



# iRidium for KNX/EIB

## *Fast Start: Connection Setting Manual for KNX/EIB equipment*

### *Review of iRidium Software Package for KNX/EIB:*

iRidium turns your iPhone/iPod/iPad or Windows device into a KNX system control panel. With the help of iRidium GUI Editor you can create interfaces with animation and magnificent visual effects.

iRidium can import channels and commands from ETS projects. Then you can quickly bind them to control items (buttons, lists, levels, etc.) by the Drag&Drop method. "iRidium for KNX" includes all the features of iRidium for AV CONTROL, that can help you to set up communication with your AV equipment.

iRidium software package for KNX includes four components:

***iRidium GUI Editor*** – a Graphic User Interface Editor, which allows you to create your own interfaces and set connection to the KNX/EIB bus.

***iRidium Transfer*** – an application for converting and uploading graphic interfaces on Apple iOS and Windows based control panels. Besides iRidium Transfer is responsible for creation of iRidium Client for Windows.

***iRidium Client*** for Windows – a component for launching and operating the application on your PC.

***iRidium Gate*** – an application, which allows you to establish simultaneous communication of several control devices (iPhone/iPod/iPad or Windows devices) with bus devices through one KNX IP-router. iRidium Gate provides connection stability between iPhone/iPod/iPad and a KNX system (after loosing connection KNX IP-router cannot reconnect for 1-2 minutes without iRidium Gate). It also enables you to reduce the cost of your KNX system as additional KNX IP-routers are not required. iRidium Gate can be installed on any PC under Windows XP / 7.

The part of the software package for installation on iOS devices is the ***iRidium for KNX*** Client application which is in charge of launching and running GUIs on target iOS devices.

In order to enable the connection of iOS "iRidium for KNX" Client to KNX it is required to purchase and activate an iRidium license Key. License files for iOS received after the Key activation are stored in the Transfer application and are transferred to target iOS devices along with GUIs. Licenses for Windows are stored in the project folder (a received license file is copied into the folder with the Client generated for Windows).

iRidium supports operation with the following devices:

- Apple iOS devices (iPad, iPhone, iPod Touch)
- Windows XP & Windows 7 based devices

***Recommendations and Requirements for Operation with iRidium Software Package***

- PC under Microsoft Windows XP / 7
- “iRidium for KNX” application installed on your iOS device from the AppStore
- Wi-Fi network for uploading and updating GUIs on iOS devices
- Internet connection for purchasing of iRidium Keys and activation of licenses
- iRidium License file for enabling the connection to the KNX/EIB bus

## Content

Installation of iRidium Software Package Components .....	4
Creating a GUI in iRidium GUI Editor .....	5
Import of KNX/EIB Channels from ETS projects .....	6
Setting of connection to KNX/EIB .....	8
<i>Connection to KNX/EIB IP-router through UDP</i> .....	8
Setting of Data Reading in ETS .....	9
Setting of Channels for Communication with the KNX/EIB bus .....	10
<i>Principles of Controlling Dimmers in the KNX/EIB Bus</i> .....	11
Dimming by Level (slider).....	11
Dimming by Increase/Decrease Channels .....	14
Dimming by Up/Down Button.....	16
<i>Setting a Fixed Value</i> .....	18
Button: Setting a Fixed Value for Dimmers and Relays .....	18
Relay Trigger Switch by iRidium .....	18
Relay Trigger Switch by KNX/EIB .....	19
<i>Reading Data about a KNX/EIB Variable State</i> .....	20
Output of temperature or channel variable values .....	20
Setting of Data Reading in ETS .....	21
Connection to the Bus over the Internet: iRidium Gate.....	22
Uploading a Project on a Panel, Connection to Equipment .....	27
<i>The Emulation Mode</i> .....	27
<i>Conversion of the Project for Its Launching on Target Devices</i> .....	28
<i>Uploading the Project on an iOS Device</i> .....	29
<i>Description of the Licensing Procedure for Control Devices</i> .....	30
Receipt of UDID for Activation of iRidium Base License for an iOS Device.....	30
Licensing Procedure for an iOS Device.....	32
Activation of the iRidium Base License Key for an iOS Device .....	34
Uploading of the License File for iOS to iRidium Transfer .....	36
<i>Setting up of iOS “iRidium for KNX” Client</i> .....	37
<i>Creation of iRidium Windows-Client</i> .....	38
Receipt of HWID for Licensing of iRidium Windows-Client.....	39

## Installation of iRidium Software Package Components

---

In order to start your work, it is necessary to install two parts of the iRidium software package. The first part, an installation package for your personal computer under Windows XP / 7, can be downloaded from the iRidium web-site, in the **Download** section:

**Link:** <http://iridiummobile.net/download>



iRidium mobile offers a convenient software complex for controlling audio/video equipment, and home automation from any devices

### iRidium Downloads

Applications

The iRidium software package for installation on PC includes several components. To operate KNX/EIB equipment you require:

**iRidium GUI Editor** – a Graphic User Interface Editor, which allows you to create your own interfaces for control panels.

**iRidium Transfer** – an application for uploading graphic interfaces created in iRidium GUI Editor on iOS devices. iRidium Transfer allows you to convert interfaces into the format for their launching on Windows XP/7 devices. It can also check iRidium license availability for iOS devices.

**iRidium Client** - a part of the iRidium software package which launches a ready graphic interface on the iPod/iPhone/iPad or Windows device. It is responsible for interaction between GUIs and equipment. In order to correctly run iRidium Client you need to obtain a license. You can download iRidium Client for Windows from our web site.

**iRidium Gate** – an application for communication with devices, which do not support simultaneous connection of several Clients (the majority of standard KNX/EIB IP-routers). iRidium Gate enables the communication of KNX/EIB equipment with several Clients. It converts TCP to the protocol which is supported by target equipment.

The second part of the iRidium software package is the Client application for its installation on panels based on the Apple iOS operation system. The application is called **iRidium for KNX** and can be downloaded from the official *AppStore*.

**iTunes Link:** [http://itunes.apple.com/us/app/iRidium for KNX/id369102319?mt=8](http://itunes.apple.com/us/app/iRidium%20for%20KNX/id369102319?mt=8)

The “iRidium for KNX” Client application is a software component of iRidium that is in charge of launching and using graphic interfaces, forming and sending commands to the controlled equipment.

## iRidium for KNX

By iRidium mobile Ltd.



### Description

iRidium for KNX is an Application that turns your iDevice into the most convenient and reliable touch panel for your control system. It is integrated with KNX or Weinzierl BAOS(770,771,772) IP interfaces and enables you to control your Audio/Video equipment. iRidium for KNX features, graphic capabilities, speed and stability will exceed your expectations.

For more details, please visit our website: <http://iridiummobile.net/knx>

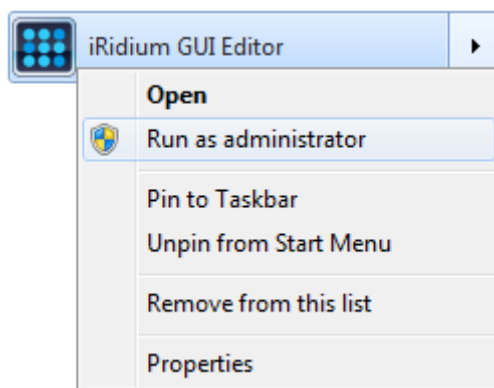
[iRidium mobile Ltd. Web Site](#) ▶ [iRidium for KNX Support](#) ▶

## Creating a GUI in iRidium GUI Editor

After the iRidium package for your PC and the “iRidium for KNX” Client are installed, you can start creating a graphic interface to control the KNX/EIB equipment. Start iRidium GUI Editor on your PC.

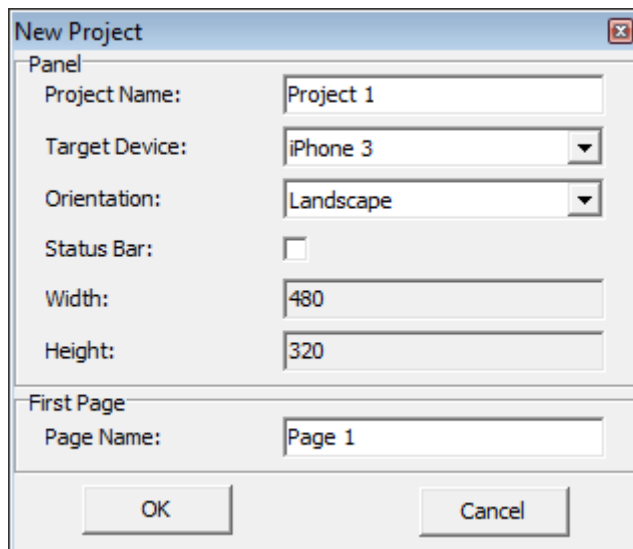


1. Do not forget that if you run Windows 7, launch of iRidium software components should be performed by Administrator rights – through the right-click menu or by using Administrator account.
2. Do not forget to close ETS before starting work with iRidium, as running ETS can cause the noninteraction between iRidium Client and the bus (if it supports only 1 connection),



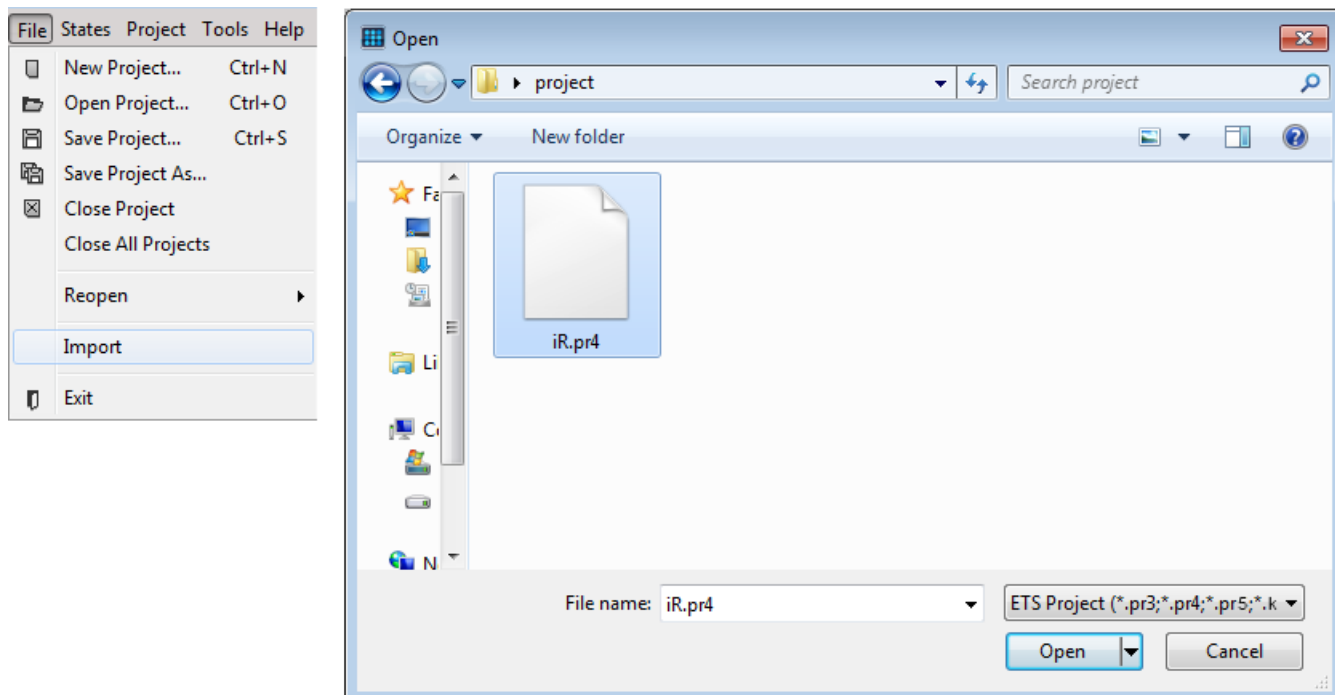
When launching iRidium GUI Editor you can see several workspaces which are responsible for different aspects of GUI creating and setting. Let’s start our work with creating a new GUI project.

