KCI 4 S0
KNX Consumption Interface

KNX-S0 Interface with 4 Inputs

ZRX-KCI4S0

Application program version: [2.0]
User manual edition: [2.0]_a

CONTENTS

DOCUMENT UPDATES

Version	Changes	Page(s)
[2.0]_a	**Changes in the application program**: • Date and time-of-day objects. • Additional consumption registers (one total register plus three partial registers). • Per-register calculation of cost and equivalent CO2 emissions. • Per-tariff consumption calculation. • Periodical reset of partial registers. • Water/gas flow, electric power and pulse monitoring alarms. • Notifications upon exceedance of specific consumption values. • Inputs lock and register reading on request. • Ten logic functions. • Possibility of maintaining the previous consumption registers after download.	-
	Major revision of the whole document due to the new functionality.	-
[1.1]_a	**Changes in the application program**: • Optimisation of the S0-pulse count algorithm.	-

1 INTRODUCTION

1.1 KCI 4 S0

KCI 4 S0 from Zennio is an S0 to KNX interface that makes it possible to integrate S0-pulse generators into the KNX installation. Specifically, **up to four S0-pulse generators** can be connected independently to the inputs of KCI 4 S0, which will interpret these S0 pulses as energy, water or gas consumption measurements, depending on the external S0-compliant meter that has been connected.

Therefore, KCI 4 S0 brings the possibility of notifying other KNX devices about the **consumption levels** measured by external S0-pulse generators. In practice, KCI may still manage flows or volumes of any other type (not only electric power, water and gas) as long as they are measured by a pulse generator (even non-S0 pulses can be recognised). However this is <u>not guaranteed</u> – previous testing is encouraged.

The most outstanding features of KCI 4 S0 are:

- 4x inputs for monitoring **up to four S0-pulse generators**.

- Possibility of **disabling each input** independently.

- **Up to four registers** per input (one **total** and three **partial**).

- **Customisable pulse frequency**, i.e., possibility of setting how many pulses are expected per unit consumption.

- Possibility of sending the **energy, water or gas consumption values** to the KNX bus both periodically or on value changes.

- Calculation of the **electric power** and of the **water/gas flow rate**, with the possibility of sending them to the KNX bus both periodically or on value changes.

- **Alarms** for power / flow excess and low power / flow.

- **Per-tariff calculation of cost and CO2 equivalent emissions**, for up to two registers per input.

- Calculation of **cost and CO2 equivalent emissions for each register**.

- Pulse and consumption **monitoring alarms**.

- **Objects to request** the accumulated consumption values.

- Possibility of customising the **initial consumption value** and of maintaining the accumulated consumption after a download.

- Auxiliary **battery pack** to ensure the pulse count even in the absence of bus power.

- LEDs and communication objects to indicate **low or empty battery**.

- 10 customisable, multi-operation **logic functions**.

1.2 INSTALLATION

KCI 4 S0 connects to the KNX bus through the on-board KNX connector. Once the device is provided with power from the KNX bus, both the individual address and the associated application program can be downloaded.

1. Battery Holder.
2. EMPTY Battery LED Indicator.
3. LOW Battery LED Indicator.
4. Prog./Test Button.
5. Prog./Test LED.
6. Input Indicator LED.
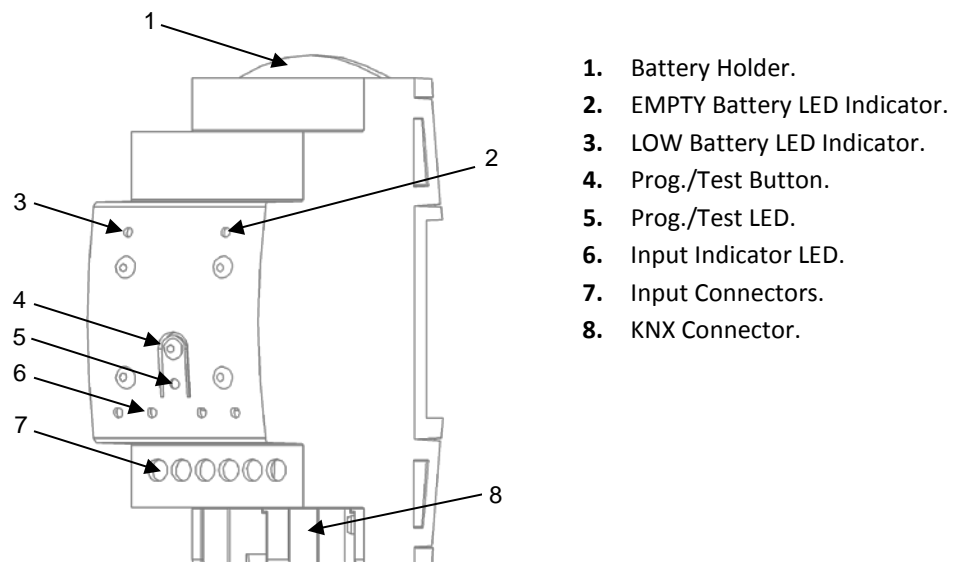7. Input Connectors.
8. KNX Connector.

Figure 1: KCI 4 S0 - Element Diagram.

The main elements of the device are described next.

- **Test/Prog. Pushbutton (4)**: a short press on this button sets the device into the programming mode, making the associated LED (5) light in red.

  Note: *if this button is held while plugging the device into the KNX bus, the device will enter into **safe mode**. In such case, the LED will blink in red.*

- **Input Connectors (7)**: slots for the connection of the input lines from the pulse generators. Whenever a pulse is received through one of the inputs, the associated LED (6) will flash briefly.

- **Battery Holder (1)**: host for the two CR2032 batteries (2 x 3V) that will provide power in the absence of the KNX bus.

- **Low-Battery Indicator (3)**: LED indicator that will blink in red if the battery level is weak. It is advisable to change the batteries when this happens.

- **Empty-Battery Indicator (2)**: LED indicator that will blink in red if the battery level is critically low. It is required to change the batteries when this happens.

Notes regarding the low/empty battery indicators:

- *The two LED indicators are mutually exclusive. Once the empty-battery indicator turns on, the low-battery indicator will turn off.*

- *In the absence of KNX bus power, the device will keep trying to count pulses even in the empty-battery state. However, once under the low-battery level, an accurate behaviour cannot be guaranteed.*

- *To extend the battery life, the empty/low battery indicators **only blink in presence of KNX bus power**. In other words, it is once the bus recovers from the power failure that KCI 4 S0 will resume the battery level check.*

To get detailed information about the technical features of the device, as well as on the installation and security procedures, please refer to the corresponding **Datasheet**, bundled with the original package of the device and also available at www.zennio.com.

1.3 CONSUMPTION DATA RESETS AFTER A DOWNLOAD

It is important to bear in mind that, by default, reprogramming the device from ETS (through either partial or complete downloads) clears the consumption variables and therefore any data accumulated so far on both partial and total registers (see sections 2.2.2 and 2.2.3).

Should these values need to be resumed after reprograming the device, the integrator should activate in ETS, prior to the download, the option to **maintain the current value** of such registers, or alternatively set an **initial value for the energy or volume**.

1.4 SETTING THE TIME OF DAY

To make use of calendar-dependent functions, such as periodical registers or the register initial dates, an external clock (such as the one in Z41 from Zennio) should be present in the installation in order to synchronise the KCI internal clock.

To that end, KCI will request the time of day during the start-up. Up to four read requests will be sent (after a customisable, initial delay) to the KNX bus. Specific **date** and **time of day** objects are provided for this.

Notes:

- *The time count is performed internally by KCI. However, it must be notified (through external events) about any time adjustments in order to ensure the synchronisation of the periodical timing of the registers .*

- *It is advisable to resynchronise the internal time count periodically to prevent precision loss after prolonged operation. Therefore, KCI sends a time request to the KNX bus every 15 minutes.*

2 CONFIGURATION

2.1 GENERAL

KCI 4 S0 incorporates four input channels for connecting up to four S0-pulse generators. Which ones will be in use is the first thing to configure.

On the other hand, two one-bit alarm objects will be available in the project by default, besides of **Date** and **Time of Day** objects (see section 1.4). Their behaviour will be analogous to that of the on-board LED indicators that inform, respectively, about **low battery** and **empty battery**. While any of these LEDs remains on, the corresponding object will be sent to the bus with the value "1" every 15 minutes. Once the alarm situation is over, the LED turns off and the object is sent once with the value "0".

