output value is Off” is selected, everytime a “0” value is sent, it will be sent cyclically. Also a new parameter will appear (see below).

• When “Always” is selected, the binary input will send any value cyclically. Also a new parameter will appear (see below).

• When “Never” is selected, cyclical sending will be disabled.

➤ Period for cyclical sending (seconds)

This parameter lets configure a time (in seconds) for the cyclical sending.

<table>
<thead>
<tr>
<th>Period for cyclical sending (seconds)</th>
<th>2</th>
</tr>
</thead>
</table>

**Figure 4.41** Parameter detail

• When “Dimming” is selected the communication objects and new parameters for the input “x” will appear as shown below.

> 90 Status_In2 - Dimming - On/Off [DPT_1.001 - 1bit] - 0-Off;1-On

> 91 Status_In2 - Dimming - Step(%) [DPT_3.007 - 4bit] - Dimming step

<table>
<thead>
<tr>
<th>Function</th>
<th>Dimming</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>No action</td>
</tr>
<tr>
<td>Mode for short (long) operation</td>
<td>Toggle: On/Off (increase/decrease)</td>
</tr>
<tr>
<td>Increasing step</td>
<td>+ 100 %</td>
</tr>
<tr>
<td>Decreasing step</td>
<td>- 100 %</td>
</tr>
<tr>
<td>Short/long operation limit (x100ms)</td>
<td>10</td>
</tr>
<tr>
<td>Cyclic sending period (x100ms) (0-No cyclical sending)</td>
<td>0</td>
</tr>
</tbody>
</table>

**Figure 4.42** Parameter detail

➤ Send telegram after bus recovery
This parameter lets select if the Binary Input “x” will send a telegram, or not, after a bus recovery, and the type of telegram sent (if enabled).

- When “No action” is selected, no telegram will be sent after a bus recovery.
- When “On” is selected, the binary input will send a telegram with a “1” value after a bus recovery. Also a new parameter will appear (see below).
- When “Off” is selected, the binary input will send a telegram with a “0” value after a bus recovery. Also a new parameter will appear (see below).

![Sending delay after bus recovery (seconds)](image)

**Figure 4.43 Parameter detail**

- **Sending delay after a bus recovery (seconds)**

  This parameter lets configure a delay (in seconds) that will be applied after a bus recovery and, after which, a telegram will be sent.

- **Mode for short (long) operation**

  This parameter lets select the value that the Binary Input “x” will send on a rising edge (contact activated), for a short and a long operation.

  - When “On (increase)” is selected, the binary input will always send telegrams with a “1” value for a short operation, and an “increase step” for a long operation.
  - When “Off (decrease)” is selected, the binary input will always send telegrams with a “0” value for a short operation, and an “decrease step” for a long operation.
  - When “Toggle: On/Off (increase/decrease)” is selected:
    o For the short operation the binary input will send a “1” value after a “0” value and viceversa.
    o For the long operation the binary input will send an “increase step” after a “decrease step” and viceversa.

  **Important:** Note that the first long operation in toggle depends on the last short operation, meaning that after a “1” value will be sent a “decrease step” and after a “0” value will be sent an “increase step”.

  **Important:** The time period between a short and a long operation is defined in the parameter “Short/long operation limit (x100ms)”. 
Increasing step

This parameter lets select the increasing step value (in %) that will be sent for a long operation.

Decreasing step

This parameter lets select the decreasing step value (in %) that will be sent for a long operation.

Short/long operation limit (x100ms)

This parameter lets introduce the time period difference for the short and the long operation.

Cycl. send. period in long oper. (x100ms)

This parameter lets configure a time (in seconds) for the cyclical sending of a long operation.

- When “Shutter/Blind” is selected the communication objects and new parameters for the Input “x” will appear as shown below.

```
92 Status_In3 - Shut/Blind - Step [DPT_1.023 - 1bit] - 0-Step Up;1-Step Down
93 Status_In3 - Shut/Blind - Move [DPT_1.023 - 1bit] - 0-Move Up;1-Move Down
```

<table>
<thead>
<tr>
<th>&gt; Function</th>
<th>Shutter/Blind</th>
</tr>
</thead>
<tbody>
<tr>
<td>&gt; Send telegram after bus recovery</td>
<td>No action</td>
</tr>
<tr>
<td>&gt; Operation</td>
<td>Toggle (Up/Down)</td>
</tr>
<tr>
<td>&gt; Method</td>
<td>Step-Move-Step</td>
</tr>
<tr>
<td>&gt; Short/long operation limit (x100ms)</td>
<td>10</td>
</tr>
<tr>
<td>&gt; Vane adjustment time (x100ms)</td>
<td>10</td>
</tr>
</tbody>
</table>

**Figure 4.44 Parameter detail**

Send telegram after bus recovery

This parameter lets select if the Binary Input “x” will send a telegram, or not, after a bus recovery and the type of telegram sent (if enabled).

- When "No action" is selected, no telegram will be sent after a bus recovery.

- When "Move Up" is selected, the binary input will send a telegram with a "0" value after a bus recovery. Also a new parameter will appear (see below).

- When "Move Down" is selected, the binary input will send a telegram with a "1" value after a bus recovery. Also a new parameter will appear (see below).
Sending delay after a bus recovery (seconds)

This parameter lets configure a delay (in seconds) that will be applied after a bus recovery and, after which, a telegram will be sent.

Operation

This parameter lets select the value that the Binary Input “x” will send on a rising edge (contact activated).

