Technical Documentation Motivity

Status: Draft
Firmware Version : v1.2
Hardware Version : PIR Snr. 141023
Order Number: MOTX

Functional Description

The Motivity is a Motion Detector with optional additional sensors

Its functions are:

- Master/Slave Mode possible
- (Optional) Internal Temperature Sensor with correction
- (Optional) Internal Light Sensor with correction
- Up to 4 different Output Actions available
  - Output: switching, dimming, recalling scenes, 1 or 2 Byte unsigned
  - With switch-off delay
  - Light dependent
  - Activation of objects at runtime, e.g. for day/night mode
  - Ventilation Control
- Blocking object

This functionality is further made complete by some functional modules

- Scene Module
- Basic Logic Functions
- Timers
- Up/Down Counters
- Thermostat

The functions and parameters are explained in more detail hereafter.

The parameters are divided into 5 Parts

1. Configuration
   Here you can set some general settings.
2. Output
   Define up to 4 actions that have to be performed upon a motion event, possibly filtered by other variables.
3. Sensors
   Configure some sensor settings
4. Temperature
   Contains the settings of the internal temperature sensor, as well as those for the thermostat.
5. Modules
   Activate additional functionality that comes with each switch. Currently supported:
   - Scene Module, supporting eight scenes with eight actuators(1 Bit/1Byte/2Byte supported)
   - Basic Logic Module (AND/OR functions), consisting of five Logic Channels that each have up to five 1-Bit inputs
   - Timers, up to four
   - Up/Down Counters, up to four

General Settings

In this page you can set some general settings, e.g. whether a Light or Temperature sensor is attached to the Motivity base electronics.

You can also specify the "Read on Init delay time". The "Read On Init" flag is a communication object flag new for System B devices. If you set this flag on a communication object, then the intensity will issue read requests upon power-up, to make sure that its status values are up-to-date.

Set this value to a time, where you are certain that every bus device is up and running and will answer read requests. In that way you assure that the internal states of the Intensity correspond with the actual values.

List of available Communication Objects and Parameters:

Name	***Value Range***	***Comment***
Light Sensor Present	Yes/No	whether Light Sensor is attached
Temperature Sensor Present	Yes/No	whether Temperature Sensor is attached
Read On Init Delay[s]	4 – 255	Time To wait before reading objects with the ROI flag

Output

The settings under Output let you define which action should be executed when Motion is detected, and what should happen when the switch-off delay is reached.

General

If this Motivity only forwards its motion detection events to another KNX device that decides which action to take, you should set this Motivity into Slave Mode. However, if this device acts as the Master, you can set here which action to perform when motion is detected. Up to 4 different actions can be defined, which can be (de)activated on the fly.

Here you can also specity whether the Motivity should control the ventilation (see further).

You can also enable a blocking object, which will cause the Motivity to ignore any motion detection in the output object. Note that if blocking has been set after motion has been detected, then no OFF action will be taken., The other functions (e.g. Under sensors) are not affected by this setting and bus communication is still possible.

List of the parameters and communication objects:

Name	***Value Range***	***Comment***
Mode	Master / Slave	Select between Master/Slave Mode
Enable Blocking Object	Yes/No	enable a general blocking object which will enable/disable all output objects
Number of Output Objects	1 to 4	Number of possible actions this motivity must perform upon motion detection
Enable Ventilation	Yes/No	Enable ventilation logic

Communication Objects :

No	***Name***	***I/O***	***DPT***	***Flags***	***Use***
1	Blocking Object	I	DPT1.2	WCTUI	enable/disable all output objects
6	Slave Motion Detector Input	I	DPT1.1	WCTUI	Input detection status from Slave Motion Detector

Output

With each Output Action you can specify which action to perform when motion is detected, allowing you to e.g.

- switch on/off a 1-Bit actuator
- adjust the dimming to a fixed absolute value
- recall a scene
- send an (unsigned) 1 or 2 Byte value
- etc...

You can also specify a switch-off delay. This is the time after the last detection that the Motivity will wait to perform the turn off action.

It's also possible to make the action depending on a maximum brightness value. When the light level, from the internal or external sensor, is above this threshold, then the motivity will ignore the motion event.

Output Actions can be selected at runtime. This allows you to specify a different behaviour at a different time or in different conditions. When an Output Action is deactivated, it might be useful to turn off the light sources it controls. If this is necessary, set "Send OFF on deactivation" to "Yes".

