

Copyright (c) 2015 Genesis Technologies AG Switzerland, all rights reserved

## **KNX generic AC driver (GTSTC4ZKNXGAC) for KNX Air-conditions**

### **1. Disclaimer**

- Genesis Technologies does provide this driver as it is. There is no commitment to support, nor does Genesis Technologies assure a decent reaction time in case of a problem. Genesis Technologies can refuse all support, discontinue support, end the driver lifetime at any time. Genesis Technologies will not refund in any case licenses
- Notwithstanding the foregoing, Genesis Technologies is not liable to licensee for any damages, including compensatory, special, incidental, exemplary, punitive, or consequential damages, connected with or resulting from this license agreement or licensee's use of this software. Licensee's jurisdiction may not allow such a limitation of damages, so this limitation may not apply.

By using this driver you accept that you have read and accepted the terms above!

### **2. What is the thermostat driver? What is it used for?**

This driver is designed to work with KNX air-Conditions or KNX air-condition modules such as Hitachi C-A16KNX or Intesis DK-RC-KNX-1. It does allow the user to direct adjusting the AC's cool, heat, dry and other function modes as well as setpoints and fan modes. In extra pages the flap/louvers directions can be managed as well.

### **3. What thermostatV2 features are supported?**

Supported features of the thermostatV2 proxy

- Setpoint
- Modes
- Presets
- Preset scheduling
- Extras
- Hold modes

Not supported

- Legacy scheduling

### **4. What KNX thermostat features are supported?**

Supported features of the usual air condition interfaces

- Set and display the actual setpoint
- Display temperature
- Modes: Off, cool, heat, auto and custom defined (See properties description)
- Fan modes: Custom defined (See properties description)

## **5. Theory of operation / general info**

### **5.1. General**

The driver does reflect what is going on the air-condition unit. Modes, Fan modes and flap/louvers sentences that are displayed are fully customizable. The driver works for simply display a temperature up to a fully functional AC system with all features.

### **5.2. Temperature**

Room temperature is measured and delivered from KNX, no matter if it's the thermostat or an external sensor, the driver does interpret the values that are delivered. Fahrenheit and Celsius can be selected to display in the scale wished.

### 5.3. Setpoints

Setpoints are read out of the air-condition and are displayed without shifts or whatever calculations.

### 5.4. Operation modes

The typical KNX modes like comfort, night etc. are not supported.

### 5.5. Language support

All preset and extra page strings are per default in English. However the driver ships with predefined translations for de\_CH, de\_DE, fr\_FR, it\_IT and en\_US. These languages are written as files into the path \\ IP of controller \drivers\lua\sandbox\driverId if they don't exist and loaded each time the driver starts. The driver id is shown by hovering with the mouse over the driver in the project tree, system design.

If for example a de\_DE translation should not match your expectations, feel free to open the de\_DE file in a text editor, edit and save.

Restart the driver and you should be set. (For Flash, press refresh Navigators after the driver restart)

As well if your system does run on a locale setting which may be missing, copy the en\_EN to a new file and translate the strings inside.

Name the new file after the locale inside. If the locale code is not known, see here [https://github.com/bobdenotter/locales/blob/master/locales\\_list.txt](https://github.com/bobdenotter/locales/blob/master/locales_list.txt) Restart the driver and you should be set. (For Flash, press refresh Navigators after the driver restart)

**REMARK:** Control4 translates per default the sentences *Auto*, *Cool*, *Heat*, and *off* into the set local language. Do set the property *Modes available* with that knowledge in mind. So for example on a system with de\_CH localization a property string set: *0=Auto,1=Heat,2=Trocknen,3=Ventilieren,4=Cool* results on Navigators as the list: *0=Auto,1=Heizen,2=Trocknen,3=Ventilieren,4=Kühlen*

## 6. Can I use this driver for KNX thermostat x?

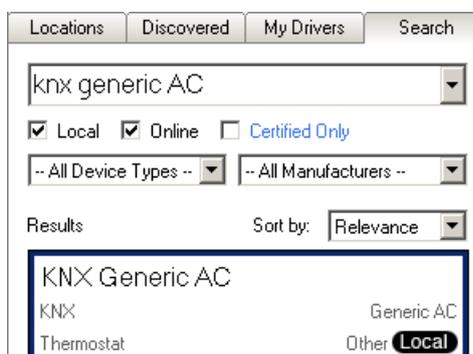
Yes. Maybe it will work. We do provide the driver as is. Test and you will see. The driver works for a few hours without license key.