- When “Up” is selected, the binary input will always send telegrams with a “0”.
- When “Down” is selected, the binary input will always send telegrams with a “1” value.
- When “Toggle (Up/Down)” is selected the binary input will send a “0” value after a “1” value and viceversa.

Method

This parameter lets select the working method for the shutter/blind.

- When “Step-Move-Step” is selected: On a rising edge (contact activated) a step/stop telegram will be sent and will begin a time called \( T_1 \). If a falling edge occurs (contact deactivated) during the \( T_1 \), no action will be performed. If the rising edge is maintained longer than \( T_1 \), a move telegram will be sent and will start a time called \( T_2 \). If a falling edge occurs during the \( T_2 \), a step/stop telegram will be sent. If a falling edge occurs after \( T_2 \) no action will be performed.

- When “Move-Step” is selected: On a rising edge a move telegram will be sent and will begin the \( T_2 \) time. If a falling edge occurs during the \( T_2 \), a step/stop telegram will be sent. If a falling edge occurs after \( T_2 \) no action will be performed.

**Important:** The \( T_1 \) time have to be defined in the “Short/long operation limit (x100ms)” parameter. Also the \( T_2 \) time have to be defined in the “Vanes adjustment time (x100ms)” parameter.

Short/long operation limit (x100ms)

This parameter lets introduce the time period difference for the short and the long operation (\( T_1 \) time).

Vanes adjustment time (x100ms)
This parameter lets introduce the time period for the vanes adjustment/blind movement (T2 time).

- When “Value” is selected the communication objects and new parameters for the Input “x” will appear as shown below.

![Figure 4.46 Parameter detail](image)

- **Send telegram after bus recovery**

  This parameter lets select if the Binary Input “x” will send a telegram, or not, after a bus recovery and the type of telegram sent (if enabled).

  - When “**No action**” is selected, no telegram will be sent after a bus recovery.
  
  - When “**Fixed value**” is selected, the binary input will send a telegram with the same value configured in the “Value on rising edge” parameter. Also a new parameter will appear (see below).

![Figure 4.47 Parameter detail](image)

- **Sending delay after a bus recovery (seconds)**

  This parameter lets configure a delay (in seconds) that will be applied after a bus recovery and, after which, a telegram will be sent.

- **DPT to be sent**

  This parameter lets select the DPT type for the value that will be defined in the next parameter. This value will be sent on a rising edge (contact activated).

![Figure 4.48 Parameter detail](image)

- **Value on rising edge (when contact activated)**
This parameter lets define a value for the DTP type configured in the “DPT to be sent” parameter. This value will be sent on a rising edge (contact activated).

- When “Execute Scene (internal)” is selected, the binary input “x” will activate the scene defined in the next parameter, on a rising edge (contact activated).

![Parameter detail](image)

- Scene when contact is activated

This parameter lets choose the scene that will be activated on a rising edge. This scene MUST be defined in the “Scene Configuration” dialog as a preset.

- When “Occupancy (internal)” is selected, the binary input “x” will have the same behavior as configured in the parameter “Enable use of Occupancy function” inside the “Switch-Off Timeouts Configuration” dialog.

![Parameter detail](image)

- When “Window Contact (internal)” is selected, the binary input “x” will have the same behavior as configured in the parameter “Enable use of Open Window / Switch off timeout function” inside the “Switch-Off Timeouts Configuration” dialog.
5. Specifications

<table>
<thead>
<tr>
<th>Specification</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Envelope</td>
<td>ABS (UL 94 HB). 2,5 mm thickness</td>
</tr>
<tr>
<td>Dimensions</td>
<td>70 X 70 X 28 mm</td>
</tr>
<tr>
<td>Weight</td>
<td>70g</td>
</tr>
<tr>
<td>Colour</td>
<td>Ivory white</td>
</tr>
<tr>
<td>Power supply</td>
<td>29V DC, 7mA</td>
</tr>
<tr>
<td></td>
<td>Supplied through KNX bus.</td>
</tr>
<tr>
<td>MHI X/Y Bus</td>
<td>Voltage: 13-18V</td>
</tr>
<tr>
<td></td>
<td>Current: 80mA</td>
</tr>
<tr>
<td>LED indicators</td>
<td>1 x KNX programming.</td>
</tr>
<tr>
<td>Push buttons</td>
<td>1 x KNX programming.</td>
</tr>
<tr>
<td>Binary inputs</td>
<td>4 x Potential-free binary inputs.</td>
</tr>
<tr>
<td></td>
<td>Signal cable length: 5m ushieldeed, may be extended up to 20m with twisted.</td>
</tr>
<tr>
<td></td>
<td>Compliant with the following standards:</td>
</tr>
<tr>
<td></td>
<td>IEC61000-4-2 : level 4 - 15kV (air discharge) - 8kV (contact discharge)</td>
</tr>
<tr>
<td></td>
<td>MIL STD 883E-Method 3015-7 : class3B</td>
</tr>
<tr>
<td>Configuration</td>
<td>Configuration with ETS.</td>
</tr>
<tr>
<td>Operating Temperature</td>
<td>From -25°C to 85°C</td>
</tr>
<tr>
<td>Storage Temperature</td>
<td>From -40°C to 85°C</td>
</tr>
<tr>
<td>Isolation Voltage</td>
<td>2500V</td>
</tr>
<tr>
<td>RoHS conformity</td>
<td>Compliant with RoHS directive (2002/95/CE).</td>
</tr>
<tr>
<td>Certifications</td>
<td>CE conformity to EMC directive (2004/108/EC) and Low-voltage directive (2006/95/EC)</td>
</tr>
<tr>
<td></td>
<td>EN 61000-6-2; EN 61000-6-3; EN 60950-1; EN 50491-3; EN 50090-2-2;</td>
</tr>
<tr>
<td></td>
<td>EN 50428; EN 60669-1; EN 60669-2-1</td>
</tr>
</tbody>
</table>

![Diagram showing IntesisBox® KNX - Mitsubishi Heavy Industries A.C. interface and specifications](attachment:image.png)
6. AC Unit Types compatibility.