The safety pause parameter will disable the motion detection for 2 seconds. This might be useful if a different light source is in the field of the detector, and switching on of this light sources causes the detector to trigger on it.

List of parameters and communication objects:

No	*Name*	*I/O*	*DPT*	*Flags*	*Use*
7	Switch Value 1bit	O	DPT1.1	CT	1 bit Output from Output Object1
7	Value 1 Byte Scaling	O	DPT5.1	CT	1 Byte Scaling Output from Output Object1, typically used for dimming
7	Value 1 Byte	O	DPT5.10	CT	Unsigned 1 Byte Output from Output Object1
7	Scene	O	DPT17.1	CT	Scene value from Output Object1
7	Value 2 Byte	O	DPT7.1	CT	Unsigned 2 Byte Output from Output Object1
8	External Lux Value	I	DPT9.4	CW	Value if external light sensor is used
9	Safety Pause Input	I	DPT1.1	CW	Input for ignoring briefly motion detector
10	Object Selection	I	DPT1.1	CW	Input for (de)activating the Output Object

The objects of the second Output Object start at 11. There is a spacing of 4 objects between each Output Action.

Name	***Value Range***	***Comment***
Value Type	1bit, 1 Byte Scaling[0-100%], 1 Byte unsigned [0-255], Scene [1-64], 2 Byte unsigned [0-65535]	type of value to send with this output action
ON Value to send on Start	depending on value type	value to send when motion is detected
OFF Value to send at End of Switch Off Delay	depending on value type	value to send when switch off delay has elapsed
Base Switch Off Delay	1 second, 1 minute, 1 Hour	Time Base for Switch Off Delay Time
Time Factor Switch Off Delay	1 to 240	factor to multiply Time Base Switch Off Delay Time
Depending on Brightness Value	Yes/No	the action is depending on the (internal or external)lux value
Lux Threshold	0-750	threshold value under which the light is switched on
Use internal sensor	Yes/No	Use internal brightness sensor, or a value from an external sensor, received over the bus
Use Safety Pause	Yes/No	ignore motion detector input for 2 seconds after a '1' telegram was received on the “Safety Pause Input” object
Activation	Always Active, Active on 1, Active on 0	(de)activate this Output Action
Send OFF Value upon deactivation	Yes/No	Send OFF value when Output Action is active and deselected

Ventilation

The motivity is able to control ventilation. After a person has entered a room, ventilation can be started, after a certain amount of time has elapsed. The person has to be present for a minimum time, at least 5 seconds. If the minimum presence time is bigger than the delay time, then the delay time will be automatically extended to be equal to the minimum presence time. Note that the blocking object will block this when active.

List of the parameters and communication objects:

Name	***Value Range***	***Comment***
Ventilation Delay Time[seconds]	5 to 240	time to wait after the first detection to start the ventilation
Ventilation Duration time [x10 seconds]	1 to 240	Duration of the venilation
Ventilation Minimum Presence time[seconds]	5 to 240	Minimum time between the first detection and last detection to start the ventilation

No	***Name***	***I/O***	***DPT***	***Flags***	***Use***
23	Ventilation Switch	O	DPT1.1	CT	Ventilation Switch object

Sensors

To obtain direc feedback about the Motion Detector and Brightness.

Motion Detector

If you only want to capture the state of the (debounced) Motion Detection sensor (e.g. For Slave – Master setup) then you can use the "Detector Status" object.

You can also enable the Green Feedback LED for feedback.

Name	***Value Range***	***Comment***
Detector Status	Do Not Send, Send 1 on every detection, Send 1 at start Detection, 0 at end	whether to use a fixed scene value, or use one from an external communication object
Enable Feedback LED	Yes/No	turn on the (Green) LED when the motion detector senses motion

No	***Name***	***I/O***	***DPT***	***Flags***	***Use***
5	Motion Detector Status	O	DPT1.1	CT	motion detector status

Brightness Sensor

To get information about the measured light intensity, you can query the brightness sensor. Due to tolerances in the brightness sensor electronics, it might be necessary to correct the measured value.

If you want to switch on a certain brightness level (e.g for determing the difference between day/night), you can use the Threshold value switch communication object. A

If the Lux Hysteresis is different from 0, then the switch object will be set to 0 only when the measured light level is below the threshold minus hysteresis value.

