Technical Manual

BX-MFB012

CONTENTS

1 General information

BX-MFB12 is a 12-channel multifunctional relay actuator.
Maximum current delivery from each channel is 16 A at 230VAC.
The device is ETS-programmable to perform a number of functions:

- control lights and other utilities
- lift and lower shutters and blinds
- pilot valves with PWM actuation
- manage 2- or 4-pipe fancoils

This device can be used for both home and industrial applications.

2 Product Features

2.1 Technical specifications

The actuator is powered by a Konnex bus via latest-generation Siemens TPUART2 - whose excellent performance in terms of current delivery enables to reach high switching speeds.
The IP20 casing is pre-set for installation on a DIN 35mm bar (DIN EN 60715).
Its width footprint is equal to 8 18mm modules.
It is equipped with 16A toggle-type relays with contact directly connected to the terminals - with no phase-sharing.
The screw-type terminals are able to accept cable cross-sections up to 5 mm^2.
The relays can be manually controlled via the keypad provided on the device front, complete with contact state indicator LEDs.
The relays used can withstand an inrush current up to 170A for the first 2ms, therefore, they are particularly suitable for piloting inductive loads typical of fluorescent or neon lamps.

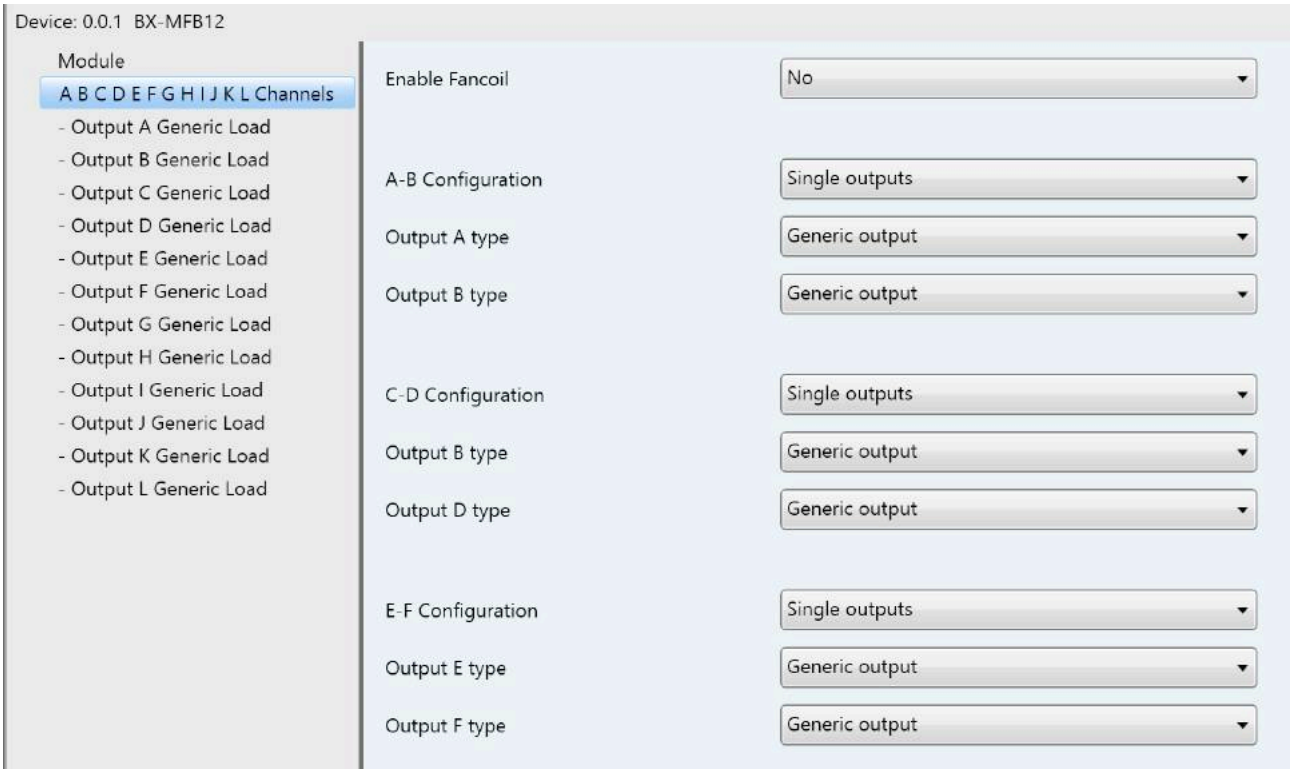
3 ETS library

The ETS Library features a number of parameters used to characterise the operation of each actuator output.
These parameters are conveniently divided into twelve pages dedicated to each channel configuration, to which two main pages are added, necessary to assign the function for which each Relay is intended.

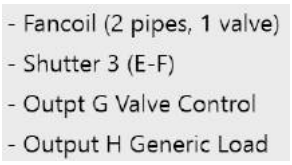
3.1 Module

This is the main page where the **Modify Parameters** window is opened.
It contains a header showing the code of the KNX device to which the library is dedicated.
The only purpose of this page is allowing the user to decide whether or not to enable **Logic Functions**.
Logic Functions are located on this general-purpose page because they are 16 resources freely assignable to any channel.
The Logic Functions are only activated when the relay output is programmed for a generic load, i.e. when the relay output is not used to carry out a specific function, such as shutter operation or valve control.
Despite this limit, enabling Logic Functions determines, however, the appearance of the communication objects designed for control variable reading.
The enabling of a logic function implies that the relay output state is no longer controlled by the switching communication object, but by the result of the enabled logic function with switching communication and logic function objects at its input.
For further details please refer to the dedicated section.

3.2 A B C D E F G H I J K L Channels

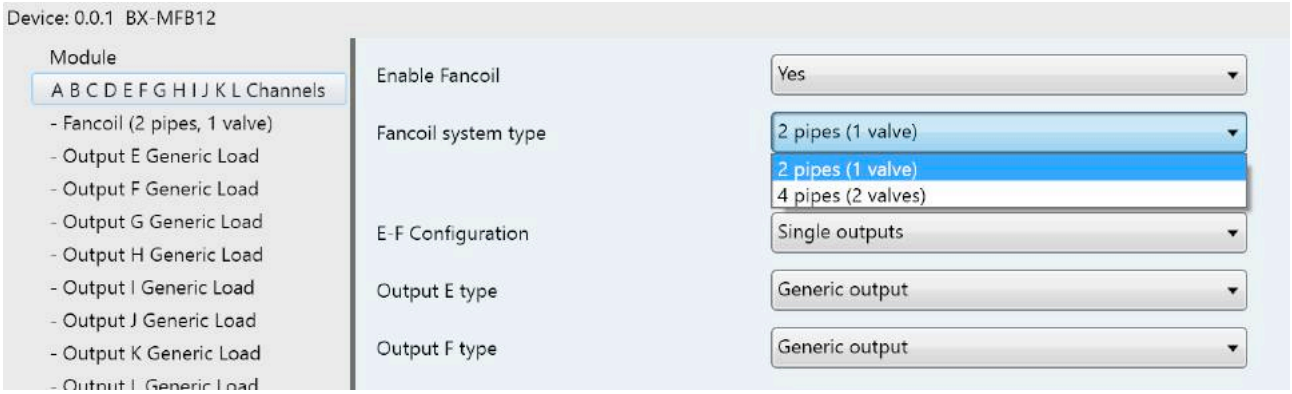


The **Channels** page enables to assign each channel to a dedicated function.



The various choices will determine a dynamic variation of the side menu with the activation of the pages necessary for the configuration of the various functions: Fancoil, Shutter, Valve Control and Generic Load.

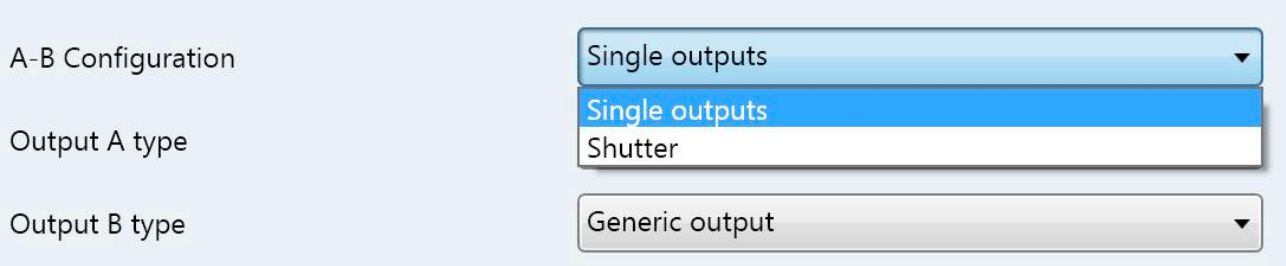
The first option should be used to decide whether or not to assign the first channels to the **Fancoil** function.