Please, refer to section 1.2 for more details about the battery warnings.

Besides, additional objects can be enabled to **request global consumption data** or to **lock the inputs**.

Finally, ten customisable, multi-operation **logic functions** can be enabled too.

ETS PARAMETERISATION

After importing the corresponding database in ETS and adding the device into the topology of the desired project, the configuration process begins by right-clicking into the device and selecting *Edit parameters*.

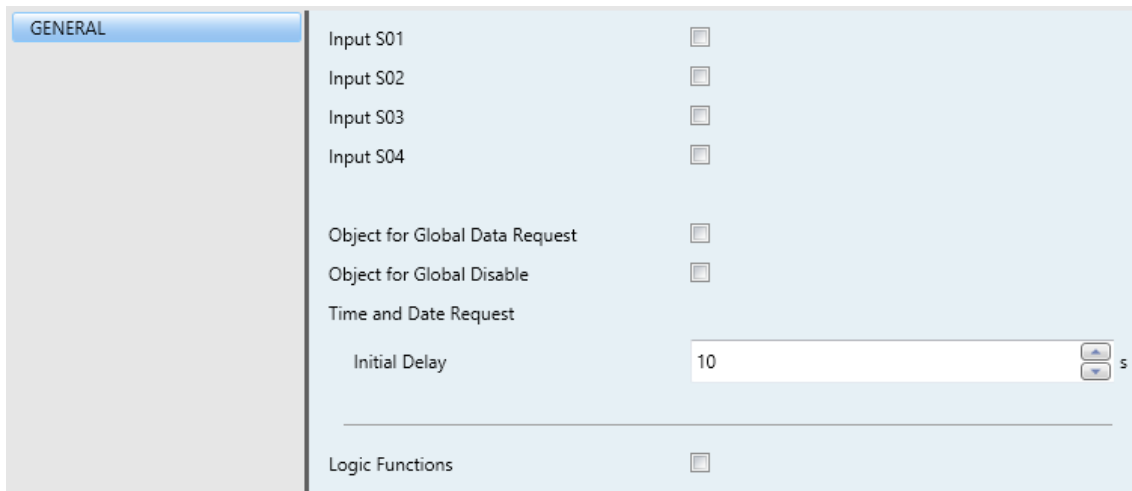The "General" parameter screen is the only shown by default.

Figure 2: General.

Four checkboxes (one per input channel) let the integrator select which input channels will be in use. Additional entries will be included in the tab menu on the left depending on the number of active inputs.

The options for each of the inputs are detailed in the next sections.

Besides, additional parameters are available to enable or disable the following communication objects:

- **Object for Global Data Request**: enables the "**Global Data Request**" one-bit object, which can be sent the value "1" to request the global consumption values from all the input channels.

- **Object for Global Disable**: enables the "**Global Disable**" one-bit object, which by default has the value "1". When it receives the value "0", KCI interrupts all actions, including the pulse count in all inputs. Only data requests will be responded.

- **Initial Delay**: time between 1 and 255 seconds the device will wait after the start-up before sending the date and time requests (see section 1.4).

- **Logic Functions**: enables the Logic Functions configuration screen (see section 2.4).

On the other hand, the following objects are available in the project topology at any time:

- "**Date**": its value will become periodically updated according to the internal date of the device. It can also receive values from an external clock for synchronisation purposes.

- "**Time of Day**": its value will become periodically updated according to the internal time of the device. It can also receive values from an external clock for synchronisation purposes.

- "**Alarm: Low Battery**": one-bit object that will send a "1" every 15 minutes if the battery level is low, and a "0" (only once) if this no longer happens. This object is analogous to the low-battery LED indicator.

- "**Alarm: Empty Battery**": one-bit object that will send a "1" every 15 minutes if the battery level is critically low (or zero), and a "0" (only once) if this no longer happens. This object is analogous to the empty-battery LED indicator.

As with the LED indicators, the two alarm objects are mutually exclusive. Once the level of the battery becomes empty, KCI 4 S0 will start sending the empty-battery object and will stop sending the low-battery object (which will be then sent with the value "0" –only once–).
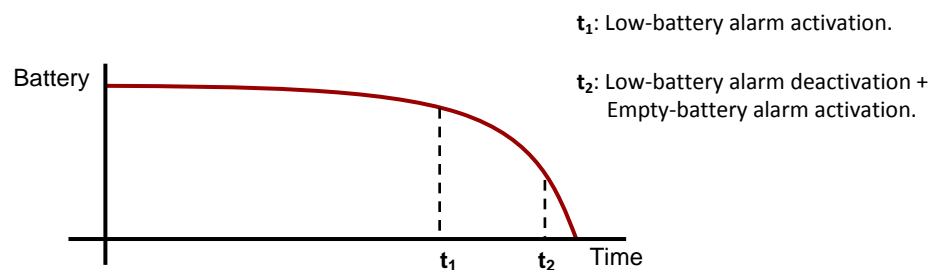
t_1: Low-battery alarm activation.

t_2: Low-battery alarm deactivation + Empty-battery alarm activation.

Figure 3: Battery Alarms

2.2 INPUT S0x

2.2.1 CONFIGURATION

Each input channel of KCI 4 S0 can be particularised for two different scenarios, depending on the type of the S0-pulse generator connected to it:

- Electrical energy consumption meters.

- Water or gas volume consumption meters.

The options available will slightly differ for each case, although are entirely analogous. Even though the compatibility is not guaranteed, KCI 4 S0 should also recognise pulses of any other kind (even non-S0); however, previous testing is highly advisable.

As electric power, water or gas are being consumed, the pulse generator will send pulses to KCI 4 S0. To interpret the pulses, KCI 4 S0 needs to know the **pulse frequency**, which is the ratio (provided by the manufacturer) between the number of pulses sent by the external meter and the amount of electrical energy (or water or gas) measured, i.e., the number of pulses sent per unit energy (or per unit volume). The higher the pulse frequency, the lower the consumption increases assumed by KCI 4 S0 on the reception of each pulse.

For each S0 input, it is possible to enable **a total register, up to three partial registers** and the **calculation of the electric power or water/gas flow.** The main difference between total and a partial register is the latter one can be reset periodically or through object (see section 2.2.3). Therefore, both the total consumption since the start-up and the partial consumption during several time intervals will be available. The energy/volume **units of measurement** can be selected independently for each register in order to allow more resolution when required. The options are: watt-hours (Wh), kilowatt-hours (kWh), litres (l) or cubic metres (m^3), depending on the meter type.

On the other hand, KCI can also calculate:

- The **kilogrammes of carbon dioxide** that have been released to the atmosphere to obtain the energy consumed. To that end, it is necessary to set the CO2 ratio, i.e., the CO2 kilogrammes per kWh (or m^3) consumed.
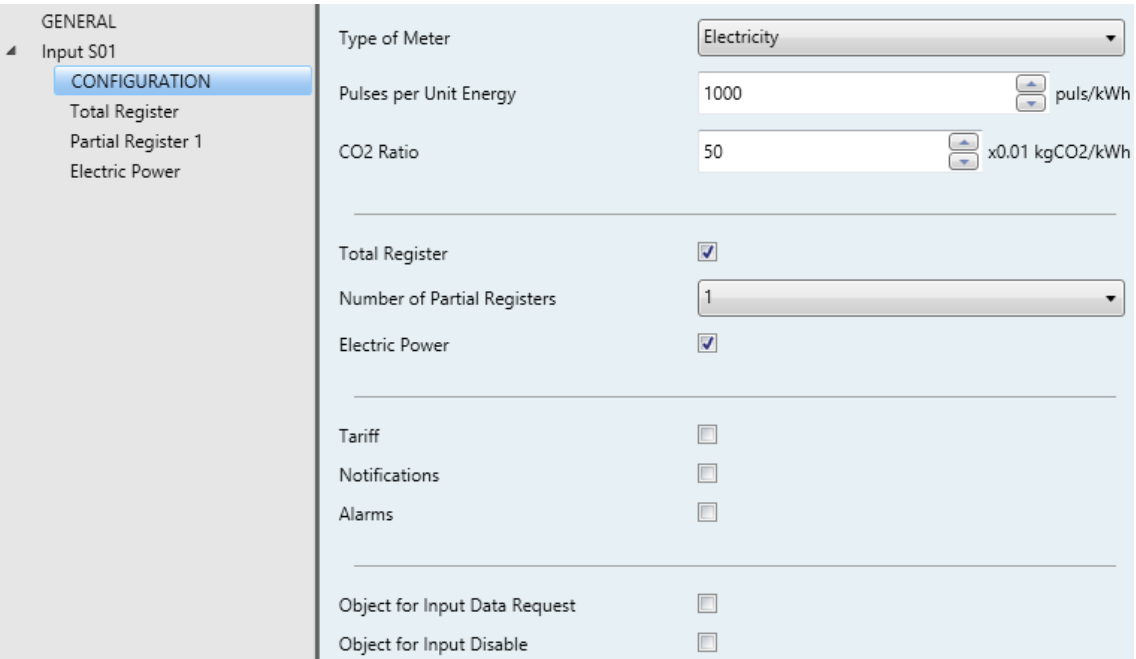
The **cost** of the consumed energy, according to a certain **tariff**.

