



# KNXnet/IP

*RTI XP Series Controllers*

*KNXnet/IP Driver*

## Driver Overview

Updated September 18, 2013

The information and ideas contained in this document are the property of Active Surroundings and are submitted in confidence solely for the consideration of the recipient. By retention of this document, the recipient agrees to maintain this document in confidence and not to duplicate, modify, distribute, publish, use or disclose the whole or any part of this document, or any of the information or ideas contained herein, for any purpose other than to evaluate this document, without prior written authorization from Active Surroundings. [info@activesurroundings.com](mailto:info@activesurroundings.com)

## Index

1	Design objectives .....	2
1.1	Overview .....	2
1.2	Demo Mode .....	2
1.3	IP Gateway Support.....	2
1.4	BAOS Gateways.....	2
2	Connectivity and Communication.....	2
3	KNX Telegrams.....	3
3.1	Telegram format.....	3
3.2	Data types .....	3
3.2.1	Current Supported Data Types .....	3
4	RTI Interface .....	4
4.1	Property Page (Configuration) .....	4
4.2	Properties.....	4
4.2.1	Debug Settings .....	4
4.2.2	KNX Communications .....	5
4.2.3	Export Communications (UDP) .....	5
4.2.4	Export Communications (TCP) .....	5
4.2.5	KNX Settings.....	5
4.2.6	Lighting Groups .....	6
4.2.7	Shutter Groups.....	7
4.2.8	Incremental Groups.....	7
4.2.9	Telegram Macro Triggers.....	7
4.2.10	Date Time Groups.....	8
4.2.11	Date Time Broadcast .....	8
4.2.12	Advanced Settings .....	8
4.2.13	Keys .....	9
4.3	System Variables .....	9
4.3.1	Debug.....	9
4.3.2	Lighting Names.....	9
4.3.3	Shutter Name .....	10
4.3.4	Lighting Values .....	10
4.3.5	Incremental Group Values .....	10
4.3.6	Date Time Values.....	10
4.4	Methods.....	10
4.4.1	General Commands .....	10
4.4.2	Lighting Commands .....	10
4.4.3	Shutter Commands.....	11
4.4.4	Incremental Commands .....	11

## 1 Design objectives

### 1.1 Overview

The KNXnet/IP for RTI driver shall be compatible with any IP gateway device that conforms to the KNX tunnelling specification. There shall be no requirement for any cross-consideration between the KNX programming and the KNXnet/IP for RTI driver (as with the BAOS based gateway with data point definition requirements). The above criteria are in order to reduce project time, permit retro implementation and to allow the RTI integrator to standardize the project to KNX design format, both in terms of GUI implementation and the programmatic interface.

### 1.2 Demo Mode

The driver will operate in a fully functional demo mode for 1 hour should no valid licence key be entered into the License Key property.

### 1.3 IP Gateway Support

The KNXnet/IP for RTI driver shall support any KNXnet/IP gateway supporting and conforming to the UDP KNXnet/IP tunnelling specification. Generally this means any KNXnet/IP for RTI driver capable of being connected via the popular ETS KNX programming environment.

### 1.4 BAOS Gateways

The KNXnet/IP for RTI driver has been tested against BAOS type gateways 770, 771 and 772 BAOS type gateways.

## 2 Connectivity and Communication

KNXnet/IP is a UDP framed protocol, whereby a client device may connect via a gateway device and be able to send and receive telegrams to and from the KNX bus. The client device requests a connection from the gateway and receives an identifier which may then be included in all subsequent transactions. All data transmitted and received must follow a strict protocol of timing and sequencing. It is the responsibility of both client and server (KNXnet/IP for RTI driver and KNX IP gateway) to fulfil their respective obligations in maintaining the protocol flow.

As per the KNX tunnelling standard, the KNXnet/IP for RTI driver maintains a heartbeat sequence of connection state confirmations with the gateway about every 60 seconds. Therefore the KNXnet/IP for RTI driver manages the connection and reconnects should the need arise for whatever reason.