Enabling the Fancoil function determines the assignment of the first 4 or 5 channels to this function.
There will be 4 channels when controlling a 2-pipe system is specified.
A 2-pipe system only has one fancoil input line.
In particular, the channels A B C D are allocated.

There will be 5 channels when controlling a 4-pipe system is specified.
A 4-pipe system has two separate input lines, one for cooling and the other one for heating.
In this case, the channel E is also allocated to control the second input line

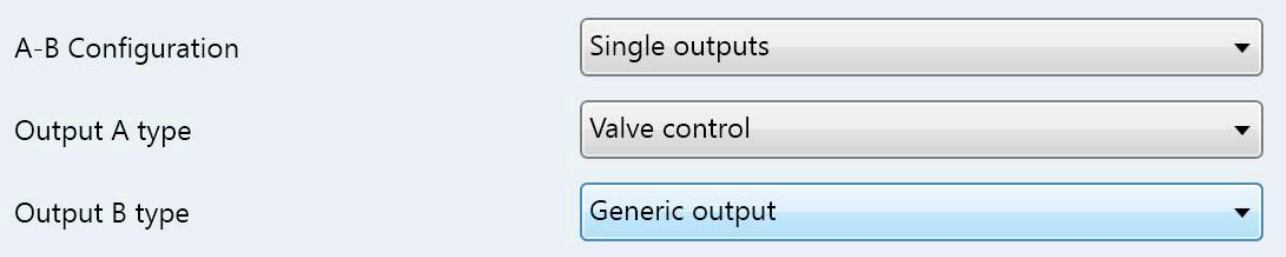
The second option concerns the possibility of deciding whether a pair of outlets should be used to control a **Shutter**.



Dedicating two outputs to shutter control means to have them work in a mutually exclusive manner, i.e. without ever being able to simultaneously close the outputs of both relays.
This important precaution is necessary to prevent the opposite motor windings from becoming burnt.

As an alternative to the Shutter mode, the channels can work as independent outputs if the **Single Outputs** mode is assigned.

This is the typical option enabling to use a single channel to control a load, by



assigning the **Generic Output** default option to the Output X type field.
As another, final alternative, there is an option enabling to allocate a generic output to **Valve control** by using the PWM algorithm.
The PWM (Pulse Width Modulation) algorithm enables to use a standard ON/OFF valve as a linear valve, so as to work with a communication object that has per cent data (Data Type 5).

3.3 Fancoil

3.3.1 Operating principle with external Thermostat

The Fancoil module is a device designed to control the operation of a thermal device and its ventilation system in safe conditions, by guaranteeing interlocking of the channels used.
Operation cycle management according to environmental conditions is ensured by an external Thermostat.
The channels A, B and C are reserved for ventilation speed piloting, while the channels D and/or E are reserved for input valve opening.
The ventilation speed value determines the heat exchange intensity and therefore influences the amount of energy transferred to or received from the environment.
The input valves can be preliminarily opened to guarantee ventilation efficiency and preliminarily closed to remove condensation.
The valve opening and ventilation speed are controlled by the external Thermostat.
Ventilation control can be achieved by directly sending the required V1, V2 or V3 speed setting control signal only or by despatching a per cent value signal indicative of the required operation intensity.
By contrast, the input valve opening and closing controls must be sent explicitly.

If the **2 pipes** option has been enabled, there will be a single input valve with its corresponding communication objects available for switching and reading its

3	Fancoil	Set valve	1 bit	C	R	W	-	-	on/off		Low
7	Fancoil	Status valve	1 bit	C	R	-	T	-	on/off		Low

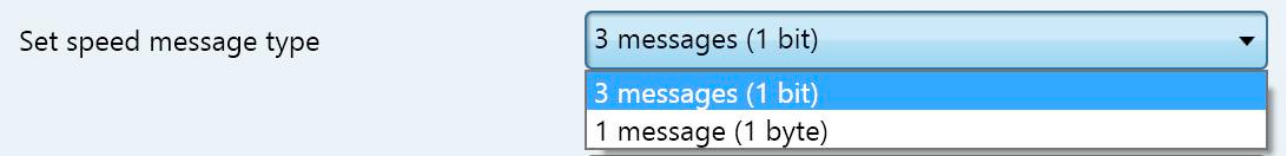
status.

If the **4 pipes** option has been enabled, there will be two, mutually exclusive input valves, one intended for **Heating** and one intended for **Conditioning**, complete with their communication objects available for switching and reading their status.

7	Fancoil	Heating valve status	1 bit	C	R	-	T	-	on/off		
3	Fancoil	Heating valve set	1 bit	C	R	W	-	-	on/off		
56	Fancoil	Conditioning valve status	1 bit	C	R	-	T	-	on/off		
48	Fancoil	Conditioning valve set	1 bit	C	R	W	-	-	on/off		

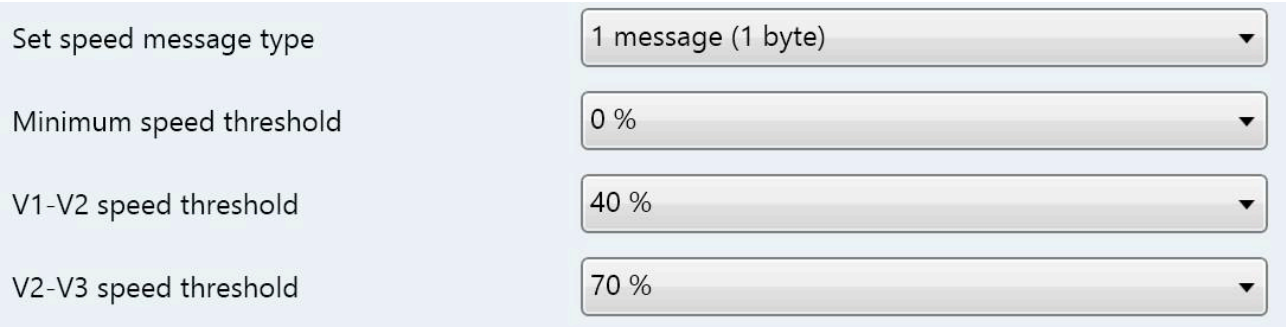
3.3.2 Configuration

The first and most important parameter to set in the Fancoil page is the format of



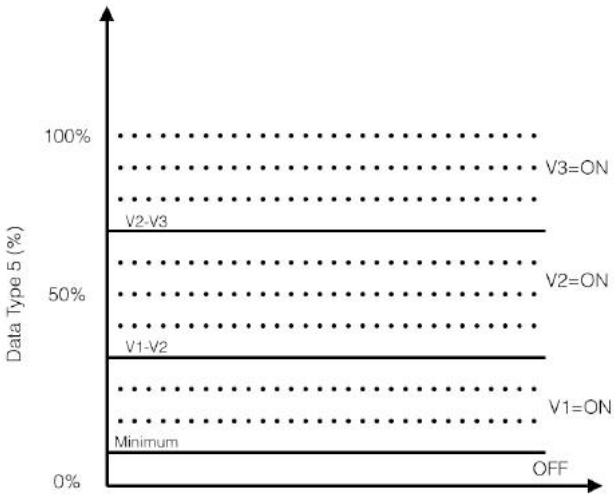
the telegram message with which the Thermostat sends its ventilation request.

If the **1 message (1 byte)** option is selected, the request will be submitted in a per cent format (Data Type 5), where 100% refers to a maximum ventilation request and 1% to a minimum ventilation request.
In this case, it will be up to the actuator to convert the per cent information into the 3 different speed settings according to the indications contained in the



programming parameters.

Minimum speed threshold is the lowest threshold below which the Fancoil is turned off.
V1-V2 speed threshold is the threshold below which V1 is turned on and above which V2 is turned on.
V2-V3 speed threshold is the threshold below which V2 is turned on and above which V3 is turned on.



In this configuration the communication objects that are active to receive external Thermostat indications

3	Fancoil	Set valve	1 bit	C	R	W	-	-	on/off	Low
7	Fancoil	Status valve	1 bit	C	R	-	T	-	on/off	Low
148	Fancoil	Set speed %	1 Byte	C	R	W	-	-	percentage (0..100%)	Low
149	Fancoil	Status speed %	1 Byte	C	R	-	T	-	percentage (0..100%)	Low

are 3, 7, 148 and 149, according to the Table here below.

If the option **3 Message (1 bit)** is selected, the request will be submitted in the ON/OFF format (Data Type 1) directly to the communication object that will switch the corresponding relay at the required speed.