## 7. Software installation Control4

### 7.1. Prerequisites

- The driver uses the Control4 KNX network driver to connect to KNX. Please make sure that this driver is installed in your project and is connected to KNX, running
- Download the driver and placed them in %USERPROFILE%\Control4\Drivers.

**7.2.** Start Composer, go to *System Design* and select the tab *Search* in the right top window. Search for *knx generic AC*



**7.3.** Drag and drop the *knx generic AC* driver into the wished room. Click on it and find the Properties page. (Hovering with the mouse over the Properties will show up hints)

-> Info	14.09.2015 10:38 - Driver/Version - 1.00.10 running
Driver name	GTSTC4ZKNXGAC
-> Licensing	14.09.2015 10:38 - Licensing/Check - Valid until: 11.09.2065
Mac of the Controller	000FF57A8CA
Activation key	2822/LRMemBgNcq-JDAK/TfDhkkZbRQ-393a2Qn7NbTluSV-Sc7ILy56aoXJvCo-GkLNZDci

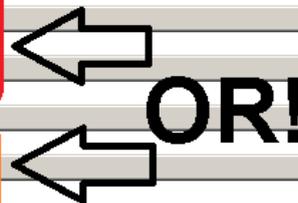
#### 7.4. Common properties

- -> Info: Does show the driver health info. If the driver experience issues this line will display it.
- Driver name: The product identification of the driver
- -> Licensing: Shows info about licensing status
- Mac of the Controller: Inserted automatically
- Activation key: Insert here the license key which is displayed in the driver dashboard after you bought a license and assigned it to the system. Once assigned in the Dashboard the license may jump in automatically once the driver connects to the online services

#### 7.5. General Config (none)

#### 7.6. KNX group addresses

Room temp. (2 Byte)	7/0/2
Setpoint (2 Byte)	7/0/0
Setpoint status (2 Byte)	7/0/1
Unit power (1 bit)	2/0/6
Unit power status (1 bit)	2/0/7
Mode (1 Byte)	0/0/7
Mode status (1 Byte)	0/0/9
Modes available	0=Auto,1=Heat,2=Trocknen,3=Ventilieren,4=Cool
Modes (1 bit)	2/0/3=Cool,2/0/0=Heat
Modes status (1 bit)	2/0/4=Cool,2/0/1=Heat
Fan (1 Byte)	0/0/12
Fan status (1 Byte)	0/0/14
Fan speeds available	0=Niedrig,31=Mittel,61=Hoch
Louvers (1 Byte)	0/0/17
Louvers status (1 Byte)	0/0/19
Louvers pos. available	0=Position 0,16=Position 1,31=Position 2, 46=Position 3,61=Position 4,76=Position 5,91=Pos
Louvers auto (1 bit)	3/0/3
Louvers auto status (1 bit)	3/0/4



- Room temp. (2 Byte): Room temperature. DPT 9.001, required Flags: C,R,T
- Setpoint (2 Byte): Setpoint. DPT 9.001, required Flags: C,W
- Setpoint status (2 Byte): In Setpoint status. DPT 9.001, required Flags: C,R,T
- Unit power (1 bit): Address that switches on/off the AC unit. DPT 1.001, required Flags: C,W,T
- Unit power status (1 bit): Status for on/off. DPT 1.001, required Flags: C,R,W,T
- Mode (1 Byte): In ETS usually as unit mode listed, to set auto, heat, cool, dry, etc.. , DPT