## 3 KNX Telegrams

The KNXnet/IP for RTI driver uses the 3 level format for KNX telegrams. The KNX telegram list should be provided by the programmer of the KNX system.

### 3.1 Telegram format

The structure of the 3 level telegram is as follows:

Main Group/Mid Group/Sub Group = value

Example:

0/0/1 = 1

It appears that there is no absolute standard for the use of KNX address usage. The experience of the author suggests that different companies use differing standards in their programmatic approach, this was therefore taken into account when designing the properties interfacing with the RTI “Integration Designer” design environment.

The author further recommends the references found at <http://knx.org/> for further information regarding KNX technology but which is considered out of scope for this document.

### 3.2 Data types

The KNXnet/IP for RTI driver currently supports a subset of the KNX data type range, these being the common types required to implement the more common operations required with RTI implementations.

#### 3.2.1 Current Supported Data Types

The current supported data types are;

DPT\_1BIT (KNX 1.\*\*) (typically used for switching)

DPT\_4BIT (KNX 3.\*\*) (typically used for dim up/down)

DPT\_8BIT (KNX 6.001) (signed counter)

DPT\_1BYTE (KNX 5.004) (typically used for brightness values/device values)

DPT\_2BYTE (KNX 9.001) (typically used for HVAC set points ETC)

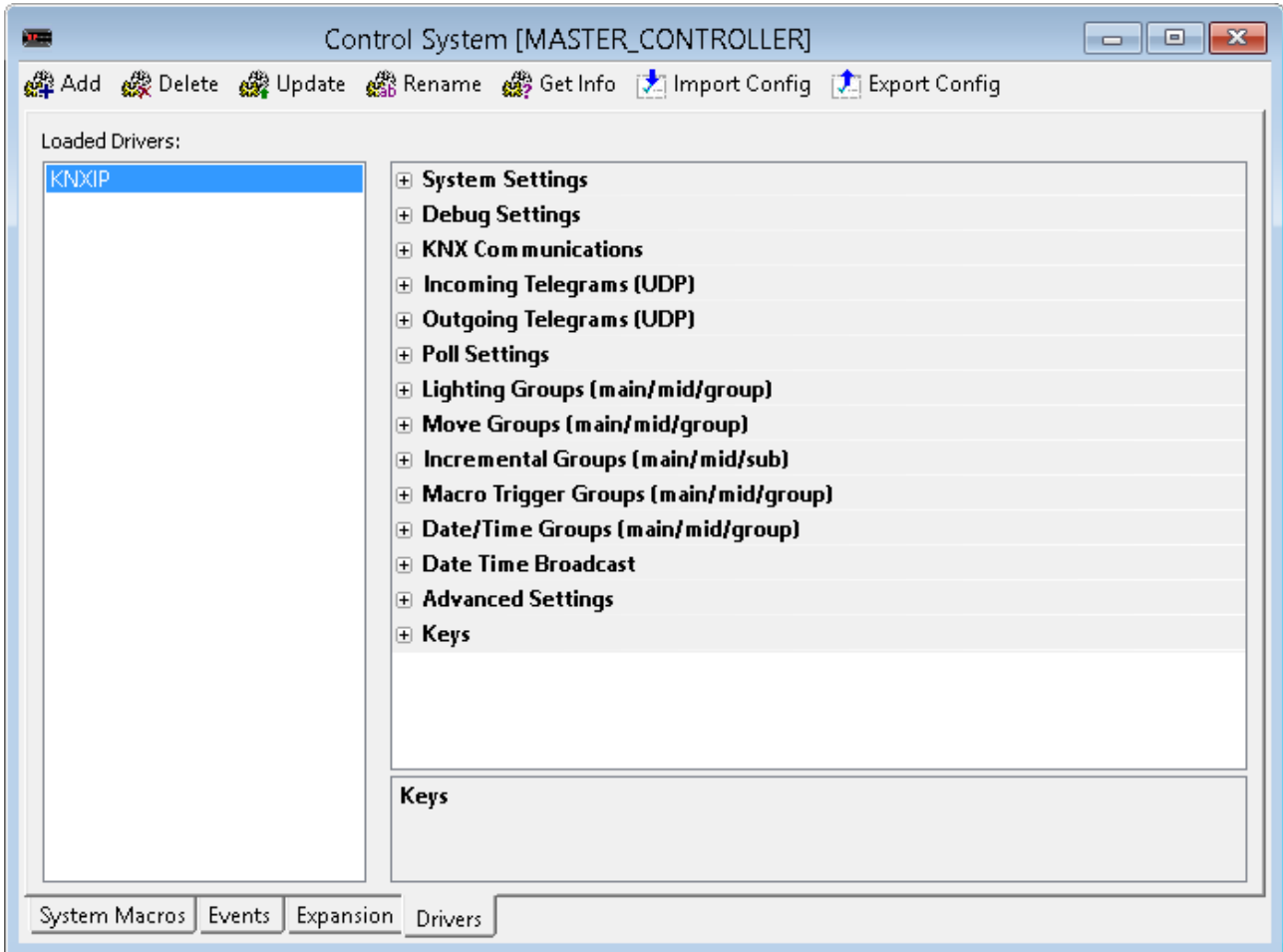
DPT\_3BYTE (KNX 10.001) (Time)