A list of Mitsubishi Heavy Industries indoor unit models compatible with MH-RC-KNX-1i and their available features can be found in:

# 7. Error Codes

<table>
<thead>
<tr>
<th>Error Code KNX</th>
<th>Error in Remote Controller</th>
<th>Error Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>N/A</td>
<td>No active error</td>
</tr>
<tr>
<td>1</td>
<td>E1</td>
<td>Remote controller communication error</td>
</tr>
<tr>
<td>2</td>
<td>E2</td>
<td>Duplicated indoor unit address</td>
</tr>
<tr>
<td>3</td>
<td>E3</td>
<td>Outdoor unit signal line error</td>
</tr>
<tr>
<td>5</td>
<td>E5</td>
<td>Communication error during operation</td>
</tr>
<tr>
<td>6</td>
<td>E6</td>
<td>Indoor heat exchanger temperature thermistor anomaly</td>
</tr>
<tr>
<td>7</td>
<td>E7</td>
<td>Indoor return air temperature thermistor anomaly</td>
</tr>
<tr>
<td>8</td>
<td>E8</td>
<td>Heating overload operation</td>
</tr>
<tr>
<td>9</td>
<td>E9</td>
<td>Drain trouble</td>
</tr>
<tr>
<td>10</td>
<td>E10</td>
<td>Excessive number of indoor units (more than 17) by controlling one remote controller</td>
</tr>
<tr>
<td>12</td>
<td>E12</td>
<td>Address setting error by mixed setting method</td>
</tr>
<tr>
<td>14</td>
<td>E14</td>
<td>Communication error between master and slave indoor units</td>
</tr>
<tr>
<td>16</td>
<td>E16</td>
<td>Indoor fan motor anomaly</td>
</tr>
<tr>
<td>19</td>
<td>E19</td>
<td>Indoor unit operation check, drain motor check setting error</td>
</tr>
<tr>
<td>28</td>
<td>E28</td>
<td>Remote controller temperature thermistor anomaly</td>
</tr>
<tr>
<td>30</td>
<td>E30</td>
<td>Unmatched connection of indoor and outdoor unit</td>
</tr>
<tr>
<td>31</td>
<td>E31</td>
<td>Duplicated outdoor unit address No.</td>
</tr>
<tr>
<td>32</td>
<td>E32</td>
<td>Open L3 Phase on power supply at primary side</td>
</tr>
<tr>
<td>33</td>
<td>E33</td>
<td>Inverter primary current error</td>
</tr>
<tr>
<td>35</td>
<td>E35</td>
<td>Cooling overload operation</td>
</tr>
<tr>
<td>36</td>
<td>E36</td>
<td>Discharge pipe temperature error</td>
</tr>
<tr>
<td>37</td>
<td>E37</td>
<td>Outdoor heat exchanger temperature thermistor anomaly</td>
</tr>
<tr>
<td>38</td>
<td>E38</td>
<td>Outdoor/Ambient air temperature thermistor anomaly</td>
</tr>
<tr>
<td>39</td>
<td>E39</td>
<td>Discharge pipe temperature thermistor anomaly</td>
</tr>
<tr>
<td>40</td>
<td>E40</td>
<td>High pressure error</td>
</tr>
<tr>
<td>41</td>
<td>E41</td>
<td>Power transistor overheat</td>
</tr>
<tr>
<td>42</td>
<td>E42</td>
<td>Current cut</td>
</tr>
<tr>
<td>43</td>
<td>E43</td>
<td>Excessive number of indoor units connected, excessive total capacity of connection</td>
</tr>
<tr>
<td>45</td>
<td>E45</td>
<td>Communication error between inverter PCB and outdoor control PCB</td>
</tr>
<tr>
<td>46</td>
<td>E46</td>
<td>Mixed address setting methods coexistent in same network</td>
</tr>
<tr>
<td>47</td>
<td>E47</td>
<td>Inverter over-current error</td>
</tr>
<tr>
<td>48</td>
<td>E48</td>
<td>Outdoor DC fan motor anomaly</td>
</tr>
<tr>
<td>49</td>
<td>E49</td>
<td>Low pressure anomaly</td>
</tr>
<tr>
<td>51</td>
<td>E51</td>
<td>Inverter anomaly</td>
</tr>
<tr>
<td>53</td>
<td>E53</td>
<td>Suction pipe temperature thermistor anomaly</td>
</tr>
<tr>
<td>54</td>
<td>E54</td>
<td>High/Low pressure sensor anomaly</td>
</tr>
<tr>
<td>55</td>
<td>E55</td>
<td>Underneath temperature thermistor anomaly</td>
</tr>
<tr>
<td>56</td>
<td>E56</td>
<td>Power transistor temperature thermistor anomaly</td>
</tr>
<tr>
<td>57</td>
<td>E57</td>
<td>Insufficient in refrigerant amount or detection of service valve closure</td>
</tr>
<tr>
<td>58</td>
<td>E58</td>
<td>Anomalous compressor by loss of synchronism</td>
</tr>
<tr>
<td>59</td>
<td>E59</td>
<td>Compressor startup failure</td>
</tr>
<tr>
<td>60</td>
<td>E60</td>
<td>Rotor position detection failure / Anomalous compressor rotor lock</td>
</tr>
<tr>
<td>61</td>
<td>E61</td>
<td>Communication error between the master unit and slave units</td>
</tr>
<tr>
<td>63</td>
<td>E63</td>
<td>Emergency stop</td>
</tr>
<tr>
<td>65532</td>
<td>N/A</td>
<td>Initialization process.</td>
</tr>
<tr>
<td>65535</td>
<td>N/A</td>
<td>Communication error between MH-RC-KNX-1i and AC unit / Remote controller</td>
</tr>
</tbody>
</table>