Consumption notifications are available for each register, together with **power and flow alarms** (provided that the calculation of the power and flow has been enabled) as well as **pulse monitoring alarms**.

A binary communication object to **request the consumption data** of such input will be available any time, as well as a binary object to **disable the input**. While an input remains disabled, it will interrupt the pulse count and will not send any objects unless requested through the aforementioned, specific request object.

ETS PARAMETERISATION

For every active input, ETS shows one primary tab, containing another one named Configuration with the following parameters:

Figure 4: Input S0x – Configuration.

Type of Meter: "Electricity" (default option) or "Water or gas", depending on what the S0-pulse generator will be measuring.

Depending on the meter type, the following parameters will refer to energy or water/gas, although they are analogous in both cases:

- **Pulses per Unit Energy** (or **volume**): sets the pulse frequency, in pulses per kilowatt-hour (pulse/kWh) or pulses per cubic metre (pulse/m3). The available range is 1 to 65535 (the default is "1000").

- **CO2 Ratio**: sets the hundredths of a kilogramme of CO_2 released to the atmosphere per unit consumption (0.01 kgCO2/kWh or 0.01 kgCO2/m^3). The available range is 1 to 255 (the default is "50").

- **Total Register**: checkbox to show or hide the total register tab (see section 2.2.2).

- **Number of Partial Registers**: sets the number of partial registers (up to three) required for this specific input. There will be as many additional parameter tabs as registers. See section 2.2.3.

- **Electric power** (or **water/gas flow**): activates or deactivates the calculation of the electric power or the gas/water flow, depending on the meter type, as well as their configuration tabs. See section 2.2.4.

- **Tariff**: shows or hides the Tariff tab. See section 2.2.5.

- **Notifications**: shows or hides the Notification tab. See section 2.2.6.

- **Alarms**: shows or hides the Alarms tab. See section 2.2.7.

- **Object for Input Data Request**: shows or hides the "**[S0x] Input Data Request**" object. If KCI receives this object with value '1', it will send the KNX bus the consumption, cost and CO2 objects of the registers corresponding to this input.

- **Object for Input Disable**: shows or hides the "**[S0x] Input Disable**" object, which disables the current input when the value '0' is received through this object. Normal operation will be resumed as soon as the value '1' is received.

2.2.2 TOTAL REGISTER

The total register stores the value of the **global consumption** of either energy consumption or water/gas volume (depending on the selected meter type) together with

the value of the corresponding cost and CO_2 emissions. Therefore, it reflects the per-input values accumulated since the device start-up.

The **initial value of the register** can be set in parameters, being either a fixed value (such as '0') or the last value that may have already been saved in the register prior to the download. After the first download, the previous value will be considered as '0'.

It is also possible to overwrite the current **value** of the total register anytime through a specific object, to reset it or to set it to a specific value.

The current values of the total register can be sent to the KNX bus as follows:

- **Cyclically**: after a certain time (configurable).

    Note: *the value sent by KCI 4 S0 is not an exact instant value, but the best known value so far. Depending on the pulse frequency, there may be a certain delay between the value being measured by the external meter and that corresponding to the last pulse.*

- **On value change**: the current value will be sent to the bus as soon as it is found to differ from the last sent by more than a certain amount (configurable). On the arrival of a pulse, the device will verify if the difference between the new value and the last sent is greater than the configured threshold; if it is, it will notify the KNX bus. It is also possible to set a minimum time between sendings, to avoid massive transmissions in case of high consumption.

- **Cyclical and on value change**: a combination of the above two options. The current value will be sent periodically, and also when it is found to differ from the last one sent by more than the specified amount. Note that the cyclical sending will be also conditioned by the minimum time between sendings (if configured), so that after a sending due to a value change there will not be cyclical sendings until the minimum time expires.

ETS PARAMETERISATION

The parameters shown in the Total Register tab are the following:

Figure 5: Input S0x - Total Register

- **Energy** (or **Volume**) **Units**: sets units of measurement of the total register for either energy ("Wh" or "kWh") or volume (litres or cubic metres).

- **Initial Consumption**: sets the initial state of the total register after an ETS download: "Keep Current Value" or "Set Value". By selecting the latter, an additional textbox will show to enter the desired **value**, in terms of the selected units of measurement.

- **Object to Set the Energy (or Volume) Value**: enables the "**[S0x] [TR] Set Consumption**" four-byte object, which permits overwriting the current value of the total register, in terms of the configured units of measurement.

- **Additional Objects**: allows enabling four additional functions related to the consumption register:

Figure 6: Total Register - Additional Objects CAMBIAR

> **Cost** (only available if Tariff has been activated in the Configuration tab corresponding to the current input; see section 2.2.1): enables the "**[S01] [TR] Cost**" two-byte object, which can be read to obtain the cost corresponding to the total consumption currently registered.

➢ **CO2 Equivalent Emissions**: enables the "**[S01] [TR] CO2**" two-byte object, which can be read to obtain the kilogrammes of CO_2 corresponding to the total consumption currently registered.

➢ **Consumption in each Tariff** (only available if Tariff has been activated in the Configuration tab corresponding to the current input; see section 2.2.1): enables the "**[S01] [TR] Consumption in Tariff x**" and "**[S01] [TR] Cost in Tariff x**" objects, which can be read to obtain how much of the accumulated consumption and cost corresponds to tariff "x".

   Note: *this feature is available for up to two registers per input.*

• **Sending Mode**: sets whether the "**[S0x] [TR] Consumption**" object, together with the above additional objects, should be sent automatically to the bus. The options are: "Disabled", "Cyclical", "On Value Change" or "Cyclical and on Value Change". Depending on this selection, there more parameters will be available:

Figure 7: Total Register - Sending modes

➢ **Period**: sets the period for the periodical sendings.

➢ **Minimum Time between Sendings**: sets a minimum period of time between consecutive sendings.

➢ **Minimum Change to Send**: sets the minimum change required between two consecutive sendings.

2.2.3 PARTIAL REGISTERS
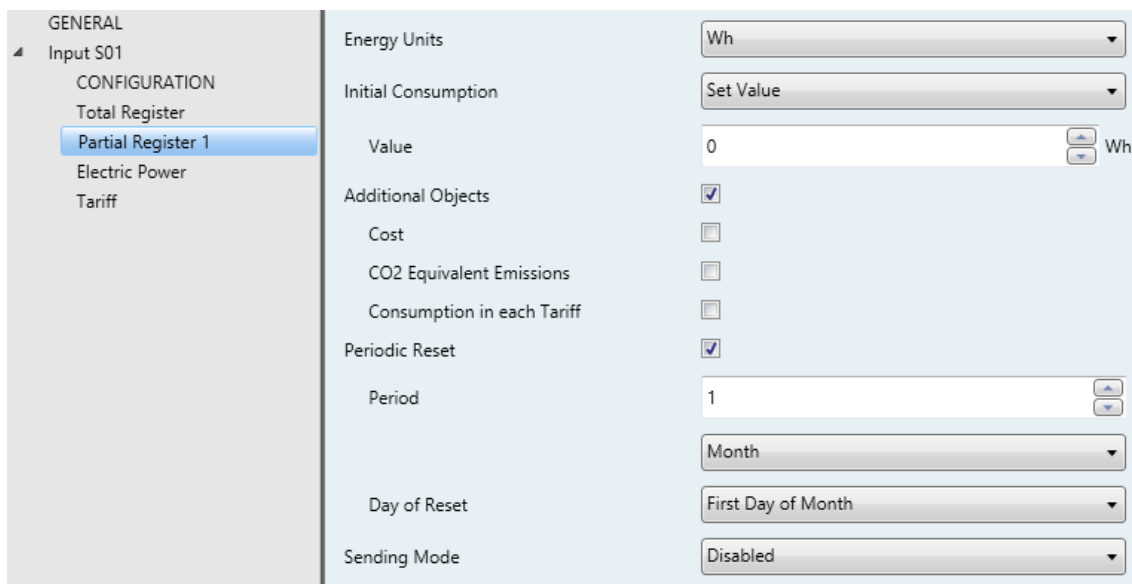
Up to three partial registers are also available per input, which will monitor consumption during different periods of time. As the total register, they can record consumption data itself, as well as other variables such as CO2 and cost.

