

“11911_HeizungforFellerEDIZIODue.hsl”

Connecting Feller’s EDIZIODue to GIRA’s QuadClient

Overview

This logic handles the connection between a Feller EDIZIODue and the Gira HS QuadClient template 1-13. It handles the Standby, Comfort and Night modes. The Anti-freeze mode hasn’t been implemented as I think it is better handled from a central group address rather than from the QuadClient directly. Yet, such an extension would be trivial to make, feel free to send your modifications as pull requests on the git repository hosting the logic.

The logic uses timers to make sure it sends the state transition messages to Feller in the right order: sending them in a random order could generate visible transition glitches on the display.

Last but not least, since the mode numbers are not the same on the 1-13 template and on the Feller, this module handles the proper mapping.

Description

This logic requires two internal communication objects to be created in Gira for each Feller that needs to be attached to the QuadControl. These objects will be used to i) display the proper status in the QC template (Status) and ii) to receive the QC control request (Control).

Then, the logic will consume and feed the following KNX Group Address:

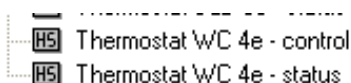
1. The Feller Status Confirmation object (object 61)
2. The Feller Comfort object (object 47)
3. The Feller Night object (object 48)

(each of those object will have to be in a separate GA).

Setup

1) Create the Internal communication objects

Create a Control and Status communication object for the Feller you want to control in QC:



Data	
Designation:	Thermostat WC 4e - control
Data type:	8-bit (0..255/EIS 2,6)
Retentive memory:	<input type="checkbox"/> Yes, save communication object retentively.
Initialisation value:	0
Min value:	0
Max value:	255
Step size:	0
List:	
Group address:	
Receive communication object	<input type="checkbox"/> Yes
Send communication object	<input type="checkbox"/> Yes

Data	
Designation:	Thermostat WC 4e - status
Data type:	8-bit (0..255/EIS 2,6)
Retentive memory:	<input type="checkbox"/> Yes, save communication object retentively.
Initialisation value:	0
Min value:	0
Max value:	255
Step size:	0
List:	
Group address:	
Receive communication object	<input type="checkbox"/> Yes
Send communication object	<input type="checkbox"/> Yes

IMPORTANT: Make sure to use a EIS 2.6 datatype (or similar), but not an RTC status type.

2) Validate Feller Status Confirmation object in GIRA Expert

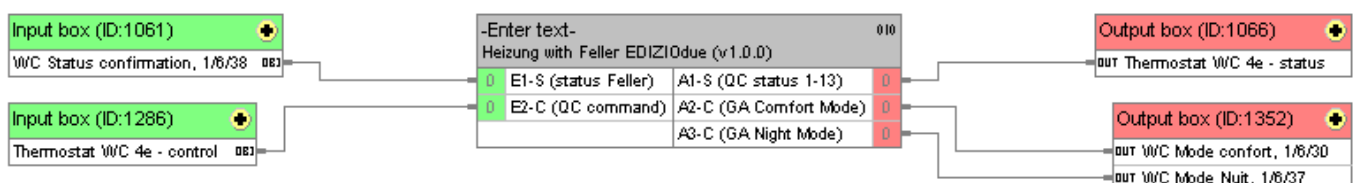
Go in Gira Expert and check the Feller Status Confirmation GA (this is the GA that gets updated by Feller whenever the mode gets changed):

Data	
Designation:	Cagibi Status confirmation
Data type:	8-bit (0..255/EIS 2,6)
Retentive memory:	<input type="checkbox"/> Yes, save communication object retentively.
Initialisation value:	0
Min value:	0
Max value:	255
Step size:	0
List:	
Group address:	1/6/68
Query during starting:	<input checked="" type="checkbox"/> Yes
Only send in case of change	<input type="checkbox"/> Yes
Read flag:	<input type="checkbox"/> Yes
Transfer flag:	<input checked="" type="checkbox"/> Yes
Receive communication object	<input type="checkbox"/> Yes
Send communication object	<input type="checkbox"/> Yes

You have to verify that i) the Data Type is set to EIS 2.6 (and not RTC Status) and that ii) the Query during starting is set to “Yes”.