In case you detect an error code not listed, contact your nearest Mitsubishi Heavy Industries technical support service for more information on the error meaning.
## Appendix A – Communication Objects Table

<table>
<thead>
<tr>
<th>TOPIC</th>
<th>OBJECT NUMBER</th>
<th>NAME</th>
<th>LENGTH</th>
<th>DATAPoint type</th>
<th>FLAGS</th>
<th>FUNCTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>On/Off</td>
<td>0</td>
<td>Control_ On/Off</td>
<td>1 bit</td>
<td>DPT_Switch</td>
<td>W T</td>
<td>0 - Off; 1-On</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>Control_ Mode</td>
<td>1 byte</td>
<td>DPT_HVACContrMode</td>
<td>W T</td>
<td>0 - Auto; 1 - Heat; 3 - Cool; 9 - Fan; 14 - Dry</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>Control_ Mode Cool/Heat</td>
<td>1 bit</td>
<td>DPT_Heat/Cool</td>
<td>W T</td>
<td>0 - Cool; 1 - Heat</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>Control_ Mode Cool &amp; On</td>
<td>1 byte</td>
<td>DPT_Scaling</td>
<td>W T</td>
<td>0% - Off; 0.1%-100% - On + Cool</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>Control_ Mode Heat &amp; On</td>
<td>1 byte</td>
<td>DPT_Scaling</td>
<td>W T</td>
<td>0% - Off; 0.1%-100% - On + Heat</td>
</tr>
<tr>
<td>Mode</td>
<td>5</td>
<td>Control_ Mode Auto</td>
<td>1 bit</td>
<td>DPT_Bool</td>
<td>W T</td>
<td>1 – Auto mode</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>Control_ Mode Heat</td>
<td>1 bit</td>
<td>DPT_Bool</td>
<td>W T</td>
<td>1 – Heat mode</td>
</tr>
<tr>
<td></td>
<td>7</td>
<td>Control_ Mode Cool</td>
<td>1 bit</td>
<td>DPT_Bool</td>
<td>W T</td>
<td>1 – Cool mode</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>Control_ Mode Fan</td>
<td>1 bit</td>
<td>DPT_Bool</td>
<td>W T</td>
<td>1 – Fan mode</td>
</tr>
<tr>
<td></td>
<td>9</td>
<td>Control_ Mode Dry</td>
<td>1 bit</td>
<td>DPT_Bool</td>
<td>W T</td>
<td>1 – Dry mode</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>Control_ Mode +/-</td>
<td>1 bit</td>
<td>DPT_Step</td>
<td>W</td>
<td>0 - Decrease; 1 - Increase</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Control_ Mode +/-</td>
<td>1 bit</td>
<td>DPT_UpDown</td>
<td>W</td>
<td>0 - Up; 1 - Down</td>
</tr>
<tr>
<td>Fan Speed</td>
<td>11</td>
<td>Control_ Fan Speed / 2 Speeds</td>
<td>1 byte</td>
<td>DPT_Scaling</td>
<td>W T</td>
<td>0%-74% - Speed 1; 75%-100% - Speed 2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Control_ Fan Speed / 2 Speeds</td>
<td>1 byte</td>
<td>DPT_Enumerated</td>
<td>W T</td>
<td>1 - Speed 1; 2 - Speed 2</td>
</tr>
</tbody>
</table>
### Fan Speed

| Control_ Fan Speed / 3 Speeds | 1 byte | DPT_Scaling | 5.001 | W T | 0%-49% - Speed 1; 50%-82% - Speed 2; 83%-100% - Speed 3 |
| Control_ Fan Speed / 3 Speeds | 1 byte | DPT_Enumerated | 5.010 | W T | 1 - Speed 1; 2 - Speed 2; 3 Speed 3 |
| Control_ Fan Speed / 4 Speeds | 1 byte | DPT_Scaling | 5.001 | W T | 0%-37% - Speed 1; 38%-62% - Speed 2; 63%-87% - Speed 3; 88%-100% - Speed 4 |
| Control_ Fan Speed / 4 Speeds | 1 byte | DPT_Enumerated | 5.010 | W T | 1 - Speed 1; 2 - Speed 2; 3 Speed 3; 4 - Speed 4 |

### Vanes Up-Down

| Control_ Vanes U-D / 4 pos | 1 byte | DPT_Scaling | 5.001 | W T | 0%-37% - Pos1; 38%-62% - Pos2; 63%-87% Pos3; 88%-100% - Pos4 |
| Control_ Vanes U-D / 4 pos | 1 byte | DPT_Enumerated | 5.010 | W T | 1 - Pos1; 2 - Pos2; 3 - Pos3; 4 - Pos4 |