Name	***Value Range***	***Comment***
Correction for Lux Value	-127,128	correct on the measured brightness value
Send at least every … minute(s)	0-60	send the lux value cyclically
Threshold value for switch object	0-750	threshold level
Hysteresis	0-255	hysteresys value

No	***Name***	***I/O***	***DPT***	***Flags***	***Use***
3	Lux Value	O	DPT9.4	CRT	measured brightness value
4	Lux Threshold Switch	O	DPT1.1	CT	threshold value has been surpassed

Temperature

All temperatures are in °C.

Internal Sensor

The Intensity contains an internal Temperature sensor which is sampled every 5 seconds. It measures temperatures from -40 °C to 125 °C. By using the parameter Temperature Correction you can correct the measured value when necessary. The temperature value can be send onto the bus after a cyclical time or when the new measured value differs too much from a previous value. The latter case will also reset the cyclical timer.

List of the parameters and communication objects:

Name	***Value Range***	***Comment***
Temperature Correction (x 0.1 °C)	-50 to 50	correction on measured value, in tenth of a degree (range -5° to +5°)
Send at a deviation of ... (x 0.1 °C)	0 to 50	send temperature automatically onto the bus as soon as the difference between the last sent temperature is bigger than the current temperature + or - this parameter. Set to 0 to disable.
Send at least every ... minute(s)	0 to 60	send temperature periodically onto the bus. Set to 0 to disable.

No	***Name***	***I/O***	***DPT***	***Flags***	***Use***
2	Internal Temperature	O	DPT9.1	RCT	(Corrected) Internal Measured Temperature

Thermostat

You can use a built-in thermostat to control the climate of your room. Heating, Cooling, Heating + Cooling with manual or automatic switch is supported. If a wait time is specified, then the thermostat will wait that period after a switch occurred before controlling the heating/cooling.
The status of heating/cooling can be monitored by the heating/cooling mode feedback, or by the corresponding bits in the RHCC Status Feedback object.

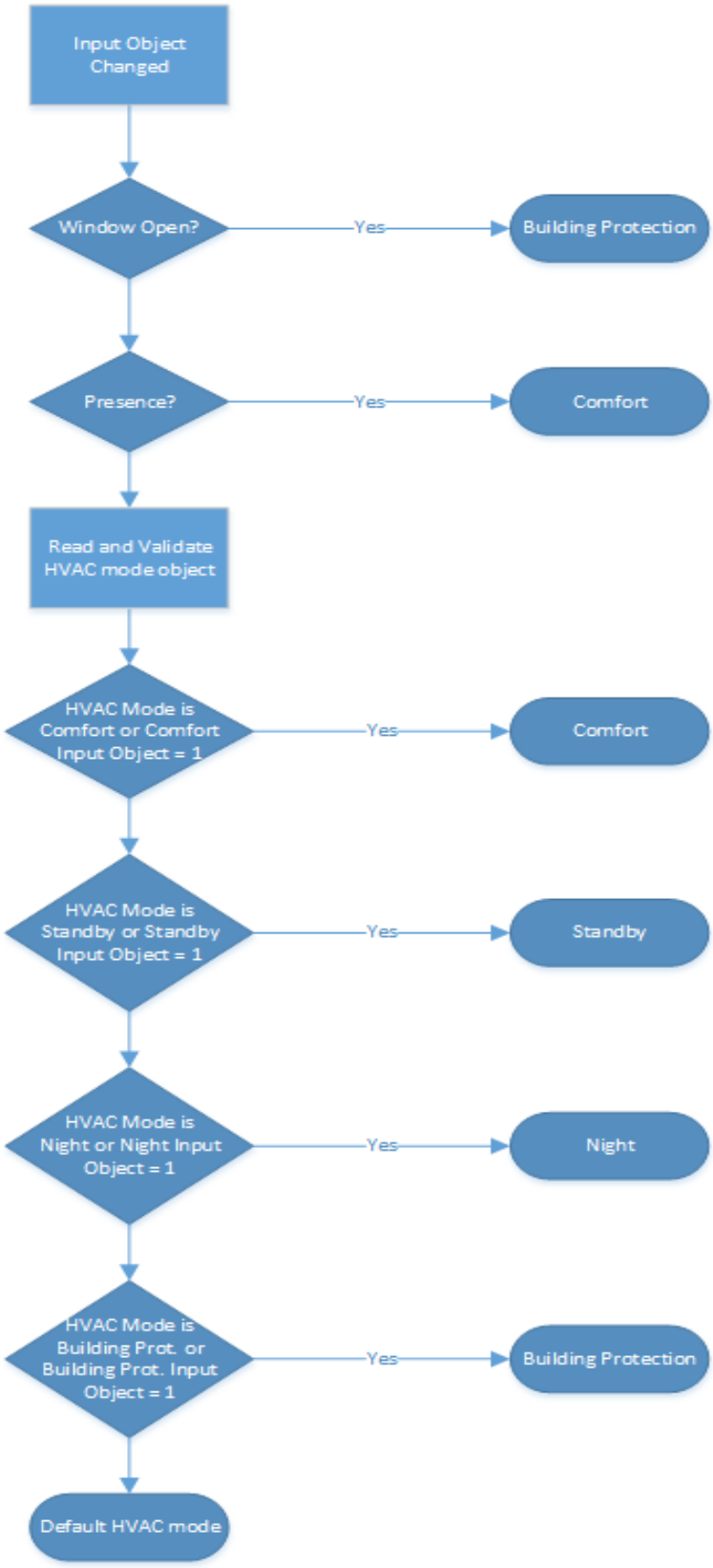
The method for controling the climate can be 2-Step switching , PI continuous or PI switching. When switching is selected, an output object Heating (or Cooling) switch will appear. In the other case, a Heating (or Cooling) value will make you able to control your heating/cooling equipment.