In this case, setting the value of the register through an object is not possible. Nevertheless, the parameters involved are analogous to those in the total register (see section 2.2.2). In addition, a **periodic reset** can be configured with a periodicity of days or months. An object is also available to perform a reset anytime.

Together with the objects that contain the consumption, CO2 and cost values accumulated so far, an additional set of objects will be provided containing the values of the **immediately previous period**. Therefore, every time the configured periodicity expires and the register is reset, these objects will be sent to the bus to inform about the final values accumulated in that period. Note that this sending will take place even if the resending restrictions that may have been configured are not met.

ETS PARAMETERISATION

The parameters to configure any of the partial registers are:

Figure 8: Partial Register

- **Energy Units, Initial Consumption, Additional Objects** and **Sending Mode**, as well as the objects involved, are all of them similar to the analogous parameters in the total register (section 2.2.2).

- **Periodic Reset**: activates or deactivates the automatic reset of the partial register whenever a certain period expires:

  ➢ **Period**: from 1 (default) to 255 days or months.

  ➢ **Day of reset**: in case the above period has been set in terms of months, sets whether the reset should take place on the first day of the month (default option) or always on the same day of the month, taking into account the day the device starts working.

  The register can also be reset upon request by sending the value "1" to the "**[S01] [PRy] Reset**" binary object.

  Please bear in mind the following:

  ➢ In the absence of bus power at the end of the cycle, the register will be reset upon power recovery..

  ➢ In case of time adjustments, the resets may be affected by a delay.

  ➢ In case of selecting "Same Day of Month", the specific day of the month will change whenever the register is manually reset through the object.

2.2.4 ELECTRIC POWER (OR WATER/GAS FLOW)

Based on the accumulated energy consumption and the time lapse between the successive increments, KCI 4 S0 can obtain (and send to the KNX bus) the corresponding **electric power**, in terms of watts or kilowatts, or the corresponding **water/gas flow**, in terms of litres per hour (l/h) or cubic metres per second (m^3/s).

The power and water/gas flow will be sent to the bus through two different objects, one in watts (or litres per hour), and the other one in kilowatts (or cubic metres per second).

The integrator can select one out of two **alternative methods** to calculate the electric power and the flow, named as "**instantaneous value**" and "**average value**". How each of them works is described in ANNEX I: Power Calculation.

When the instantaneous value method has been selected, the electric power (or water/gas flow) will be calculated whenever a pulse arrives. The result of such calculation, therefore, can never be zero as there must be some consumption. Moreover, if no pulses are received (for example in case the power is actually zero), no power values will be sent to the bus. It is therefore necessary to set a **threshold in order to consider power (or flow) equal to zero**. In case the time since the last pulse is greater than such limit, the power or flow will be considered zero.

Note: *in practice, instead of configuring a time threshold, the integrator will be required to configure a power (or flow) threshold, so when the time that would correspond to that power/flow threshold expires, the latter will be considered zero.*

Depending on the selected calculation method, different possibilities to send the value to the bus are available:

- **With value change**: the value will be sent to the KNX bus as soon as it is calculated, provided that it differs from the last value sent. This option is only available for the instantaneous value method as the sendings depend on the arrival of the pulses. A minimum time and a minimum value change between sendings can be parameterised.

- **Periodic**: the value will be sent periodically. This option is only available for the average value method.

- **Periodic with minimum value change**: the value will be sent periodically although it will be discarded in case it does not differ from the last value sent in more than a certain amount. This option is only available for the average value method.

  Note: *bear in mind that the electric power (and water/gas flow) sent to the bus is always an estimate. Depending on the pulse frequency, even if there is some consumption, a power of 0 W may be estimated if no pulses have been received since the last sending.*

ETS PARAMETERISATION

Once the **electric power** or **water/gas flow** functions have been activated in the input (see section 2.2.1), the "**[S0x] Power (4 bytes)**" and "**[S0x] Power (2 bytes)**" objects (or the analogous for water/gas flow) will be shown in ETS, as well as a specific screen containing the following parameters. Note that these parameters may differ slightly depending on the meter type, although they are entirely analogous.

Figure 9**: Input S0x – Electric power** – Instantaneous **Value**

- **Calculation Method**: "Instantaneous Value (with the last two pulses)" (default option) or "Average Value (average value since the last sending)". Please refer to ANNEX I: Power Calculation for details.

The remaining parameters will depend on the above selection.

Under the "Instantaneous Value" method:

- **Minimum Power** (or **Flow**): sets the power (or flow) lower limit, which will determine the maximum time without pulse receptions the device will allow before considering that the power (or flow) is actually zero.

- **Sending Mode**: sets when the power (or flow) value will be sent to the bus. The only option for this method is "With Value Change".

  - ➢ **Minimum Time between Sendings**: sets the minimum time space (from 1 to 255 seconds) between consecutive transmissions.

  - ➢ **Minimum Change to Send**: sets the minimum increment or decrement over the last value sent (0 - 65535 W or l/h) that will trigger a new sending.

Note: *the assumption of null power (or flow) is always performed according to the minimum power (or flow) value configured above. However, these two values may condition when it is actually reported to the bus.*

Example: *null power may not be sent depending on the parameterisation.*

If the pulse frequency in an input is set to 1000 pulses/kWh and the minimum power is set to 100 W (under this value, the power object will take value 0), then 36 seconds with no pulses will be required to consider the current power as zero.

$$P_{min} = \frac{E}{t} \rightarrow 100W = \frac{1\,kW * 3600s}{t} \rightarrow t = 36s$$

If the minimum time between sendings is set to a value greater than 36, the value "0" of the power object will not be sent to the bus, as it is calculated 36 seconds after the last value was sent.

Under the "Instantaneous Value" method:

Figure 10: Input S0x - Electric power - Average Value

- **Sending Mode**: sets when the power (or flow) value will be sent to the bus. The options for this method are "Periodic" and "Periodic with Minimum Value Change".

  - ➤ **Period**: sets the periodicity of the sending. The available range is 10 to 255 seconds, 1 to 255 minutes and 1 to 255 hours.

  - ➤ **Minimum Change to Send** (only available for "Periodic with Minimum Value Change") sets the minimum increment or decrement over the last value sent (0 - 65535 W or l/h) that will trigger a new sending, provided the period configured above has lapsed.

2.2.5 TARIFF

For every register, it is possible to calculate the monetary cost of the accumulated consumption, together with the consumption corresponding to each tariff (for up to two registers per input), in case of a time-slot based tariff system.

Thus, it will be possible to configure up to **four tariffs** per input, being also possible to modify them through object. To allow tariff switchovers, the device will also provide a one-byte object will, so it can be notified about the tariff (1 to 4) to be adopted at the end of each time slot. In addition, another object will be available per input to inform about the current tariff.

PLEASE NOTE THAT THE ACTIVE TARIFF BY DEFAULT IS TARIFF NUMBER 1. THEREFORE, ANY INITIAL CONSUMPTIONS SET IN PARAMETERS WILL COMPUTE IN SUCH TARIFF. ETS PARAMETERISATION

The Tariff tab will become visible once the Tariff function has been enabled in the Configuration screen (see section 2.2.1).

The configurable parameters are:

Figure 11: Input S0x – Tariff

- **Number of Tariffs**: sets the number of available tariff slots, up to four.