3) Setup the logic in the Logic Editor

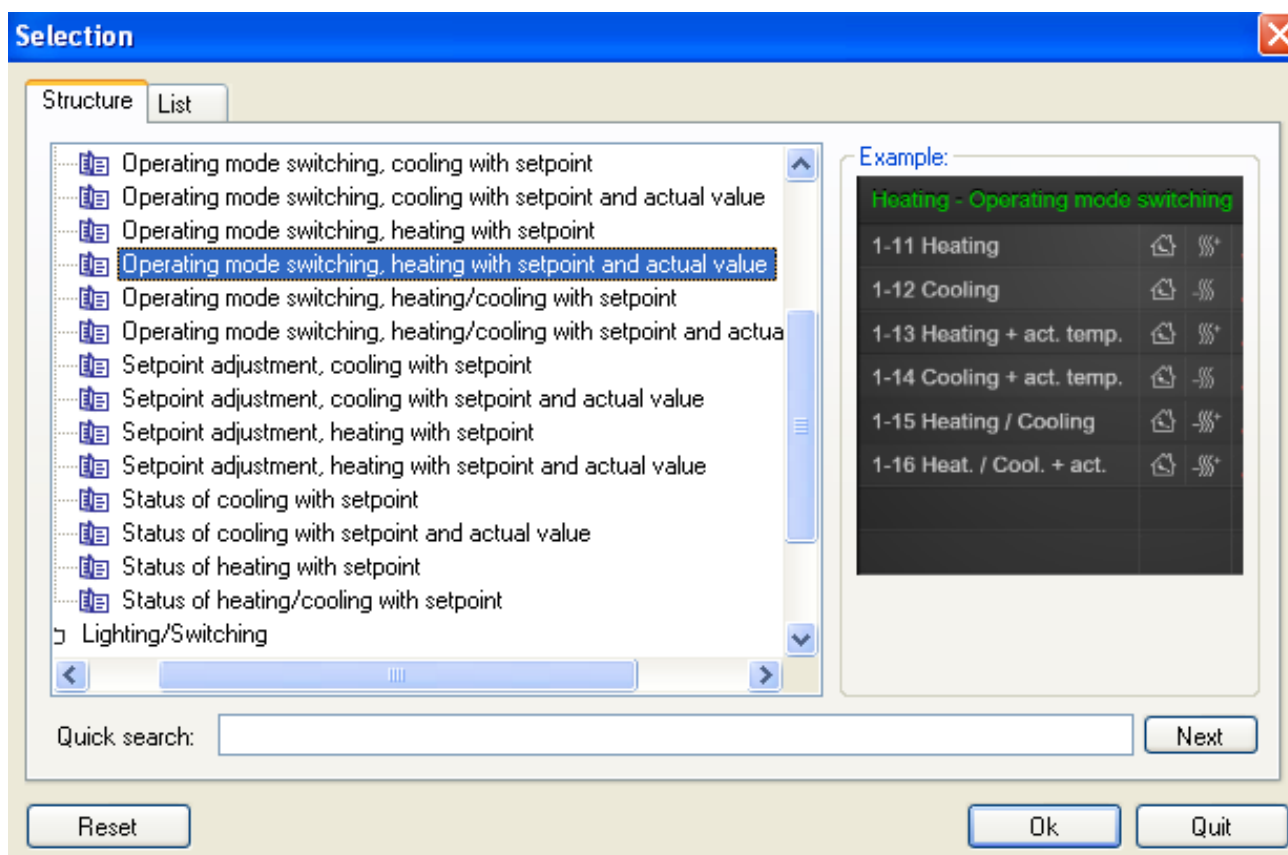
Go in the logic editor, and wire the proper GA and internal GA properly:



IMPORTANT: Make sure to use TWO separate Input Boxes for the two Input: do not group the two GA in the same Input Box.

4) create your Function in GIRA QuadConfig tool

In GIRA Expert, go in the QuadConfig tool, and create a 1-13 template:



Then, feed the template with the proper GA and internal GA create previously:

General	
Text in room	Thermostat
User groups	All user groups
Export to iPhone®/iPad®	<input checked="" type="checkbox"/>
Function	
Text	Building structure\4e étage\WC\Thermostat WC 4e
Function template	Operating mode switching, heating with setpoint and actual value
Use time clock	<input type="checkbox"/>
Interfaces	
RTC status	Thermostat WC 4e - status
Operating mode switching as per KNX (1 byte)	Thermostat WC 4e - control
Heating message	WC Chauffeur (Heizen) (1/6/33)
Party/Comfort extension	WC Mode confort (1/6/30)
Setpoint temperature	WC Valeur de consigne pour la temp. ambiante (1/6/35)
Actual temperature	WC Température de la pièce (1/6/36)

And you should be all set!

License

This piece of logic is licensed under the LGPL. In summary: use it as much as you want (including commercially), but if you are going to modify it, provide the source code of your modifications so others

can benefit from it.