5.010, Flags: C,R,W,T

**REMARK:** Do not mix modes/status from 1 Byte and 1 bit addresses. See picture above.

- Mode status (1 Byte): In ETS usually as unit mode listed, to read auto, heat, cool, dry, etc.. , DPT 5.010, Flags: C,R,T  
**REMARK:** Do not mix modes/status from 1 Byte and 1 bit addresses. See picture above.
- Modes available: Comma separated string of the available operation modes. The notation is related to the value that activates the mode. Example definition: *0=Auto,1=Cool,2=Heat,3=Off*. When *Auto* is pushed on Navigators the driver will send the value 0 to the set address in the property *Mode (1 Byte)*  
**REMARK #1:** Control4 translates per default the sentences *Auto, Cool, Heat, and off* into the set local language. So for example on a system with de\_CH localization a property string set: *0=Auto,1=Heat,2=Trocknen,3=Ventilieren,4=Cool* results on Navigators as the list: *Auto, Heizen, Trocknen, Ventilieren, Kühlen*.  
**REMARK #2:** Do not mix modes/status from 1 Byte and 1 bit addresses. See picture above.
- Modes (1 bit): Comma delimited list of available modes and KNX group addresses. Format: "Group address" = "mode to show on Navigators". Example: *1/1/1=Auto, 1/1/2=Heat,1/1/3=Cool,1/1/4=Fan*. The driver sends out a 1 to the matched mode's address. To all other a 0 will be sent before.  
**REMARK:** Control4 translates per default the sentences *Auto, Cool, Heat, and off* into the set local language. So for example on a system with de\_CH localization a property string set: *1/1/1=Auto,1/1/2=Heat,1/1/3=Cool,1/1/4=Fan* results on Navigators as the list: *Auto, Heizen, Kühlen, Fan*  
**REMARK #2:** Do not mix modes/status from 1 Byte and 1 bit addresses. See picture above.
- Modes status (1 bit): Comma delimited list of available modes and KNX group addresses. Format: "Group address" = "status to show on Navigators". Example: *1/1/1=Auto, 1/1/2=Heat,1/1/3=Cool,1/1/4=Fan*. The driver interprets the last sent address with a value of 1 as the active mode.  
**REMARK:** Do not mix modes/status from 1 Byte and 1 bit addresses. See picture above.
- Fan (1 Byte): In ETS usually as fan mode listed, to set low, medium, high, super high, etc.. , DPT 5.010, Flags: C,R,W,T
- Fan status (1 Byte): In ETS usually as fan mode listed, to read low, medium, high, super high, etc.. , DPT 5.010, Flags: C,R,T
- Fan speeds available: Comma delimited list of available fan speeds. The notation is related to the value that activates the mode. Example definition: *0=Low,31=Mid,61=High*. When *High* is pushed on Navigators the driver will send the value 61 to the set address in the property *Fan (1 Byte)*
- Louvers (1 Byte): (Louvers = flaps). In ETS usually as louvers scale listed, to set positions of the louvers, DPT 5.010, Flags: C,R,W,T
- Louvers status (1 Byte): In ETS usually as louvers scale listed, to read positions of the louvers, DPT 5.010, Flags: C,R,T
- Louvers pos. available: Comma delimited list of available fan speeds. The notation is related to the value that activates the mode. Example definition: *0=Pos 0,16=Pos 1,31=Pos 2*. When *Pos 2* is pushed on Navigators the driver will send the value 31 to the set address in the property *Louvers (1 Byte)*
- Louvers auto (1 bit): Address that switches on/off auto mode for the louvers/flaps. DPT 1.001, required Flags: C,W,T
- Louvers auto status (1 bit): Status for on/off auto flap/louvers. DPT 1.001, required Flags: C,R,W,T

Addressing example (1 Byte addressing mode) in KNX, how it should look like in ETS. (Addresses may vary)

0	Unit-1 On/Off [1bit]	1=On,0=Off	6/0/0	1 bit	C	R	W	T	-
1	Unit-1 Mode [1byte]	0=Aut,1=Heat,2=Dry,3=Fan,4=Coo	6/0/7, 6/0/6	1 Byte	C	R	W	T	-
2	Unit-1 Fan [1byte]	0%..100%	6/0/5, 6/0/4	1 Byte	C	R	W	T	-
3	Unit-1 Louver Scale [1byte]	0%..100%	6/0/9	1 Byte	C	R	W	T	-
4	Unit-1 Louver Auto [1bit]	1= Automatic	6/0/10, 6/0/8	1 bit	C	R	W	T	-
5	Unit-1 Set Temperature [2byte]	17°C..30°C	6/0/3, 6/0/1	2 Byte	C	R	W	T	-
6	Unit-1 Ambient Temperature [2byte]	17°C..30°C	6/0/2	2 Byte	C	R	-	T	-

## 7.7. Other properties

-> Status	
Wait for bus status (ms)	800
Action on status timeout	Nothing, wait for bus status to refresh Navigators
-> Presets	
Enabled options	Operating modes, setpoint and fan