4 Modes are supported :

- Comfort
- Standby
- Night
- Building Protection (a.k.a. Heat/Frost Protection)

The modes can be set over the HVAC object (DPT 20.102) , or over the corresponding 1 Bit object. The Presence or Window open object can also alter the current mode.

For a schematic overview of how the current HVAC mode is determined, see below.


Input Object Changed
Window Open?
Yes
Building Protection
Presence?
Yes
Comfort
Read and Validate HVAC mode object
HVAC Mode is Comfort or Comfort Input Object = 1
Yes
Comfort
HVAC Mode is Standby or Standby Input Object = 1
Yes
Standby
HVAC Mode is Night or Night Input Object = 1
Yes
Night
HVAC Mode is Building Prot. or Building Prot. Input Object = 1
Yes
Building Protection
Default HVAC mode

Feedback about the current mode can be obtained through the mode feedback status objects or through the HVAC status object (DPT_HVACStatus).

Status and Errors

Through the RHCC Status Feedback object (DPT 22.101) you can track the status of the thermostat. Following bits are implemented:

Bit0 : Fault

Bit7 : Heating Disabled

Bit8 : HeatCool mode

Bit11 : Cooling Disabled

Bit12 : DewPoint status

Bit13 : FrostAlarm

Bit14 : Overheat Alarm

Bit 6 (Controller Status) of the HVAC Status object indicates whether the thermostat is actively heating/cooling.

The "Dewing Point Alarm" will disable the cooling immediately.

Frost Alarm will be set if the temperature drops below the frost protection setpoint. Overheat alarm when temperature is higher than the heat protection temperature.

Setpoint

The setpoint can be controlled in 2 ways

- by writing a new temperature to the Setpoint object
- Using the Setpoint Adjustment object. This can either be a floating point offset or a 1 bit switch object. The latter you can use to change the setpoint using button input. Writing a 0(up) will increase the offset with 0,5 K, writing a 1 (Down) will decrease it.

You can also limit the values that are written directly to the setpoint object, by setting "New Setpoint within Adjustment bounds" to "Yes".

Feedback of the setpoint , or the adjustment, will be given through the feedback objects.

Actual Temperature

It's possible to use an external temperature sensor to determine the actual temperature of the room, optionally for a certain proportion. Feedback on the calculated temperature can be obtained through the "Actual Temperature Feedback" object.

