Chorus

Easy 4-channel contacts interface
Easy 2-channel contacts interface

GW 90834A
GW 90833

Technical manual

Contents

1 Introduction

This manual explains the functions of the "**Easy 4-channel contacts interface**" (GW 90834A) and the "**Easy 2-channel contacts interface**" (GW90833), and how they are set and configured using the ETS configuration software.

The GW90833 device has two input channels, while the GW90834A has four.

The two devices are identical from a functional viewpoint, apart from the fact that the "temperature sensor" function is only available in the GW90834A 4-channel version. Otherwise, the functional description and relative parameterising with ETS are the same, and the indications given in this manual refer to the 4-channel version (GW 90834A).

2 Application

The Easy 4-channel contacts interface is an input device that can be inserted in flush-mounting boxes using, for example, a blanking module (GW10751, GW12751, GW14751). It has 4 independent inputs to which potential-free contacts (push-buttons, one-way switches, sensors, etc.) can be connected, and 4 outputs that can command supplementary LEDs powered at 3.3 Volt, with a maximum current of 1 mA. The interrogation voltage needed to establish the closure/opening of the contact is supplied directly by the device and is equal to 3.3 Volt. There are certain device functions which, to be managed, require the use of two linked channels (for example to command a roller shutter with an up button and a down button), and others for which the use of a single channel is sufficient.
The device can perform the following functions:

- ON / OFF commands for loads
- timed commands
- alarm management (wind, rain)
- forcing management
- dimmer management (single or double push-button)
- curtain / roller shutter management (single or double push-button)
- scene management
- commands to the temperature adjustment system
- interface for temperature sensors (GW90834A only)

A function can be associated with each input channel by means of a specific parameter, as described below.

2.1 Association limits

Maximum number of group addresses: 40
Maximum number of associations: 40

This means that up to 40 group addresses can be defined, and up to 40 associations can be made between communication objects and group addresses.

3 *“Settings ”* menu

The ***Settings*** menu contains only the parameter for configuring the programming mode - either ETS (“System” mode) or Easy - via the Easy software controller.
In this Easy version of the device, the signalling LED commands are linked to each input channel for specific functions only where light signalling is envisaged; otherwise the LEDs can be used to indicate the status (ON or OFF) of any loads independent of the input channel, via the relative switching object.

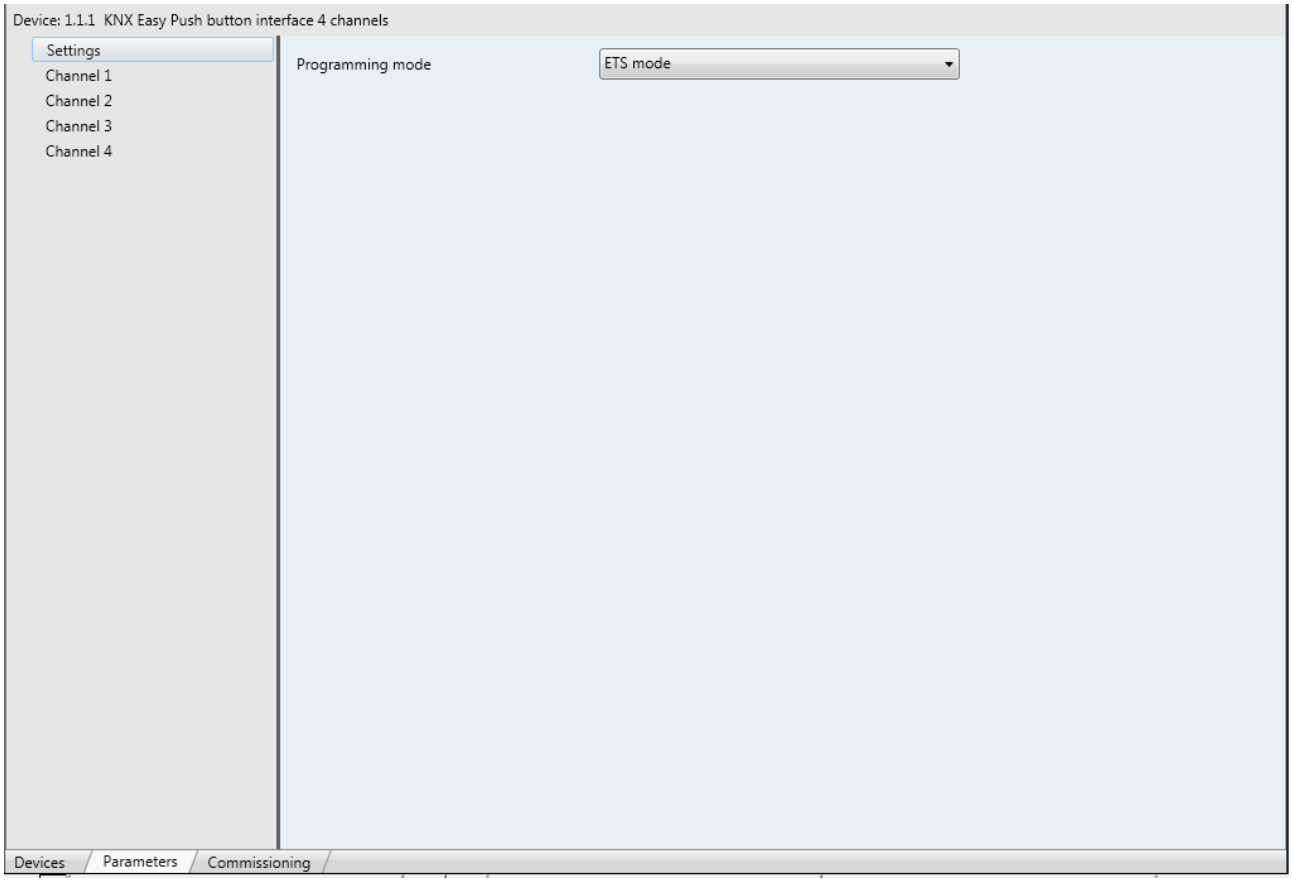

Fig. 3.1

3.1 Parameters

➢ *3.1.1 Programming mode*

This determines the device programming mode:

- ***Easy mode (default value)***
  This option must be selected if the device is to be configured with the Easy controller software.
  If the device has previously been configured with ETS and it is to be included in an Easy project, download the application via ETS with this parameter selected in “Easy mode” so that the Easy controller software can subsequently configure it.

- ***ETS mode***
  This option must be selected if the device is configured with ETS (“System Mode”).

3.2 Communication objects

By default, the *Ch.x-Light signalling* command objects of the LED outputs that can be used for on/off LED switching (to indicate a load status or other information) are enabled. Depending on the function associated with each channel, these objects can remain free for a command of the relative LED, or else they are not visualised by ETS for those functions that expressly require the use of the LED output for a signal associated with the relative input channel.

The functions enabled on each input channel (via the ***Matched function*** parameter) that do not manage light signalling (and which therefore leave the *Ch.x-Light signalling* object free for a signal independent of the input) are:

- Default (no function associated with channel X)
- Single button shutter
- Double button shutter
- Edges
- Priority command
- Wind sensor input
- Rain sensor input
- HVAC mode
- Thermoregulation enabling

All the other functions enabled on channel X require a signal on the LED that may be connected to output X, whose command is associated with the receiving of a status object from the actuator commanded by the push-button or sensor connected to the relative input of the contact interface. Consequently, the *Ch.x – Light signalling* object is no longer available for the command of the channel X LED via the BUS.

Number	Name	Object Function	Length	...	R	W	T	U	Data Ty...	Priority
26	Ch.3 - Light signalling	1=on/0=off	1 bit	C	-	W	T	U	switch	Low
27	Ch.4 - Light signalling	1=on/0=off	1 bit	C	-	W	T	U	switch	Low
24	Ch.1 - Light signalling	1=on/0=off	1 bit	C	-	W	T	U	switch	Low
25	Ch.2 - Light signalling	1=on/0=off	1 bit	C	-	W	T	U	switch	Low

➢ 3.2.1 Ch.x – Light signalling

This communication object is used by the device to command output X for switching a signalling LED.
The enabled flags are C (communication), W (writing from BUS), T (transmission) and U (update).
The standardised object format is *1.001 DPT_Switch*, the object dimension is *1 bit*, and the values it may assume are 1/0 (*ON/OFF*).
By default, the values are 1=ON and 0= OFF. The LED command value can be inverted using the ***LED output functioning*** parameter in the menu of each channel (specified below).

4 *"Channel x"* menu

This chapter describes in general the communication objects and parameters relating to channels 1, 2, 3, 4, (indicated generically as *channel x*) (fig. 4.1).
The value set for the first item (***Matched function***) determines the menu structure.
The configuration parameters of the ***LED output functioning*** associated with each channel are listed below.