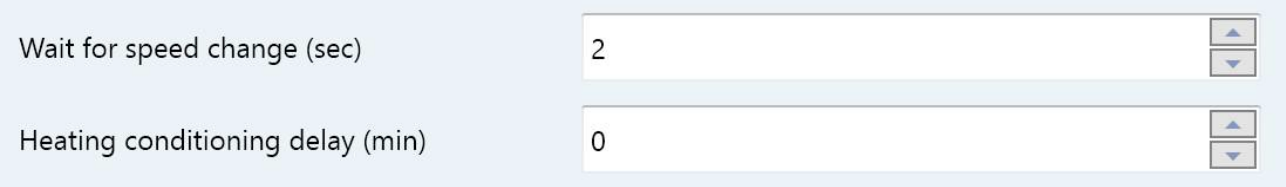
In this case, the device will preliminarily disable the relay that had been previously closed to guarantee interlocking of the motor winding control channels.
The Table below shows the Communication Objects active in this configuration.

Number	Name	Object Function	Description	Group Addresses	Length	C	R	W	T	U	Data Type	Priority
0	Fancoil	Set speed V1			1 bit	C	R	W	-	-	on/off	Low
1	Fancoil	Set speed V2			1 bit	C	R	W	-	-	on/off	Low
2	Fancoil	Set speed V3			1 bit	C	R	W	-	-	on/off	Low
3	Fancoil	Set valve			1 bit	C	R	W	-	-	on/off	Low
4	Fancoil	Status speed V1			1 bit	C	R	-	T	-	on/off	Low
5	Fancoil	Status speed V2			1 bit	C	R	-	T	-	on/off	Low
6	Fancoil	Status speed V3			1 bit	C	R	-	T	-	on/off	Low
7	Fancoil	Status valve			1 bit	C	R	-	T	-	on/off	Low

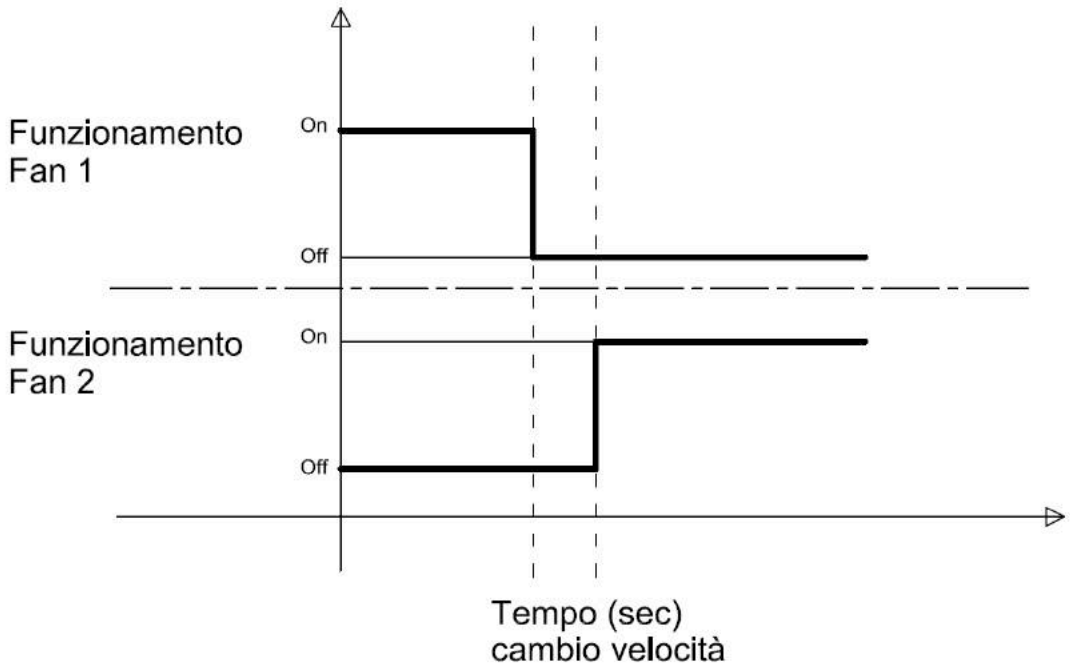
The Tables shown refer to a circuit with a single valve.

3.3.3 Safety functions

The configuration of Fancoil-dedicated parameters also makes it possible to enable a number of safety functions.



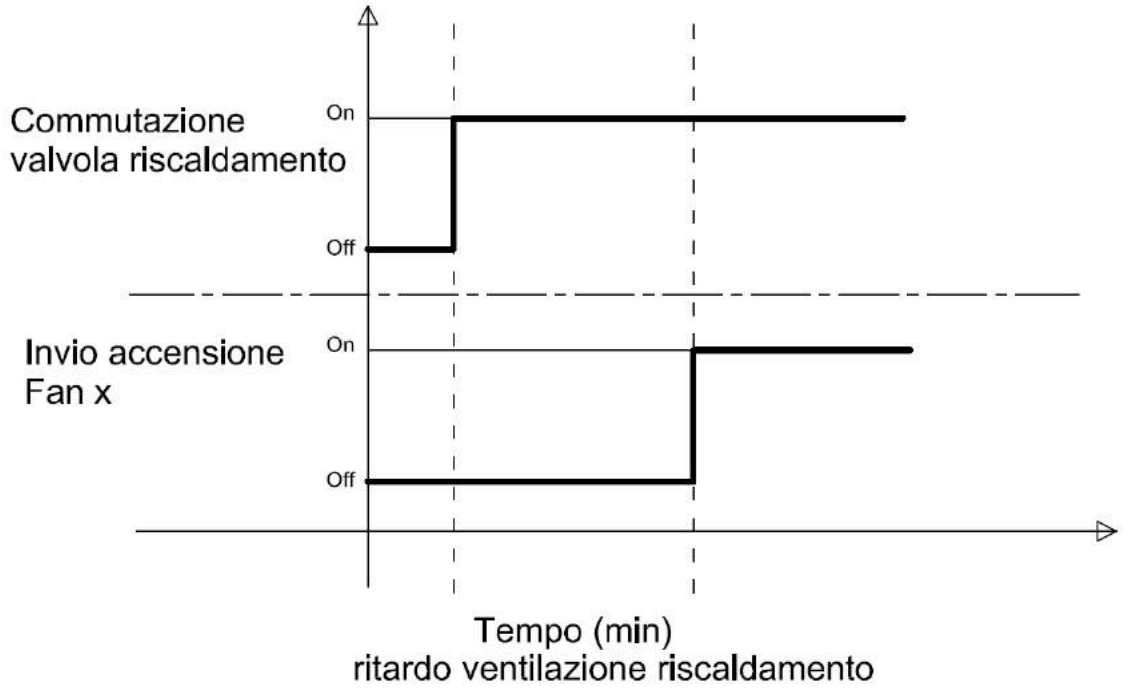
The first one, named **Wait for speed change (sec)** makes it possible to allow for a safety delay between disabling the old speed and enabling the new speed when the ventilation speed is changed.



Fan 1 Operation
Fan 2 Operation
Speed change time (sec)

This delay is even prescribed by some ventilation system installation manuals as it guarantees a longer life for the motor.
In Blumotix modules, this parameter default setting is 2 seconds.

The second function is named **Heating conditioning delay (min);** it is a protection system that is activated on turning on the heating valve, and it prevents ventilation until the heat exchanger circuit has warmed up.
This protection device ensures that no cold air is ventilated due to split heating-up failure.



Heating valve switching
Fax x power-on send
Heating ventilation delay time (min)

When ventilation override is activated, the device will keep any changes received on the KNX bus stored in its memory and re-activate the current ventilation value once the pre-set delay time has elapsed.

3.3.4 General parameters

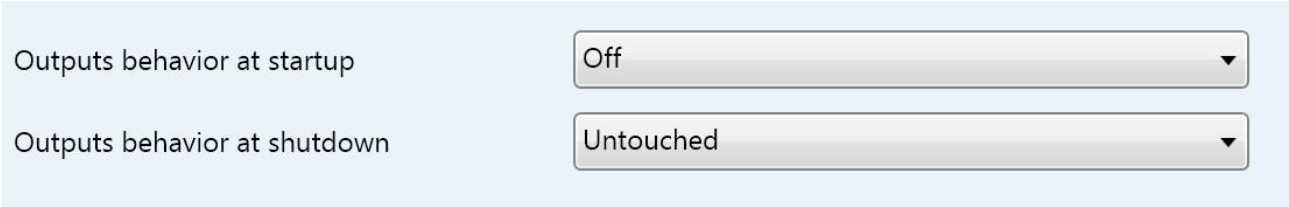
On the subject of input valve use, please be reminded that it is possible to define whether the Relay contact should be normally open (by default) or closed.
This choice enables to adjust the Relay operation to the type of valve installed in the system.