After pressing the “New Project” button a window for customizing the main project settings (Project Name, resolution and First Page Name) will open. Choose the resolution corresponding to the resolution of the target device where the project will be launched. You cannot change it while working on the project.



## *Import of KNX/EIB Channels from ETS projects*

It is necessary to import channel data from an ETS project (supported formats of project files: \*.pr3; \*.pr4; \*.pr5; \*.knxproj) to bind channels to GUI items. You can use a special tool for importing channels from ETS projects to iRidium GUI Editor:



Channel import includes the following steps:

1. Create a new project in iRidium GUI Editor
2. Select Import in the menu (File→ Import)
3. Select an ETS project file which contains channel data (\*.pr3, \*.pr4, \*.pr5, \*.knxproj)
4. Import ETS channels into [iRidium GUI Editor](#)
5. Check and edit (if it is necessary) channel settings in iRidium GUI Editor according to description of KNX/EIB channels.

After the import is completed, indicate the settings of connection to the KNX/EIB bus and drag channels to graphic items.

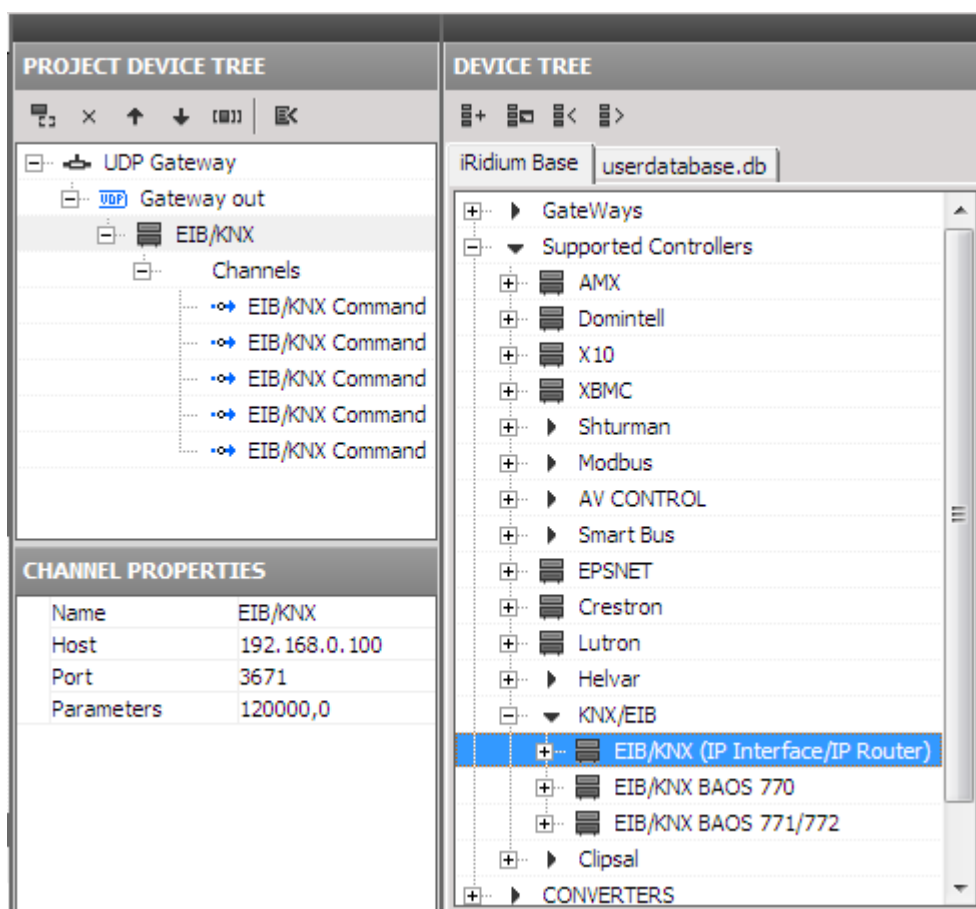
## Setting of connection to KNX/EIB

### Connection to KNX/EIB IP-router through UDP

Setting of connection to the KNX/EIB gateway through UDP protocol:

Import the controlled equipment: **KNX/EIB (IP interface / IP-router)** from iRidium Base to Project Device Tree by the Drag&Drop method. At that, the UDP gateway and the channel are added automatically (the rest channels can be dragged to Project Device Tree or created by cloning the first one).

Project Device Tree will look as follows:



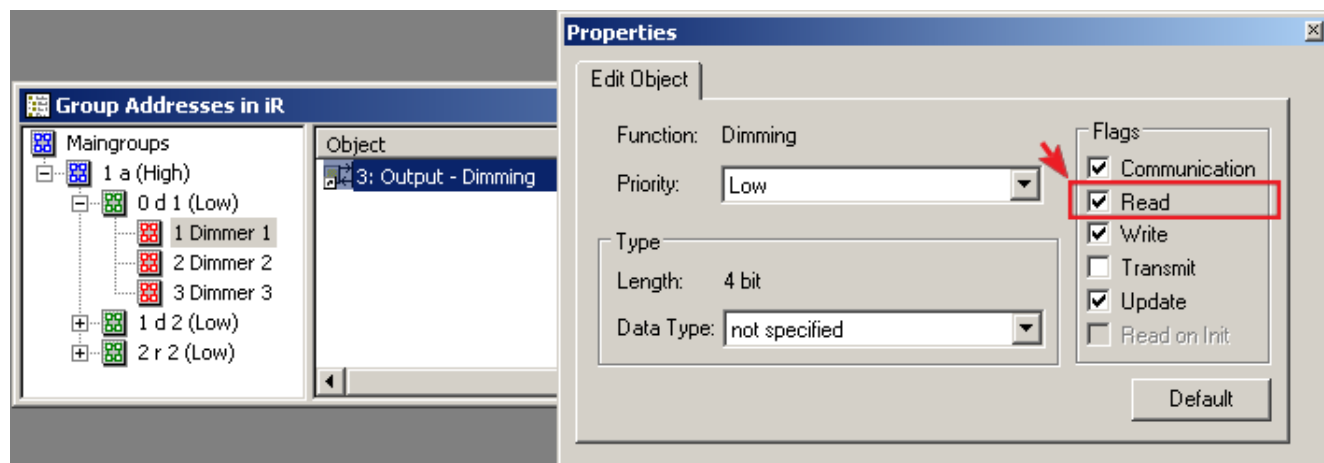
#### Properties of connection to the gateway:

- **Name** – a device name
- **Host** - an IP-address of the KNX/EIB gateway
- **Port** – a port number
- **Parameters** – time out – the time of pause between connections if a connection error occurs (indicated in ms). Time interval between commands which are being sent is indicated after comma.



## Setting of Data Reading in ETS

Any channel can be configured for any data transfer direction: Write, Read, Read/Write. Channels configured for reading need the additional setting. In the ETS development environment mark the checkbox "Read" for the KNX/EIB channels you need to read the data from. Otherwise the reading of a current state in iRidium is impossible.



All channels regardless of their type (Buttons, Levels, etc.) and data transfer direction (Write, Read, Read/Write) operate with the set **Action** property: **Press** (on pressing), **Release** (on releasing) or **Move** (on moving). **Receive** is set only for data reading (Read).

## Setting of Channels for Communication with the KNX/EIB bus

After finishing connection setting you can proceed to setting properties for project channels. The bus contains devices with various types and properties, the operation of which is set in a different way. Each device type has its own control commands. Setting principles of different devices will be provided in the next sections.

CHANNEL PROPERTIES	
Name	EIB/KNX Command
Type	Boolean
Property	Value
Direction	Write
Action	Press
Sub Type	SWITCH
Address	0/0/0
Value	Off

- **Name** – a channel name , set at random
- **Type** – an EIB type. It is set similarly to the ETS development environment
  - *Boolean* (1/0) - for DPT\_B1
  - *Signed 32-bit* (information) – for the rest EIB types.
- **Property** – a pattern of signal change. Set *Value* for all channels.
- **Direction** – a direction of data transfer
  - *Read* - reading only
  - *Write* – writing only
  - *Read/Write* – reading and writing
- **Action** – a way of sending a command
  - *Receive* – receives all changes of a channel state (for reading channels)
  - *Press* – sends a command on pressing
  - *Release* – sends a command on releasing
  - *Move* – for Level – sends each value (which a slider takes when moving the slider bar). It is not recommended when operating with big volume of processed data.
- **Sub Type** – an EIB sub type. It is set similarly to the variable sub type in the ETS development environment
- **Address** – a group address. It should correspond to the actual group address of the cell in ETS. 1/1/1 – this value is taken from the ETS development environment.
- **Value** – a set variable value (level)

## Principles of Controlling Dimmers in the KNX/EIB Bus

Admissible values for a dimmer are in the range from 0 to 255 or from 0 to 100. Dimmer variables can be controlled in a few ways:

- Control by *Level (slider)*
- Control by *Up/Down Button*
- Specifying a fixed level value by *Button*
- Control by *Trigger Button*

### Dimming by Level (slider)

Control of variable value is possible when using different variable types and sub types. On the whole the settings of dimming are identical.

#### Example 1: 8-bit variables with VALUE & SCALING sub types

The image displays two side-by-side screenshots of a configuration interface for KNX/EIB Bus dimmers. Each screenshot shows a 'CHANNEL PROPERTIES' table and a 'PROPERTIES' table.

**Left Screenshot (Item 12):**

CHANNEL PROPERTIES	
Name	Dimmer
Type	Unsigned 8-bit
Property	Value
Direction	Read/Write
Action	Move
Sub Type	SCALING
Address	1/1/1
Value	0

PROPERTIES		
General	Programming	States
Name	Item 12	
Type	Level	
Left	23	
Top	22	
Width	83	
Height	175	
Min	0	
Max	100	
Direction	Vertical	
Active	True	
Page Flip	Commands Count: 0	
Focus Lock Receive	True	
Hit	Active	
Invert	False	
Slider	None	
Slider Color	■ #FF000000	
Sound		

**Right Screenshot (Item 1):**

CHANNEL PROPERTIES	
Name	Dimmer
Type	Unsigned 8-bit
Property	Value
Direction	Read/Write
Action	Move
Sub Type	VALUE
Address	1/1/1
Value	0

PROPERTIES		
General	Programming	States
Name	Item 1	
Type	Level	
Left	130	
Top	22	
Width	83	
Height	175	
Min	0	
Max	255	
Direction	Vertical	
Active	True	
Page Flip	Commands Count: 0	
Focus Lock Receive	True	
Hit	Active	
Invert	False	
Slider	None	
Slider Color	■ #FF000000	
Sound		

8-bit channels with VALUE or SCALING sub types are used to change the dimmer value.

**VALUE:** an 8-bit channel, which enables the change of the dimmer value in the range from 0 to 255.

**SCALING:** an 8-bit channel, which enables the change of the dimmer value in the range from 0 to 100.

**Type** - a variable type. Select *Unsigned 8-bit*.

**Direction** – a direction of channel data transfer. Select *Write* for setting a dimmer output state without displaying its changes (without reading data about the dimmer state) or *Read/Write* for displaying the dimmer state at any external change (also at the project launch when having a request).

**Sub Type** – a variable EIB sub type: *VALUE* (0..255) or *SCALING* (0..100)

**Value** – a dimmer level.

- If a channel will be bound to Button, i.e. will be used for displaying a fixed value in the range from 0 to 100 (from 0 to 255) on pressing Button, indicate the value which the dimmer should take.
- If a channel will be bound to Level, leave 0. The dimmer will be controlled by changing the value of the slider scale.

The created channel can be bound to Button or Level.

Level should have the following properties:

The “General” tab:

**Type** – an item type: *Level* (slider).

**Min ... Max** – minimum and maximum values of the Level scale. The other values Level takes are limited by these ones. For a KNX/EIB dimmer they are: 0/100 or 0/255.

**Direction** – Level positioning: Horizontal or Vertical.

The “Programming” tab:

**Feedback** – a graphic item feedback type. It is always *Channel* for *Level*.

Button should have the following properties:

The “General” tab:

**Type** – an item type: *Button*.

The “Programming” tab:

**Feedback** – a graphic item feedback type: *Momentary*.

If it is a channel with the *Read/Write* direction of data transfer, set *Channel* as the graphic item feedback type regardless of its other settings.