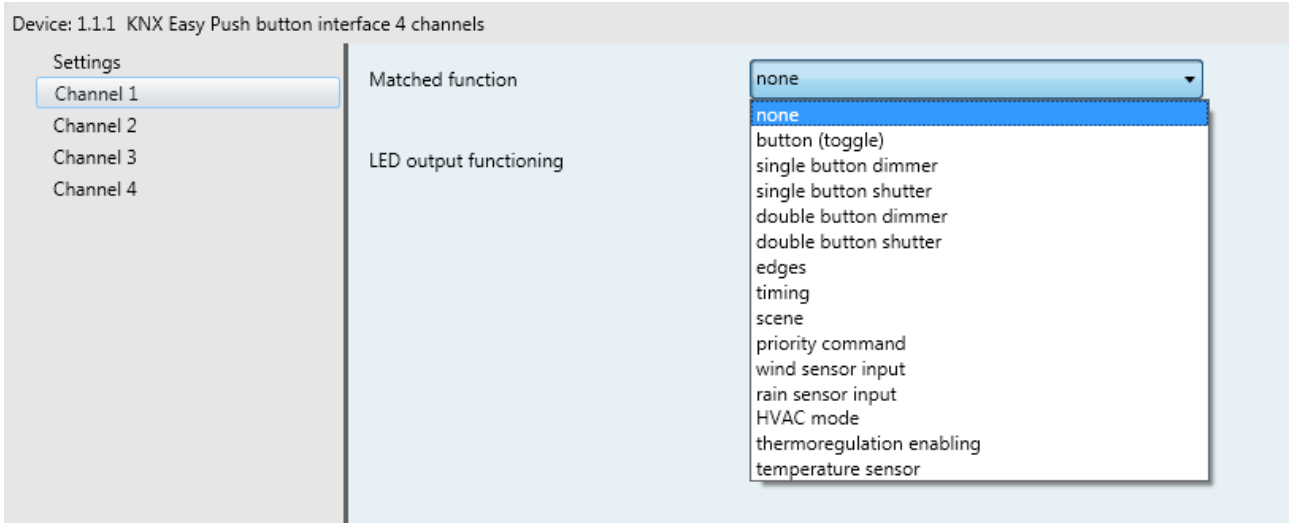

Fig 4.1

4.1 Parameters

➢ *4.1.1 Matched function*

This determines the function associated with the general channel x. Depending on the value set with this parameter, the ***Channel x*** menu will appear differently. The values that can be set are:

- ***none (default value)***

No function is associated with the general channel x, so it is disabled.

- ***button (toggle)***

See chapter 5 - ***"Button (toggle)"* function**

- ***single button dimmer***

See chapter 6 - ***"Single button dimmer"*** function

- ***single button shutter***

See chapter 7 - ***"Single button shutter"* function**

- ***double button dimmer***

See chapter 8 - **"Double button dimmer" function**

- ***double button shutter***

See chapter 9 - ***"Double button shutter"* function**

- ***edges***

See chapter 10 - ***"Edges"* function**

- ***timing***

See chapter 11 - ***"Timing"* function**

- ***scene***

See chapter 12 - ***"Scene"*** **function**

- ***priority command***

See chapter 13 - ***"Priority command"*** **function**

- ***wind sensor input***

See chapter 14 - ***"Wind sensor input" function***

- ***rain sensor input***

See chapter 15 - ***"Rain sensor input" function***

- ***temperature adjustment mode***

See chapter 16 - ***"HVAC mode"*** **function**

- ***thermoregulation enabling***

See chapter 17 - ***"Thermoregulation enabling"*** **function**

- ***temperature sensor***

See chapter 18 - ***"Temperature sensor"*** **function .**This function is only available in the 4-channel version of the device (GW90834A).

➢ *4.1.2 LED output functioning*

This determines the management of any LEDs that may be connected to the LED command outputs available (which in turn may be associated with the respective input channels), whose function envisages light signalling.

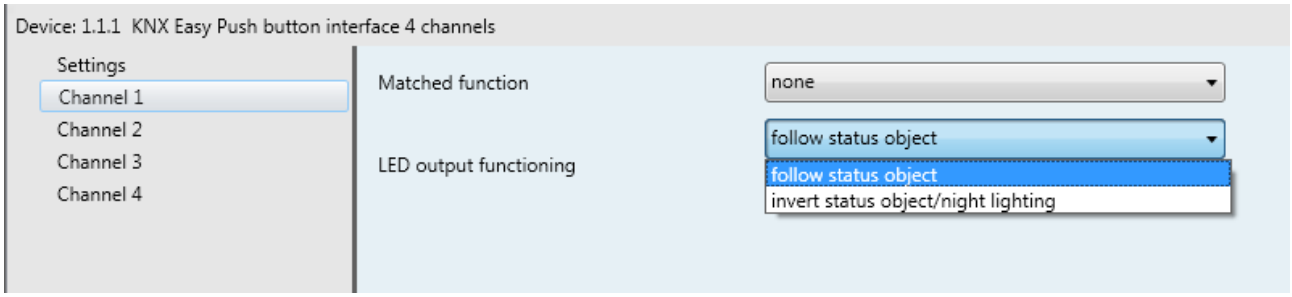

Fig 4.2

- ***follow status object (default value)***

The LED status follows the value (ON/OFF) of the status object associated with the respective channel.

- ***Invert status object / night lighting***

The LED status inverts the value (ON/OFF) of the status object associated with the respective channel. In this case, the LED can be used for night-time localisation of the push-button in which it may be fitted.

5 *"Button (toggle)"* function

This function is used to set the sending of an cyclic ON/OFF switching command. When a button key is pressed, the device sends a telegram on the BUS containing the logic value opposite the value of the status assumed by the commanded actuator.
The value (ON or OFF) evaluated by the device in order to send the next inverted status is the last value received via the ***Ch.x - Status feedback*** communication object. The device uses this value to determine, for example, the current status of the output channel of the actuator that is commanded (by itself or by other devices). In this way, the next command sent by the device will be the opposite of the current status of the output channel. If the actuator status changes (e.g. following the execution of a scene), the contact interface can know the current actuator status thanks to the ***Ch.x – Status feedback*** communication object. It can therefore send the correct command without having to realign itself with the actuator status (which could create vacuum pressure).

NB: if an input channel is configured as a "button (toggle)", it is assumed that it has been connected to an external push-button that sends an ON/OFF switching command to a load (connected to an actuator output channel) every time it is pressed. The switching value (ON or OFF) to be sent when the push-button is pressed is evaluated by inverting the value received via the ***Ch.x – Status feedback*** status object from the actuator, so it is essential to attribute the same group address to the two status objects (***Ch.x – Status feedback*** and ***Status*** of the output channel of the commanded actuator). If this is not the case, switching will be impossible.

If there is a LED associated with channel X, it will light up when ON status feedback is received by the actuator, and switch off when OFF feedback is received in the case of direct functioning, or the opposite reaction in the case of inverted functioning - depending on the setting of the ***LED output functioning*** parameter.

Fig. 5.1

5.1 Parameters

There is no parameter associated with this function.

5.2 Communication objects

The "***button (toggle)***" function displays the following communication objects:

	Number	Name	Object Function	Length	...	R	W	T	U	Data Ty...	Priority
	0	Ch.1 - Status feedback	On/Off status	1 bit	C	-	W	T	U	switch	Low
	1	Ch.1 - Switch	On/Off	1 bit	C	R	-	T	-	switch	Low

Fig. 5.2

➢ *5.2.1 Ch.x – Switch*

The device uses this communication object to send ON/OFF cyclical switching commands on the BUS.
The enabled flags are C (communication), R (reading from BUS), T (transmission).
The standardised object format is *1.001 DPT_Switch*, the object dimension is *1 bit*, and the values it may assume are 1/0 (*ON/OFF*).

➢ *5.2.2 Ch.x - Status feedback*

Via this communication object, the device receives feedback about the status of the devices it commands (usually actuators) so it is always updated in order to be able to command them correctly. Remember to always attribute the same group address to this object - ***Ch.x – Status feedback*** - and the ***Status*** object of the output channel of the commanded actuator.
The enabled flags are C (communication), W (writing from BUS), T (transmission) and U (update).
The standardised object format is *1.001 DPT_Switch*, the object dimension is *1 bit*, and the values it may assume are *ON/OFF* status or, more generally, *1/0*.

6 *"Single button dimmer"* function

This is used to configure the channel to control a dimmer with a single push-button, increasing and decreasing dimmer brightness with the same button.
It is possible to send on/off telegrams and brightness control telegrams.
As there is only one input for managing the On/Off and brightness control functions, the operation is managed so that pressing of the button triggers the sending of the opposite command compared to the last one sent. In addition, there is a distinction between short presses and long presses:

- if the button is pressed for longer than 0.5 sec, it is recognised as a long operation and, in this case, is interpreted as a brightness control command. If the last command was an off command or a decrease brightness command, the new one will be an increase brightness command; vice versa, if the last command was an on command or an increase brightness command, the new one will be a decrease brightness command. In both cases, when the button is released, a control stop telegram is sent to stop the dimmer brightness increase/decrease operation and to fix the brightness value reached at the moment the control stop command was received. In order to function correctly, the device must know the status of the channel of the dimmer commanded via the ***Ch.x - Dimmer status feedback*** object.
- if the button is pressed for 0.5 sec or less, it is recognised as a short operation and, in this case, is interpreted as an on/off command. The command to be sent on the BUS is calculated on the basis of the value of the ***Ch.x - Dimmer status feedback*** object, so that the opposite value to the commanded dimmer status can be sent. The brightness increase/decrease commands do not have any effect in determining the command to be sent.