  - **Tariff "y"**: sets the fee (in hundredths of the currency unit) per unit consumption (kWh or m^3) for tariff "y", from 0 to 65535.

Bear in mind that only after enabling the Tariff function the **Cost** and **Consumption in each tariff** parameters will be available under **Aditional objects**, within the parameter screens of the registers.

Enabling such function also brings the following objects to the project topology:

- "**[S0x] Set Tariff**", which allows setting the current tariff (from 1 to 4).

- "**[S0x] Tariff (Status)**", which reflects the currently active tariff (1 to 4).

- "**[S0x] Tariff 'y'**", which allows overwriting the fee initially set in parameters.

Important: *the per-tariff consumption objects are only available for up to two registers per input. It is up to the integrator the selection of these registers.*

2.2.6 NOTIFICATIONS

Notifications in KCI consist in sending a binary object to the KNX bus when the selected variables, stored in either total or partial registers, reach a pre-set value. This applies to **consumption** variables, as well as to **CO2 emissions** and **cost** variables. Up to eight notification objects can be configured per input.

Example: *Notifications.*

In an input configured as water / gas type, two notifications are enabled regarding the consumption stored in a partial register being reset every day. Their trigger values are, respectively, 100 litres and 200 litres.

On a daily basis, KCI will send one binary object as soon as the consumption exceeds the first value (100 litres) and another binary object as soon as it exceeds the second one (200 litres).

ETS PARAMETERISATION

The Notifications tab will be available after enabling such function in the Configuration screen (see section 2.2.1). It contains the following parameters, provided that at least one register has been enabled for the input.

Figure 12: Input S0x – Notifications

- **Notification 'y'**: activates or deactivates notification number 'y' and therefore the "**[S0x] Notification 'y'**" binary object.

  ➢ **Register**: sets the register containing the variable to be monitored. Only currently enabled registers will be displayed.

  ➢ **Variable**: sets the variable to be monitored. Only variables currently enabled for the above register will be displayed.

  ➢ **Limit**: sets the trigger value for the notification. The units of measurement will match those configured for the selected variable.

  ➢ **Action**: sets the value ("<u>0</u>" or "<u>1</u>") to be sent when the notification is triggered.

Note: *values between the maximum multiple of the pulse frequency and the maximum value allowed by the object should not be configured as limits. For instance, given a pulse frequency of 1000 W per pulse and considering that the maximum value allowed by the consumption DPT is 2147483647, it is important to avoid values greater than 2147483000 for consumption variables, as they will be never reached.*

2.2.7 ALARMS

In addition to notifications and battery alarms, KCI implements two electric power (or water/gas flow) alarms and one pulse monitoring alarm.

Power and flow alarms

Every input channel incorporates an alarm function to monitor the power (or flow) according to an upper limit (**excess power**) and a lower limit (**low power**), which can be enabled and parameterised independently and also modified dynamically through two specific objects.

In the event of being the calculated power greater or lower than such limits, KCI will send a binary value, respectively through two separate objects. The **values** to be sent can be configured in parameters, together with some additional options:

- A certain **hysteresis**, to prevent sending the alarm and no-alarm values repeatedly in case the power or flow remains close to the limit value for a long period of time.

- Whether the **no-alarm value** (i.e., the inverse of the configured alarm value) should be sent to the bus or not once the alarm condition is over.

- Re-sending **periods** for the alarm value and the no-alarm value, so the alarm object is re-sent periodically to the bus.

Pulse monitoring alarm

To be able to transmit the consumption information to the KNX system, KCI needs to count the pulses generated by an external meter. It is therefore advisable to ensure these **pulses are actually being received**.

To meet this need, each input channel in KCI provides an alarm to monitor the arrival of the S0 pulses from the external meters. A configurable binary value will start being sent to the bus (periodically) in case no pulses are received for more than a certain time, also configurable in parameters. Once the pulse reception is resumed, the contrary value will be sent to the bus (once).

ETS PARAMETERISATION

The Alarms tab will be available after enabling such function in the Configuration screen (see section 2.2.1). The available parameters are:

Figure 13: Input S0x – Alarms

Electric Power (or Water / Flow) Alarms

These parameters will only show after enabling the power (or flow) calculation function in the Configuration screen.

- **Excess Power / Flow**: enables or disables the excess power or flow alarm.

  - ➢ **Limit**: upper limit that will trigger the alarm in case the calculated power or flow exceeds it. The available range is 0 to 65535 (litres per hour or tens of watts, depending on the meter type). This value can be replaced afterwards by writing to the "**[S0x] Alarm: Excess Power Limit**" or "**[S0x] Alarm: Excess Power Flow**" objects.

> ➤ **Action**: sets the value to be sent ("1" or "0") through the "**[S0x] Excess Power**" or "**[S0x] Excess Flow**" objects on the activation of the alarm.

> ➤ **Periodic sending**: activates or deactivates the periodic resending of the alarm object.

> - **Period**: resending time from 1 to 255 seconds, minutes or hours.

> ➤ **Notification of No Alarm**: indicates if the alarm object should be sent with inverse value of the configured alarm value) once the power or flow are under the upper limit. Such value will also be periodically re-sent in case the above parameter has been activated.

> ➤ **Hysteresis under the Limit**: offset in litres per hour or tens of watt to postpone the deactivation of the alarm, which will not take place until the power or flow is lower than the upper limit minus the hysteresis. The available range is 0 to 255.

⦁ **Low Power (or Flow)**: enables or disables the low power flow alarm.

> ➤ **Limit**: lower limit that will trigger the alarm in case the calculated power or flow are below it. The available range is 0 to 65535 (litres per hour or tens of watts, depending on the meter type). This value can be replaced afterwards by writing to the "**[S0x] Alarm: Low Power Limit**" or "**[S0x] Alarm: Low Power Flow**" objects.

> ➤ **Action**: sets the value to be sent ("1" or "0") through the "**[S0x] Low Power**" or "**[S0x] Low Flow**" objects on the activation of the alarm.

> ➤ **Periodic sending**: analogous to the periodic sending of the excess power alarm.

> ➤ **Notification of No Alarm**: analogous to the no-alarm notification of the excess power alarm.

> ➤ **Hysteresis above the Limit**: offset in litres per hour or tens of watt to postpone the deactivation of the alarm, which will not take place until the power or flow is greater than the lower limit plus the hysteresis. The available range is 0 to 255.

Pulse Monitoring Alarm

- **Monitoring Period**: maximum time (in the range 1 to 65535 seconds, minutes or hours) the absence of pulses will not be considered an alarm situation. The value "0" disables this function.

- **Action**: sets the value to be sent to the bus ("Send 1" or "Send 0") through the "**[S0x] Alarm: Pulse Receiving – Error**" binary object when the alarm is triggered.

2.3 INITIALISATION AND POWER LOSS

After an ETS download (as long as bus power is present):

- The **initial energy or volume accumulated value** of each register will be the one set in parameters (zero by default), unless the option to keep the previous value has been parameterised (see section 2.2).

- The value of the **power and flow** objects will be zero until they can be calculated (i.e., until KCI 4 S0 receives the pulses required for calculation).

- The **battery charge level** will start being monitored, so the associated LED indicators and alarm objects will be activated when corresponding.

When a KNX bus power loss takes place:

- The **low-battery and empty-battery LED indicators will remain switched off** to extend the life of the battery. KCI 4 S0 will start again informing about the battery level after the recovery of the bus.

- In the absence of bus power and while there is battery charge enough, KCI 4 S0 **will still count any pulses** arriving from the pulse generator.

  Note: *if the battery results to be discharged, the pulse count cannot be guaranteed during the bus failures. Therefore, after the bus recovery, KCI 4 S0 may not reflect the corresponding energy or volume increments.*

Once the KNX bus power is recovered:

- The **low-battery or empty-battery LED indicator will start blinking** if the battery level is respectively low or very low. In such case, the corresponding communication object will be sent with the value "1". On the contrary, if the battery level is high enough, then the LEDs will stay off and the objects will remain with value "0".

- All **timers** will be reset to zero, and so will be their periods and the spacing between consecutive sendings.

- The energy and volume objects will either recover their previous values or be updated **according to the increments measured during the absence of bus power** – in such case, they will be sent to the bus if the option to send them on a value change is active and if the increment has been greater than the minimum change parameterised.