<p>| Control_ Vanes U-D Pos1 | 1 bit | DPT_Bool | 1.002 | W T | 1 - Set Position 1 |
| Control_ Vanes U-D Pos2 | 1 bit | DPT_Bool | 1.002 | W T | 1 - Set Position 2 |
| Control_ Vanes U-D Pos3 | 1 bit | DPT_Bool | 1.002 | W T | 1 - Set Position 3 |</p>
<table>
<thead>
<tr>
<th></th>
<th>Description</th>
<th>Type</th>
<th>DPT</th>
<th>Value</th>
<th>W</th>
<th>T</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>21</td>
<td>Control_ Vanes U-D Pos4</td>
<td>1 bit</td>
<td>DPT_Bool</td>
<td>1.002</td>
<td>W</td>
<td>T</td>
<td>1 – Set Position 4</td>
</tr>
<tr>
<td>22</td>
<td>Control_ Vanes U-D Swing</td>
<td>1 bit</td>
<td>DPT_Bool</td>
<td>1.002</td>
<td>W</td>
<td>T</td>
<td>0 – Off; 1 – Swing</td>
</tr>
<tr>
<td>23</td>
<td>Control_ Vanes U-D -/+</td>
<td>1 bit</td>
<td>DPT_Step</td>
<td>1.007</td>
<td>W</td>
<td></td>
<td>0 - Decrease; 1 - Increase</td>
</tr>
<tr>
<td></td>
<td>Control_ Vanes U-D -/+</td>
<td>1 bit</td>
<td>DPT_UpDown</td>
<td>1.008</td>
<td>W</td>
<td></td>
<td>0 - Up; 1 - Down</td>
</tr>
<tr>
<td>24</td>
<td>Control_ Setpoint Temperature</td>
<td>2 byte</td>
<td>DPT_Value_Temp</td>
<td>9.001</td>
<td>W</td>
<td>T</td>
<td>(ºC)</td>
</tr>
<tr>
<td>25</td>
<td>Control_ Setpoint Temp -/+</td>
<td>1 bit</td>
<td>DPT_Step</td>
<td>1.007</td>
<td>W</td>
<td></td>
<td>0 - Decrease; 1 - Increase</td>
</tr>
<tr>
<td></td>
<td>Control_ Setpoint Temp -/+</td>
<td>1 bit</td>
<td>DPT_UpDown</td>
<td>1.008</td>
<td>W</td>
<td></td>
<td>0 - Up; 1 - Down</td>
</tr>
<tr>
<td>26</td>
<td>Control_ Ambient Temperature</td>
<td>2 byte</td>
<td>DPT_Value_Temp</td>
<td>9.001</td>
<td>W</td>
<td>T</td>
<td>(ºC)</td>
</tr>
<tr>
<td>27</td>
<td>Control_ Reset Filter</td>
<td>1 bit</td>
<td>DPT_Reset</td>
<td>1.015</td>
<td>W</td>
<td>T</td>
<td>1 – Reset filter</td>
</tr>
<tr>
<td>28</td>
<td>Control_ Reset Error</td>
<td>1 bit</td>
<td>DPT_Reset</td>
<td>1.015</td>
<td>W</td>
<td>T</td>
<td>1 – Reset error</td>
</tr>
<tr>
<td>29</td>
<td>Control_ Window Contact Status</td>
<td>1 bit</td>
<td>DPT_OpenClose</td>
<td>1.009</td>
<td>W</td>
<td>T</td>
<td>0 - Open; 1 - Closed</td>
</tr>
<tr>
<td></td>
<td>Control_ Switch Off Timeout</td>
<td>1 bit</td>
<td>DPT_Start</td>
<td>1.010</td>
<td>W</td>
<td>T</td>
<td>0 - Stop; 1 - Start</td>
</tr>
<tr>
<td>30</td>
<td>Control_ Occupancy</td>
<td>1 bit</td>
<td>DPT_Occupancy</td>
<td>1.018</td>
<td>W</td>
<td>T</td>
<td>0 - Not Occupied; 1 - Occupied</td>
</tr>
<tr>
<td>31</td>
<td>Control_ Sleep Timeout</td>
<td>1 bit</td>
<td>DPT_Start</td>
<td>1.010</td>
<td>W</td>
<td>T</td>
<td>0 - Stop; 1 - Start</td>
</tr>
<tr>
<td>32</td>
<td>Control_ Lock Remote Control</td>
<td>1 bit</td>
<td>DPT_Bool</td>
<td>1.002</td>
<td>W</td>
<td>T</td>
<td>0 - Unlocked; 1 - Locked</td>
</tr>
<tr>
<td>33</td>
<td>Control_ Lock Control Objects</td>
<td>1 bit</td>
<td>DPT_Bool</td>
<td>1.002</td>
<td>W</td>
<td>T</td>
<td>0 - Unlocked; 1 - Locked</td>
</tr>
<tr>
<td>34</td>
<td>Control_ Power Mode</td>
<td>1 bit</td>
<td>DPT_Start</td>
<td>1.010</td>
<td>W</td>
<td>T</td>
<td>0 - Stop; 1 - Start</td>
</tr>
<tr>
<td>35</td>
<td>Control_ Econo Mode</td>
<td>1 bit</td>
<td>DPT_Start</td>
<td>1.010</td>
<td>W</td>
<td>T</td>
<td>0 - Stop; 1 - Start</td>
</tr>
<tr>
<td>36</td>
<td>Control_ Additional Heat</td>
<td>1 bit</td>
<td>DPT_Start</td>
<td>1.010</td>
<td>W</td>
<td>T</td>
<td>0 - Stop; 1 - Start</td>
</tr>
<tr>
<td>37</td>
<td>Control_ Additional Cool</td>
<td>1 bit</td>
<td>DPT_Start</td>
<td>1.010</td>
<td>W</td>
<td>T</td>
<td>0 - Stop; 1 - Start</td>
</tr>
</tbody>
</table>
### Scenes