Name	***Value Range***	***Comment***
Control Mode	Heating / Cooling / Heating and Cooling (Automatic Switch) / Heating and Cooling (Manual Switch)	Heating or Cooling Mode of the thermostat. When using the manual switch you can use the heating/cooling Selection object to set the desired mode(0 = Cooling, 1 = Heating). When the switch is done automatically, then the current mode will depend on the current temperature. When lower than the heating setpoint, heating mode will be activated. Higher than the cooling setpoint will set cooling mode active
Wait Time after switch (min)	0 – 240	time to wait after a heating/cooling mode switch has occurred to actually control the HVAC equipment
Default HVAC Mode	Comfort / Standby / Night / Building Protection	when the current HVAC mode cannot be determined (when the other objects have not be written or at 0), the thermostat will switch to this mode
Use external Temperature Sensor	Yes/No	whether you want to use an external temperature sensor or the internal one. The actual temperature the termostat is using can be read from the Actual Temperature Feedback object.
Proportion external sensor	20 % / 40 % / 50 % / 60 % / 80 % / 100 %	proportion of the the external sensor that is used to calculate the actual temperature.
Maximum Adjustment up	0°C to 7°C	maximum upward offset that can be set on the Setpoint Adjustment Object
Maximum Adjustment down	0°C to 7°C	maximum downward offset that can be set on the Setpoint Adjustment Object
Setpoint Adjustment Over	2 Byte Floating Point Object / 1 bit Object	wheter you want to use a 1 bit object (0 = +0.5°, 1 = -0.5°) or a floating point object to set the offset.
Setpoint Comfort Mode	5°C – 40 °C	Setpoint in HVAC mode “Comfort”
Standby Offset	+/- 0°C - 7°C	Offset applied to Setpoint in Standby Mode
Night Offset	+/- 0°C - 7°C	Offset applied to Setpoint in Night Mode
Setpoint Frost/Heat Protection	5°C – 40 °C	Setpoint in HVAC mode “Building Protection”
Control Method	2-Step Switching / PI Switching / PI Continuous	method to determine the heating/cooling. The best method depends on the type of HVAC equipment
Hysteresis Up	0.3 °C to 2.0 °C	the difference the Setpoint should be surpassed to stop heating/cooling
Hysteresis Up	0.3 °C to 2.0 °C	the difference the Setpoint should be surpassed to start heating/cooling
PWM Cycle Time (minutes)	1 to 60 minutes	the period of time of the PWM cycle duration when PI Switching is selected
Cooling System	Cooling Ceiling (5K / 240 min) / Fan Air Convector (4K / 90 min) / Split Unit (4K / 90 min) / User Defined	the type of cooling system. The differential and proportial factors for the PI controller are derived from this.
Heating System	Warm Water Heating (5K / 150 min) / Underfloor Heating (5K / 240 min) / Electric Heating (4K / 100 min) / Fan Convector (4K / 90 min) / Split Unit (4K / 90 min) / User Defined	the type of heating system. The differential and proportial factors for the PI controller are derived from this.
Proportional Range (x 0.1K)	10 – 50	the proportional factor of the PI controller
Reset Time (minutes)	0 – 240	the reset time of the PI controller