DPT\_11BYTE (KNX 11.001) (Date)

These types are appropriate for most common command and control requirements.

## 4 RTI Interface

### 4.1 Property Page (Configuration)



### 4.2 Properties

#### 4.2.1 Debug Settings

*Enable Trace:* When checked, the KNXnet/IP for RTI driver trace will indicate the group address being sent and received, along with any relevant information.

#### 4.2.2 KNX Communications

The properties below set the required network information for gateway connection.

*Gateway IP Address:* IPV4 address of the KNX IP Gateway

*Gateway UDP Port:* 3671 is the default KNX tunnelling port. The property is provided as the port may change in devices in some manufacturers.

#### 4.2.3 Export Communications (UDP)

When selected, the KNXnet/IP for RTI driver sends KNX telegrams received from the KNX bus to the specified UDP connection. Valid KNX telegrams received via the connection are despatched to the KNX bus.

The properties below set the required network information for gateway connection.

*Enable UPD Export:* When checked, the connection will be opened and maintained by the KNXnet/IP for RTI driver, with all telegrams being sent via the connection.

*Export IP Address:* IPV4 endpoint address (e.g. 192.168.1.100)

*Export IP Port:* Remote port number opened at remote address.

#### 4.2.4 Export Communications (TCP)

When selected, the KNXnet/IP for RTI driver sends KNX telegrams received from the KNX bus to the specified TCP connection. Valid KNX telegrams received via the connection are despatched to the TCP bus.

When selected, the KNXnet/IP for RTI driver sends KNX telegrams received from the KNX bus to the specified UDP connection. Valid KNX telegrams received via the connection are despatched to the KNX bus.

The properties below set the network information required for a gateway connection.

*Enable UPD Export:* If checked, the connection will be opened and maintained by the KNXnet/IP for RTI driver, with all telegrams being sent via the connection.

*Export IP Address:* IPV4 endpoint address (e.g. 192.168.1.101)

*Export IP Port:* Remote port number opened at remote address.

#### 4.2.5 KNX Settings

This section contains various KNX implementations and KNXnet/IP for RTI driver settings.

*Enable Background Polling:* If checked, all specified item group addresses will be background read according to the criteria below:

*Poll On Startup:* If checked, all specified item group addresses will be read only when a new connection between the KNXnet/IP for RTI driver and KNX IP gateway is made. This property is typically set when the RTI controller is initialized for the status reading of lights, etc.

*Polling Rate:* Numeric value expressed in milliseconds, denotes the polling request frequency used to dispatch the reading telegrams. Valid values are 2000 – 10000 ms (such telegrams are usually regarded as low priority and care should be taken not to load a possibly already busy KNX bus).

*Poll Switch Activation Status:* All declared switch statuses will be read.

*Poll Brightness Value:* All declared brightness values will be read.