Bind the preset dimmer channel to an item by using the Drag&Drop method. It will appear in the “Programming” tab.

### Example 2: 4-bit variable with VALUE sub type

A dimmer can only send data to the KNX/EIB bus (Write, Channel) or perform a two-direction data exchange – send values and display an actual current state (Read/Write, Channel):

CHANNEL PROPERTIES	
Name	EIB/KNX Command
Type	Signed 32-bit
Property	Value
Direction	Read/Write
Action	Release
Sub Type	VALUE
Address	1/1/1
Value	0

CHANNEL PROPERTIES	
Name	EIB/KNX Command
Type	Signed 32-bit
Property	Value
Direction	Write
Action	Move
Sub Type	VALUE
Address	1/1/1
Value	0

The following properties are set in Channel Properties:

**Type** – a variable type. Select *Signed 32-bit* for a dimmer – an information channel - as values 0 and 100 are operated with. *Boolean*, a logic variable, can be used when working with 0 and 1 only.

**Direction** – a direction of channel data transfer. Select *Write* for setting a dimmer output state without displaying its changes (without reading data about the dimmer state) or *Read/Write* for displaying the dimmer state at any external change (also at the project launch when having a request).

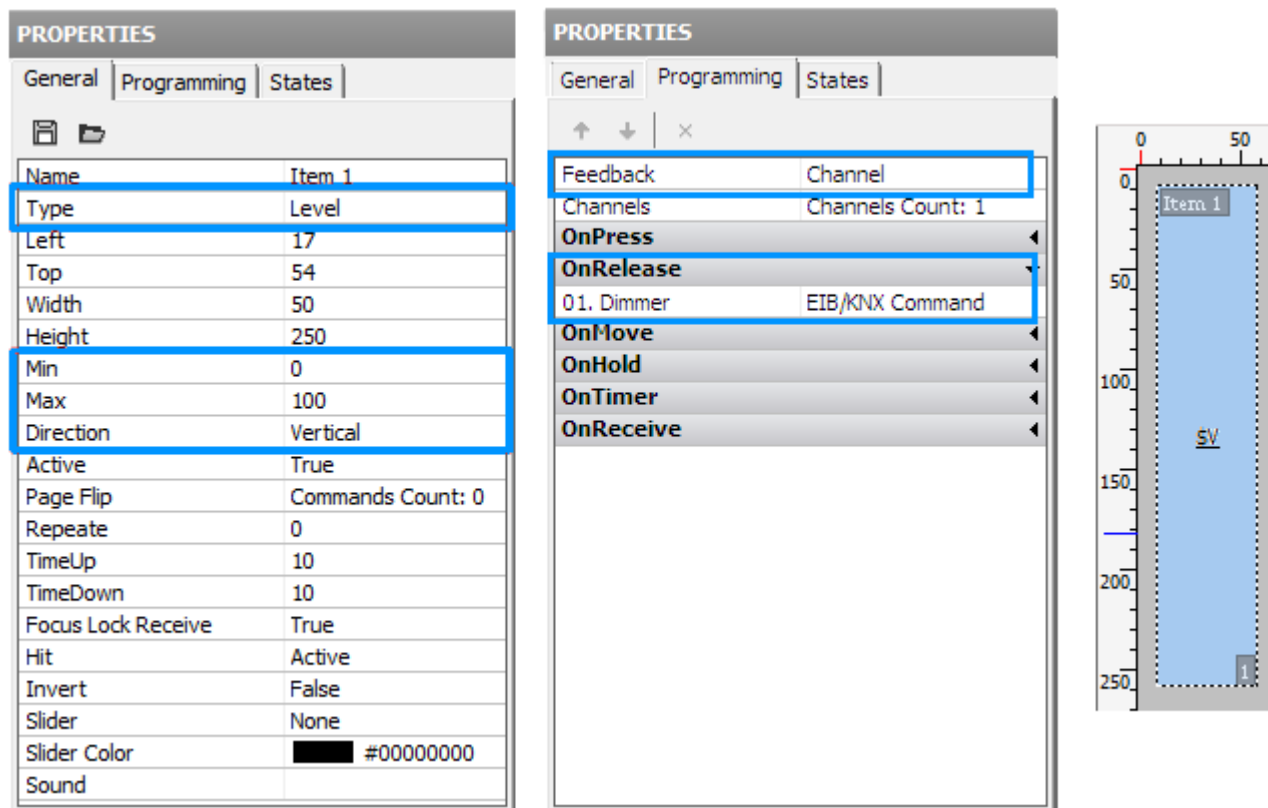
**Action** – a way of sending a command. You can select *Move* as the command is sent when the slider is actuated.

**Sub Type** – a variable EIB sub type.

**Address** – an address of the device communicating with the gateway

**Value** – a dimmer level (0 ... 100). As it is set by the slider this property is not specified, leave 0.

The created channel is bound to a graphic item with the following properties:



### The "General" tab:

**Type** – an item type: *Level* (slider).

**Min ... Max** – minimum and maximum values of the Level scale. The other values Level takes are limited by these ones. For a KNX/EIB dimmer they are: 0/100.

**Direction** – Level positioning: Horizontal or Vertical.

### The "Programming" tab:

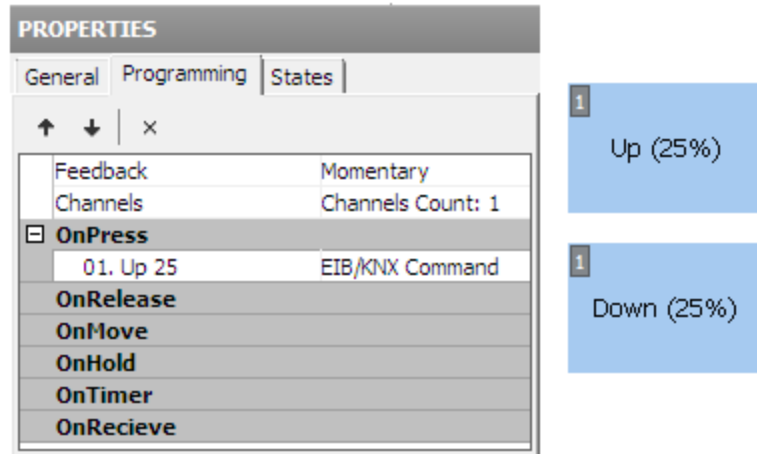
**Feedback** – a graphic item feedback type. It is always *Channel* for *Level*.

Bind a preset dimmer channel to an item by using the Drag&Drop method. It will appear in the "Programming" tab.

### ***Dimming by Increase/Decrease Channels***

It can be done by Buttons with Increase/Decrease KNX/EIB channels:

CHANNEL PROPERTIES		CHANNEL PROPERTIES	
Name	EIB/KNX Command	Name	EIB/KNX Command
Type	4 bit	Type	4 bit
Property	Value	Property	Value
Direction	Write	Direction	Write
Action	Press	Action	Press
Sub Type	CONTROL_DIMMING	Sub Type	CONTROL_DIMMING
Address	1/1/1	Address	1/1/1
Value	(3) 25% Decrease	Value	(11) 25% Increase



To change the value by some preset value in percentage terms, create a channel with the following properties:

**Type** – a variable type. Select *4-bit*.

**Direction** – a direction of channel data transfer. Select *Write* for setting a dimmer output state without displaying its changes (without reading data about the dimmer state) or *Read/Write* for displaying the dimmer state at any external change (also at the project launch when having a request).

**Sub Type** – a variable EIB sub type. Select *CONTROL\_DIMMING* (as in ETS).

**Address** – a group address of the controlled variable.

**Value** – a value and direction of level changing (upward or downward in percentage terms).

The created channel is bound to a graphic item with the following properties:

The “General” tab:

**Type** – an item type: *Button*.

The “Programming” tab:

**Feedback** – a graphic item feedback type: *Momentary* or *None*.

Bind a preset dimmer channel to an item by using the Drag&Drop method. It will appear in the “Programming” tab.

There have to be two channels to change the dimmer value in two directions. Their settings are different only by the Value property where you should indicate the direction of value change (*Decrease/Increase*).

## Dimming by Up/Down Button

It is often necessary to change the dimmer value by some fixed value up or down from the current one, for example, to change the dimmer value by 5, 10%, 20% or other.

KNX/EIB dimmer values change in the range from 0 to 255 or from 0 to 100 depending on the channel type. You can set a step of value change within the range and regulate it with two buttons. The same channel is bound to both buttons. Value change direction and change limits are set in the graphic item Properties where the “*Up/Down Button*” type is selected which is aimed especially for this type of value regulation.

You don't need to create a separate channel for binding to the *Up/Down Button* type graphic items. You can bind to them the same channel which was set for dimming by *Level*.

CHANNEL PROPERTIES	
Name	Dimmer
Type	Unsigned 8-bit
Property	Value
Direction	Read/Write
Action	Move
Sub Type	SCALING
Address	1/0/2
Value	0

Then you can proceed with the setting of graphic items. The properties of *Up Button* and *Down Button* are different and should be set separately:

**PROPERTIES**

General | Programming | States

Name	Item 11
Type	Up/Down Button
Left	206
Top	84
Width	75
Height	75
Active	True
Page Flip	Commands Count: 0
Up/Down Value	10
Max/Min Value	100
Sound	

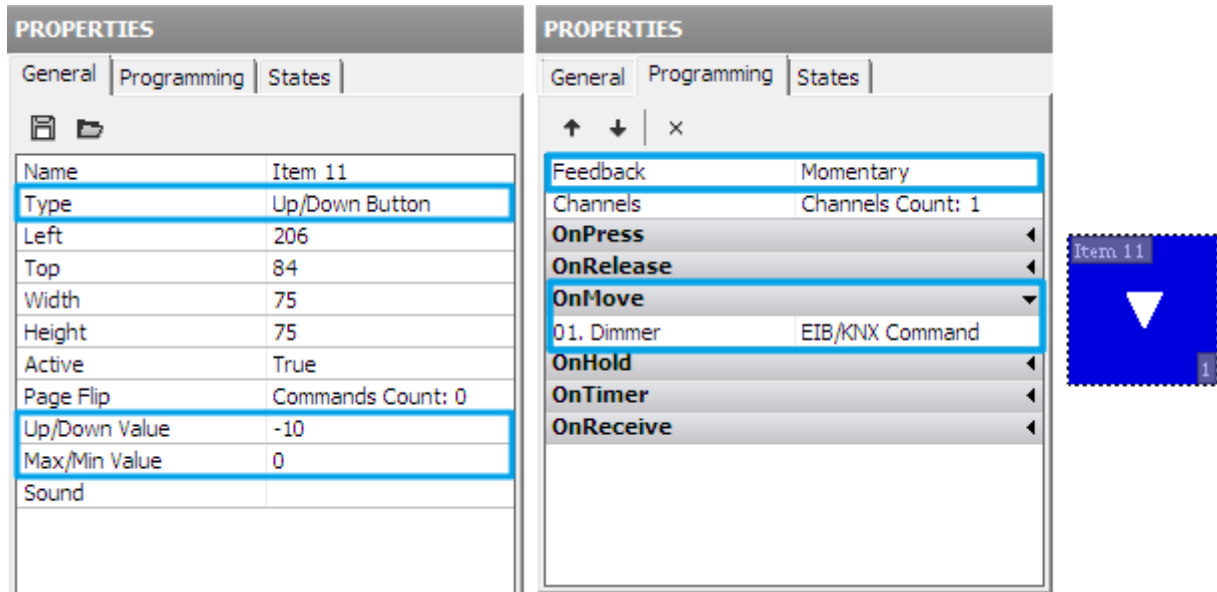
**PROPERTIES**

General | Programming | States

↑ ↓ ×

Feedback	Momentary
Channels	Channels Count: 1
<b>OnPress</b>	◀
<b>OnRelease</b>	◀
<b>OnMove</b>	▼
01. Dimmer	EIB/KNX Command
<b>OnHold</b>	◀
<b>OnTimer</b>	◀
<b>OnReceive</b>	◀





The “General” tab:

**Type** – an item type: *Up/Down Button*.

**Up/Down Value** – a step of value increase/decrease for a single pressing. To increase the value use positive value, to decrease – negative. For example, for *Up Button* set «10», for *Down Button* «-10».