No	Name	I/O	DPT	Flags	Use
24	External Temperature	I	DPT9.1	WCTUI	Temperature from external sensor
25	Setpoint	I	DPT9.1	WCTUI	Current Setpoint
26	Setpoint Adjustment	I	DPT9.1	WC	Setpoint adjustment value (float value)
27	Setpoint Adjustment	I	DPT1.8	WC	Setpoint adjustment value (1 Bit input – Up/Down)
28	Dewpoint Alarm	I	DPT1.2	WC	dewpoint alarm for Thermostat in Cooling Mode
29	Presence	I	DPT1.2	WC	Presence object for determing the HVAC mode. Normally HVAC mode will switch to Comfort mode
30	Window Open	I	DPT1.2	WC	Window Open object for determing the HVAC mode. Normally HVAC mode will switch to Building Protection mode
31	HVAC mode	I	DPT20.102	WC	HVAC mode object for controlling the HVAC mode according to the values defined in DPT_HVACMode [0 .. 4]
32	Frost/Heat protection Mode	I	DPT1.2	WC	switches the thermostat in Frost/Heat protection mode
33	Comfort Mode	I	DPT1.2	WC	switches the thermostat into comfort mode
34	Standby Mode	I	DPT1.2	WC	switches the thermostat into standby mode
35	Night Mode	I	DPT1.2	WC	switches the thermostat into night mode
36	Heating/Cooling selection	I	DPT1.2	WCTUI	if heating/cooling mode is set to switch manually, then writing 1 onto this object activates the heating mode
37	RHCC Status Feedback	O	DPT22.101	RCT	status information, bits defined according to DPT_RHCCStatus
38	Actual Temperature Feedback	O	DPT9.1	RCT	the calculated temperature, from internal and external sensors, used by the thermostat
39	Current Setpoint Feedback	O	DPT9.1	RCT	the actual setpoint, adjustment included
40	Setpoint Adjustment Feedback	O	DPT9.1	RCT	the adjustment applied
41	HVAC Status Feedback	O	--	RCT	HVAC mode feedback, bits according to DPT_HVACStatus
42	Frost/Heat Protection Feedback	O	DPT1.2	RCT	HVAC mode feedback, whether thermostat is in Frost/Heat Protection mode
43	Comfort Mode Feedback	O	DPT1.2	RCT	HVAC mode feedback, whether thermostat is in Comfort mode
44	Standby Mode Feedback	O	DPT1.2	RCT	HVAC mode feedback, whether thermostat is in Standby mode
45	Night Mode Feedback	O	DPT1.2	RCT	HVAC mode feedback, whether thermostat is in Night mode
46	Cooling Switch	O	DPT1.1	CT	Cooling object, active when the cooling is actively cooling the room
47	Cooling Value	O	DPT5.1	CT	The calculated PWM value from the PI controller.
48	Heating Switch	O	DPT1.1	CT	Heating object, active when the thermostat is actively heating the room
49	Heating Value	O	DPT5.1	CT	The calculated PWM value from the PI controller.
50	Cooling Mode feedback	O	DPT1.2	RCT	feedback whether the thermostat is in cooling mode
51	Heating Mode feedback	O	DPT1.2	RCT	feedback whether the thermostat is in heating mode

Modules

Under modules you can activate additional functionality that comes with each switch.

- Scene Module, supporting eight scenes with eight actuators(1 Bit/1Byte/2Byte supported)
- Basic Logic Module (AND/OR functions), consisting of five Logic Channels that each have up to five 1-Bit inputs
- Timers, up to four
- Up/Down Counters, up to four

Scene Module

The scene module is a matrix of 8 actuator groups, with 8 scenes that **optionally** have a value for every actuator. If you want an actuator not to change with a scene, you can specify not to use this value for this scene.
For actuator 1-6 the type must be 1 Bit or 1 Byte. Actuator 7 and 8 can additionally contain a 2 Byte value.

It is also possible to save scenes. When the scene module receives a request to save a scene, it will issue a read request for the corresponding actuators and wait 1 second to receive all the read responses. It then saves and start using the received values.

It's also possible to choose not to overwrite the existing scene parameters. This is useful in the case the end user already changed the scene theirselves using the scene save functionality after a long touch. If you change the type of an actuator, then you must set this parameter to "No".

List of Parameters and Communication objects:

Name	***Value Range***	***Comment***
Overwrite Existing Scenes	Yes/No	Overwrite the scenes that were specified in the past. Otherwise, use the scene values from the parameters.
Actuator Type Group *1-6*	Switch 1 bit / Value 1 Byte	Value type of actuator
Actuator Type Group *7-8*	Switch 1 bit / Value 1 Byte / Value 2 Byte	Value type of actuator
Scene number	0 – 63	Number of this Scene. Writing this value to the com object "Scene Function" will activate this scene.
Use Value *X*	Yes/No	whether to sent the value to the actuator in this scene, or ignore it
Value	On / Off	value for 1 bit actuator
Value	0-255	value for 1 Byte actuator
Value	0-65535	value for 2 Byte actuator