- Wait for bus status (ms): Depending on the KNX bus configuration or AC unit the status values for a executed action can be delayed. Navigators jump back to the former state when the status message does not arrive in a decent time. To avoid Navigators flip around a status can be simulated by the driver itself. This property set the time to wait until this action happens.
- Action on status timeout: Possible values:
  - Nothing, wait for bus status to refresh Navigators: After the *Wait for bus status (ms)* expires nothing happen
  - Simulate bus status, refresh Navigators: After the *Wait for bus status (ms)* expires the driver does simulate a status to get the Navigators happy
  - Simulate bus status, refresh Navigators and lookup status after x seconds: After the *Wait for bus status (ms)* expires the driver does simulate a status to get the Navigators happy and initiate after the x seconds a lookup to get the real bus status.
- Enabled options: Defines what options should be available for Presets

-> Log	
Lua window	None
Online logserver	None
Log period (h)	1
Log auth	

- -> Log / Lua window: Selects the log level printed out in Lua window from Composer. Use the level *Debug* to see what the driver does
- -> Log / Online log server: If a persistent problem appears that cannot be solved from the installer support may ask you to activate a decent level. Do not use until support ask you to do so
- Log period (h): The time frame the log should output
- Log auth: Developer can give you a code to see more logs. Usually not really interesting, only for in deep debug

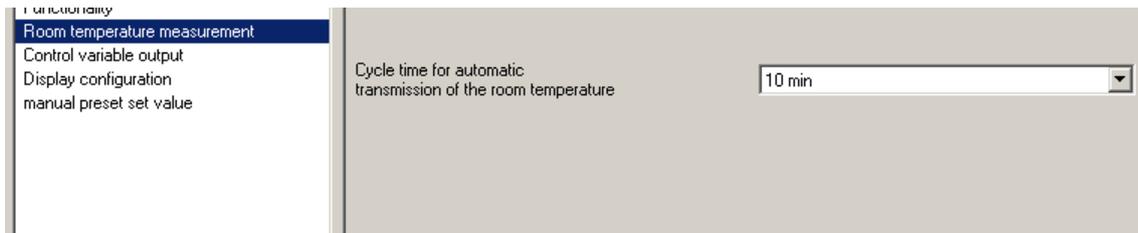
## 7.8. Connections

The KNX connection should auto connect. Room bindings can be used to assign the thermostat to the rooms

KNX Thermostat Feller RTH				
Name	Type	Connection	Input/Output	Connected To
<b>Control Inputs</b>				
KNX Control	Control	KNX_DEVICE	Input	KNX Network->KNX Device
<b>Room Control</b>				
Room Selection	RoomControl	TEMPERATURE	Output	Fix Room->Temperature, C4 KNX Test->Temperat...
Room Selection	RoomControl	TEMPERATURE_CON...	Output	Fix Room->Temperature Control, C4 KNX Test->T...

## 8. Additional settings in ETS

Right click on the air-condition module and select *Edit parameters*. Click on *Room temperature measurement* and change *Cycle time for automatic transmission of the room temperature* to 10 min. (or similar)



## 9. Known issues

- Depending on what language the system is set temperatures and setpoints are maybe not shown with decimal values (22.5 C). en\_US and de\_CH are tested to fully work, it\_IT, de\_DE and fr\_FR do not show decimal places
- Changes on the extra page setting will not be shown on Flash Navigators without refreshing them over *refresh Navigators*
- Temperature on Navigators may not refresh if Thermostat is selected for over 5 days
- List navigators are often not showing the setpoints; also if cooling is active adjusting the heat setpoint will not work. Same for heating and cool setpoint.
- Programming: Using the variables SINGLE\_SETPOINT\_C and SINGLE\_SETPOINT\_F the value needs to be multiplied by 10. So if the setpoint should be 25C then you set it to 250.0
- With OS2.7.2 the single setpoint cannot be set in Programming in the thermostat page. Use the device variable instead
- SR-250 and SR-260 may not show the single setpoint. Only heat setpoint which is equal to the single setpoint

## 10. Updates / Support

Updates are available on: <https://technet.genesis-technologies.ch>

Bugs should be reported to: [info@genesis-technologies.ch](mailto:info@genesis-technologies.ch)