**Max/Min Value** – maximum or minimum values taken by a variable. For *Up Button* set dimmer maximum: 100, for *Down Button* set its minimum: 0. The limits have to correspond with the actual extreme values of the variable or can be less to make a cut-off.

The “Programming” tab:

**Feedback** – a graphic item feedback type. It is always *Momentary*.

## Setting a Fixed Value

### Button: Setting a Fixed Value for Dimmers and Relays

You can set the level value in the available range by *Button*. The channel, which setting is performed similarly to a dimmer or relay channel, is used for that. The only difference in setting is indication of the fixed Value (i.e. the value the variable has to take).

The screenshot displays two panels. The left panel, titled 'CHANNEL PROPERTIES', shows the following details for a channel named 'Dimmer Scaling50':

CHANNEL PROPERTIES	
Name	Dimmer Scaling50
Type	Unsigned 8-bit
Property	Value
Direction	Write
Action	Press
Sub Type	SCALING
Address	1/0/2
Value	49

The right panel, titled 'PROPERTIES', shows the 'Programming' tab with a list of actions:

- Feedback: Momentary
- Channels: Channels Count: 1
- OnPress: 01. Dimmer Scaling50 EIB/KNX Command
- OnRelease
- OnMove
- OnHold
- OnTimer
- OnReceive

To the right of the programming list, a small blue box labeled 'Item 11' contains the text 'Dimmer scaling 50'.

**Value** – the value, the dimmer has to be set up by *Button*.

### Relay Trigger Switch by iRidium

There is often a necessity to change a relay state with one button (and not two buttons sending the opposite values) – trigger switch. A *Trigger Button* graphic item is used to implement a button sending the value opposite to the current channel variable value at each pressing. There have to be a channel sending data to the bus with the *Read/Write* direction for operation with this item: it has to have a possibility to send data to the bus and receive the current state of a variable (any of the presented below). You can also use two channels with the *Read* and *Write* directions. EIB sub type should be set either *SCALING* or *SWITCH*.

The image shows three 'CHANNEL PROPERTIES' panels side-by-side, illustrating different configurations for a trigger switch:

CHANNEL PROPERTIES	
Name	Trigger1
Type	Unsigned 8-bit
Property	Value
Direction	Read/Write
Action	Press
Sub Type	SCALING
Address	1/0/2
Value	0

CHANNEL PROPERTIES	
Name	TriggerBoolean
Type	Boolean
Property	Value
Direction	Read
Action	Press
Sub Type	SWITCH
Address	1/0/2
Value	On

CHANNEL PROPERTIES	
Name	TriggerBoolean
Type	Boolean
Property	Value
Direction	Write
Action	Press
Sub Type	SWITCH
Address	1/0/2
Value	On

## Relay Trigger Switch by KNX/EIB

Unlike *Trigger Button* the graphic item of this type receives the actual channel state only at the project launch and then changes its state to the opposite by KNX/EIB means. The drawback of using this type item is that when you change the trigger position from another panel or through ETS, a variable with the TRIGGER sub type will not receive any data about that. And the channel will not send data about changing the trigger state. You have to press the button of state switching twice to synchronize trigger positions.

Therefore, we recommend using the iRidium *Trigger Button* item as it enables the monitoring of the variable state in real time.

CHANNEL PROPERTIES	
Name	TriggerBoolean
Type	Boolean
Property	Value
Direction	Read/Write
Action	Release
Sub Type	TRIGGER
Address	1/0/2
Value	1

Channel type of response *-Action-* can be *Press* or *Release*. The set channel has to be bound to a graphic item with the following properties:

### The “General” tab:

**Type** – an item type. Select *Trigger Button*

**Trigger Value 1 / Trigger Value 2** – values the trigger can take (upper and lower). For a relay they are: **0** (OFF) and **1** (ON).

### The “Programming” tab:

**Feedback** – a graphic item feedback type. Select *Channel* to process the received data and send commands to the bus. This feedback type is obligatory for *Trigger Button*.

## Reading Data about a KNX/EIB Variable State

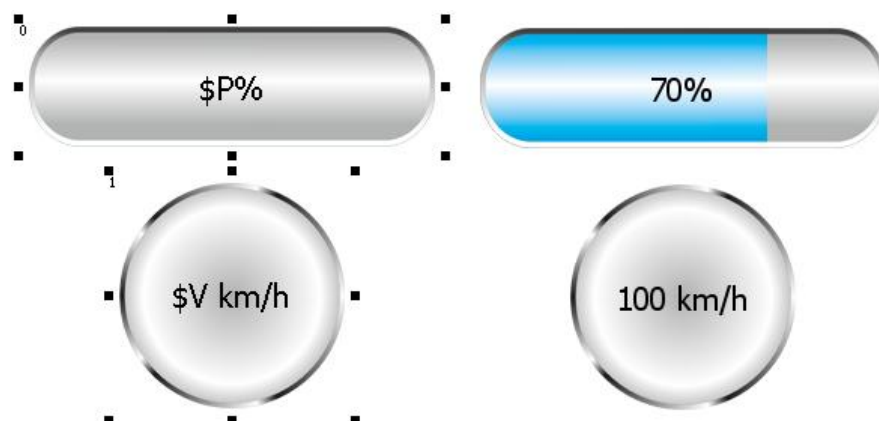
### Output of temperature or channel variable values

If a graphic item is used not only to send but also to receive data about a current variable state, it is necessary to output its current value to that item. It is essential for the variables taking values in the wide range. The values, which may require the numerical display, are: temperature, pressure, time, date, filling level, speed, etc.

There is a number of Templates of value output to display a numerical variable value. A template is a combination of symbols. The indication of these symbols in the item text initiates the display of the current value of the variable bound to the item instead of the symbols. You can enter a random text before and after the template (separation by comma is not necessary).

Template	Function
\$P	Variable value in percentage terms from the maximum value
\$V	Current variable value
\$F1, \$F2, \$F3, ...	Current value with the fixed number of symbols after the point
\$S	Current state
\$L	Lower level value
\$H	Upper level value
\$A	Current value minus lower level value
\$R	Upper value minus lower value
\$\$	Displaying of the "dollar" symbol

Let us give an example of one of the templates of value output to graphic items: the **\$V km/h** template for a Level item with the Channel feedback, allows you to output a current speed value that is visualized by Level position and the changing number. Data on the current variable state can be received from the channel or set manually if the channel is set to data writing.



Templates of value output to graphic items can be bound both to Buttons and Levels. They can display a value sent to a channel (e.g., Level that sets a channel value equal to 70% from the maximum), as well as a value received as a feedback (e.g., Button that displays a value received from a channel).

The example of a template displaying in the Text input field of a graphic item:

The image shows the configuration interface for a graphic item. On the left, the 'PROPERTIES' panel is open to the 'States' tab, showing 'State 1' with various settings. The 'Text' field is highlighted with a blue border and contains the value '\$V'. On the right, the 'CHANNEL PROPERTIES' panel shows settings for 'Dimmer 32', including 'Type: Signed 32-bit', 'Property: Value', 'Direction: Read/Write', 'Action: Release', 'Sub Type: VALUE', 'Address: 1/0/2', and 'Value: 0'. To the right of these panels, a visual representation of the graphic item is shown. It consists of a vertical rectangle divided into two sections: a top green section and a bottom blue section. The text '\$V' is displayed in the blue section, and the value '111' is displayed in the green section. A dashed box around the top section is labeled 'Item 21'.

### Setting of Data Reading in ETS

Any channel can be configured for any data transfer direction: Write, Read, Read/Write. Channels configured for reading need the additional setting. In the ETS development environment mark the checkbox "Read" for the KNX/EIB channels you need to read the data from. Otherwise the reading of a current state in iRidium is impossible.

The image shows the 'Properties' dialog box for a 'Dimming' object. The 'Function' is set to 'Dimming', 'Priority' is 'Low', 'Type' is '4 bit', and 'Data Type' is 'not specified'. The 'Flags' section is expanded, and the 'Read' checkbox is checked and highlighted with a red box. Other flags include 'Communication', 'Write', 'Transmit', 'Update', and 'Read on Init'. A red arrow points to the 'Read' checkbox. The 'Default' button is visible at the bottom right.

## Connection to the Bus over the Internet: iRidium Gate

**iRidium Gate** – an application for communication with devices, which do not support simultaneous connection of several Clients or remote communication (Ethernet/Wi-Fi). iRidium Gate enables the communication of KNX/EIB equipment with several Clients through one IP-router. It converts TCP to the protocol which is supported by target equipment.

**Features** of the iRidium Gate application for KNX/EIB:

Suppose , a JUNG IPR 100 REG type IP-router is used to communicate with the KNX/EIB equipment. There are some limitations at such connection:





1. Communication of Clients with the equipment (connected to PC through the COM-port) via Wi-Fi or 3G is not possible.
2. Only one connection is possible. You have to add additional KNX/EIB IP-routers if you want to add the second or the third iPhone/iPad.
3. IP-router does not respond to control commands for several minutes after the incorrect disconnection.

iRidium Gate provides the stable connection between iPhone/iPod/iPad and a KNX/EIB system (after loosing connection KNX/EIB IP-router cannot reconnect for 1-2 minutes without iRidium Gate). It also eliminates the limitation on the number of devices connected to the bus simultaneously.

iRidium Gate can be installed on any PC and enables you to establish multiple connection through one KNX/EIB IP Router and provide connection permanency. It makes communication from iPhone/iPad more stable and reduces the cost of your KNX system as additional KNX/EIB IP-routers are not required.

**IMPORTANT:** You should indicate the IP-address of the KNX/EIB gateway in iRidium Gate settings; and the IP-address of PC with the running iRidium Gate application in the settings of a project for iPad/iPhone/iPod.

**Interface** of the iRidium Gate application:

	Adding a new Gate (a gateway for data converting and transferring )
	Setting of the added Gate
	Enabling Gate (disabled Gate does not transfer data)
	Disabling Gate



## Deleting Gate

### **Creation and setting** of Gate for operation with KNX/EIB:

Create a new gateway for data converting and transferring from the Extranet or Intranet to the KNX/EIB bus by clicking the *Add Gate* button. You can set Gate in several ways. Different settings are used when operating through UDP:

#### **Gate section:**

Gate  
Name: EIB/KNX Gate  
Comment: TCP to UDP

**Name** – Gate name, random

**Comment** – Gate description, if required

#### **Server section:**

Server  
Type: EIB/KNX  
Transport: Tcp  
Port: 3671  
Max Connection: 4

**Type** – a Gate type. Select KNX/EIB to work with the bus equipment

**Transport** – a type of transport which is used for connection of Clients (control panels with iRidium) to Gate. TCP is always used.

**Port** – external Gate TCP-port, which iRidium KNX/EIB Clients are connected to. It can be set at random.

**Max Connection** – maximum number of Clients connected to Gate concurrently. It is set at random (do not forget that an abundant number of Clients can overload the bus).

#### **Device section:**

Device  
Type: EIB/KNX  
Transport: Udp  
Host: 192.168.0.99  
Port: 3671  
Idle: 70000 ms

**Type** – a data format, which they are transferred to the bus through Gate in.

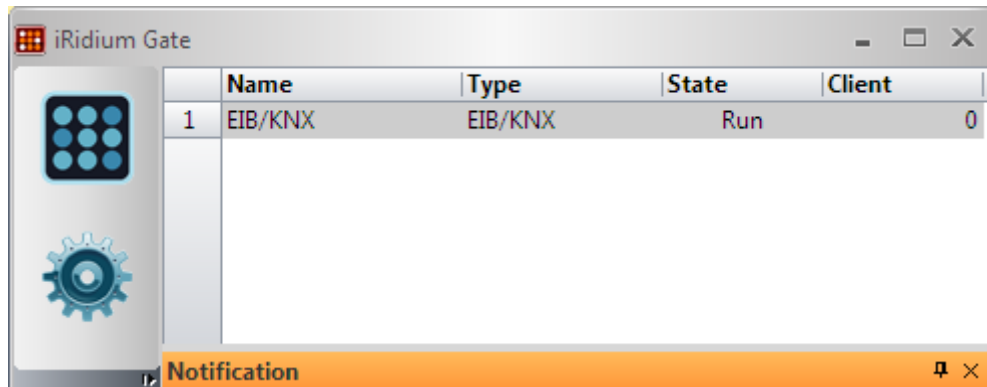
**Transport** – a type of transport, which is used for transferring data to the KNX/EIB bus.