No	*Name*	*I/O*	*DPT*	*Flags*	*Use*
52	Scene Function	I	DPT18.1	WC	Input object of scene number of type DPT_SceneControl
53	Actuator 1 – Switch 1 Bit	I/O	DPT1.1	WCTU	1 bit value to be sent / saved when a scene is recalled / saved
54	Actuator 1 – Value 1 Byte	I/O	DPT5.10	WCTU	1 Byte value to be sent / saved when a scene is recalled / saved
55	Actuator 2 – Switch 1 Bit	I/O	DPT1.1	WCTU	1 bit value to be sent / saved when a scene is recalled / saved
56	Actuator 2 – Value 1 Byte	I/O	DPT5.10	WCTU	1 Byte value to be sent / saved when a scene is recalled / saved
57	Actuator 3 – Switch 1 Bit	I/O	DPT1.1	WCTU	1 bit value to be sent / saved when a scene is recalled / saved
58	Actuator 3 – Value 1 Byte	I/O	DPT5.10	WCTU	1 Byte value to be sent / saved when a scene is recalled / saved
59	Actuator 4 – Switch 1 Bit	I/O	DPT1.1	WCTU	1 bit value to be sent / saved when a scene is recalled / saved
60	Actuator 4 – Value 1 Byte	I/O	DPT5.10	WCTU	1 Byte value to be sent / saved when a scene is recalled / saved
61	Actuator 5 – Switch 1 Bit	I/O	DPT1.1	WCTU	1 bit value to be sent / saved when a scene is recalled / saved
62	Actuator 5 – Value 1 Byte	I/O	DPT5.10	WCTU	1 Byte value to be sent / saved when a scene is recalled / saved
63	Actuator 6 – Switch 1 Bit	I/O	DPT1.1	WCTU	1 bit value to be sent / saved when a scene is recalled / saved
64	Actuator 6 – Value 1 Byte	I/O	DPT5.10	WCTU	1 Byte value to be sent / saved when a scene is recalled / saved
65	Actuator 7 – Switch 1 Bit	I/O	DPT1.1	WCTU	1 bit value to be sent / saved when a scene is recalled / saved
66	Actuator 7 – Value 1 Byte	I/O	DPT5.10	WCTU	1 Byte value to be sent / saved when a scene is recalled / saved
67	Actuator 7 – Value 2 Byte	I/O	DPT7.1	WCTU	2 Byte value to be sent / saved when a scene is recalled / saved
68	Actuator 8 – Switch 1 Bit	I/O	DPT1.1	WCTU	1 bit value to be sent / saved when a scene is recalled / saved

Timer

A Timer object can be used to start an action after another one has occurred, with a delay time. It is also possible to send out a value cyclically. In the latter case, a value will be sent each time the timer expires, as long as the input Activation Object is 1.

The timer period is Factor x Time Base, allowing you to specify a period from 100 milliseconds up to 255 hours.

You can choose when to activate the timer; wheter a 1, or a 0 is written to the object, or on both edges. There is no way to cancel a non-cyclical timer that has been activated.

The value that is sent can be freely chosen.
The copy/invert of a 1 Bit object will use the "Input Value" when the timer expires, not the value at the time of activation of the timer.

If a timer is "Resetable", then an incoming telegram on the activation object will reset the timer period to 0.

List of parameters and communication objects:

Name	***Value Range***	***Comment***
Time Base	100 milliseconds / 1 second / 1 minute / 1 hour	base of time calculation
Time Factor	1 – 255	multiplied by Time Base to get the timer Period
Value	On / Off / Invert / Copy	Value to be transmitted. When Invert or Copy is selected than an 1 bit input object will be available onto which the input value must be written so that the desired operation can be carried out.
Value	0 – 255	1 Byte Value to be transmitted after timer expiry
Value	0 – 65535	2 Byte Value to be transmitted after timer expiry
Use Value from External Object	Yes/No	Available when 1 / 2 Byte output value type is selected. Provides a means to sent out a copy of a value when the timer expires.
Cyclic	Yes/No	timer is cyclic, thus will be restarted automatically every time the timer expires. When the activation object is set to 0 , then the timer will stop.
Use Value from External Object	Yes/No	Available when 1 / 2 Byte output value type is selected. Provides a means to sent out a copy of a value when the timer expires.
Activation on	On / Off / Both Edges	Edge on which the timer has to start
Resetable	Yes/No	a new write onto the input object that matches the Activation Parameters will reset the timer

These are the communication objects for Timer 1. Timer 2 starts at 113, with 5 objects intermittently for subsequent timers.

No	*Name*	*I/O*	*DPT*	*Flags*	*Use*
108	Activate	I	DPT1.1	WC	Activate the timer
109	Input Value – 1 Bit	I	DPT1.1	WC	Input value of a 1 Bit timer
109	Input Value – 1 Byte	I	DPT5.10	WCTUI	Input value of a 1 Byte timer
109	Input Value – 2 Byte	I	DPT7.1	WCTUI	Input value of a 2 Byte timer
110	Switch 1 Bit	O	DPT1.1	CT	Output value of a 1 Bit timer
111	Output Value – 1 Byte	O	DPT5.10	CT	Output value of a 1 Byte timer
112	Output Value – 2 Byte	O	DPT7.1	CT	Output value of a 2 Byte timer

Up/Down Counter

The Up/Down counter makes it possible to maintain a 1Byte unsigned value centrally, which can be increased/decreased with the specified step value by writing a 1(decrease)/0(increase) onto the input object. The bounds of this value will limit its range. It is also possible to reset the value by writing a 1 to the Reset Object.