<table>
<thead>
<tr>
<th></th>
<th>Description</th>
<th>Size</th>
<th>Protocol</th>
<th>Index</th>
<th>W</th>
<th>T</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>38</td>
<td>Control_Save/Exec Scene</td>
<td>1 byte</td>
<td>DPT_SceneControl</td>
<td>18.001</td>
<td>W</td>
<td>T</td>
<td>0 to 4 – Control_Exec Scene 1 to 5; 128 to 132 – Control_Save Scene 1 to 5</td>
</tr>
<tr>
<td>39</td>
<td>Control_Store Scene1</td>
<td>1 bit</td>
<td>DPT_Bool</td>
<td>1.002</td>
<td>W</td>
<td></td>
<td>1 - Store Scene 1</td>
</tr>
<tr>
<td>40</td>
<td>Control_Store Scene2</td>
<td>1 bit</td>
<td>DPT_Bool</td>
<td>1.002</td>
<td>W</td>
<td></td>
<td>1 - Store Scene 2</td>
</tr>
<tr>
<td>41</td>
<td>Control_Store Scene3</td>
<td>1 bit</td>
<td>DPT_Bool</td>
<td>1.002</td>
<td>W</td>
<td></td>
<td>1 - Store Scene 3</td>
</tr>
<tr>
<td>42</td>
<td>Control_Store Scene4</td>
<td>1 bit</td>
<td>DPT_Bool</td>
<td>1.002</td>
<td>W</td>
<td></td>
<td>1 - Store Scene 4</td>
</tr>
<tr>
<td>43</td>
<td>Control_Store Scene5</td>
<td>1 bit</td>
<td>DPT_Bool</td>
<td>1.002</td>
<td>W</td>
<td></td>
<td>1 - Store Scene 5</td>
</tr>
<tr>
<td>44</td>
<td>Control_Execute Scene1</td>
<td>1 bit</td>
<td>DPT_Bool</td>
<td>1.002</td>
<td>W</td>
<td>T</td>
<td>1 - Execute Scene 1</td>
</tr>
<tr>
<td>45</td>
<td>Control_Execute Scene2</td>
<td>1 bit</td>
<td>DPT_Bool</td>
<td>1.002</td>
<td>W</td>
<td>T</td>
<td>1 - Execute Scene 2</td>
</tr>
<tr>
<td>46</td>
<td>Control_Execute Scene3</td>
<td>1 bit</td>
<td>DPT_Bool</td>
<td>1.002</td>
<td>W</td>
<td>T</td>
<td>1 - Execute Scene 3</td>
</tr>
<tr>
<td>47</td>
<td>Control_Execute Scene4</td>
<td>1 bit</td>
<td>DPT_Bool</td>
<td>1.002</td>
<td>W</td>
<td>T</td>
<td>1 - Execute Scene 4</td>
</tr>
<tr>
<td>48</td>
<td>Control_Execute Scene5</td>
<td>1 bit</td>
<td>DPT_Bool</td>
<td>1.002</td>
<td>W</td>
<td>T</td>
<td>1 - Execute Scene 5</td>
</tr>
</tbody>
</table>

### Disabling

<table>
<thead>
<tr>
<th></th>
<th>Description</th>
<th>Size</th>
<th>Protocol</th>
<th>Index</th>
<th>W</th>
<th>T</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>49</td>
<td>Control_Disable Input 1</td>
<td>1 bit</td>
<td>DPT_Bool</td>
<td>1.002</td>
<td>W</td>
<td>T</td>
<td>0 - Enable; 1 - Disable</td>
</tr>
<tr>
<td>50</td>
<td>Control_Disable Input 2</td>
<td>1 bit</td>
<td>DPT_Bool</td>
<td>1.002</td>
<td>W</td>
<td>T</td>
<td>0 - Enable; 1 - Disable</td>
</tr>
<tr>
<td>51</td>
<td>Control_Disable Input 3</td>
<td>1 bit</td>
<td>DPT_Bool</td>
<td>1.002</td>
<td>W</td>
<td>T</td>
<td>0 - Enable; 1 - Disable</td>
</tr>
<tr>
<td>52</td>
<td>Control_Disable Input 4</td>
<td>1 bit</td>
<td>DPT_Bool</td>
<td>1.002</td>
<td>W</td>
<td>T</td>
<td>0 - Enable; 1 - Disable</td>
</tr>
<tr>
<td></td>
<td>Control_Disable Input 1</td>
<td>1 bit</td>
<td>DPT_Enable</td>
<td>1.003</td>
<td>W</td>
<td>T</td>
<td>0 - Disable; 1 - Enable</td>
</tr>
<tr>
<td></td>
<td>Control_Disable Input 2</td>
<td>1 bit</td>
<td>DPT_Enable</td>
<td>1.003</td>
<td>W</td>
<td>T</td>
<td>0 - Disable; 1 - Enable</td>
</tr>
<tr>
<td></td>
<td>Control_Disable Input 3</td>
<td>1 bit</td>
<td>DPT_Enable</td>
<td>1.003</td>
<td>W</td>
<td>T</td>
<td>0 - Disable; 1 - Enable</td>
</tr>
<tr>
<td></td>
<td>Control_Disable Input 4</td>
<td>1 bit</td>
<td>DPT_Enable</td>
<td>1.003</td>
<td>W</td>
<td>T</td>
<td>0 - Disable; 1 - Enable</td>
</tr>
</tbody>
</table>
### On/Off