Valve relay contact type	Normally open

We will have a single valve configuration when the 2-pipes option is enabled
or we will have a twin configuration for heating and conditioning when the 4-pipes option is enabled.

Heating valve relay contact type	Normally open
Conditioning valve relay contact type	Normally open

Finally, there are two parameters to tell the device how to behave when it is powered on (**Outputs behavior at startup**) and when voltage input to bus KNX is



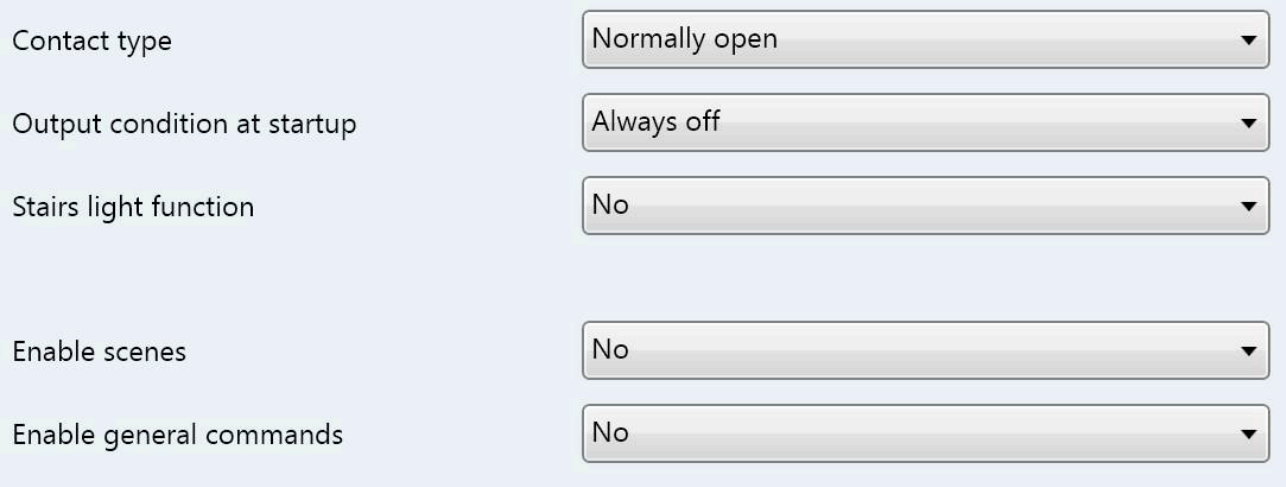
disconnected (**Ouputs behavior at shutdown**).
Upon Startup, the device may set all the outputs to their off position (**OFF**) or it can restore the outputs to their position before power-off (**Previous status**).

At the time of KNX bus voltage drop, the device may also not change the output status (**Untouched**) or it can turn all the controls off (**OFF**).
If the outputs are turned off, this action will not affect the possibility of restoring their active state upon power-on.

The Fancoil function does not involve any scenes.

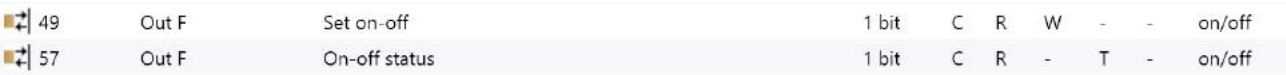
3.4 Generic Load

The load control generic output features the traditional functions to define the Contact Logic, activating the Stairs Light Timer, associating the output with Scenes and as was mentioned earlier on, the possibility of working in conjunction with



Logic Functions.
Contact Type defines whether the Relay contact logic condition OFF must be of the open or closed type. The typical default value is open, i.e. non-conductive circuit.

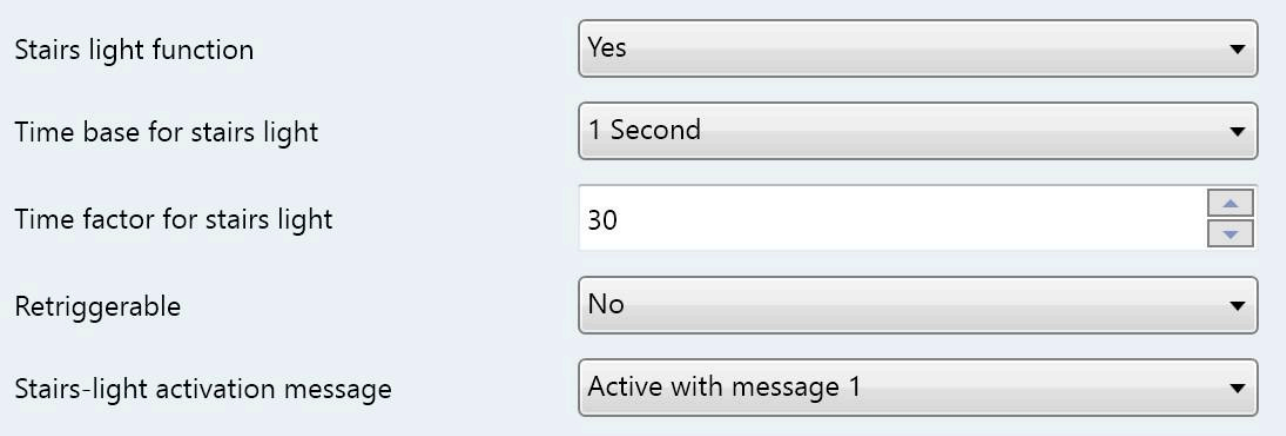
Output condition at startup defines the Relay behaviour upon system power-on.
Always off is the precautionary condition that will keep the output turned off.
Alternatively, it is possible to set the opposite condition, **Always on**.
Or, **Previous condition** can be set to restore the setting existing before the power cut. This condition is especially suitable for use with lighting.



For each output parametrized as Generic Load, two communication objects are always assigned, the first one named **Set on-off** to switch the output, and the second one named **On-off status** to notify the output state changes.

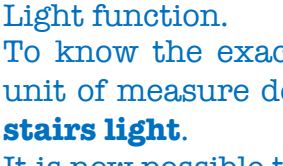
3.4.1 Stairs light function

When the Stair Light function is activated the following configuration parameters



are displayed.

Time factor for stairs light is the parameter defining the duration of the Stairs Light function.
To know the exact value, this factor must be multiplied by the unit of measure defined in the field above, named **Time base for stairs light**.
It is now possible to select a second, minute or hour value.
The product of these two factors is the time during which the Stairs Light remains

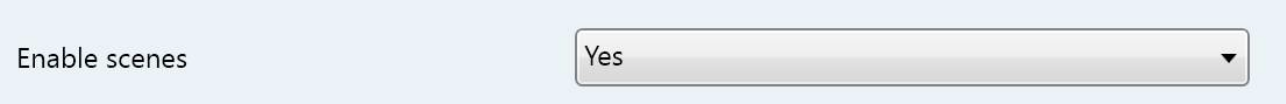


lit.

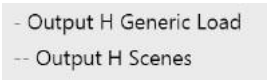
By enabling the Stairs Light function a dedicated communication object is introduced to activate this function.
This means that it will be possible to switch on the Stairs Light in a timed manner with the communication object **Set stair light**, or switch it on permanently with the standard object **Set on off**.
By enabling the **Retriggerable** parameter, you can make the time count start every time that the start telegram message is sent; otherwise, if this function is not enabled, time will continue to run to the end without any possibility of extending the activation time.

Stairs light activation message indicates which value will determine the activation of the Stairs light function. In this, and in other, cases different values may trigger function activation: it could be the 1 value sent when a button is closed or the 0 value generated by the opening of a door contact.