Using this type of function, brightness control depends on the so-called brightness control characteristic curve, which varies from one device to another depending on how the manufacturer designed the curve that regulates the power (and therefore the brightness) on the dimmer actuator. This means that the speed at which the brightness reaches its maximum and minimum value does not depend on the commands sent by the 4-channel interface, but on the setting of specific parameters that are usually present on the actuators.
If there is a signalling LED associated with the channel, it will only light up when the corresponding dimmer channel is ON if the Status signalling object is connected to the ***Ch.x -Dimmer status feedback*** communication object.
The ***Channel X*** menu appears as shown in fig. 6.1.

Fig. 6.1

6.1 Parameters

There are no parameters to be set for this function.

6.2 Communication objects

The communication objects used to manage the ***single button dimmer*** function are displayed when this function has been set in the ***Matched function*** item of the ***Channel X*** menu (refer to fig. 6.2).

Number	Name	Object Function	Length	...	R	W	T	U	Data Type	Priority
0	Ch.1 - Dimmer status feedback	On/Off status	1 bit	C	-	W	T	U	switch	Low
1	Ch.1 - Switch	On/Off	1 bit	C	R	-	T	-	switch	Low
4	Ch.1 - Brightness dimming	Increase/Decrease	4 bit	C	R	-	T	-	dimming control	Low

Fig. 6.2

➢ *6.2.1 Ch.x – Switch*

The device uses this communication object to send on/off commands on the BUS following brief contact closure operations (short presses on the push-button). As we will see in paragraph ***6.2.2***, the value sent is always the opposite of the last value received on the ***Ch.x – Dimmer status feedback*** object.
The enabled flags are C (communication), R (reading from BUS), T (transmission).
The standardised object format is *1.001 DPT_Switch*, the object dimension is *1 bit*, and the values it may assume are *ON/OFF.*

➢ *6.2.2 Ch.x – Dimmer status feedback*

Via this communication object, the device receives feedback telegrams from the BUS about the status of the loads controlled by the dimmer (in turn managed via the general channel x). Remember to always attribute the same group address to this object - ***Ch.x – Dimmer status feedback*** - and the ***Status*** object of the output channel of the commanded dimmer. The use of this object and the relative addressing is essential for correct functioning and when, for instance, the status of the load controlled by the dimmer actuator changes for scene execution following commands received from other devices (e.g. a display panel). The value of the command to be sent is calculated as the opposite of the last value received on the object in question.
The enabled flags are C (communication), W (writing from BUS), T (transmission) and U (update).
The standardised object format is *1.001 DPT_Switch*, the object dimension is *1 bit*, and the values it may assume are *ON/OFF*.

➢ *6.2.3 Ch.x – Brightness dimming*

The device uses this communication object to send brightness increase/decrease commands on the BUS following long contact closure operations (long presses on the push-button). The value sent via this object is always the opposite of the last command sent, or it is an increase command if the last value received on the ***Ch.x – Dimmer status feedback*** object was “OFF”, or it is a decrease command if the last value received on the ***Ch.x – Dimmer status feedback*** object was “ON”. To ensure this functioning, it is essential that the ***Ch.x – Dimmer status feedback*** object and the ***Status*** signalling object of the commanded dimmer have the same group address.
The coding of this type of command allows the sending of commands to “increase up to 100% (decrease as far as the minimum value)” the brightness value. When the contact re-opens (the push-button is released), a control stop command is sent. In this way, the dimming effect is faster or slower depending on the construction characteristics of the commanded device (the dimmer actuator).
The enabled flags are C (communication), R (reading from BUS) T (transmission).
The standardised object format is *3.007 DPT_Control_Dimming*, the object dimension is *4 bit*, and the values it may assume are *increase/decrease and control stop*.

7 *"Single button shutter"* function

This is used to configure the channel to control a shutter with a single push-button, regulating the upward and downward movement of the shutter and controlling the opening/closing of the slats (in the case of a Venetian blind).
It is possible to send up/down telegrams and slat tilt control telegrams.
As the up/down and slat control functions are implemented via a single input or push-button, the operation is managed so that a pressing of the button triggers the sending of the opposite command compared to the last movement signal received by the actuator that manages the shutter. In addition, there is a distinction between short presses and long presses:

- if the button is pressed for longer than 0.5 sec, it is recognised as a long operation and, in this case, is interpreted as an up/down command. If the last received movement signal was "up", the new command will be a down command, and vice versa.
- if the button is pressed for 0.5 sec or less, it is recognised as a short operation and, in this case, is interpreted as a STOP command if the shutter is moving, or a slat control command if a Venetian blind is being controlled. If the last received movement signal was "up", the new command will be a slat closure control command; if the last received movement signal was "down", the new command will be a slat opening control command. If the shutter is moving, the slat stop/control command will only stop the shutter up/down movement; slat control is only carried out when the Venetian blind is stationary.

The ***Channel X*** menu appears as shown in fig. 7.1.

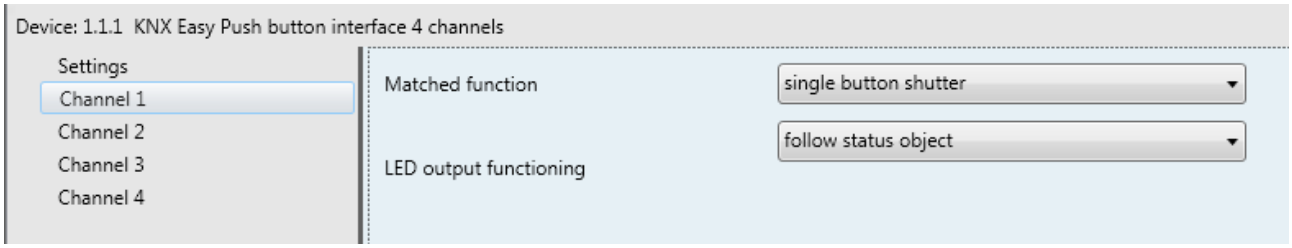

Fig. 7.1

7.1 Parameters

There are no parameters to be set for this function.

7.2 Communication objects

The communication objects used to manage the ***single button shutter*** function are displayed when this function has been set in the ***Matched function*** item of the general ***Channel X*** menu (refer to fig. 7.2).

	Number	Name	Object Function	Length	...	R	W	T	U	Data Type	Priority
	0	Ch.1 - Movement feedback	Increase/Decrease	1 bit	C	-	W	-	-	up/down	Low
	1	Ch.1 - Shutter stop/Louvres step	Stop/Step	1 bit	C	R	-	T	-		Low
	2	Ch.1 - Shutter movement	Up/Down	1 bit	C	R	-	T	-	up/down	Low

Fig. 7.2

➢ *7.2.1 Ch.x - Shutter movement*

The device uses this communication object to send up/down commands on the BUS following prolonged contact closure operations (long presses on the push-button). As shown in paragraph 7.2.3, the value sent is always the opposite of the last value received on the ***Ch.x – Movement feedback*** object.
The enabled flags are C (communication), R (reading from BUS), T (transmission).
The standardised object format is *1.008 DPT_UpDown*, the object dimension is *1 bit*, and the values it may assume are *up/down*.

➢ *7.2.2 Ch.x - Shutter stop/Louvre step*

The device uses this communication object to send slat opening/closure control commands on the BUS following brief contact closure operations (short presses on the push-button). If the shutter is moving, this command stops the shutter up/down movement (Stop command); slat control is only carried out when the Venetian blind is stationary.
The enabled flags are C (communication), R (reading from BUS), T (transmission).
The standardised object format is *1.007 DPT_Step*, the object dimension is *1 bit*, and the values it may assume are *opening/closure control* or *stop movement.*

➢ *7.2.3 Ch.x – Movement feedback*

Via this communication object, the device receives feedback from the BUS about the movement of the shutter commanded by the shutter actuator.
The use of this communication object is essential for the correct functioning of the device, as its value conditions the choice of the next commands to be sent via the ***Ch.x–Shutter movement*** and ***Ch.x–Shutter stop/Louvre step*** objects according to the type of press on the button (inverting the status or direction, as described above). For this reason, you must remember to assign the same group address to the ***Ch.x – Movement feedback*** object and the corresponding ***Movement feedback*** object relating to the shutter actuator motor commanded.
The enabled flags are C (communication) and W (writing from BUS).
The standardised object format is *1.008 DPT_UpDown*, the object dimension is *1 bit*, and the values it may assume are *up/down*.

8 *"Double button dimmer"* function

Used to configure an input channel of the device, to control a dimmer; the dimmer brightness will be regulated (increased or decreased) according to the setting of the ***Regulation direction*** parameter. This option assumes that another input channel is configured in the same way, with the opposite regulation function (e.g. for one single lamp, an increase/ON regulation push-button is connected to channel 1, and a decrease/OFF regulation push-button is connected to channel 2).
ON/OFF and brightness increase/decrease telegrams can be sent according to the setting of the ***Regulation direction*** parameter, as described below. The distinction between an ON (or OFF) command and a regulation command is always made by the long or short pressing of the push-button, as follows:

- if the button is pressed for longer than 0.5 sec, it is recognised as a long operation and, in this case, is interpreted as a brightness control command. If the set control direction is "increase", the control will only be increasing, otherwise if the set control direction is "decrease" the control will be decreasing. In both cases, when the button is released, a control stop telegram is sent to stop the dimmer brightness increase/decrease operation and to fix the brightness value reached at the moment the control stop command was received.
- if the button is pressed for 0.5 sec or less, it is recognised as a short operation and, in this case, is interpreted as an ON or OFF command depending on the setting of the ***Regulation direction*** parameter.
  If the regulation direction set is "increase", the push-button connected to this channel will only send ON commands. If the regulation direction is "decrease", the push-button connected to the channel will only send OFF commands.