- The value of the **power and flow** objects will be zero until they can be calculated again (i.e., until KCI 4 S0 receives the pulses required for calculation).

2.4 LOGIC FUNCTIONS

This module makes it possible to perform numeric and binary operations to incoming values received from the KNX bus, and to send the results through other communication objects specifically enabled for this purpose.

KCI 4 S0 can implement **up to 10 different and independent functions**, each of them entirely customisable and consisting in **up to 4 consecutive operations each one**.

The execution of each function can depend on a configurable **condition**, which will be evaluated every time the function is **triggered** through specific, parameterisable communication objects. The result after executing the operations of the function can also be evaluated according to certain **conditions** and afterwards sent (or not) to the KNX bus, which can be done every time the function is executed, periodically or only when the result differs from the last one.

Please refer to the "**Logic Functions**" user manual (available within the KCI 4 S0 product section at the Zennio homepage, www.zennio.com) for detailed information about the functionality and the configuration of the related parameters

ANNEX I: POWER CALCULATION

KCI 4 S0 can dynamically obtain the value of the electric power or the water / gas flow rate by monitoring the pulse reception.
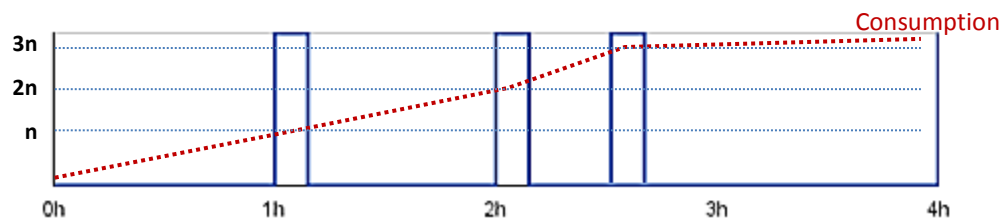
Figure 14: Pulse Reception.

The scenario shown in Figure 5 represents an example. KCI 4 S0 will be receiving discrete pulses at certain instants, as the energy, water or gas consumption increases. Assuming "n" energy (or volume) units per pulse (that is, <u>a pulse frequency of "1/n"</u>) and that the power / flow estimation is required to be re-calculated <u>at the end of every hour</u>, it is possible to proceed as follows:

- Before receiving any pulses, the **accumulated energy** (or volume) is supposed to be zero, assuming that no initial value has been parameterised.

- After receiving the **first pulse**, the accumulated energy (or volume) is becomes "n". Although that value is only true when the pulse arrives, during the second hour "n" will remain as the last known value.

- On the other hand, after the second hour, it is possible to estimate that there has been a **power** (or flow rate) of one pulse per hour (i.e., of "n" units per hour), as only one pulse was received between instants t = 1h and t = 2h.

  - Analogously, at the end of the **third hour**, one possible estimation would be that during the last hour there has been a power of two pulses per hour (2n units/hour), as two pulses arrived between t = 2h and t = 3h.

  - At t = 4h, as no pulses have been received since t = 3h, the **power estimation** could be 0 units/hour. However, in practice this does not mean that there was no energy consumption, but that it was low enough as to not receive pulses for the moment, given the pulse frequency of the meter.

Note that the lower the power calculation intervals (and the higher the pulse frequency), the **higher the accuracy** of these results. In other words, to obtain the instant power value (and not an approximation) it would be necessary to take infinitesimal intervals and to have a continuous (non-discrete) reception of consumption increments.

Said that, KCI 4 S0 offers **two methods** for the power calculation:

- Average value since the last sending:

  This is the method applied in the above example.

  A certain time period is defined so that KCI 4 S0 calculates the power by taking into account the number of **pulses received during such interval**.

  $$\text{Power} = \Delta_{\text{Energy during the interval}} / \Delta_{\text{time of the interval}}$$

  $$\text{Flow} = \Delta_{\text{Volume during the interval}} / \Delta_{\text{time of the interval}}$$

  In this case, as the power is only re-calculated after specific intervals, it only makes sense to send the value to the bus cyclically (subject, if desired, to a minimum change over the last value sent).

- Instantaneous value with the last pulses:

  The result is also an average value although it is the most accurate approach to the actual instant power that KCI 4 S0 can achieve, as in this case the value is re-calculated **whenever a new pulse arrives** by taking into account the time lapsed from the last pulse.

  $$\text{Power} = \Delta_{\text{Energy due to a single pulse}} / \Delta_{\text{time since the last pulse}}$$

  $$\text{Flow} = \Delta_{\text{Volume due to a single pulse}} / \Delta_{\text{time since the last pulse}}$$

  By using this method, it only makes sense to send the result at the moment of calculation (subject, if desired, to a minimum time spacing between sendings and to a minimum increment over the last value sent).

  Under this method, long intervals between pulses (either because of configuring a very low pulse frequency or due to a very limited consumption) can cause the power not to be calculated at all. To force that the value "0" is

actually sent to the bus, a power threshold can be parameterised so that, after the time period corresponding to that power elapses with no pulse reception, the power is assumed to be zero.

Example: *Power Calculation Methods.*

*In Figure 5, assuming that the power has been parameterised to be sent to the bus only cyclically every two hours, the value at instant t = 2h will depend on the calculation method. For "**average**", the estimated power at t = 2h will be "n / 2" (one pulse divided by two hours); on the other hand, for "**instantaneous**", the estimated power at t = 2h will be "n" (one pulse divided by one hour), which had already been calculated at t = 1h.*

ANNEX II. COMMUNICATION OBJECTS

- **"Functional range"** shows the values that, with independence of any other values permitted by the bus according to the object size, may be of any use or have a particular meaning because of the specifications or restrictions from both the KNX standard or the application program itself.

Number	Size	I/O	Flags	Data type (DPT)	Functional Range	Name	Function
1	3 Bytes	I	C T - W U	DPT_Date	01/01/1990 - 31/12/2089	Date	Set Date
2	3 Bytes	I	C T - W U	DPT_TimeOfDay	00:00:00 - 23:59:59	Time of Day	Set Time of Day
3	1 Bit	I	C - - W -	DPT_Trigger	0/1	Global Data Request	0=No Action; 1=Request
4	1 Bit	I	C - - W -	DPT_Enable	0/1	Global Disable	0=Disable; 1=Enable
5	1 Bit	O	C T R - -	DPT_Alarm	0/1	Alarm: Low Battery	0=Normal; 1=Low Battery
6	1 Bit	O	C T R - -	DPT_Alarm	0/1	Alarm: Empty Battery	0=Normal; 1=Empty Battery
7, 12, 20, 28, 75, 80, 88, 96, 143, 148, 156, 164, 211, 216, 224, 232	3 Bytes	O	C T R - -	DPT_Date	01/01/1990 - 31/12/2089	[S0x] [yR] Starting Date	Date when the register was started
8, 13, 21, 29, 76, 81, 89, 97, 144, 149, 157, 165, 212, 217, 225, 233	4 Bytes	O	C T R - -	DPT_ActiveEnergy	0 - 2147483647	[S0x] [yR] Consumption	Electrical Energy (Wh)
	4 Bytes	O	C T R - -	DPT_Value_4_Count	0 - 2147483647	[S0x] [yR] Consumption	Water/Gas Volume (l)
	4 Bytes	O	C T R - -	DPT_ActiveEnergy_kWh	0 - 2147483647	[S0x] [yR] Consumption	Electrical Energy (kWh)
	4 Bytes	O	C T R - -	DPT_Value_Volume	0 - 2147483647	[S0x] [yR] Consumption	Water/Gas Volume (m3)
9, 14, 22, 30, 77, 82, 90, 98, 145, 150, 158, 166, 213, 218,	2 Bytes	O	C T R - -	9.xxx	0 - 670760.96	[S0x] [yR] Cost	Cost corresponding to the consumption