**Host** – an IP-address of the KNX/EIB router, which Client is connected to

**Port** – a UDP-port of the EIB gateway, which is used for communication with the equipment

**Idle** – an interval between connections in case of connection loss

When Gate is set, save the settings. It will have the following view in the iRidium Gate application window:



Gate enabling/disabling is performed through the application menu (menu buttons on the left). If you want to connect Clients, Gate has to be enabled. The number of the Clients connected at the moment is displayed in the *Client* window.

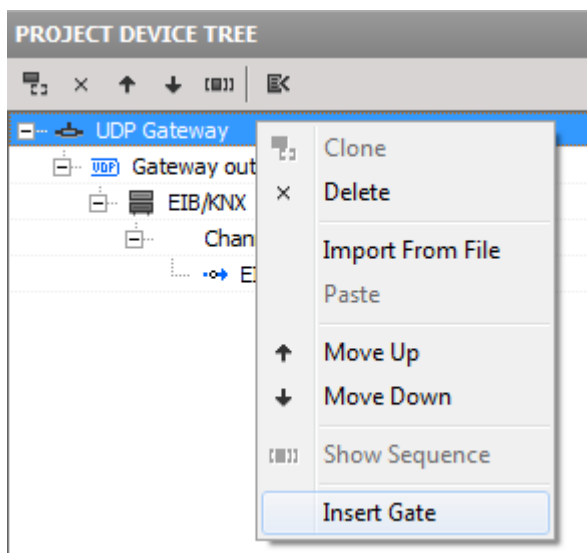
When the iRidium Gate application is set, it is necessary to create or customize the iRidium GUI Editor project for KNX/EIB, which is aimed for connection with the bus devices. If the project is not set properly it cannot be connected to the iRidium Gate application.

### **Setting of the iRidium GUI Editor Project for Connection to Gate**

The TCP/IP protocol connection is used to enable the connection of several control devices (iPhone/iPod/iPad or Windows devices) one KNX/EIB IP- router. The TCP/IP protocol connection is made to the iRidium Gate application.

Gate setting in iRidium GUI Editor is performed by adding Gate (gateway) into the new or existing Project Device Tree.





You need to indicate properties of connection to your PC (server) with the running iRidium Gate application in the gateway settings:

CHANNEL PROPERTIES	
Name	TCP to UDP Gate
Host	192.168.0.100
Port	6000
Parameters	

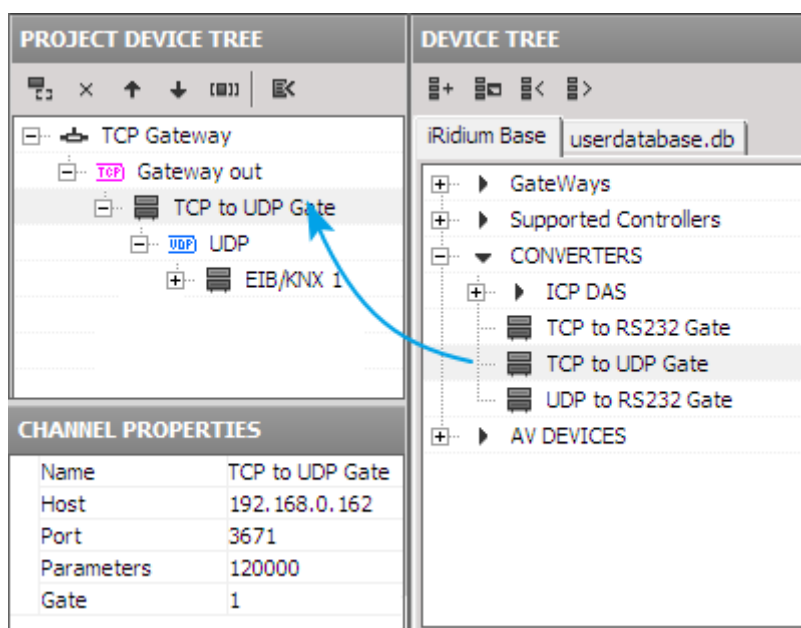
**Name** – a gateway name

**Host** – an IP-address of PC with the running iRidium Gate application

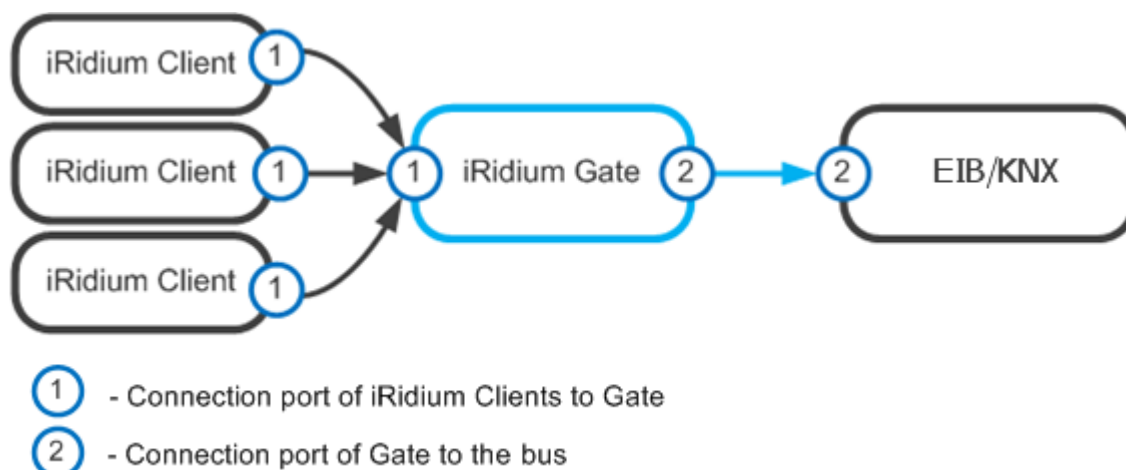
**Port** – a port number, set at random, but should be the same as the one set in Gate

**Parameters** – time out - the time of pause between connections if a connection error occurs (indicated in ms).

To add Gate manually to the new project, which is created without import, you should add the gateway into Project Device Tree from iRidium Base:

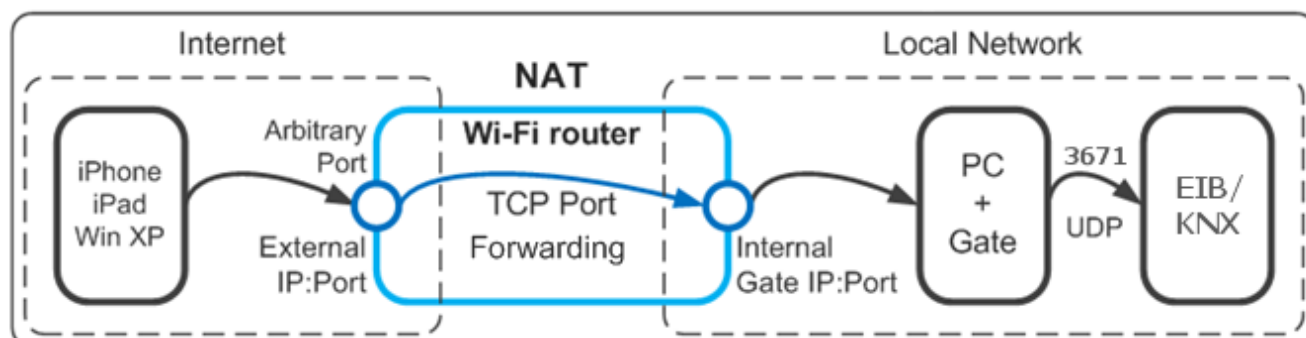


Thus, we get the following connection diagram of Client to the KNX/EIB bus through the iRidium Gate application:



The connection port of Clients to Gate is chosen at random and is set in Gate first and then in GUI Editor. The connection port of Gate to the bus (3671 by default) is indicated in the iRidium Gate application.

*For operation over the Internet* it is necessary to set the forwarding port on your Wi-Fi router, so the commands could be forwarded from the external network to the local network:



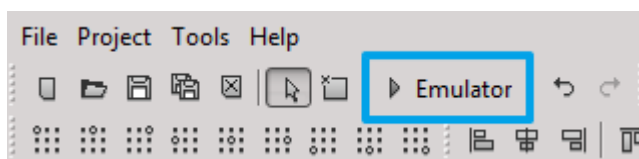
\* refer to your system manager to set the port forwarding

## Uploading a Project on a Panel, Connection to Equipment

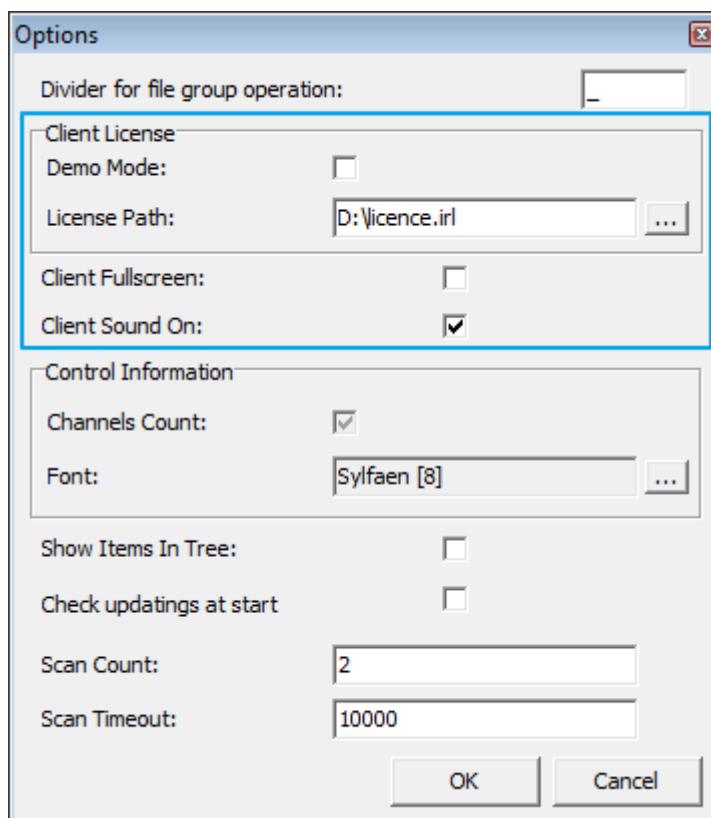
When GUI creation is completed it is necessary to check its operation and upload a ready GUI on an iOS or Windows control device.

### The Emulation Mode

The Emulation mode allows you to check the proper operation of the project graphic interface and connection to the KNX/EIB bus. But the connection with the bus can be established only when having an iRidium license whereas the graphic interface is fully functional without the license. Click the *Emulator* button in the toolbar of the Editor to start the Emulator:



The setting of license connection, enabling/disabling sound-on buttons and switching on/off the full screen mode when running Emulator are performed through the Options window (Tools → Options):

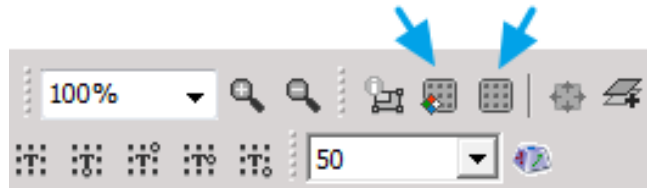


When there is a tick in the *Demo Mode* field, there is no connection to the gateway and there will be no messages about operation in the demo mode at each start of Emulator. To operate in the fully functional mode indicate a path to the license which is activated for Windows PC where Emulator is running in the License Path field.