In this case too, the brightness control depends on the so-called characteristic brightness control curve which may be fixed or programmed on the dimmer actuator.
If there is a signalling LED associated with the channel, it will only light up when the corresponding dimmer channel is ON if the ***Status*** signalling object is connected to the ***Ch.x -Dimmer status feedback*** communication object (same group address).
The ***Channel X*** menu appears as shown in fig. 8.1.

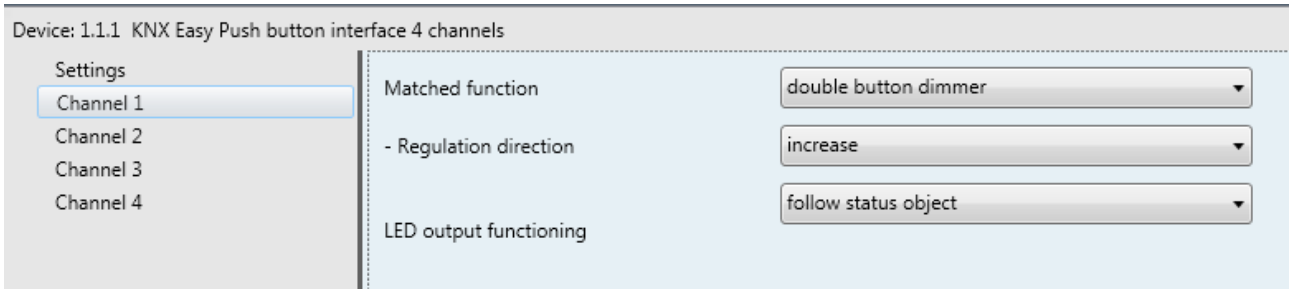

Fig. 8.1

8.1 Parameters

➢ 8.1.1 Regulation direction

Used to set the type of regulation you want to associate with the push-button connected to channel X.
The values that can be set are:

- ***increase (odd channel default value)***

The closure of the contact (pressing of the push-button) triggers the sending of an "increase" regulation telegram (in the case of prolonged pressing) or an ON telegram (short pressing) on the BUS.

- ***decrease (even channel default value)***

The closure of the contact (pressing of the push-button) triggers the sending of a "decrease" regulation telegram (in the case of prolonged pressing) or an OFF telegram (short pressing) on the BUS.

8.2 Communication objects

The communication objects used to manage the ***double button dimmer*** function are displayed when this function has been set in the ***Matched function*** item of the ***Channel X*** menu (refer to fig. 8.2).

Number	Name	Object Function	Length	...	R	W	T	U	Data Type	Priority
0	Ch.1 - Dimmer status feedback	On/Off status	1 bit	C	-	W	T	U	switch	Low
1	Ch.1 - Switch	On/Off	1 bit	C	R	-	T	-	switch	Low
4	Ch.1 - Brightness dimming	Increase/Decrease	4 bit	C	R	-	T	-	dimming control	Low

Fig. 8.2

➢ 8.2.1 Ch.x – Switch

The device uses this communication object to send on/off commands on the BUS following brief contact closure operations (short presses on the push-button). The value sent is always ON if the channel was configured via the ***Regulation direction*** parameter with “increase”; if it was configured with “decrease”, the command sent is always OFF.
The enabled flags are C (communication), R (reading from BUS), T (transmission).
The standardised object format is *1.001 DPT_Switch*, the object dimension is *1 bit*, and the values it may assume are *ON/OFF.*

➢ 8.2.2 Ch.x – Dimmer status feedback

Via this communication object, the device receives feedback telegrams from the BUS about the status of the loads controlled by the dimmer (in turn managed via the general channel x).
The use of this object and the relative addressing is essential when, for instance, the status of the load controlled by the dimmer changes for scene execution following commands received from other devices (e.g. a display panel), or if the dimmer output channel itself is controlled by several push-buttons simultaneously. In the latter case, it is a good idea to give the ***Dimmer status feedback*** object the same group address associated with the identical communication object on the output channel of the dimmer actuator. In this case, it is possible to manage a LED that may be associated with the channel.
The enabled flags are C (communication), W (writing from BUS), T (transmission) and U (update).
The standardised object format is *1.001 DPT_Switch*, the object dimension is *1 bit*, and the values it may assume are *ON/OFF*.

➢ 8.2.3 Ch.x – Brightness dimming

The device uses this communication object to send brightness increase/decrease commands on the BUS, depending on how the ***Regulation direction*** parameter is configured and following long contact closure operations (long presses on the push-button). The value sent via this object is increasing regulation if the set regulation direction is “increase”, or decreasing if the set regulation direction is “decrease” and the control STOP command.

The enabled flags are C (communication), R (reading from BUS), T (transmission).
The standardised object format is *3.007 DPT_Control_Dimming*, the object dimension is *4 bit*, and the values it may assume are *increase/decrease* and *control stop*.

9 *"Double button shutter"* function

This is used to configure the channel to control a shutter with a double push-button, regulating the upward and downward movement of the shutter and controlling the opening/closing of the slats.
It is possible to send up/down telegrams and slat control telegrams.
If there are two push-buttons for commanding one shutter, you must assign one movement direction to the button connected to one input channel and the opposite direction to the button connected to the second input channel of the device. If the ***Movement direction*** parameter is set at "**up**", the push-button will only send upward movement commands; if the parameter is set at "**down**", the push-button will only send downward movement commands.
Each push-button manages the up/down and slat control functions by distinguishing between long and short presses:

- if the button is pressed for longer than 0.5 sec, it is recognised as a long operation and, in this case, is interpreted as an up or down command depending on the setting of the ***Movement direction*** parameter.
- if the button is pressed for 0.5 sec or less, it is recognised as a short operation and, in this case, is interpreted as a STOP command if the shutter is moving, or a slat control command if a Venetian blind is being controlled. In this case, if the ***Movement direction*** assigned is "**up**", the command will be a slat control command on opening; vice versa, if the ***Movement direction*** assigned is "**down**", the command will be a slat control command on closure. If the shutter is moving, the slat stop/control command will only stop the shutter up/down movement; slat control is only carried out when the Venetian blind is stationary.

The ***Channel X*** menu appears as shown in fig. 9.1.

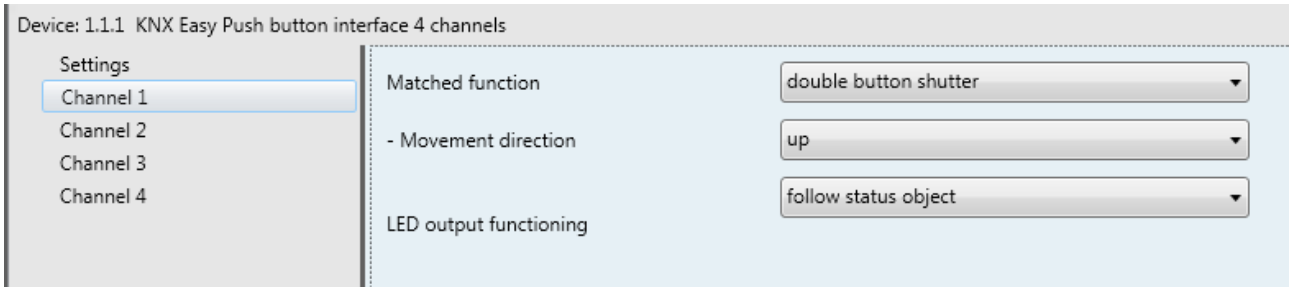

Fig. 9.1

9.1 Parameters

➢ *9.1.1 Movement direction*

Used to set the shutter movement direction you want to associate with the push-button connected to channel X. The values that can be set are:

- ***up (odd channel default value)***

The closure of the contact (pressing of the push-button) triggers the sending of an "up" movement telegram (in the case of prolonged pressing) or a "stop" movement telegram (short pressing) on the BUS.
In the case of a Venetian blind, a short press can control the opening slat tilt once the blind has stopped moving.

- ***down (even channel default value)***

The closure of the contact (pressing of the push-button) triggers the sending of a "down" movement telegram (in the case of prolonged pressing) or a "stop" movement telegram (short pressing) on the BUS.
In the case of a Venetian blind, a short press can control the closure slat tilt once the blind has stopped moving.

9.2 Communication objects

The communication objects used to manage the ***double button shutter*** function are displayed when this function has been set in the ***Matched function*** item of the general ***Channel X*** menu (refer to fig. 9.2).