<table>
<thead>
<tr>
<th></th>
<th>Status_ On/Off</th>
<th>Data Type</th>
<th>DPT</th>
<th>Access</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>53</td>
<td>1 bit</td>
<td>DPT_Switch</td>
<td>1.001</td>
<td>R</td>
<td>0 - Off; 1-On</td>
</tr>
</tbody>
</table>

### Mode

<table>
<thead>
<tr>
<th></th>
<th>Status_ Mode</th>
<th>Data Type</th>
<th>DPT</th>
<th>Access</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>54</td>
<td>1 byte</td>
<td>DPT_HVACContrMode</td>
<td>20.105</td>
<td>R</td>
<td>0 - Auto; 1 - Heat; 3 - Cool; 9 - Fan; 14 - Dry</td>
</tr>
<tr>
<td>55</td>
<td>1 bit</td>
<td>DPT_Heat/Cool</td>
<td>1.100</td>
<td>R</td>
<td>0 - Cool; 1 - Heat</td>
</tr>
<tr>
<td>56</td>
<td>1 bit</td>
<td>DPT_Bool</td>
<td>1.002</td>
<td>R</td>
<td>1 - Auto mode is active</td>
</tr>
<tr>
<td>57</td>
<td>1 bit</td>
<td>DPT_Bool</td>
<td>1.002</td>
<td>R</td>
<td>1 - Heat mode is active</td>
</tr>
<tr>
<td>58</td>
<td>1 bit</td>
<td>DPT_Bool</td>
<td>1.002</td>
<td>R</td>
<td>1 - Cool mode is active</td>
</tr>
<tr>
<td>59</td>
<td>1 bit</td>
<td>DPT_Bool</td>
<td>1.002</td>
<td>R</td>
<td>1 - Fan mode is active</td>
</tr>
<tr>
<td>60</td>
<td>1 bit</td>
<td>DPT_Bool</td>
<td>1.002</td>
<td>R</td>
<td>1 - Dry mode is active</td>
</tr>
<tr>
<td>61</td>
<td>14 byte</td>
<td>DPT_String_8859_1</td>
<td>16.001</td>
<td>R</td>
<td>ASCII String</td>
</tr>
</tbody>
</table>