## Conversion of the Project for Its Launching on Target Devices

When the operation of the project is checked in the Emulation mode you can start its uploading on the iOS or Windows target device. Conversion of the project to the format for its launching on the panel is made in iRidium Transfer. It is an application for converting and uploading projects on control devices and also for storing and binding iRidium licenses. The transfer of the project, which is created in iRidium GUI Editor, to iRidium Transfer can be made in several ways:

### 1. Transfer of the design by the *Send to Transfer* buttons in the iRidium GUI Editor toolbar



- a) The first type – the *Send to Transfer* button – opens the project in Transfer for the quick upload on an iOS device (iPad / iPhone / iPod Touch). The window for selecting the iOS device, which the design is to be bound, from the list of Panels in Transfer will open when transferring. After selecting the panel you can start design updating on the iOS device, additional settings – by default.
- b) The second type – the *Send to Transfer 1* button – adds the project in the list of Designs in Transfer and opens the menu of its basic settings. This transfer type is convenient when creating a Windows project. Right after the design transferring and indicating its basic properties you can click the Generate for Windows button and receive a ready Windows Client.

### 2. Transfer of the saved design file

iRidium GUI Editor can save designs in \*.irp и \*.irpz format. Files of these formats can be uploaded in the list of Designs in Transfer by the *Add Design* button or by the Drag & Drop method:



After the addition of the design a window with its basic settings will open, select the required ones.

When the GUI is added in the list of Designs in Transfer you can start its conversion and uploading in the iOS or Windows target control device.

The upload of projects on iOS and Windows devices is made differently. To convert a project for Windows you need to press the *Generate for Windows* button. And for uploading on iOS device you have to add the iOS device, where the project will be uploaded on, to the list of Panels in Transfer. This process is described in the following section.

## Uploading the Project on an iOS Device

In order to upload a design on the control device based on Apple iOS operation system, it is necessary to establish connection between the iRidium Transfer application installed on your personal computer, and the “iRidium for KNX” Client application installed on the iOS device.

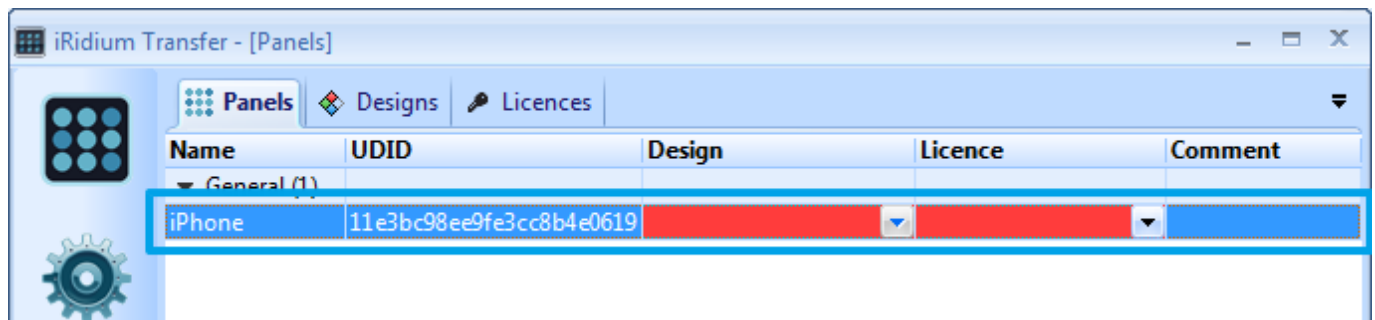
### Addition of iOS Devices to the Transfer Panels List

In order to establish the connection between Transfer and “iRidium for KNX”, it is needed to add the control device with the installed “iRidium for KNX” Client into the list of Panels in Transfer. Addition of the iOS based device into the list of Panels in Transfer is available through the Device Search option. For a successful search, please make sure that your PC and iOS device are connected to the same Wi-Fi network (it can be checked with ping request). Discovery of the device cannot be done over USB connection.

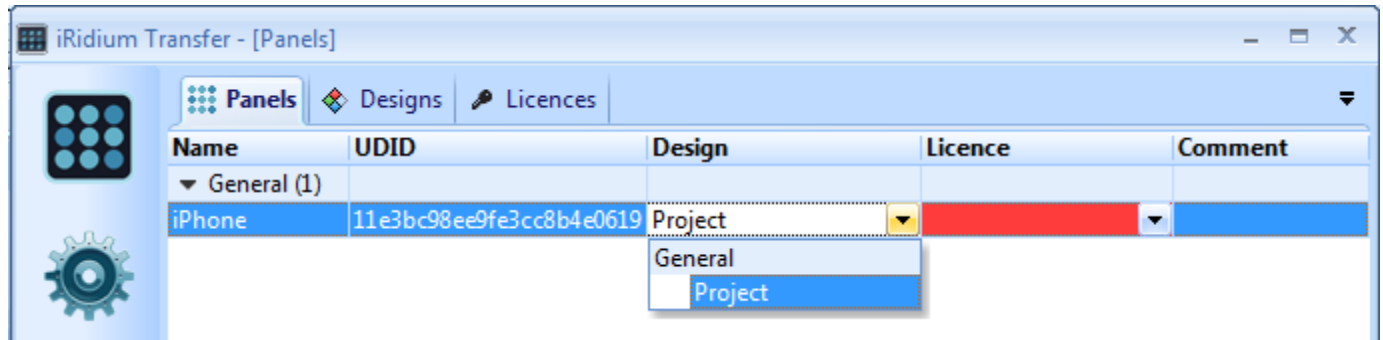


To start discovery of the devices, press the *Search* button in the Panels tab of the Transfer application, and open the “iRidium for KNX” Client application on your iOS device. Please make sure you are opening the application, and not restoring it from the minimized state which is possible on the devices with multitasking support. Before starting the discovery, delete “iRidium for KNX” Client from the list of the launched programs.

If Wi-Fi is active and the search port for panels in the Transfer application (10000) is not blocked or busy by another application, your iOS device will appear in the list of panels in Transfer:



Here you can see two fields highlighted red. The first field is for selection of the graphic interface that will be uploaded on the selected panel. It is possible to select a required design from the dropdown list containing all added designs in the Transfer application.



The second field is for the attaching the license file that will enable the graphic interface to connect to the controlled equipment. In case if a license file is absent, there will be no connection between “iRidium for KNX” and the equipment, you will not be able to neither send control commands nor receive the feedback. At this stage you can generate your project for Windows but there will be no connection to the equipment without the license.

### *Description of the Licensing Procedure for Control Devices*

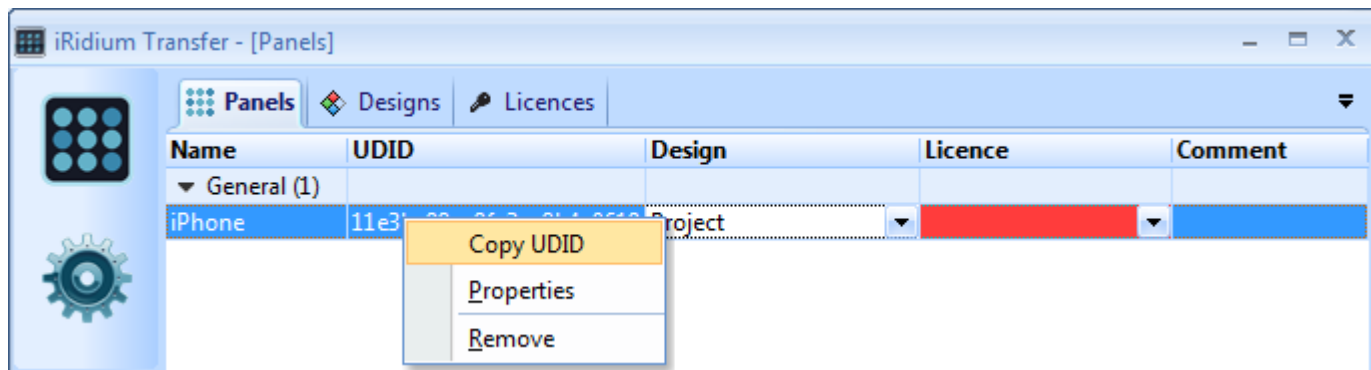
In order to receive a license file, you are required to purchase iRidium Activation Key and go through the licensing process. The license process includes an activation of the purchased Key, after which you will receive the license file for an iOS device that needs to be added into the Transfer application or the license file for a Windows device that needs to be added to the folder with the ready project.

iRidium Keys differ both by operating systems of the control devices they can be bound to and by their type. There are two types of licenses: Base and Ultimate. To operate with KNX/EIB equipment you require Base licenses.

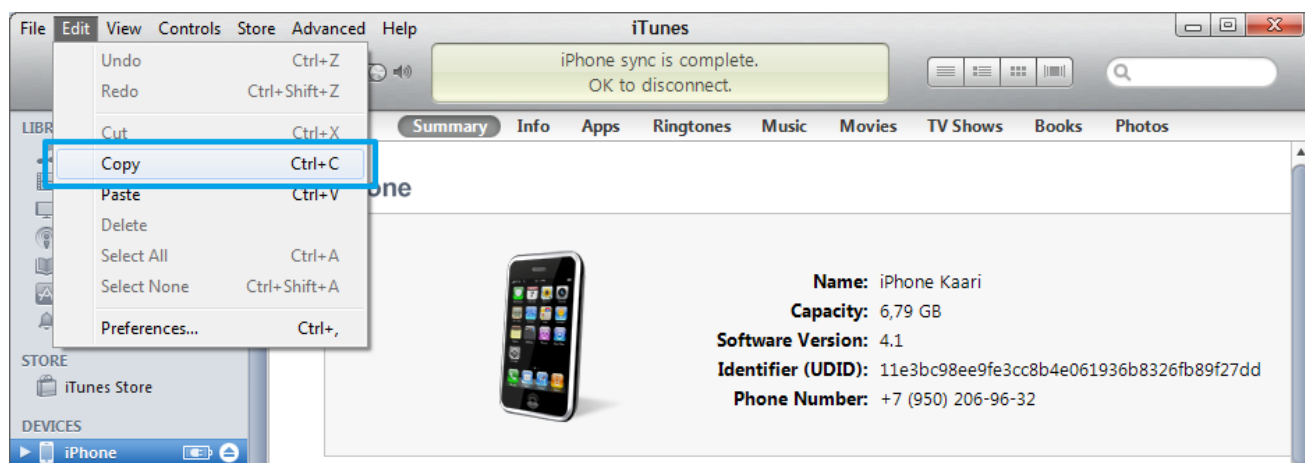
**Base licenses** give a control device (iPad/iPhone/iPod, Windows panels) permission for connection to the controlled equipment. Such license is bound to a particular panel (iOS or Windows).

### *Receipt of UDID for Activation of iRidium Base License for an iOS Device*

For receiving UDID of the iOS panel from the list of Panels in Transfer, copy it from the *Panels* tab in the device parameter line:



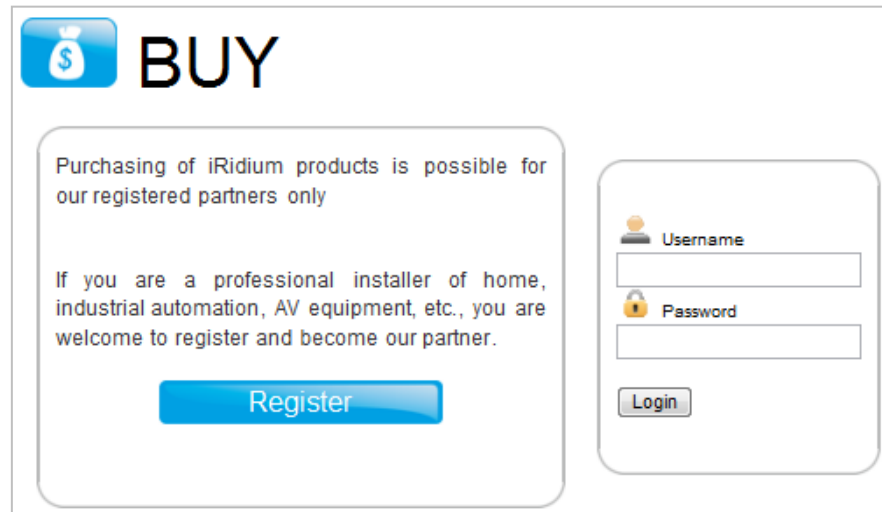
Besides, UDID of the iOS device can be copied from the iTunes application. It can be helpful when device is added manually or when you work via Internet:



## Licensing Procedure for an iOS Device

The licensing procedure of an iOS device begins with purchasing iRidium Activation Key which will be necessary for creation of the license file. The iRidium activation Key can be purchased in two ways:

1. You can register at our web site, wait till your registration is approved and purchase Activation Key directly from the web site, in section BUY: <http://iridiummobile.net/buy2>



**BUY**

Purchasing of iRidium products is possible for our registered partners only

If you are a professional installer of home, industrial automation, AV equipment, etc., you are welcome to register and become our partner.