Number	Name	Object Function	Length	...	R	W	T	U	Data Type	Priority
1	Ch.1 - Shutter stop/Louvres step	Stop/Step	1 bit	C	R	-	T	-	step	Low
2	Ch.1 - Shutter movement	Up/Down	1 bit	C	R	-	T	-	up/down	Low

Fig. 9.2

➢ *9.2.1 Ch.x - Shutter movement*

The device uses this communication object to send up/down commands on the BUS, depending on how the ***Movement direction*** parameter is configured and following long contact closure operations (long presses on the push-button).
The enabled flags are C (communication), R (reading from BUS), T (transmission).
The standardised object format is *1.008 DPT_UpDown*, the object dimension is *1 bit*, and the values it may assume are *up/down*.

➢ *9.2.2 Ch.x - Shutter stop/Louvre step*

The device uses this communication object to send commands for slat control on opening and closure on the BUS, depending on how the ***Movement direction*** parameter is configured and following short contact closure operations (short presses on the push-button).
If the shutter is moving, this command stops the shutter up/down movement (Stop command); slat control is only carried out when the Venetian blind is stationary.
The enabled flags are C (communication), R (reading from BUS), T (transmission).
The standardised object format is *1.007 DPT_Step*, the object dimension is *1 bit*, and the values it may assume are *opening/closure control* or *stop movement.*

10 *"Edges"* function

This function is used to set the type of ON/OFF switching command to send following a variation in the contact status (up/down edge). It is possible to differentiate between the type of command depending on the edge that is detected (from contact open to contact closed, and vice versa).
If there is a signalling LED associated with the channel, it will always be OFF (the actuator return status object is not managed).

Figure 10.1 shows the parameters that define the behaviour of the individual channels.

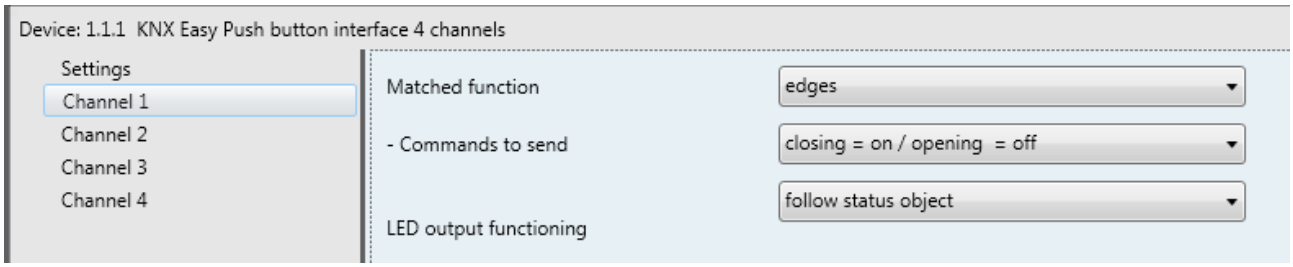

Fig. 10.1

10.1 Parameters

➢ *10.1.1 Commands to send*

Used to set the ON or OFF switching value to be sent following a variation on the contact (from closed to open, or vice versa).
The values that can be set are:

- ***closing = on / opening = off (default value)***

When a variation from open to closed is detected, a telegram with the logic value "1" (ON) is sent on the BUS via the ***Ch.x – Switch*** communication object.
When a variation from closed to open is detected, a telegram with the logic value "0" (OFF) is sent on the BUS via the ***Ch.x – Switch*** communication object.

- ***closing = off / opening = on***

When a variation from open to closed is detected, a telegram with the logic value "0" (OFF) is sent on the BUS via the ***Ch.x – Switch*** communication object.
When a variation from closed to open is detected, a telegram with the logic value "1" (ON) is sent on the BUS via the ***Ch.x – Switch*** communication object.

- ***closing = on / opening = no effect***

When a variation from open to closed is detected, a telegram with the logic value "1" (ON) is sent on the BUS via the ***Ch.x – Switch*** communication object.
When a variation from closed to open is detected, there is no effect.

- ***closing = off / opening = no effect***

When a variation from open to closed is detected, a telegram with the logic value "0" (OFF) is sent on the BUS via the ***Ch.x – Switch*** communication object.
When a variation from closed to open is detected, there is no effect.

10.2 Communication objects

The “edges” function displays the following communication objects:

Number	Name	Object Function	Length	...	R	W	T	U	Data Type	Priority
1	Ch.1 - Switch	On/Off	1 bit	C	R	-	T	-	switch	Low

Fig. 10.2

➢ 10.2.1 Ch.x – Switch

The device uses this communication object to send ON or OFF switching commands on the BUS, depending on the option assigned in the ***Commands to send*** parameter.
The enabled flags are C (communication), R (reading from BUS), T (transmission).
The standardised object format is *1.001 DPT_Switch*, the object dimension is *1 bit*, and the values it may assume are 1/0 (*ON/OFF*).

11 *"Timing"* function

This function is used to configure a push-button in order to send a timed ON command to an actuator output channel.
For this purpose, two communication objects - ***Ch.x – Timed switch*** and ***Ch.x – Status feedback*** are made visible. The first object sends the ON command to the actuator, which then returns the feedback on the addressed actuator status on the second object. The timing of the load for the OFF status is configured as normal on the actuator (activation time).
This function may be handy if you decide to assign a timed load activation function to a command (generally a push-button) - e.g. for the "stair lights" function.
If there is a signalling LED connected to the input channel, it will remain ON throughout the activation if the actuator ***Status*** object is connected to the ***Ch.x -Status feedback*** communication object (same group address).

Figure 11.1 shows the parameters that define the behaviour of the individual channels.

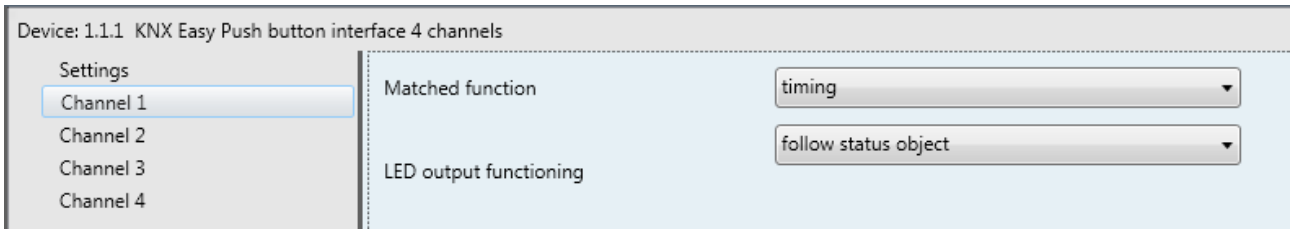

Fig. 11.1

11.1 Parameters

There are no parameters to be set for this function.

11.2 Communication objects

The "timing" function displays the following communication objects:

	Number	Name	Object Function	Length	...	R	W	T	U	Data Type	Priority
	0	Ch.1 - Status feedback	On/Off status	1 bit	C	-	W	T	U	switch	Low
	1	Ch.1 - Timed switch	Activate timing	1 bit	C	R	-	T	-	start/stop	Low

Fig. 11.2

➢ *11.2.1 Ch.x – Timed switch*

The device uses this communication object to send the ON switching command on the BUS. It is assumed that load deactivation is programmed on the actuator output channel, notifying the device of actual switch-off via the ***Ch.x - Status feedback*** object.
The enabled flags are C (communication), R (reading from BUS), T (transmission).
The standardised object format is *1.010 DPT_Start*, the object dimension is *1 bit*, and the values it may assume are 1/0 *Start/Stop* (in this case, only *Start* is transmitted).

➢ *11.2.2 Ch.x - Status feedback*

Via this communication object, the device receives feedback about the status of the devices it commands (usually actuators). In this case, the feedback is used to manage a LED that may be associated with the channel. Remember to always attribute the same group address to this object of Status feedback and the Status object of the commanded actuator.
The enabled flags are C (communication), W (writing from BUS), T (transmission) and U (update).
The standardised object format is *1.001 DPT_Switch*, the object dimension is *1 bit*, and the values it may assume are *ON/OFF* status or, more generally, *1/0*.

12 *"Scene"* function

This is used to configure the channel in order to send scene memorising and execution commands, with the possibility of memorising the scene following a command received from the BUS. Only one scene can be managed for each channel.
In this case too, the scene memorising and execution commands are distinguished by the type of action on the contact:

- if the contact remains closed (i.e. the button is pressed) for longer than 3 sec, it is recognised as a long operation and, in this case, is interpreted as a scene learning command.
- if the contact remains closed (i.e. the button is pressed) for 3 sec or less, it is recognised as a short operation and, in this case, is interpreted as a scene execution command.

If there is a LED connected to the corresponding channel, it will blink briefly if a scene learning command is being sent.

The menu associated with the general ***Channel x*** is as shown in Fig. 12.1.