226, 234							
10, 15, 23, 31, 78, 83, 91, 99, 146, 151, 159, 167, 214, 219, 227, 235	2 Bytes	O	**C T R - -**	9.xxx	0 - 670760.96	[S0x] [yR] CO2	CO2 Emissions (KgCO2)
11, 79, 147, 215	4 Bytes	I	**C - - W -**	DPT_ActiveEnergy	0 - 2147483647	[S0x] [yR] Set Consumption	Set Electrical Energy Value (Wh)
	4 Bytes	I	**C - - W -**	DPT_Value_4_Count	0 - 2147483647	[S0x] [yR] Set Consumption	Set Water/Gas Volume Value (l)
	4 Bytes	I	**C - - W -**	DPT_Value_Volume	0 - 2147483647	[S0x] [yR] Set Consumption	Set Water/Gas Volume Value (m3)
	4 Bytes	I	**C - - W -**	DPT_ActiveEnergy_kWh	0 - 2147483647	[S0x] [yR] Set Consumption	Set Electrical Energy Value (kWh)
16, 24, 32, 84, 92, 100, 152, 160, 168, 220, 228, 236	4 Bytes	O	**C T R - -**	DPT_Value_Volume	0 - 2147483647	[S0x] [PRz] Consumption (Previous Period)	Water/Gas Volume (m3)
	4 Bytes	O	**C T R - -**	DPT_Value_4_Count	0 - 2147483647	[S0x] [PRz] Consumption (Previous Period)	Water/Gas Volume (l)
	4 Bytes	O	**C T R - -**	DPT_ActiveEnergy_kWh	0 - 2147483647	[S0x] [PRz] Consumption (Previous Period)	Electrical Energy (kWh)
	4 Bytes	O	**C T R - -**	DPT_ActiveEnergy	0 - 2147483647	[S0x] [PRz] Consumption (Previous Period)	Electrical Energy (Wh)
17, 25, 33, 85, 93, 101, 153, 161, 169, 221, 229, 237	2 Bytes	O	**C T R - -**	9.xxx	0 - 670760.96	[S0x] [PRz] Cost (Previous Period)	Cost corresponding to the consumption
18, 26, 34, 86, 94, 102, 154, 162, 170, 222, 230, -238 -	2 Bytes	O	**C T R - -**	9.xxx	0 - 670760.96	[S0x] [PRz] CO2 (Previous Period)	CO2 Emissions (KgCO2)
19, 27, -35, 87, 95, 103, 155, 163, 171, 223, 231, 239-	1 Bit	I	**C - - W -**	DPT_Reset	0/1	[S0x] [PRz] Reset	0=No Action; 1=Reset
36, 104, 172, 240	4 Bytes	O	**C T - - -**	DPT_Value_Power	0 - 2147483647	[S0x] Power (4 bytes)	Electric power (W)

	4 Bytes	O	**C T - - -**	DPT_Value_Volume_Flux		[S0x] Flow (4 bytes)	Water/Gas Flow (m3/s)
37, 105, 173, 241	2 Bytes	O	**C T - - -**	DPT_Power	0 - 670760 kW	[S0x] Power (2 bytes)	Electric power (kW)
	2 Bytes	O	**C T - - -**	DPT_Value_Volume_Flow	0 - 670760.96	[S0x] Flow (2 bytes)	Water/Gas Flow (l/h)
38, 106, 174, 242	1 Byte	O	**C T R - -**	DPT_Tariff	1 – 4	[S0x] Tariff (Status)	Enumerated Value
39, 107, 175, 243	1 Byte	I	**C - - W -**	DPT_Tariff	1 – 4	[S0x] Set Tariff	Change to Tariff N
40, 108, 176, 244	2 Bytes	I	**C - - W -**	9.xxx	0 - 670760.96	[S0x] Tariff 1	Tariff 1 Value
41, 109, 177, 245	2 Bytes	I	**C - - W -**	9.xxx	0 - 670760.96	[S0x] Tariff 2	Tariff 2 Value
42, 110, 178, 246	2 Bytes	I	**C - - W -**	9.xxx	0 - 670760.96	[S0x] Tariff 3	Tariff 3 Value
43, 111, 179, 247	2 Bytes	I	**C - - W -**	9.xxx	0 - 670760.96	[S0x] Tariff 4	Tariff 4 Value
44, 46, 48, 50, 112, 114, 116, 118, 180, 182, 184, 186, 248, 250, 252, 254	4 Bytes	O	**C T R - -**	DPT_Value_Volume	0 - 2147483647	[S0x] [PR3] Consumption in Tariff T	Water/Gas Volume (m3)
	4 Bytes	O	**C T R - -**	DPT_ActiveEnergy	0 - 2147483647	[S0x] [PR3] Consumption in Tariff T	Electrical Energy (Wh)
	4 Bytes	O	**C T R - -**	DPT_Value_4_Count	0 - 2147483647	[S0x] [PR2] Consumption in Tariff T	Water/Gas Volume (l)
	4 Bytes	O	**C T R - -**	DPT_ActiveEnergy_kWh	0 - 2147483647	[S0x] [PR2] Consumption in Tariff T	Electrical Energy (kWh)
	4 Bytes	O	**C T R - -**	DPT_ActiveEnergy	0 - 2147483647	[S0x] [PR2] Consumption in Tariff T	Electrical Energy (Wh)
	4 Bytes	O	**C T R - -**	DPT_Value_Volume	0 - 2147483647	[S0x] [PR2] Consumption in Tariff T	Water/Gas Volume (m3)
	4 Bytes	O	**C T R - -**	DPT_Value_4_Count	0 - 2147483647	[S0x] [PR3] Consumption in Tariff T	Water/Gas Volume (l)
	4 Bytes	O	**C T R - -**	DPT_ActiveEnergy_kWh	0 - 2147483647	[S0x] [PR3] Consumption in Tariff T	Electrical Energy (kWh)
	4 Bytes	O	**C T R - -**	DPT_ActiveEnergy	0 - 2147483647	[S0x] [PR1] Consumption in Tariff T	Electrical Energy (Wh)
	4 Bytes	O	**C T R - -**	DPT_ActiveEnergy_kWh	0 - 2147483647	[S0x] [PR1] Consumption in Tariff T	Electrical Energy (kWh)
	4 Bytes	O	**C T R - -**	DPT_ActiveEnergy	0 - 2147483647	[S0x] [TR] Consumption in Tariff T	Electrical Energy (Wh)
	4 Bytes	O	**C T R - -**	DPT_ActiveEnergy_kWh	0 - 2147483647	[S0x] [TR] Consumption in Tariff T	Electrical Energy (kWh)

	4 Bytes	O	C T R - -	DPT_Value_4_Count	0 – 2147483647	[S0x] [TR] Consumption in Tariff T	Water/Gas Volume (l)
	4 Bytes	O	C T R - -	DPT_Value_Volume	0 – 2147483647	[S0x] [PR1] Consumption in Tariff T	Water/Gas Volume (m3)
	4 Bytes	O	C T R - -	DPT_Value_4_Count	0 – 2147483647	[S0x] [PR1] Consumption in Tariff T	Water/Gas Volume (l)
	4 Bytes	O	C T R - -	DPT_Value_Volume	0 - 2147483647	[S0x] [TR] Consumption in Tariff T	Water/Gas Volume (m3)
45, 47, 49, 51, 113, 115, 117, 119, 181, 183, 185, 187, 249, 257, 253, 255	2 Bytes	O	C T R - -	9.xxx	0 - 670760.96	[S0x] [PR3] Cost in Tariff T	Cost corresponding to the consumption
	2 Bytes	O	C T R - -	9.xxx	0 - 670760.96	[S0x] [PR2] Cost in Tariff T	Cost corresponding to the consumption
	2 Bytes	O	C T R - -	9.xxx	0 - 670760.96	[S0x] [TR] Cost in Tariff T	Cost corresponding to the consumption
	2 Bytes	O	C T R - -	9.xxx	0 - 670760.96	[S0x] [PR1] Cost in Tariff T	Cost corresponding to the consumption
52, 54, 56, 58, 120, 122, 124, 126, 188, 190, 192, 194, 256, 258, 260, 262	4 Bytes	O	C T R - -	DPT_Value_Volume	0 - 2147483647	[S0x] [PR3] Consumption in Tariff T	Water/Gas Volume (m3)
	4 Bytes	O	C T R - -	DPT_Value_4_Count	0 - 2147483647	[S0x] [PR1] Consumption in Tariff T	Water/Gas Volume (l)
	4 Bytes	O	C T R - -	DPT_ActiveEnergy	0 - 2147483647	[S0x] [PR2] Consumption in Tariff T	Electrical Energy (Wh)
	4 Bytes	O	C T R - -	DPT_Value_4_Count	0 - 2147483647	[S0x] [PR2] Consumption in Tariff T	Water/Gas Volume (l)
	4 Bytes	O	C T R - -	DPT_ActiveEnergy_kWh	0 - 2147483647	[S0x] [PR2] Consumption in Tariff T	Electrical Energy (kWh)
	4 Bytes	O	C T R - -	DPT_ActiveEnergy_kWh	0 - 2147483647	[S0x] [PR1] Consumption in Tariff T	Electrical Energy (kWh)
	4 Bytes	O	C T R - -	DPT_ActiveEnergy	0 - 2147483647	[S0x] [PR1] Consumption in Tariff T	Electrical Energy (Wh)
	4 Bytes	O	C T R - -	DPT_Value_Volume	0 - 2147483647	[S0x] [PR2] Consumption in Tariff T	Water/Gas Volume (m3)
	4 Bytes	O	C T R - -	DPT_Value_4_Count	0 - 2147483647	[S0x] [PR3] Consumption in Tariff T	Water/Gas Volume (l)
	4 Bytes	O	C T R - -	DPT_ActiveEnergy_kWh	0 - 2147483647	[S0x] [PR3] Consumption in Tariff T	Electrical Energy (kWh)
	4 Bytes	O	C T R - -	DPT_ActiveEnergy	0 - 2147483647	[S0x] [PR3] Consumption in Tariff T	Electrical Energy (Wh)
	4 Bytes	O	C T R - -	DPT_Value_Volume	0 - 2147483647	[S0x] [PR1] Consumption in Tariff T	Water/Gas Volume (m3)
53, 55, 57, 59, 121, 123,	2 Bytes	O	C T R - -	9.xxx	0 - 670760.96	[S0x] [PR3] Cost in Tariff T	Cost corresponding to the consumption