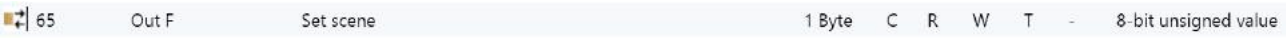
3.4.2 Scenes



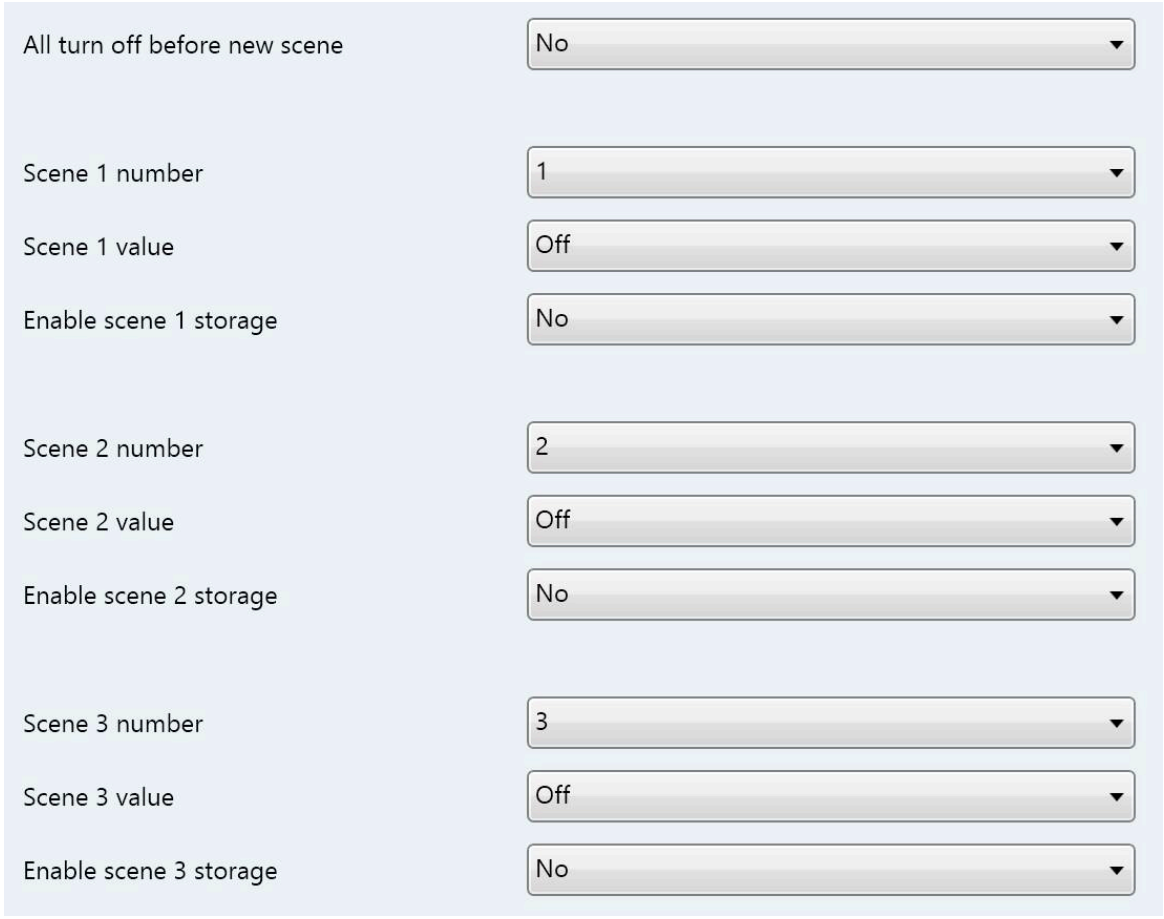
The Generic Load function enables to activate the Scenes KNX.
In this case the side menu is configured for access to the page **Output X Scenes**.



A 'scene' is a pre-determined number sent via the bus to synchronise all the devices that have been activated to recognise it.
It is a very powerful syncing technique for communication objects having different Data Types - as the value to assign to the state has been previously stored to the device as a configuration parameter.
Every channel enabled to use the scenes will have its own 1 byte **Set scene** communication object (Data Type 5), through which it can receive a syncing telegram message containing the number of the scene to refer to for status



assignment.
The available scenes in Konnex are 64, numbered in a sequence from 0 to 63.



The Blumotix actuators enable to assign to each output 8 different actions to assign to one of 64 available scenes.

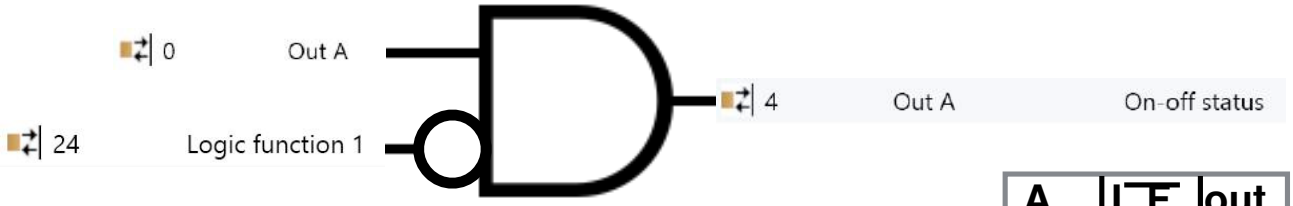
Scene X number defines the scene number to associate with that given action.
Scene X value defines the state to assign to the action that you wish to carry out.

Enable scene X storage allows the memory storage function to be enabled.
It is possible to send via the bus a special control to ask the devices to store their contingent state as the new value to assign to that scene.
The new value will replace the value stored in the configuration parameters.
This option enables the end customer to autonomously configure scenes without having to resort to installation programming.

Finally, something about the first parameter that can be configured in the table.
It is named **All turn off before new scene** and requires that - before the output state can be modified due to a scene activation - the module sets all outputs to the OFF mode.
It is a precautionary measure very similar to the interlocked mode, to stop two relays with opposed functions from becoming closed during the configuration change-over.

3.4.3 Logic Function

As we mentioned earlier on, logic functions are initially only available in the Generic Load mode.
Logic functions enable to make a channel state conditional on the occurrence of certain situations, defined through the configuration of logic operators.
There are 16 generic logic operators available, to be assigned to the desired channels without restrictions.
If a channel is assigned a logic function, its state will no longer be determined by its communication object only, but by the result of the logic operation between its communication object and the one assigned to the logic function named **Logic function X**.

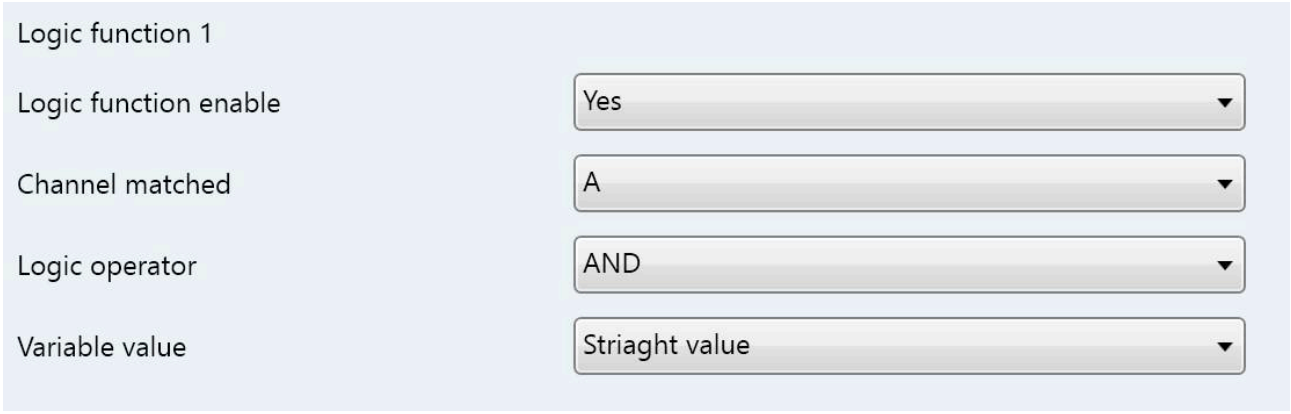


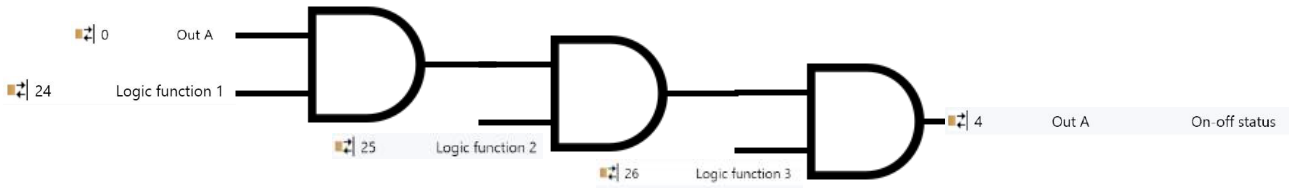
A	L.F	out
0	1	0
1	1	0
0	0	0
1	0	1

At the top we can see a clear example of a locking function.
It is a function obtained by reversing the variable logic input,

modifying the option **Variable value**.

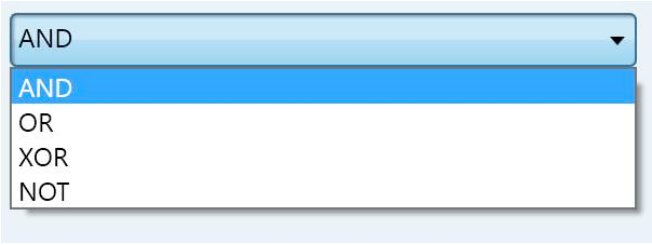
The output A is conditional upon the logic operator AND associated with it.
This means that the output will match the input only if the variable logic is equal to 0 (please compare to the truth table opposite).