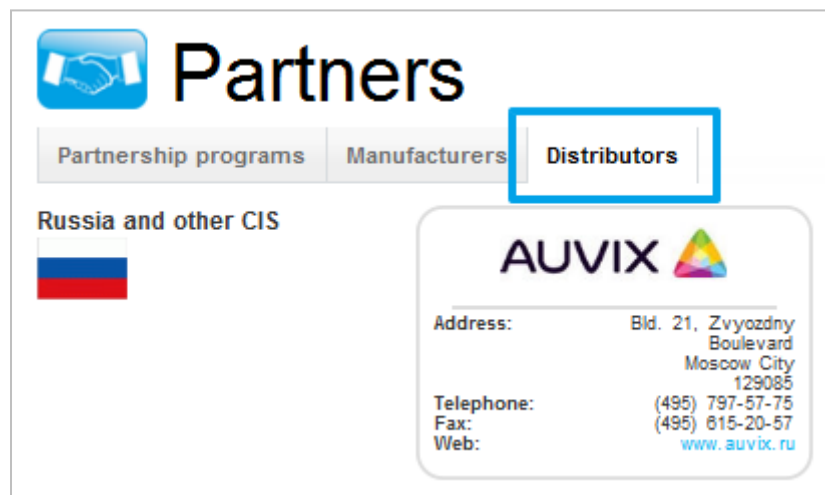
[Register](#)

Username

Password

[Login](#)


2. You can contact the official iRidium distributor in your country. A list of distributors is located in the Partners section at our web site <http://iridiummobile.net/partners>



**Partners**

Partnership programs | Manufacturers | **Distributors**

Russia and other CIS



**AUVIX**

Address: Bld. 21, Zvyozdny Boulevard  
 Moscow City  
 129085

Telephone: (495) 797-57-75  
 Fax: (495) 615-20-57  
 Web: [www.auvix.ru](http://www.auvix.ru)

If you are an independent installer for automation systems, you can also receive demo Activation Keys for free, after filling in a separate registration at our web site: <http://iridiummobile.net/registration>

[Get Free License  
HERE](#)



After the purchase is made, you will receive an e-mail with Activation Key that will be used to make the iRidium license, i.e. to generate a license file for a selected device:

**Dear User,**

We appreciate your interest in our product **iRidium**.

In order to start **iRidium** testing, please, do the following :

1) **Receive your license** ~~(your name - Base password - iPad)~~ using the following demo activation key for your:

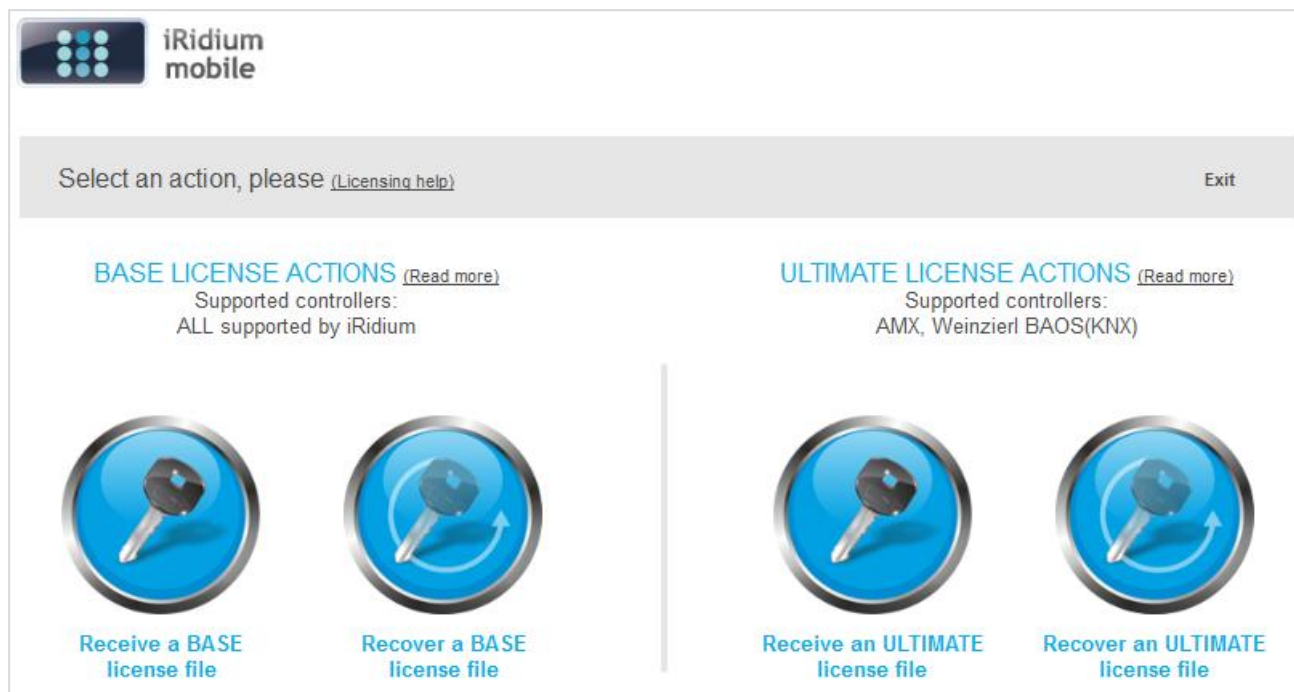
**iPad - 532b38fbdf23e1392b5d0a0d0e184c7990225784cbe793b3c5b183f4395b5d0a4439060d472b453d6eae74e2eb  
a7ee93**

**Activation of your key is made at the following web site:**

<http://iridium.license-maker.com/>

You will receive login information to access the activation web site in the e-mail.

At this web site: <http://iridium.license-maker.com/> you can also recover your license file in case it was lost:



The screenshot shows the iRidium mobile website interface. At the top left is the iRidium mobile logo. Below it is a navigation bar with the text "Select an action, please" and a link to "Licensing help", and an "Exit" button on the right. The main content area is divided into two sections: "BASE LICENSE ACTIONS" and "ULTIMATE LICENSE ACTIONS". Each section has a "Supported controllers:" list and two circular icons representing license actions. The "BASE LICENSE ACTIONS" section lists "ALL supported by iRidium" and has icons for "Receive a BASE license file" and "Recover a BASE license file". The "ULTIMATE LICENSE ACTIONS" section lists "AMX, Weinzierl BAOS(KNX)" and has icons for "Receive an ULTIMATE license file" and "Recover an ULTIMATE license file".

### *Activation of the iRidium Base License Key for an iOS Device*

In order to activate the Base license Key you will be required to enter the following information in corresponding fields: your e-mail address where the license file will be sent to, Activation Key that you received in e-mail, and identification of the specific device (Device UDID) for which the license is generated:

**Process of generating the BASE license file.**
Menu    Exit

**Enter required data, please:**

E-mail (your license file will be sent to this e-mail) \*

License Activation Key (string of 96 symbols) \*

UDID for iPad/iPhone/iPod touch or HWID for Windows XP/7 \* ([How to receive UDID or HWID?](#))

License file name (by default - License.irl)

 .irl

\* This field is required

After completing all required fields in the Activation window, you will be asked to fill out a user form. This short form needs to be filled out only once and will be used in the future for activation of new licenses from the same e-mail address:

**Please, register to continue making a license file.**
Menu    Exit

Name \*

Surname \*

Company Name

E-mail \*

\* This field is required

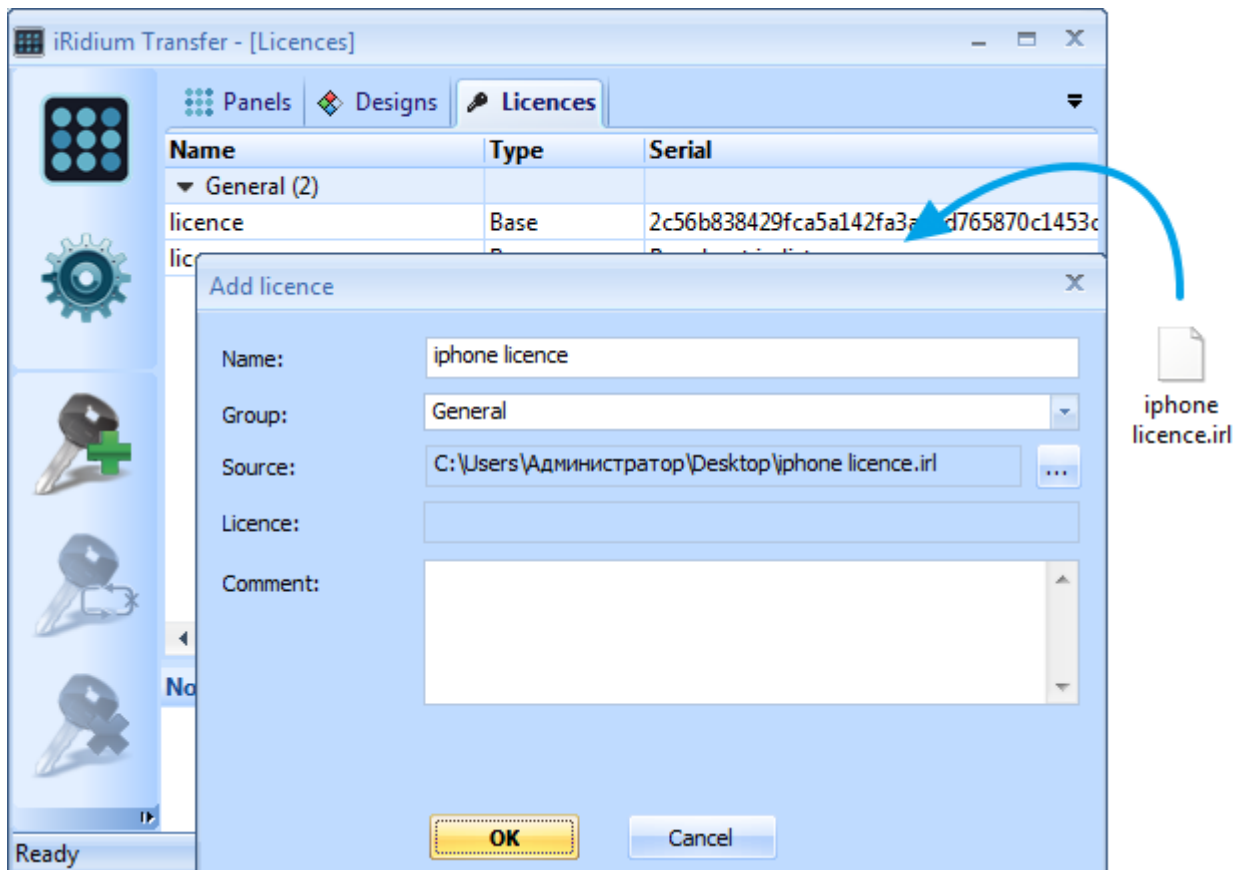
The next step will be receiving of your License File.