*Poll Incremental Group Status:* All Incremental groups will be read.

## 4.2.6 Lighting Groups

### GUI Integration (Lighting)

This section contains the group address definitions for lighting implementation of KNX groups. Furthermore, this section exists so as to take advantage of the RTI slider, button and switch objects. With this in mind it has been designed to **minimize implementation time** for the RTI integrator.



This demo implementation illustrates how an RTI slider object, a two-state button and an RTI switch object are used.

*Slider object:* On|Off, position sets brightness value.

*Button Object:* On|Off, Holding initiates dim up|down, while releasing stops dim.

*Switch Object:* On|Off only.

All objects receive feedback from the driver variable associated with the specified group address.

Integrators are free to use custom buttons, objects or other driver methods to achieve their lighting control objectives.

*Address Label:* Free text value corresponding to driver variable. This property is used for:

1. Visual ordering and record keeping of KNX telegrams.
2. Setting the text property of associated button objects or labels.

*Address Switch (Set):* Group address corresponding to the KNX activation.

*Address Switch (Status):* Group address corresponding to the KNX .

*Address Dim:* Group address corresponding to the KNX start/stop dim.

*Address Brightness Value (Set):* Group address corresponding to the KNX set brightness value.

*Address Brightness Value (Status) [optional]*: Group address corresponding to the KNX status of brightness value if different from the brightness value group. May be left empty, in which case the group specified in *Address Brightness Value (Set)* will be used for status.

#### **4.2.7 Shutter Groups**

*Address Move*: Group address for up/down (KNX move)

*Address Stop*: Group address for KNX stop

#### **4.2.8 Incremental Groups**

These groups allow the integrator to control various KNX devices that require step up/down values to be sent, examples of this include HVAC set points and fan speeds etc.

*Address Name*: Label

*Address Activation*: Group Address for activation.

*Address Status (optional)*: Group Address for status if different from above.

*Data Type*: Drop down list with options: 8 Bit(6.\* signed), 1 Byte(5.\* unsigned), 2 Byte(9.001 signed)

#### **4.2.9 Telegram Macro Triggers**

This section defines the group addresses whose values should be evaluated against the specified conditional expression. Should the value (v) match the conditions of the expression, the specified macro will be called.

*Macro Trigger Address*: Group address whose current value will be evaluated.

*Macro Trigger Value*: Allows the definition of condition under which when met will cause the triggering of a system macro.

Example:

The below is an example where v represents the value received from the KNX bus.

```
v > 100 && v < 200 || v == 10
```

So the above expression defines that when the value of v is greater than 100 AND v is less than 200 OR the value equals 10, then call the macro.

*Macro Index*: Macro to call.



#### 4.2.10 Date Time Groups

Dates and times are 2 byte data types in KNX. The 2 bytes of data do not specify its type (date or time). This section defines how specific group addresses will interpret a 2 byte telegram when it is received.

*Name:* Label

*Group:* Group address to be watched.

*DPT:* Drop down with Time and Date options, select the option the data should be translated into.

#### 4.2.11 Date Time Broadcast

It is possible to define 2 groups addresses to which the current controller time and date can be sent to the KNX bus.

*Group Address (Time):* The group address to which the KNX telegram containing the time (KNX 10.001) will be sent.

*Broadcast Schedule (Time):* Options are “Disabled” – no telegrams will be sent. “On Startup Only” – the time telegram will be sent only when the KNX connection is initialized. “Every Minute” / “Every Hour” the time telegram will be sent periodically.

*Group Address (Date):* The group address to which the KNX telegram containing the date (KNX 11.001) will be sent.

*Broadcast Schedule (Date):* Options are “Disabled” – no telegrams will be sent. “On Startup Only” – the date telegram will be sent only when the KNX connection is initialized. “Every Minute” / “Every Hour” the date telegram will be sent periodically.

#### 4.2.12 Advanced Settings

*DPT Localization:* Changes the language of display data for groups outputting textual elements in the data, such as Day (“Monday”, “Montag” etc)

*Dimming Scale:* Drop down menu specifying if a 0-255 or 0-100 dim range should be used.