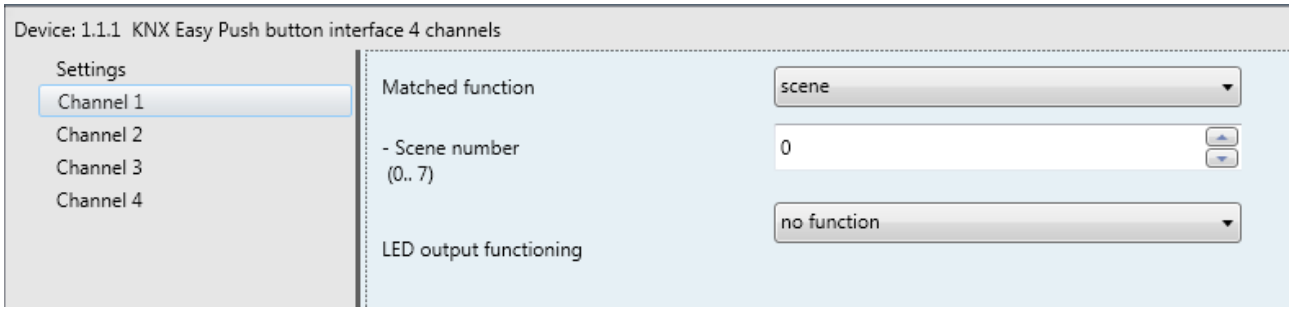

Fig. 12.1

12.1 Parameters

➢ *12.1.2 Scene number (0.. 7)*

Used to set the value of the scene you want to activate/deactivate.
Remember that the 4-channel contact interface can only manage one scene per channel.
The value set for this item varies from 0 to 7 **(default value 0)**, and is important because the output devices (actuators, dimmers, etc.) can usually manage more than one scene, identified by the scene number received; make sure this item is configured correctly, assigning the number on the basis of the scene you want to manage with channel X.

12.2 Communication objects

The communication objects used to manage the ***scene*** function are displayed when this function has been set in the ***Matched function*** item of the general ***Channel X*** menu (refer to fig. 12.2).

	Number	Name	Object Function	Length	...	R	W	T	U	Data Type	Priority
	5	Ch.1 - Scene	Execute/Store	1 Byte	C	R	-	T	-	scene control	Low

Fig. 12.2

➢ *12.2.1 Ch.x - Scene*

The device uses this communication object to send the scene execution command on the BUS following short contact closures, and scene storage commands following long contact closures.
The scene number stored or subsequently called up is the one indicated in the ***Scene number*** parameter.
The enabled flags are C (communication), R (reading from BUS), T (transmission).

The standardised object format is *18.001 DPT_SceneControl*, the object dimension is *1 byte*, and the values it may assume are *scene y execution/storage* (where *y* indicates the general number of the scene associated with the commands - i.e. the value set for the ***Scene number (0.. 7)***) parameter.

13 *"Priority command"* function

Used to configure the channel for sending a priority command. The value you want to send with the priority command is assigned via the ***Command to send on closing contact*** parameter, as described below. When the contact closes, the device sends the 2-bit ***Ch.x – Priority command*** object containing the ON or OFF value (corresponding to the UP or DOWN movement in the case of forcing sent to a motor command actuator) specified in the ***Command to send on closing contact*** parameter. When this object is received, the actuator forces the output channel (commanded by a similar 2-bit object and assigned with the same group address) to the forcing status indicated in this parameter.
This function may be handy for forcing a load status (to ON or OFF, or for forcing a shutter UP or DOWN) in certain situations such as an emergency, maintenance, presence or absence of people, etc.
Any other switching or movement commands received by the actuators during the forcing will be ignored.
When the contact opens, the device sends the forcing removal command via the same object - ***Ch.1 – Priority command***.

At the end of the forcing period, the actuators are usually in the status corresponding to the last command received, or else they restore the previous value (NB: this option is normally available as an actuator configuration parameter).

The menu associated with the general ***Channel X*** for this function is as shown in Fig. 13.1.

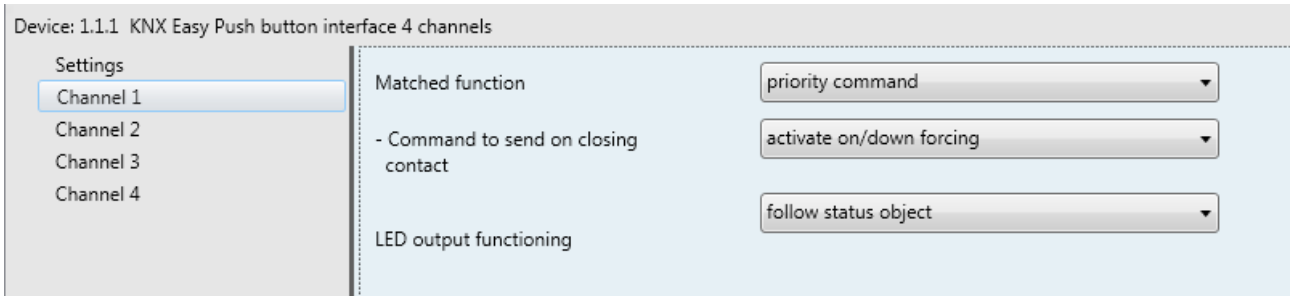

Fig. 13.1

13.1 Parameters

➢ 13.1.1 Command to send on closing contact

Used to set the value with which you want to send the forcing command.

The values that can be set are:

- ***activate on/down forcing (default value)***

When a contact closure is detected, the command is sent on the BUS on the 2-bit ***Ch.x – Priority command*** communication object, with the forcing value at "1" (ON/down).
When a contact opening is detected, a command to remove the forcing is sent on the BUS on the 2-bit ***Ch.x – Priority command*** communication object.

- ***activate off/up forcing***

When a contact closure is detected, the command is sent on the BUS on the 2-bit ***Ch.x – Priority command*** communication object, with the forcing value at "0" (OFF/up).
When a contact opening is detected, a command to remove the forcing in sent via on the BUS on the 2-bit ***Ch.x – Priority command*** communication object.

13.2 Communication objects

The communication objects used to manage the ***priority command*** function are displayed when this function has been set in the ***Matched function*** item of the general ***Channel X*** menu (refer to fig. 12.2).

Number	Name	Object Function	Length	...	R	W	T	U	Data Type	Priority
3	Ch.1 - Priority command	On/Off forced positioning	2 bit	C	R	-	T	-	switch control	Low

Fig. 13.2

➢ 13.2.1 Ch.x - Priority command

The device uses this communication object to send the forcing ON or OFF (or DOWN and UP) activation command on the BUS, depending on the setting of the ***Command to send on closing contact*** parameter and the forcing deactivation command.
The enabled flags are C (communication), R (reading from BUS), T (transmission).
The standardised object format is *2.001 DPT_Switch_Control*, the object dimension is *2 bit*, and the commands it sends are *on/off forcing enabled, forcing disabled*.

14 *"Wind sensor input"* function

Used to configure an input channel for controlling a wind sensor that may be connected to it.
When the contact is closed by an external wind sensor, the ***Ch.x – Wind alarm*** communication object is sent with a value of 1 (alarm). The information can be managed by specific actuators as a priority, with the aim - for instance - of immediately closing the external motorised curtains in the event of an alarm. For the management of the wind alarm command in relation to any other normal or priority ON/OFF switching commands that might be configured on the same output channel, refer to the actuator technical characteristics.
In the same way, the same object - ***Ch.x – Wind alarm*** - is sent with a value of 0 (end of alarm) when the contact is opened. If there is a status signalling LED connected to the output corresponding to channel X, it remains OFF.

Figure 14.1 shows the parameters that define the behaviour of the individual channels.

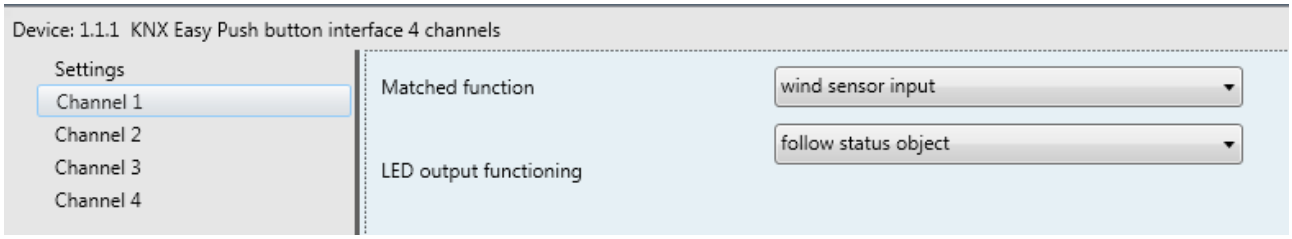

Fig. 14.1

14.1 Parameters

There are no parameters to be set for this function.

14.2 Communication objects

The "wind sensor input" function displays the following communication objects:

	Number	Name	Object Function	Length	...	R	W	T	U	Data Type	Priority
	1	Ch.1 - Wind alarm	Alarm input	1 bit	C	R	-	T	-	alarm	Low

Fig. 14.2

➢ *14.2.1 Ch.x - Wind alarm*

The device uses this communication object to send the wind alarm command (with value 1) on the BUS when contact closure is detected on the input channel.
When the contact opens, the same object is again sent, but with a value of 0 (end of alarm).
The enabled flags are C (communication), R (reading from BUS), T (transmission).
The standardised object format is *1.005 DPT_Alarm*, the object dimension is *1 bit*, and the values it may assume are 1/0 (*Alarm/No alarm)*.