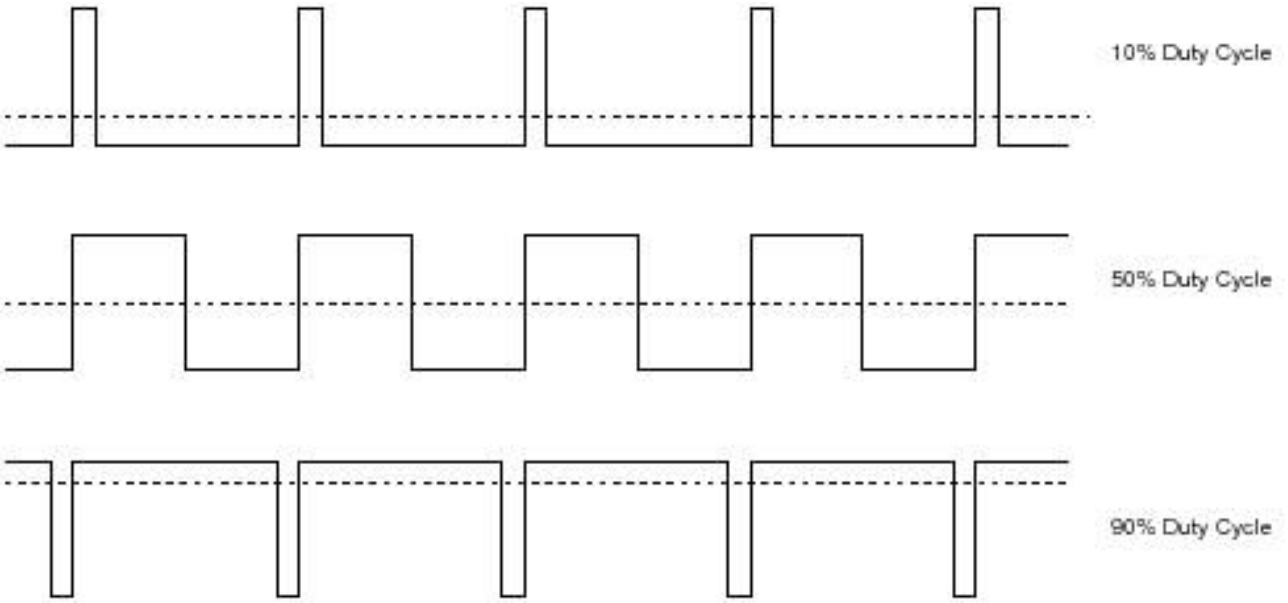
We have already repeated several times that logic functions are resources that can be freely associated with any channel.
Therefore, several may happen to be allocated to the same channel, like in the example here above.
In this case, the final result should be calculated by arranging the ports in series, from the one with the lowest ID to the one with the highest ID, by applying the output of the former to an input of the next one.

The available ports are AND, OR, XOR and NOT.

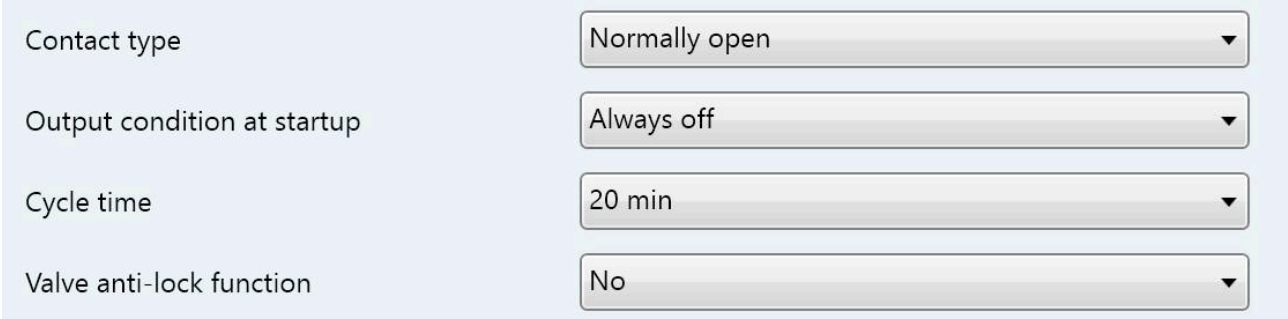


3.5 Valve control

The Valve control option is an alternative to Generic Load.
This option was named after the fact that it is generally used to control the valves feeding water to the dedicated heating radiators.
The idea is to alternate between ON/OFF valve power on/power off to obtain an average value of the water flow matching the desired one.
To do this, the PWM (Pulse Width Modulation) technique is used, consisting in changing the Duty Cycle of the ON/OFF squared wave to obtain the desired average value expressed as a per cent value.



The most important parameter to characterise the operation of the PWM technique is the duration of its cycle which, in our configuration, is named **Cycle time**.



It should be noted that the duration of the cycle does not in any way influence the hot water flow, but can improve the plant efficiency.
The liquid supply only and exclusively depends on the Duty Cycle, therefore, it can always range between 0% and 100%.
The duration of the cycle shouldn't be too short to prevent the valve becoming damaged due to excessive switching, or too long to prevent the radiator cooling down.

40	Out valve A	Set duty cycle %	1 Byte	C	R	W	-	-	percentage (0..100%)
44	Out valve A	Duty cycle % status	1 Byte	C	R	-	T	-	percentage (0..100%)
0	Valve A	Activation	1 bit	C	R	W	-	-	on/off
4	Valve A	Activation status	1 bit	C	R	-	T	-	on/off

The communication objects featured by the device to manage the procedure are really simple and user-friendly:
Valve X is the switch turning PWM control on and off,
Set duty cycle % is the percentage value (Data Type 5) regulating the Duty Cycle, i.e. the water to radiator flow rate.

Both these functions have a corresponding communication object to notify state changes.

3.6 Shutter

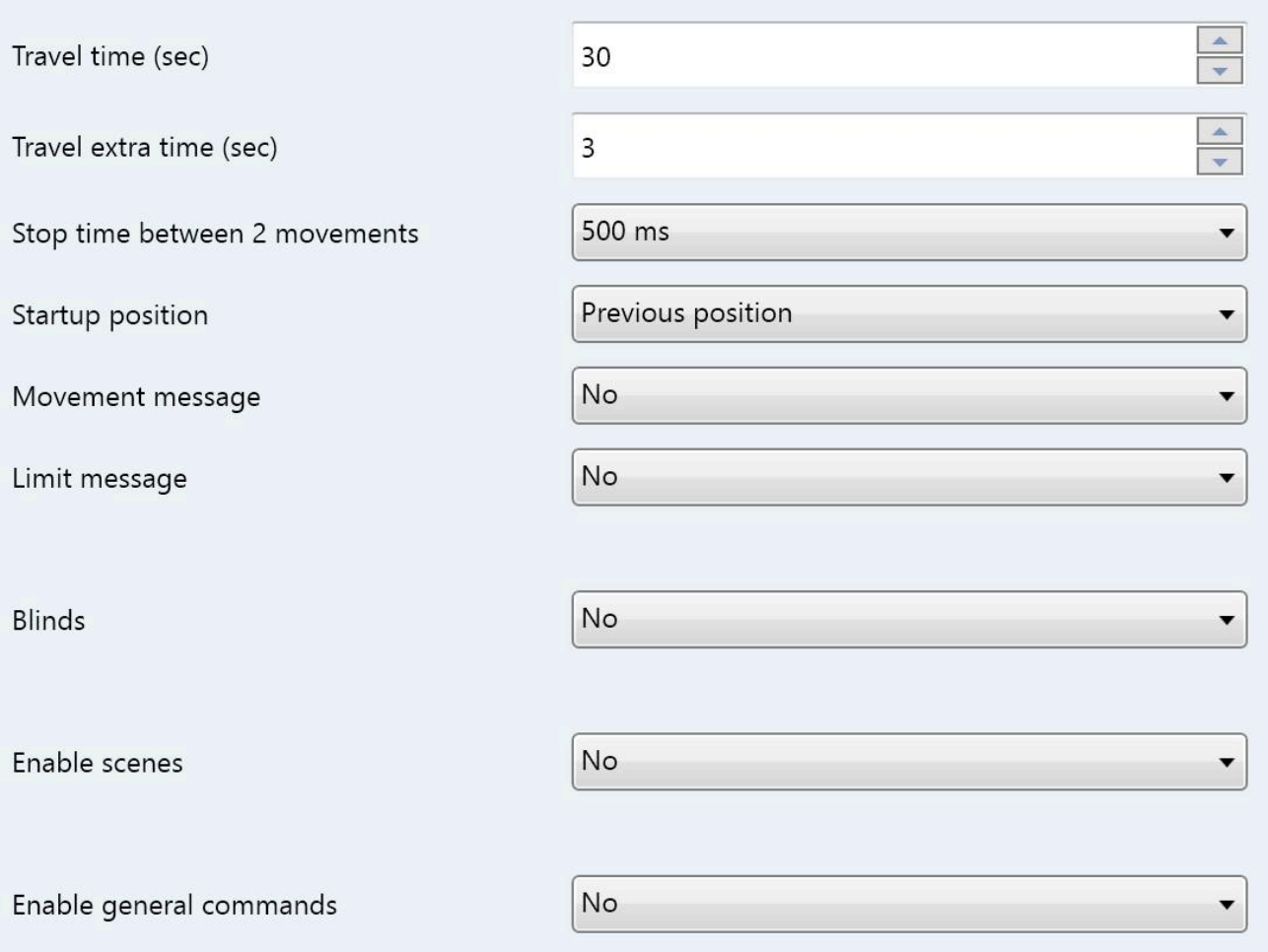
The last subject to deal with concerns rolling shutters and blinds management.
To better understand the entire procedure, we should always bear in mind that this device controls the operation of an AC motor, with two opposed windings to be powered individually in order to control the direction of rotation.
Therefore, according to the procedure, two relays should be reserved for each motor control: one to actuate the movement in the clockwise direction and the other one to actuate the movement in the anti-clockwise direction, programmed so that they can never be closed simultaneously to prevent circuit blowing.
The first relay in the pair is the one designed to raise the shutter, while the second relay in the pair is intended to lower the shutter.
Note that there is no relationship between the motor direction of rotation and the shutter raising or lowering effect, as what happens exclusively depends on how the motor is installed.
The management of shutters in a Konnex environment requires that two objects of