*Anti Slider Skip:* Checking this, enables a smoothing algorithm that reduces jumping or skipping of the slider which might otherwise occur under heavy bus loads.

*Default Data Types (Undefined 3 Byte):* Drop down with “Date” and “Time” options that define how data from a group **not defined in the Date Time Groups section** will have the data translated into a time or a date.

*Truncate 2 Byte Values:* Allows 2 byte floating point values to be truncated when received from the bus. For example, a value of 23.24 degrees centigrade received from the bus may be truncated on reception and displayed as 23.2 (option “1 Decimal Place”) or just 23 (option “Integer Value”)

### 4.2.13 Keys

*Support Keys:* Reserved for future use

*Licence Keys:* This is where you paste your purchased licence key

## 4.3 System Variables

### 4.3.1 Debug

The debug variables may be useful for the integrator for diagnostic purposes.



*ERR LED:* (Boolean) Value is true when the connection state is in an error condition.

*TX LED:* (Boolean) Value is true when a KNX frame is transmitted to the KNX gateway.

*RX LED:* (Boolean) Value is true when a KNX frame is received from the KNX gateway.

*LINK LED:* (Boolean) Value is true when a connection is being maintained between the KNXnet/IP driver and the KNX gateway.

### 4.3.2 Lighting Names

The number of Lighting groups depends on the version of the driver.

*Light Name:* (String) Value is set to the configuration *Address Label* setting in the configuration section.

### 4.3.3 Shutter Name

The number of Lighting groups depends on the version of the driver.

*Shutter Name:* (String)

### 4.3.4 Lighting Values

*Address Status:* (Boolean) On|Off = true|false.

*Address Brightness Value:* (Integer) Absolute brightness values expressed in accordance with the KNX object specification.

### 4.3.5 Incremental Group Values

*Address Value:* (String) Current value, expressed as a string so as to accept multiple KNX types.

### 4.3.6 Date Time Values

*Date Time Value:* Formatted value of the time or date last received from the corresponding group.

## 4.4 Methods

### 4.4.1 General Commands

*Send Telegram (Group Address, Command, Value, DataType)*

Parameters:

*Group Address:* 3 level KNX group address e.g. 01/1/001

*Command:* "Read" or "Write" value choice to specify whether the command is a KNX read object or write to object.

*Value:* String value containing any valid KNX data type expression.

*DataType:* "1 Bit", "4 Bit", "1 Byte", "2 Byte", "3 Byte" choice reflecting the appropriate data type.

### 4.4.2 Lighting Commands

*Set Level (Index, Level)*

Parameters;

*Address Index:* Numeric value representing the index of the lighting group.

*Level:* Numeric Value representing the absolute brightness value. This may be expressed either with 0-255 (off-full brightness) or as a percentage value 0-100 (off-full brightness) depending on the KNX implementation.

### 4.4.3 Shutter Commands

#### *Activate Shutter*

*Shutter Index*: Numeric value representing the index of the shutter group.

*Operation*: “Up (Open)”, “Down (Close)”, “Stop” choice.

### 4.4.4 Incremental Commands

These commands allow the integrator to control various KNX devices that require step up/down values to be sent, examples of this include HVAC set points and fan speeds etc. The value may be increased or decreased by a specified step within specified boundaries (optional).

#### *Increment Value (Address Index, Step, Limit, DataType)*

##### Parameters

*Address Index*: Numeric value representing the index of the incremental group.

*Step*: Amount to increment by.

*Limit*: Absolute allowable value limit.

*DataType*: “1 Byte”, choice reflecting the appropriate data type. Currently only 1 byte types are supported.

#### *Decrement Value (Address Index, Step, Limit, DataType)*

##### Parameters

*Address Index*: Numeric value representing the index of the incremental group.

*Step*: Amount to increment by.

*Limit*: Absolute allowable value limit.

*DataType*: “1 Byte”, choice reflecting the appropriate data type. Currently only 1 byte types are supported.