125, 127, 189, 191, 193, 195, 257, 259, 261, 263	2 Bytes	O	**C T R - -**	9.xxx	0 - 670760.96	[S0x] [PR2] Cost in Tariff T	Cost corresponding to the consumption
	2 Bytes	O	**C T R - -**	9.xxx	0 - 670760.96	[S0x] [PR1] Cost in Tariff T	Cost corresponding to the consumption
60, 128, 196, 264	1 Bit	O	**C T R - -**	DPT_Trigger	0/1	[S0x] Notification 1	Sending of 0
	1 Bit	O	**C T R - -**	DPT_Trigger	0/1	[S0x] Notification 1	Sending of 1
61, 129, 197, 265	1 Bit	O	**C T R - -**	DPT_Trigger	0/1	[S0x] Notification 2	Sending of 0
	1 Bit	O	**C T R - -**	DPT_Trigger	0/1	[S0x] Notification 2	Sending of 1
62, 129, 197, 266	1 Bit	O	**C T R - -**	DPT_Trigger	0/1	[S0x] Notification 3	Sending of 0
	1 Bit	O	**C T R - -**	DPT_Trigger	0/1	[S0x] Notification 3	Sending of 1
63, 130, 198, 267	1 Bit	O	**C T R - -**	DPT_Trigger	0/1	[S0x] Notification 4	Sending of 0
	1 Bit	O	**C T R - -**	DPT_Trigger	0/1	[S0x] Notification 4	Sending of 1
64, 131, 199, 268	1 Bit	O	**C T R - -**	DPT_Trigger	0/1	[S0x] Notification 5	Sending of 0
	1 Bit	O	**C T R - -**	DPT_Trigger	0/1	[S0x] Notification 5	Sending of 1
65, 132, 200, 269	1 Bit	O	**C T R - -**	DPT_Trigger	0/1	[S0x] Notification 6	Sending of 0
	1 Bit	O	**C T R - -**	DPT_Trigger	0/1	[S0x] Notification 6	Sending of 1
66, 133, 201, 270	1 Bit	O	**C T R - -**	DPT_Trigger	0/1	[S0x] Notification 7	Sending of 0
	1 Bit	O	**C T R - -**	DPT_Trigger	0/1	[S0x] Notification 7	Sending of 1
67, 134, 202, 271	1 Bit	O	**C T R - -**	DPT_Trigger	0/1	[S0x] Notification 8	Sending of 0
	1 Bit	O	**C T R - -**	DPT_Trigger	0/1	[S0x] Notification 8	Sending of 1
68, 135, 203, 272	1 Bit	O	**C T R - -**	DPT_Alarm	0/1	[S0x] Alarm: Overconsumption	0=Alarm; 1=No Alarm
	1 Bit	O	**C T R - -**	DPT_Alarm	0/1	[S0x] Alarm: Overconsumption	1=Alarm; 0=No Alarm
69, 136, 204, 273	2 Bytes	I	**C - - W -**	DPT_Power	0 - 670760.96	[S0x] Alarm: Overconsumption Limit	Overconsumption Limit Value (kW)
	2 Bytes	I	**C - - W -**	DPT_Value_Volume_Flow	0 - 670760.96	[S0x] Alarm: Overconsumption Limit	Overconsumption Limit Value (l/h)
70, 137, 205, 274	1 Bit	O	**C T R - -**	DPT_Alarm	0/1	[S0x] Alarm: Low Consumption	1=Alarm; 0=No Alarm
	1 Bit	O	**C T R - -**	DPT_Alarm	0/1	[S0x] Alarm: Low Consumption	0=Alarm; 1=No Alarm

71, 138, 206, 275	2 Bytes	I	**C - - W -**	DPT_Power	0 - 670760.96	[S0x] Alarm: Low Consumption Limit	Low Consumption Limit Value (kW)
	2 Bytes	I	**C - - W -**	DPT_Value_Volume_Flow	0 - 670760.96	[S0x] Alarm: Low Consumption Limit	Low Consumption Limit Value (l/h)
72, 139, 207, 276	1 Bit	O	**C T R - -**	DPT_Alarm	0/1	[S0x] Alarm: Pulse Receiving - Error	0=Alarm; 1=No Alarm
	1 Bit	O	**C T R - -**	DPT_Alarm	0/1	[S0x] Alarm: Pulse Receiving - Error	1=Alarm; 0=No Alarm
73, 140, 208, 277	1 Bit	I	**C - - W -**	DPT_Trigger	0/1	[S0x] Data Request	0=No Action; 1=Request
74, 141, 209, 278	1 Bit	I	**C - - W -**	DPT_Enable	0/1	[S0x] Disable	0=Disable; 1=Enable
279 - 310	1 Bit	I	**C - - W -**	DPT_Bool	0/1	[LF] (1 bit) Data Entry n	Binary Data Entry (0/1)
311 - 326	1 Byte	I	**C - - W -**	DPT_Value_1_Ucount	0 - 255	[LF] (1 byte) Data Entry m	1 byte Data Entry (0-255)
327 - 342	2 Bytes	I	**C - - W -**	DPST-7-1 DPST-8-1 DPST-9-1	0 - 65535 -32768 − 32767 -273.00 − 670760.00	[LF] (2 bytes) Data Entry m	2 bytes Data Entry
343 - 350	4 Bytes	I	**C - - W -**	DPT_Value_4_Count	0 - 2147483647	[LF] (4 bytes) DataEntry r	4 bytes Data Entry
351 - 360	1 Bit	S	**C T R - -**	DPT_Bool	0/1	[LF] Function f - Result	(1 bit) Boolean
	1 Byte	S	**C T R - -**	DPT_Value_1_Ucount	0 - 255	[LF] Function f - Result	(1 byte) Unsigned
	2 Bytes	S	**C T R - -**	DPT_Value_2_Ucount	0 - 65535	[LF] Function f - Result	(2 bytes) Unsigned
	4 Bytes	S	**C T R - -**	DPT_Value_4_Count	0 - 2147483647	[LF] Function f - Result	(4 bytes) Signed
	1 Byte	S	**C T R - -**	DPT_Scaling	0% - 100%	[LF] Function f - Result	(1 byte) Percentage
	2 Bytes	S	**C T R - -**	DPT_Value_2_Count	-32768 - 32767	[LF] Function f - Result	(2 bytes) Signed
	2 Bytes	S	**C T R - -**	DPT_Value_Temp	-273.00 - 670760.00	[LF] Function f - Result	(2 bytes) Float

Join and send us your inquiries
about Zennio devices:

http://support.zennio.com