12	Shutter 1 (A-B)	Set up/down	1 bit	C	R	W	-	-	up/down
14	Shutter 1 (A-B)	Up/down status	1 bit	C	R	-	T	-	up/down
16	Shutter 1 (A-B)	Set % position	1 Byte	C	R	W	-	-	percentage (0..100%)
18	Shutter 1 (A-B)	Position % status	1 Byte	C	R	-	T	-	percentage (0..100%)
20	Shutter 1 (A-B)	Set stop	1 bit	C	R	W	-	-	1-bit

communication are featured, designed to control motor movement and motor stop.
Set up/down is the control setting the motor in motion; if the 0 value is assigned, the shutter must be raised, while if the 1 value is assigned, the shutter must be lowered.
When the shutter is operated, its movement continues until it reaches the end of its stroke, or until a stop control is applied.
The shutter can also be set in motion by specifying the position where it should stop. This action can be performed by using the communication object **Set % position** with which it is possible to indicate the destination position in percentage terms - where 0% is the maximum opening position and 100% is the total closing position.
Set stop is the control used to control a motor stop, whichever value is attached to the telegram.

These 3 communication objects to control shutter movement have 3 corresponding items to notify status changes.
In particular, **Position % status** notifies the shutter change of position once, when the motor stops.

The shutter position is calculated by the module starting from the known travel time to movement completion.
This information should be programmed in the configuration parameters by assigning a value to the field **Travel time (sec)**.
It is always advisable to calculate the up-stroking time which is generally slightly slower as the motor must work against gravity.
It is also possible to assign the **Travel extra time (sec)** field to compensate any delays.
The extra time will not be considered in the position calculation.
Finally, it is possible to assign a pause between one movement and the next when a change of direction is performed, by programming the **Stop between 2 movements**



field. This is a precaution recommended by most motor manufacturers.
Startup position is a parameter designed to check the shutter position upon system start. For certain designers, it is preferable that, upon system start, all the shutters are in their closed position due to anti-intrusion concerns, while for others, the all shutters open position is preferable due to safety issues - to make premises evacuation easier.

3.6.1 Interaction with curtain movement

Blumotix makes a few parameters available to better illustrate the state of its actuator and improve its efficiency when used to control indoor curtain movements.
In particular, we enabled a few communication objects to notify when the motor is in motion and when it reaches its travel limit.
This information is very useful to stop the automatic movement of a transom window, for example, from being completed at an unsuitable time damaging the curtains. In this case, the curtain fully open information can be used to enable

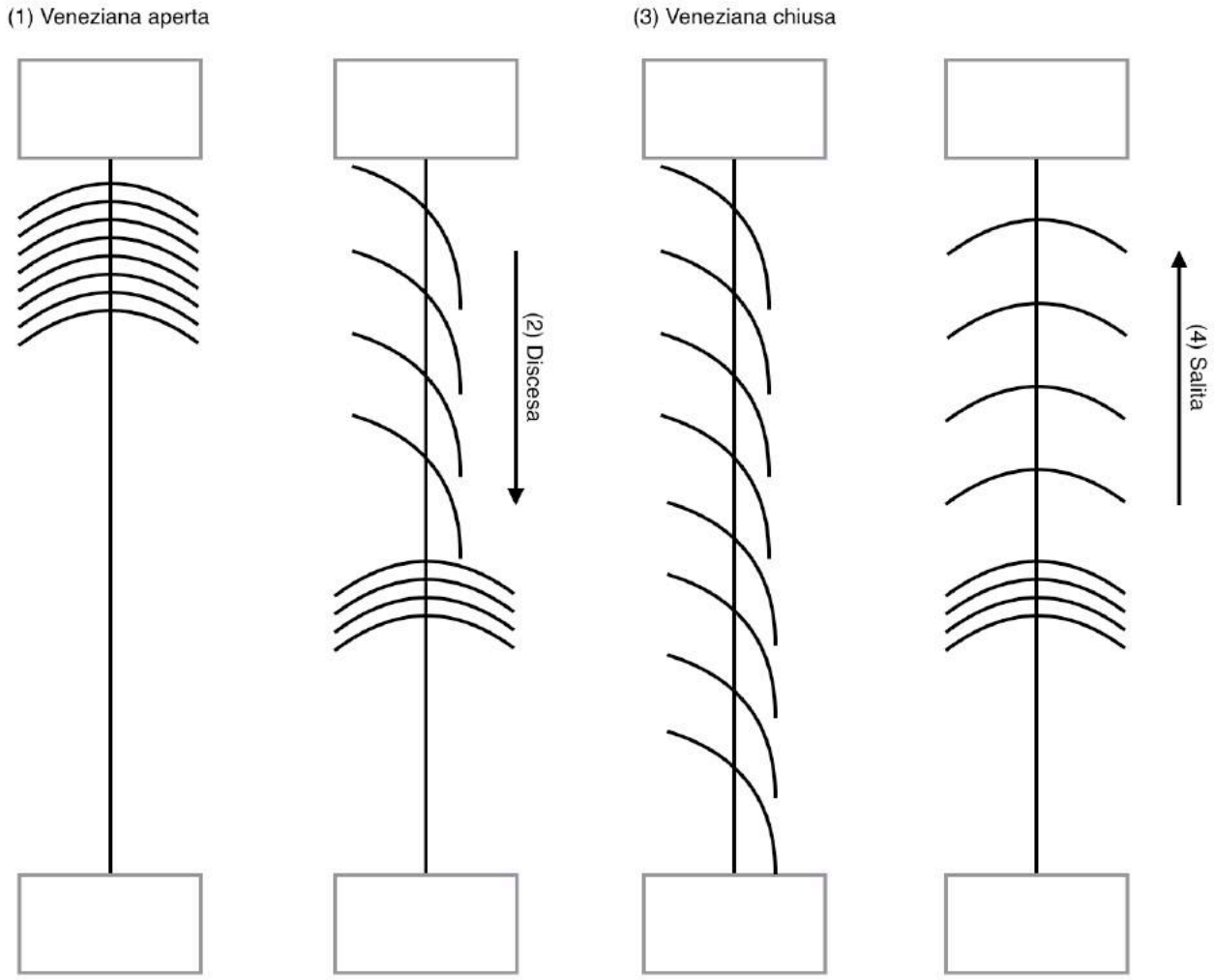
112	Shutter 1 (A-B)	Moving up	1 bit	C	R	-	T	-	boolean
113	Shutter 1 (A-B)	Moving down	1 bit	C	R	-	T	-	boolean
124	Shutter 1 (A-B)	Up limit	1 bit	C	R	-	T	-	boolean
125	Shutter 1 (A-B)	Down limit	1 bit	C	R	-	T	-	boolean

transom window opening without any risk of causing damages.
The **Movement message** field enables to actuate the communication objects **Moving up** and **Moving down**, through which the module notifies when it is moving up or down.
The assigned values are Boolean-type values, therefore, they should be assumed to have a true or false meaning.