You can directly download it from the web site, right after activation:

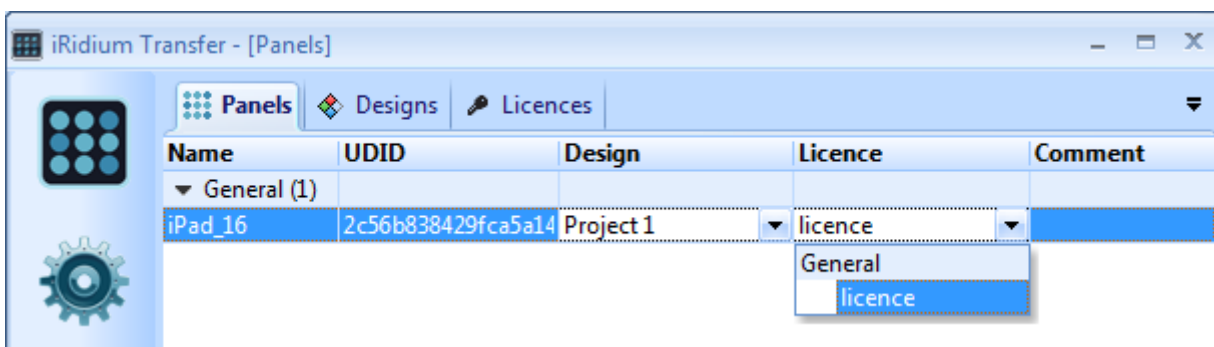


## Uploading of the License File for iOS to iRidium Transfer

The iRidium license is the file with the extension .irl received after the Key activation. After the license file has been downloaded it is necessary to add it into the list of licenses in the Transfer application (Do not forget that in Transfer you can add licenses for iOS devices only. Licenses for Windows devices are stored separately). Addition of the license file is done by the *Add License* button or by dragging and dropping it into the Transfer application window:



After the file has been added to it can be attached to the iOS device in the *Panels* tab. Do not forget that in the *Licences* tab you can add licenses for iOS devices only. You can bind only that license file to the device (in the *Panels* tab) which was activated for that particular panel.



When the design and license are ready they can be uploaded on an iOS panel from Transfer.

## Setting up of iOS “iRidium for KNX” Client

Indication of parameters of the connection to the controller and updating of the design is done in the Settings window of the “iRidium for KNX” Client application which can be accessed from the general Settings window of your iOS device:



### **Update:**

*GUI update* – enabling/disabling design update at restart of the “iRidium for KNX” application

*Update host* - the IP-address of the PC where the Transfer application is running

*Update port* – the design update port (leave by default)

### **Search (by default):**

*Server port* – a server update port

*Client port* - a Client update port

### **Version:**

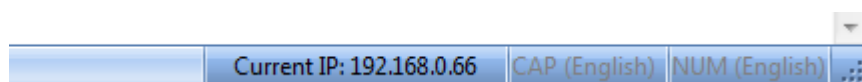
*Version*- a version of the “iRidium for KNX” application. The number of Client version should be the same as the number of the iRidium version on PC

### **Other:**

*View* – GUI orientation (vertical, horizontal, screen auto-rotation)

*Don't sleep* - enabling/disabling the “Don't sleep” mode at lengthy standby of the running Client

You can find the local IP- address of the computer in the right lower corner of Transfer:



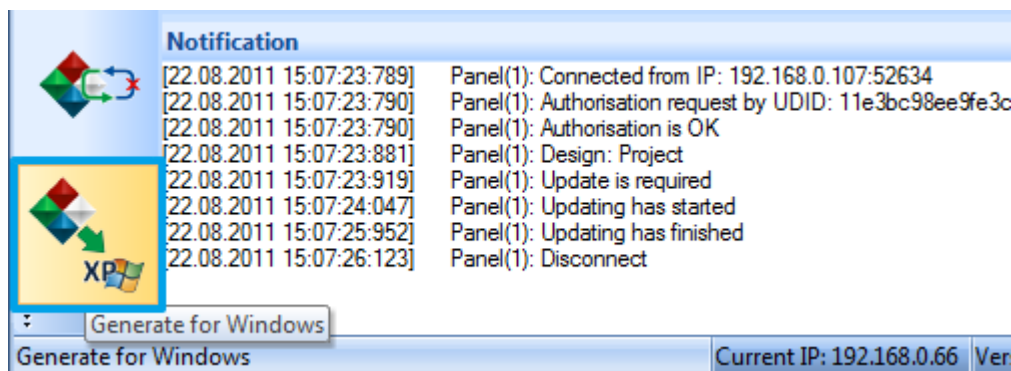
GUI update can be done via either local network or the Internet.

When all the settings are completed – all that is required is to restart the “iRidium for KNX” Client application on your iOS device in order to start the update. Please note that for devices that support multitasking, you need to fully close the software, as minimizing – maximizing of the software will not work. In order to do a full close of “iRidium for KNX” Client, you need to fully delete it from the list of running applications, as it is shown in our [instruction at Wiki](#)

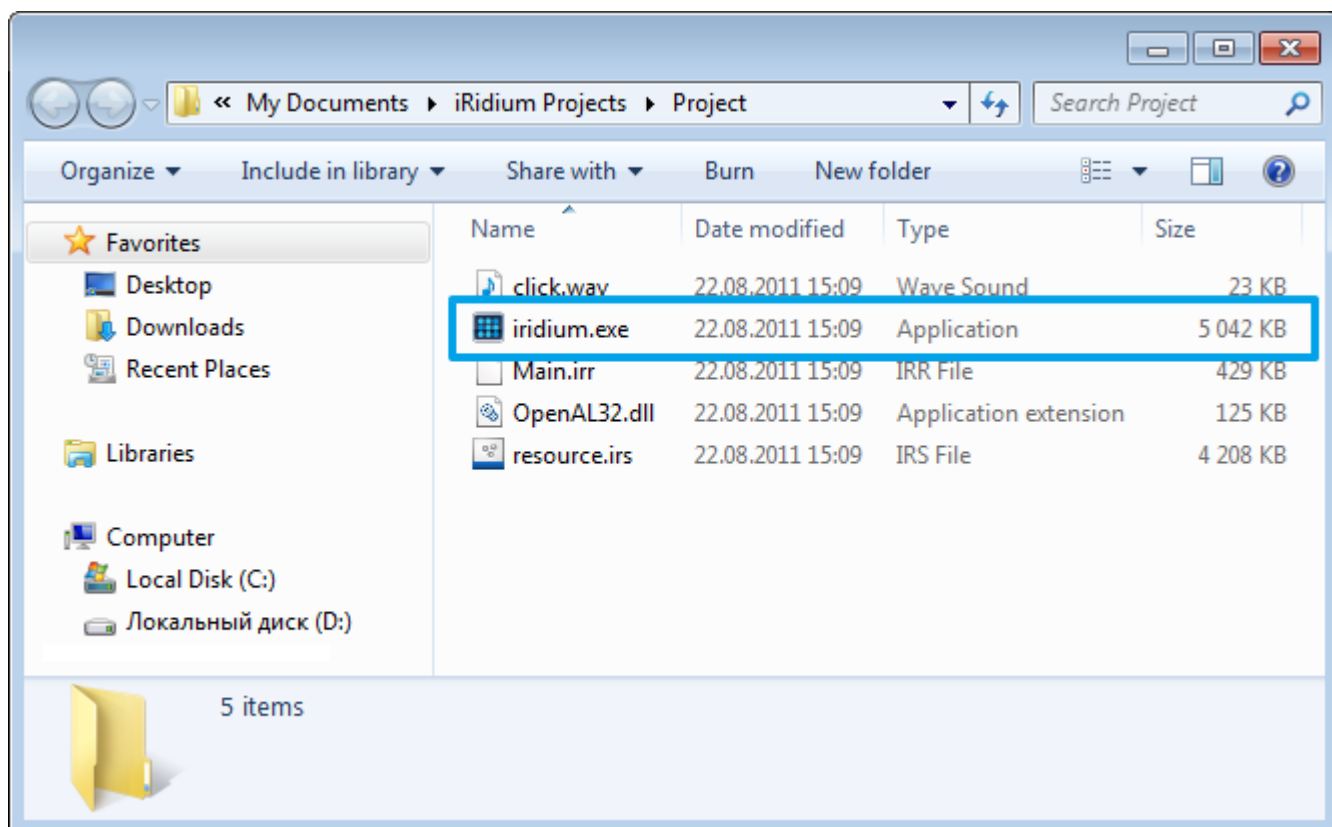
Opening of the configured application will start the update process from Transfer. After it's completed your design should be fully functional and you can disable GUI update ability in Settings.

## Creation of iRidium Windows-Client

Now let's create **iRidium Windows based Client** – an application file for launching under Windows XP/7. The design added into the Transfer application, and intended to be launched under Windows, is necessary to be selected in the list of designs in Transfer. Then click the *Generate for Windows* button in the *Designs* tab:



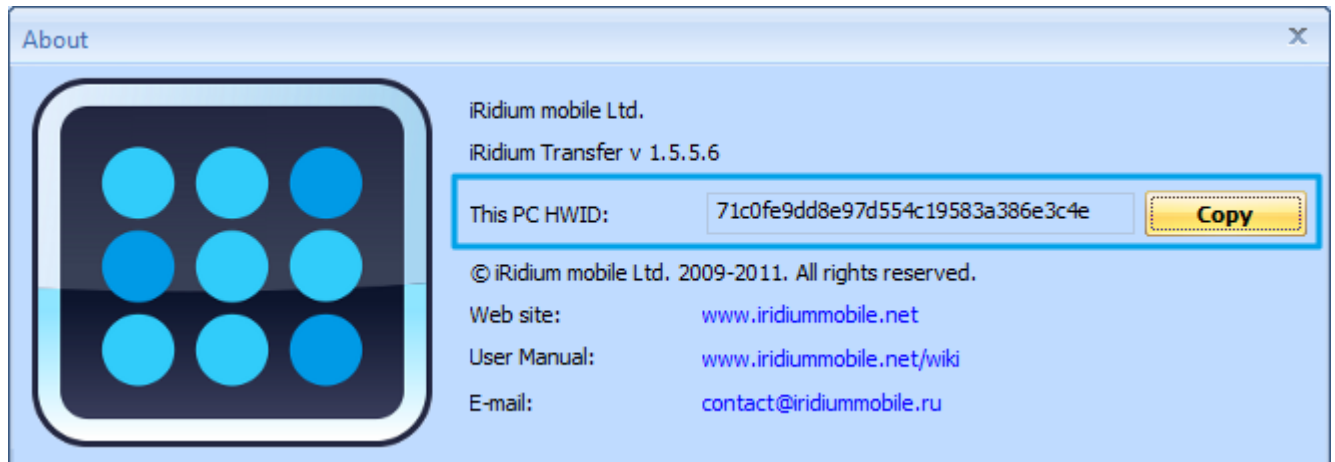
You will receive a file folder with the Windows Client application and additional files:



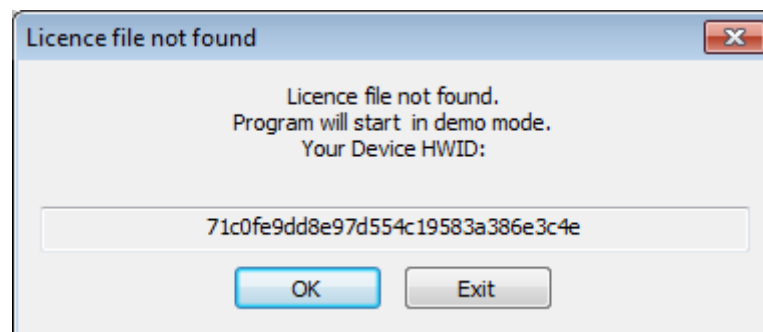
Now in order to make the Client application to be fully functional it is necessary to add a license file for Windows into the same folder. Receipt of a license file is done exactly in the same way as for iOS devices described above, except for the process of receiving the unique identifier for the panel.

## Receipt of HWID for Licensing of iRidium Windows-Client

For Windows devices a unique identifier which is required for license file receipt is **HWID**. It's attached to the computer's hard disk drive. You can find HWID of your computer where the Transfer application is installed through the *About* menu:



Or if you open any iRidium project (for example an iRidium demo-project without a license) on the control device you will be able to receive HWID of your device from a warning popup window:



After the license file has been received and added into the generated project folder the Client application can be opened. If you don't have any warnings when you are opening the Windows based Client, it means your license was accepted and your project will run in a fully functional mode.

Additional information about operation with the iRidium software package can be found at our Wiki.: <http://iridiummobile.net/wiki/>