15 *"Rain sensor input"* function

Used to configure an input channel for controlling a rain sensor that may be connected to it. When the contact is closed by an external rain sensor, the ***Ch.x – Rain alarm*** communication object is sent with a value of 1 (alarm). The information can be managed by specific actuators as a priority, with the aim - for instance - of immediately closing windows or motorised skylights in the event of an alarm. For the management of the rain alarm command in relation to any other normal or priority ON/OFF switching commands that might be configured on the same output channel, refer to the actuator technical characteristics.
In the same way, the same object - ***Ch.x – Rain alarm*** - is sent with a value of 0 (end of alarm) when the contact is opened. If there is a status signalling LED connected to the output corresponding to channel X, it remains OFF.

Figure 15.1 shows the parameters that define the behaviour of the individual channels.

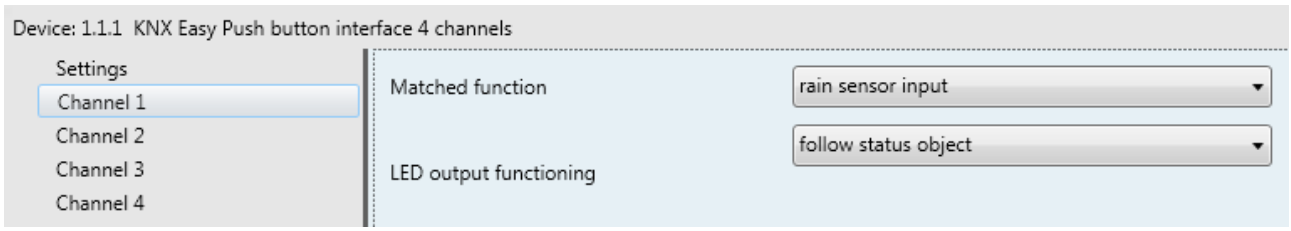

Fig. 15.1

15.1 Parameters

There are no parameters to be set for this function.

15.2 Communication objects

The "rain sensor input" function displays the following communication objects:

	Number	Name	Object Function	Length	...	R	W	T	U	Data Type	Priority
	1	Ch.1 - Rain alarm	Alarm input	1 bit	C	R	-	T	-	alarm	Low

Fig. 15.2

➢ *15.2.1 Ch.x - Rain alarm*

The device uses this communication object to send the rain alarm command (with value 1) on the BUS when contact closure is detected on the input channel.
When the contact opens, the same object is again sent, but with a value of 0 (end of alarm).
The enabled flags are C (communication), R (reading from BUS), T (transmission).
The standardised object format is *1.005 DPT_Alarm*, the object dimension is *1 bit*, and the values it may assume are 1/0 (*Alarm/No alarm)*.

16 *"HVAC mode"* function

Used to configure the channel to trigger - for instance - a mode change (via a one-way switch) in the thermostats or timed thermostats connected to the BUS so that they switch from one mode to the other.
When the input contact opens and closes, a temperature adjustment mode is associated. The device sends this to the temperature adjustment devices (e.g. Easy thermostats or Easy timed thermostats) via the 1-byte ***Ch.x – HVAC mode*** communication object.
The combinations available on mode change can be selected via the ***HVAC mode selection*** parameter.
With this function, when the contact opens or closes, the device sends the ***Ch.x – HVAC mode*** communication object, associating a temperature adjustment mode (chosen via the ***HVAC mode selection*** parameter) with the two events.
If there is a status signalling LED connected to channel X, it remains OFF.

Figure 16.1 shows the parameters that define the behaviour of the individual channels.

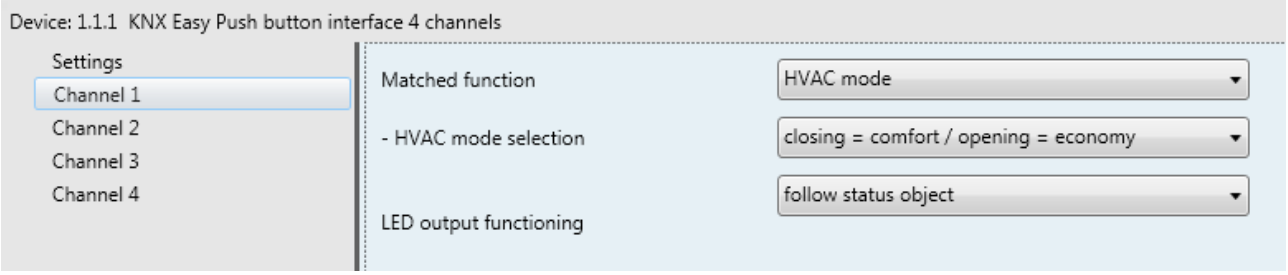

Fig. 16.1

16.1 Parameters

➢ *16.1.1 HVAC mode selection*

Used to set the mode to be sent to the temperature adjustment devices when the input contact opens or closes.
The values that can be set are:

- ***closing = comfort / opening = economy (default value)***

When a contact variation from open to closed is detected, this triggers the sending on the BUS of the ***Ch.x – HVAC mode*** communication object with a ***comfort*** value.
When a contact variation from closed to open is detected, this triggers the sending on the BUS of the ***Ch.x – HVAC mode*** communication object with an ***economy*** value.

- ***closing = comfort / opening = no effect***

When a contact variation from open to closed is detected, this triggers the sending on the BUS of the ***Ch.x – HVAC mode*** communication object with a ***comfort*** value.
When a variation from closed to open is detected, there is no effect.

- ***closing = economy / opening = no effect***

When a contact variation from open to closed is detected, this triggers the sending on the BUS of the ***Ch.x – HVAC mode*** communication object with an ***economy*** value.
When a variation from closed to open is detected, there is no effect.

- ***closing = building protection / opening = auto***

When a contact variation from open to closed is detected, this triggers the sending on the BUS of the ***Ch.x – HVAC mode*** communication object with an ***OFF = building protection*** (anti-freeze/high temperature protection) value.
When a contact variation from closed to open is detected, this triggers the sending on the BUS of the ***Ch.x – HVAC mode*** communication object with an ***auto*** (automatic) value.

16.2 Communication objects

The “HVAC mode” function displays the following communication objects:

Number	Name	Object Function	Length	...	R	W	T	U	Data Type	Priority
5	Ch.1 - HVAC Mode	Auto/Eco/Precom/Comf/Off	1 Byte	C	R	-	T	-	HVAC mode	Low

Fig. 16.2

➢ 16.2.1 Ch.x – HVAC mode

The device uses this communication object to send a mode change command on the BUS to the temperature adjustment devices (thermostats and timed thermostats).
Depending on the setting of the ***HVAC mode selection*** parameter, this object sends the corresponding mode on contact opening or closure.
With the ***HVAC mode*** function, the contact interface can only send the mode changes envisaged by the ***HVAC mode selection*** parameter.
The enabled flags are C (communication), R (reading from BUS), T (transmission).
The standardised object format is *20.102 DPT_HVACMode*, the object dimension is *1 byte*, and the commands it sends are *Operating mode: Auto/Economy/Comfort/Off.*

17 *"Thermoregulation enabling"* function

Used to configure the channel to trigger the enabling/disabling of the temperature adjustment system (heating or air-conditioning) via a command (a one-way switch, for instance).
When the input contact closes, the 1-bit ***Ch.x – Thermoregulation enabling*** object is sent with a value of 1 (ENABLE). When the contact opens, the same object is again sent, but with a value of 0 (DISABLE). If there is a status signalling LED, it is always OFF.

Figure 17.1 shows the parameters that define the behaviour of the individual channels.

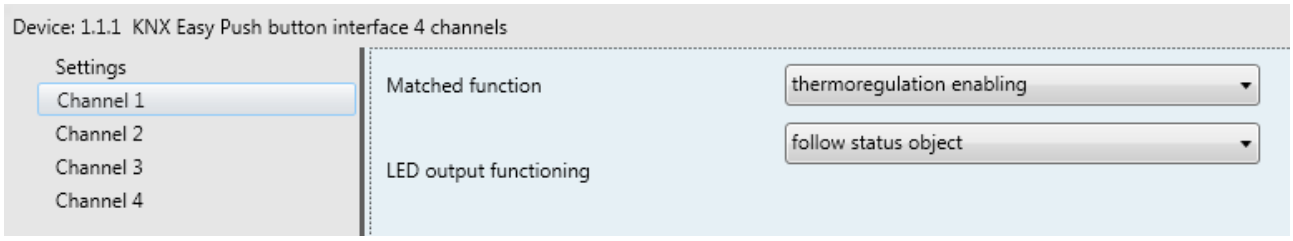

Fig. 17.1

17.1 Parameters

There are no parameters to be set for this function.

17.2 Communication objects

The "thermoregulation enabling" function displays the following communication objects:

Number	Name	Object Function	Length	...	R	W	T	U	Data Type	Priority
1	Ch.1 - HVAC enabling	Enable/Disable	1 bit	C	R	-	T	-	enable	Low

Fig. 17.2

➢ *17.2.1 Ch.x – Thermoregulation enabling*

The device uses this communication object to send the Thermoregulation enabling command (with value 1) on the BUS when contact closure is detected on the input channel.
When the contact opens, the same object is again sent, but with a value of 0 (Disable).
The enabled flags are C (communication), R (reading from BUS), T (transmission).
The standardised object format is *1.003 DPT_Enable*, the object dimension is *1 bit*, and the values it may assume are *enable/disable*.