### Fan Speed

<table>
<thead>
<tr>
<th></th>
<th>Status_ Fan Speed / 2 Speeds</th>
<th>Data Type</th>
<th>DPT</th>
<th>Access</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>62</td>
<td>1 byte</td>
<td>DPT_Scaling</td>
<td>5.001</td>
<td>R</td>
<td>50% - Speed 1; 100% - Speed 2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DPT_Enumerated</td>
<td>5.010</td>
<td>R</td>
<td>1 - Speed 1; 2 - Speed 2</td>
</tr>
<tr>
<td></td>
<td>Status_ Fan Speed / 3 Speeds</td>
<td>DPT_Scaling</td>
<td>5.010</td>
<td>R</td>
<td>33% - Speed 1; 67% - Speed 2; 100% - Speed 3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DPT_Enumerated</td>
<td>5.010</td>
<td>R</td>
<td>1 - Speed 1; 2 - Speed 2; 3 Speed 3</td>
</tr>
<tr>
<td></td>
<td>Status_ Fan Speed / 4 Speeds</td>
<td>DPT_Scaling</td>
<td>5.010</td>
<td>R</td>
<td>25% - Speed 1; 50% - Speed 2; 75% - Speed 3; 100% - Speed 4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DPT_Enumerated</td>
<td>5.010</td>
<td>R</td>
<td>1 - Speed 1; 2 - Speed 2; 3 Speed 3; 4 - Speed 4</td>
</tr>
<tr>
<td>63</td>
<td>1 bit</td>
<td>DPT_Bool</td>
<td>1.002</td>
<td>R</td>
<td>1 - Fan is in speed 1</td>
</tr>
<tr>
<td>64</td>
<td>1 bit</td>
<td>DPT_Bool</td>
<td>1.002</td>
<td>R</td>
<td>1 - Fan is in speed 2</td>
</tr>
<tr>
<td>65</td>
<td>1 bit</td>
<td>DPT_Bool</td>
<td>1.002</td>
<td>R</td>
<td>1 - Fan is in Speed 3</td>
</tr>
<tr>
<td>66</td>
<td>1 bit</td>
<td>DPT_Bool</td>
<td>1.002</td>
<td>R</td>
<td>1 - Fan is in Speed 4</td>
</tr>
<tr>
<td>67</td>
<td>14 byte</td>
<td>DPT_String_8859_1</td>
<td>16.001</td>
<td>R</td>
<td>ASCII String</td>
</tr>
<tr>
<td></td>
<td>Description</td>
<td>Type</td>
<td>DPT</td>
<td>Value</td>
<td>R</td>
</tr>
<tr>
<td>---</td>
<td>-------------------------------------------------</td>
<td>------------</td>
<td>--------------</td>
<td>-------</td>
<td>---</td>
</tr>
<tr>
<td>68</td>
<td>Status_ Vanes U-D / 4 pos</td>
<td>1 byte</td>
<td>DPT_Scaling</td>
<td>5.001</td>
<td>R</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>69</td>
<td>Status_ Vanes U-D Pos1</td>
<td>1 bit</td>
<td>DPT_Bool</td>
<td>1.002</td>
<td>R</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>70</td>
<td>Status_ Vanes U-D Pos2</td>
<td>1 bit</td>
<td>DPT_Bool</td>
<td>1.002</td>
<td>R</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>71</td>
<td>Status_ Vanes U-D Pos3</td>
<td>1 bit</td>
<td>DPT_Bool</td>
<td>1.002</td>
<td>R</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>72</td>
<td>Status_ Vanes U-D Pos4</td>
<td>1 bit</td>
<td>DPT_Bool</td>
<td>1.002</td>
<td>R</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>73</td>
<td>Status_ Vanes U-D Swing</td>
<td>1 bit</td>
<td>DPT_Bool</td>
<td>1.002</td>
<td>R</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>74</td>
<td>Status_ Vanes U-D Text</td>
<td>14 byte</td>
<td>DPT_String_8859_1</td>
<td>16.001</td>
<td>R</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>75</td>
<td>Status_ AC Setpoint Temp</td>
<td>2 byte</td>
<td>DPT_Value_Temp</td>
<td>9.001</td>
<td>R</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>76</td>
<td>Status_ AC Referene Temp</td>
<td>2 byte</td>
<td>DPT_Value_Temp</td>
<td>9.001</td>
<td>R</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>77</td>
<td>Status_ Only Centrally Ctrl</td>
<td>1 bit</td>
<td>DPT_Bool</td>
<td>1.002</td>
<td>R</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>78</td>
<td>Status_ Filter Status</td>
<td>1 bit</td>
<td>DPT_Bool</td>
<td>1.005</td>
<td>R</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>79</td>
<td>Status_ Error/Alarm</td>
<td>1 bit</td>
<td>DPT_Alarm</td>
<td>1.005</td>
<td>R</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>80</td>
<td>Status_ Error Code</td>
<td>2 byte</td>
<td>Enumerated</td>
<td></td>
<td>R</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>81</td>
<td>Status_ Error Text Code</td>
<td>14 byte</td>
<td>DPT_String_8859_1</td>
<td>16.001</td>
<td>R</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Special Modes</td>
<td>Status_ Power Mode</td>
<td>1 bit</td>
<td>DPT_Switch</td>
<td>1.001</td>
<td>R</td>
</tr>
<tr>
<td>---------------</td>
<td>-------------------</td>
<td>-------</td>
<td>------------</td>
<td>-------</td>
<td>---</td>
</tr>
<tr>
<td>Status_ Econo Mode</td>
<td>1 bit</td>
<td>DPT_Switch</td>
<td>1.001</td>
<td>R</td>
<td>T</td>
</tr>
<tr>
<td>Status_ Additional Heat</td>
<td>1 bit</td>
<td>DPT_Switch</td>
<td>1.001</td>
<td>R</td>
<td>T</td>
</tr>
<tr>
<td>Status_ Additional Cool</td>
<td>1 bit</td>
<td>DPT_Switch</td>
<td>1.001</td>
<td>R</td>
<td>T</td>
</tr>
<tr>
<td>Counter</td>
<td>Status_ Operation Hour Counter</td>
<td>2 byte</td>
<td>DPT_Value_2_Ucount</td>
<td>7.001</td>
<td>R</td>
</tr>
<tr>
<td>Scene</td>
<td>Status_ Current Scene</td>
<td>1 byte</td>
<td>DPT_SceneNumber</td>
<td>17.001</td>
<td>R</td>
</tr>
<tr>
<td>Binary Inputs</td>
<td>Status_ Inx - Switching</td>
<td>1 bit</td>
<td>DPT_Switch</td>
<td>1.001</td>
<td>R</td>
</tr>
<tr>
<td>Status_ Inx - Dimming - On/Off</td>
<td>1 bit</td>
<td>DPT_Switch</td>
<td>1.001</td>
<td>R</td>
<td>T</td>
</tr>
<tr>
<td>Status_ Inx - Shut/Blind - Step</td>
<td>1 bit</td>
<td>DPT_ShutterBlinds</td>
<td>1.023</td>
<td>R</td>
<td>T</td>
</tr>
<tr>
<td>Status_ Inx - Value</td>
<td>1 byte</td>
<td>DPT_Value_1_Ucount</td>
<td>5.010</td>
<td>R</td>
<td>T</td>
</tr>
<tr>
<td>Status_ Inx - Value</td>
<td>2 byte</td>
<td>DPT_Value_2_Ucount</td>
<td>7.001</td>
<td>R</td>
<td>T</td>
</tr>
<tr>
<td>Status_ Inx - Value</td>
<td>2 byte</td>
<td>DPT_Value_2_Count</td>
<td>8.001</td>
<td>R</td>
<td>T</td>
</tr>
<tr>
<td>Status_ Inx - Value</td>
<td>2 byte</td>
<td>DPT_Value_Temp</td>
<td>9.001</td>
<td>R</td>
<td>T</td>
</tr>
<tr>
<td>Status_ Inx - Value</td>
<td>4 byte</td>
<td>DPT_Value_4_Ucount</td>
<td>12.001</td>
<td>R</td>
<td>T</td>
</tr>
<tr>
<td>Status_ Inx - Dimming - Step(%)</td>
<td>4 bit</td>
<td>DPT_Control_Dimm.</td>
<td>3.007</td>
<td>R</td>
<td>T</td>
</tr>
<tr>
<td>Status_ Inx - Shut/Blind -Move</td>
<td>1 bit</td>
<td>DPT_ShutterBlinds</td>
<td>1.023</td>
<td>R</td>
<td>T</td>
</tr>
</tbody>
</table>