The **Limit message** field makes it possible to enable the **Up limit** and **Down limit** communication objects, through which the module notifies whether it has reached its travel limit in its high or low position.
In this case, too, the assigned values are Boolean-type values, therefore, they should be assumed to have a true or false meaning.

3.6.2 Venetian blinds

The **Blinds** field is used to enable Venetian blind control.
Venetian blinds are a special blind design with adjustable elements to block out part of the light.
These daylight-control elements are called slats.
The slat movement is not autonomous, as it is controlled by the same mechanism lifting and lowering the Venetian blind.
Usually, in this design the individual elements are gathered into packs of slats before going up and then released before going down.
Creating packs of slats (see 4-Up stroke) means positioning the slats horizontally (fully open position corresponding to the 0% value), so that when the blind is at rest in its top position, they take up the smallest possible amount of space.
Releasing the slats (see 2-Down stroke) means positioning the stacks vertically, blocking out the light completely (fully closed position corresponding to the 100% value), to improve the light control effect when the Venetian blind is closed.
Therefore, the slats only move when a change of direction is performed in the movement to lift and lower the blind.



(1) Blind open (2) Down-stroke (3) Blind closed (4) Up-stroke

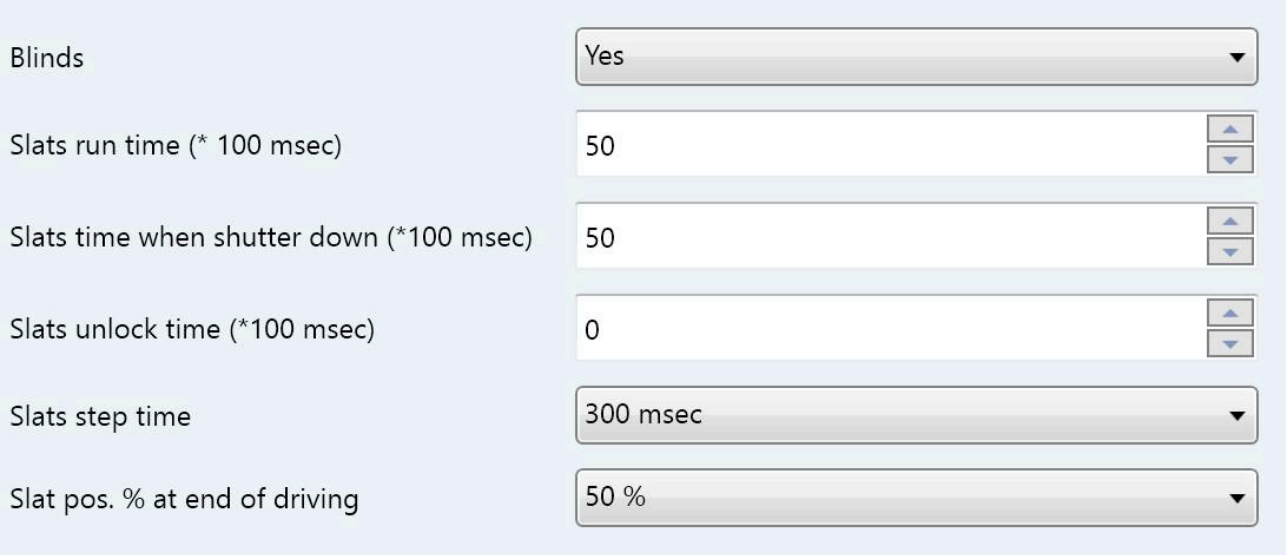
This operation mode generally allows for a slat movement reduction by approximately 90°
There are special models which, after reaching the fully lowered position, release the slats from the fixings allowing for up/down movements, so as to allow for much wider positioning angles up to 180°.

How can the slats movement be controlled?
The same motor actuator must be used to move the Venetian blind up and down.
The Set stop control is used.
If the Blinds control is enabled the Set stop control has a twofold function:
if the motor is moving it always causes it to stop,
while if the motor has stopped every incoming telegram generates a small motor movement.
If the movement is in the opposite direction with respect to the direction followed until then, it will cause a limited-extent slats repositioning.
In this way, slats can be positioned by small, programmable length steps.

20	Shutter 1 (A-B)	Stop/slats step	1 bit	C	R	W	-	-	1-bit

In this case, the Set stop control is named **Stop/slats step**.
By sending 0 an upward movement is controlled to open the blind, causing the slats to switch from their closed (100% darkening) to their open (0% darkening) position.
By sending 1 a downward movement is controlled to close the blind, causing the slats to switch from their open (100% darkening) to their closed (0% darkening) position.
The Slats movement is consistent with the Venetian blind movement: 0 to open and 1 to close.

We are illustrating here below the meaning of the parameters available in the configuration panel.



Slats run time (* 100 msec) is the parameter defining the duration in milliseconds of the slats movement following a change of direction.

We mentioned that some Venetian blinds after closure allow for a wider movement of the slats.
In this case, two new parameters must be defined, too, to correctly characterise the end part of the movement.
Slats time when shutter down (* 100 msec) defines the duration in milliseconds of the wider slats movement and **Slats unlock time (* 100 msec)** is the time necessary to release the slat mechanism fixings.

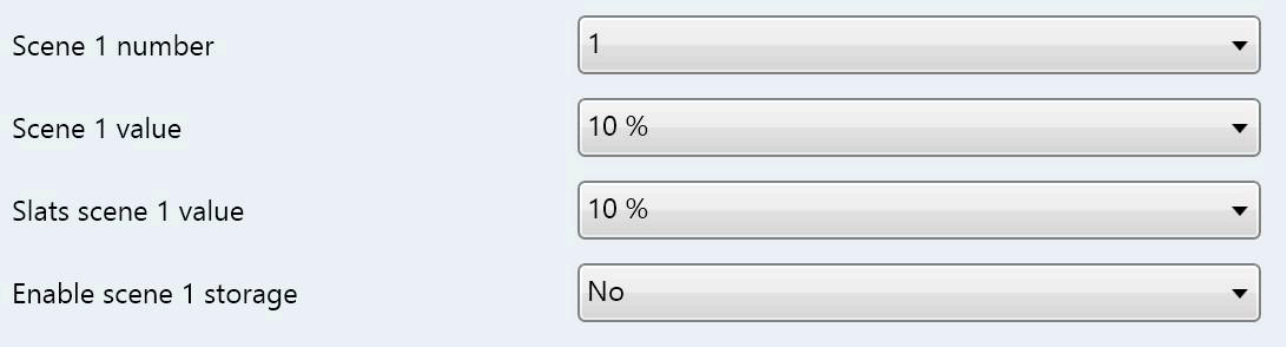
Slats step time is the parameter defining the length of a slat step, i.e. the short movement obtained by sending the **Stop/slats step** control signal.

Finally, **Slat pos. % at end of driving** defines the position that the slats should have when a movement is ended.

3.6.3 Scenes

The Shutter function enables to activate the Scenes KNX.
For general information about the meaning of Scenes in Konnex please refer to section 3.4.2, where we dealt with scenes applied to the Generic Load configuration.

According to whether or not the Blinds function is activated, parametrization can



also contain the Slats position.
Scene X value indicates the desired Shutter position, where 0% means fully open and 100% means fully closed.

Slats scene X value indicates the desired Slats position, where 0% refers to the fully open position and 100% refers to the maximum light control position.

3.7 General Controls

162	All single outs	Set on-off	1 bit	C	R	W	-	-	on/off
163	All single stairs lights	Set on-off	1 bit	C	R	W	-	-	on/off
164	All shutters	Set up/down	1 bit	C	R	W	-	-	up/down
165	All shutters	Set % position	1 Byte	C	R	W	-	-	percentage (0..100%)
166	All shutters/all slats	Set stop/step position	1 bit	C	R	W	-	-	1-bit
167	All slats	Set % position	1 Byte	C	R	W	-	-	percentage (0..100%)

No mention has been made yet of certain communication objects named **All,** able to simultaneously modify the state of several outputs of our actuator - if preliminarily enabled.

We have an **All single outs** control to switch programmed outputs like Generic Load and an **All single stairs light** for the stairs light function.
One Generic Load output responds to the All control if the **Enable general commands** option is enabled.

A similar situation applies to Rolling shutters.
We have an **All shutters Set up/down** control to set the shutters in motion.
We have an **All shutters Set stop/step position** to stop the shutters movement or control the slats.
We have an **All shutters Set % position** to move all the shutters to a predetermined position.
Similarly, we have **All slats Set % position** for the slats.
One Shutter output responds to the All control if the **Enable general commands** option is enabled.