List of parameters and communication objects:

Name	*Value Range*	*Comment*
Reset Value	0-255	initial value, or value to be set when the reset object is set to 1 .
Step Value	0-255	value to be added/subtracted each time the input object is set.
Minimum Value	0-255	the minimum value the counter can have. The counter will stop subtracting values once this value has been reached.
Maximum Value	0-255	the maximum value the counter can have. The counter will stop adding values once this value has been reached.

These are the communication objects for Up/DownCounter 1. Object for subsequent Up/DownCounters have 3 objects intermittently.

No	*Name*	*I/O*	*DPT*	*Flags*	*Use*
128	Input Value	I	DPT1.8	WC	Input value , 0 (Up) from adding the step value to the counter, 1(Down) to subtract it.
129	Reset	I	DPT1.1	WC	Input value
130	Output Value	O	DPT5.10	CT	Output value of the counter

Logic Module

Up to 5 logic channels can be defined. Each channel has up to 5 logic inputs , which can be inverted individually. You can use those to make an AND/OR comparison. The result of the function will be set onto the KNX bus, depending on the chosen setting:

1. only when the result of the logic function changes.

2. every time something is written onto an input object

4 output types can be selected: Switch 1 bit, 1 Byte, 2 Byte and 3 Byte RGB object (DPT232.600).
You can also use this mechanisme as a value converter, enabling you to generate a value from the above types, starting from a single 1 bit communication object.

At startup the initial values of the logic inputs (0 by default, 1 if they are inverted) will be evaluated and the result of the function will always be sent onto the bus. Enable the ROI flag if you want to read effectively the object's value at startup. Upon reception of the value the logic function will be executed.

List of parameters and communication objects:

Name	*Value Range*	*Comment*
Logic Function	And / Or	type of logic function to apply
Number of Input Objects	1 – 5	how many 1 bit inputs the function uses
Invert Input 1	Yes/No	whether to invert input 1
Invert Input 2	Yes/No	whether to invert input 2
Invert Input 3	Yes/No	whether to invert input 3
Invert Input 4	Yes/No	whether to invert input 4
Invert Input 5	Yes/No	whether to invert input 5
Sending Condition	Not Automatic / When Input Object is Written / When Result Changes	when to send the result of the logic function
Output Value Type	Switch 1 bit / Value 1 Byte / Value 2 Byte / Value 3 Byte	value type of the result
Send value when expression is True	Yes/No	whether a value is to be sent when the expression evaluates to True
Send value when expression is False	Yes/No	whether a value is to be sent when the expression evaluates to True
Value	On / Off	1 Bit result
Value	0 – 255	1 Byte result
Value	0 – 65535	2 Byte result
Value Byte 1	0 – 255	1 Byte part when value Type is 3 Byte. This byte corresponds to Red when using DPT232.600 (RGB value)
Value Byte 2	0 – 255	1 Byte part when value Type is 3 Byte. This byte corresponds to Green when using DPT232.600 (RGB value)
Value Byte 3	0 – 255	1 Byte part when value Type is 3 Byte. This byte corresponds to Blue when using DPT232.600 (RGB value)

List of communciation objects for Logic Function 1.

No	*Name*	*I/O*	*DPT*	*Flags*	*Use*
78	Input 1	I	DPT1.2	WC	Logic Input Object 1
79	Input 2	I	DPT1.2	WC	Logic Input Object 2
80	Input 3	I	DPT1.2	WC	Logic Input Object 3
81	Input 4	I	DPT1.2	WC	Logic Input Object 4
82	Input 5	I	DPT1.2	WC	Logic Input Object 5
83	Switch – 1 Bit	O	DPT1.1	CT	1 Bit Switch Output Object
83	Value – 1 Byte	O	DPT5.1	CT	1Byte Output value
83	Value – 2 Byte	O	DPT7.1	CT	2Byte Output value
83	Value – 3 Byte	O	DPT232.600	CT	3Byte Output value