18 *"Temperature sensor"* function

Various temperature sensors can be connected to the input contacts. Given the different characteristics of each transducer, the ***"Type of NTC sensor connected"*** parameter is used to define which of the possible sensors can be connected to the device contacts, in order to interface correctly. The contact interface sends the temperature value measured by the sensor on the BUS, according to the settings of the ***Measured temperature*** and ***Unit of measurement*** parameters.

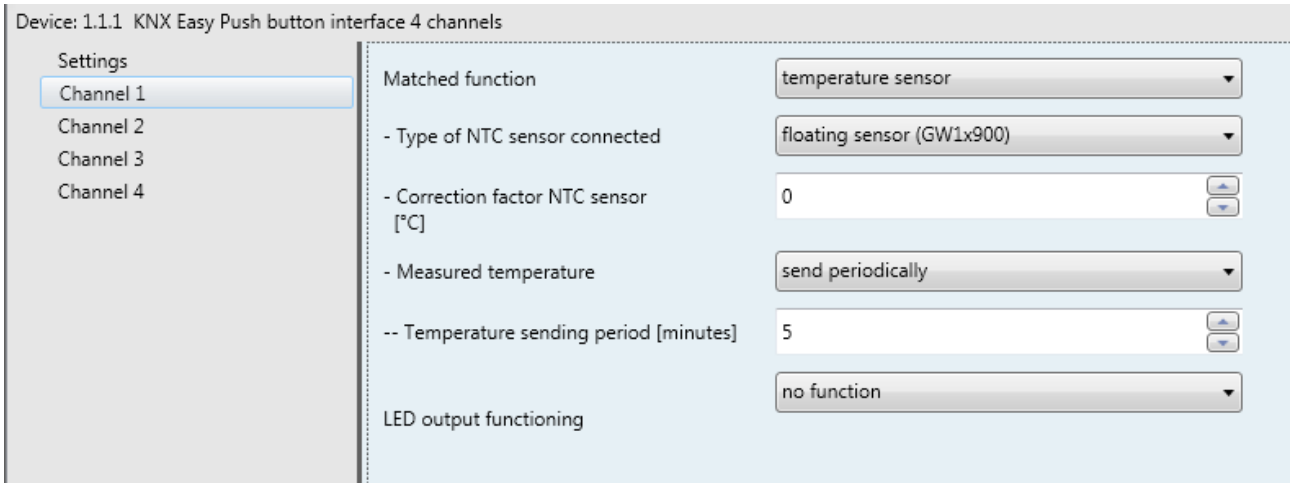

Fig. 18.1

18.1 Parameters

➢ *18.1.1 Type of NTC sensor connected*

Used to define the types of NTC sensors that can be connected to the device. The types that can be set are:

- ***wired sensor (GW10800) - (default value)***

The temperature sensor type GW10800 is connected to the input channel

- ***1-module flush-mounting sensor (GW1x900)***

A 1-module flush-mounting sensor, type GW1x900, is connected

➢ *18.1.2 Correction factor NTC sensor [0.1°C]*

Used to set the correction factor to be applied to the temperature value measured by the NTC sensor connected to the input, in order to eliminate the heat contribution generated by the installation site. The values that can be set are:

- from -20 to + 20 in steps of 1 *(default value 0)*

➢ *18.1.3 Measured temperature*

This is used to define the conditions for sending the temperature value measured by the device. The values that can be set are:

- ***send on demand only***

The value is only sent in response to a value read request command from the BUS (e.g. by a display)

- ***send in case of change - (default value)***
  The value is only sent in response to a detected temperature variation specified in the ***Minimum temperature variation for sending value [± 0.1°C]*** parameter

- ***send periodically***
  The value is sent periodically, with a frequency specified in the ***Temperature sending period [minutes]*** parameter that appears when this option is selected

- ***send in case of change and periodically***
  The value is sent in response to a detected temperature variation specified in the ***Minimum temperature variation for sending value [± 0.1°C]*** parameter, and also periodically as specified in the ***Temperature sending period [minutes]*** parameter.

➢ *18.1.4 Minimum temperature variation for sending value [± 0.1°C]*

This parameter is visible if the temperature sending conditions include "in case of change", and is used to define the minimum temperature variation (in relation to the last temperature value sent) that causes the new measured value to be spontaneously sent. The values that can be set are:

- from 1 to 10 in steps of 1 [± 0.1°C] *(default value 5)*

➢ **18.1.5** ***Temperature sending period [minutes]***

This parameter is visible if the temperature sending conditions include “periodically”, and is used to define the frequency for spontaneously sending the measure temperature telegrams. The values that can be set are:

- from 1 to 255 in steps of 1 [minute] *(default value 5)*

18.2 Communication objects

The “temperature sensor” function displays the following communication objects:

	Number	Name	Object Function	Length	...	R	W	T	U	Data Type	Priority
	28	Ch.1 - Temperature sensor	Measured value [°C]	2 Byte	C	R	-	T	-	temperature (°C)	Low

Fig. 17.2

➢ *18.2.1 Ch.x - Temperature sensor*

The device uses this communication object to send the measured temperature value on the BUS according to the method set with the previous parameters.
The enabled flags are C (communication), R (reading from BUS), T (transmission).
The standardised object format is *9.001 DPT_Value_Temp*, the measurement unit in this case is **degrees Celsius (°C)**, and the object dimension is *2 byte*.

19 Communication objects

The communication objects are listed in the tables in the following paragraphs, divided according to functions.

19.1 Communication objects with output functions

For the *GW90833 - KNX Easy 2-channel contacts interface*, the objects relating to channels 3 and 4 are not present.

#				Object name	Object function	Description	Datapoint type
Ch 1	Ch 2	Ch 3	Ch 4				
1	7	13	19	Ch.x - Switch	On/Off	Sends switching ON/OFF commands	1.001 DPT_Switch
1	7	13	19	Ch.x - Timed switch	Activate timing	Sends timed ON/OFF switching commands	1.010 DPT_Start
1	7	13	19	Ch.x – Shutter stop /Louvres step	Stop/Step	Sends shutter stop and louvre step commands	1.007 DPT_Step
1	7	13	19	Ch.x – Wind alarm	Alarm input	Sends wind alarm/no wind alarm signalling	1.005 DPT_Alarm
1	7	13	19	Ch.x – Rain alarm	Alarm input	Sends rain alarm/no rain alarm signalling	1.005 DPT_Alarm
1	7	13	19	Ch.x – HVAC enabling	Enable/Disable	Sends enable/disable HVAC commands	1.003 DPT_Enable
2	8	14	20	Ch.x – Shutter movement	Up/Down	Sends Up/Down movement shutter commands	1.008 DPT_UpDown
3	9	15	21	Ch.x - Priority command	On/Off forced positioning	Sends enable/disable priority commands	2.001 DPT_Switch_Control
4	10	16	22	Ch.x – Brightness dimming	Increase/Decrease	Sends brightness regulation commands	3.007 DPT_Control_Dimming
5	11	17	23	Ch.x - Scene	Execute/Store	Sends command to execute or store scenes	18.001 DPT_SceneControl
5	11	17	23	Ch.x – HVAC mode	Auto/Eco/Precom/Comf/Off	Sends HVAC modes (auto/comfort/precomfort/economy/off)	20.102 DPT_HVACMode

19.2 Communication objects with input functions

For the *GW90833 KNX Easy 2-channel contacts interface*, the objects relating to channels 3 and 4 are not present and the objects relating to light signalling have a scaled index:

- Ch.1 - Light signalling 24 → 12
- Ch.2 - Light signalling 25 → 13

#				Object name	Object function	Description	Datapoint type
Ch 1	Ch 2	Ch 3	Ch 4				
0	6	12	18	Ch.x – Status feedback	On/Off status	Receives status feedback from binary actuators	1.001 DPT_Switch
0	6	12	18	Ch.x – Dimmer status feedback	On/Off status	Receives status feedback from dimmer actuators	1.001 DPT_Switch
0	6	12	18	Ch.x – Movement feedback	Increase/Decrease	Receives current movement direction feedback from the shutter actuator	1.008 DPT_UpDown
24	25	26	27	Ch.x – Light signalling	1=on/0=off	Enable/Disable light signalling	1.001 DPT_Switch

code 7.01.7.579.9

Ai sensi dell'articolo 9 comma 2 della Direttiva Europea 2004/108/CE si informa che responsabile dell'immissione del prodotto sul mercato Comunitario è:
According to article 9 paragraph 2 of the European Directive 2004/108/EC, the responsible for placing the apparatus on the Community market is:
GEWISS S.p.A Via A. Volta, 1 - 24069 Cenate Sotto (BG) Italy Tel: +39 035 946 111 Fax: +39 035 945 270 E-mail: qualitymarks@gewiss.com

SAT
+39 035 946 111
8.30 - 12.30 / 14.00 - 18.00
lunedì ÷ venerdì - monday ÷ friday
24h
+39 035 946 260
sat@gewiss.com
www.gewiss.com

LAST REVISION 01/2015