iRidium for TECO

Fast Start: Connection Setting Manual for TECO Controllers

Review of iRidium Software Package for TECO:

iRidium turns your iPhone/iPod/iPad or Windows device into a TECO 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 TECO projects. Then you can quickly bind them to control items (buttons, lists, levels, etc.) by the Drag&Drop method.

iRidium software package for TECO includes several components:

iRidium GUI Editor – a Graphic User Interface Editor, which allows you to create your own interfaces and set connection to the TECO 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.

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

In order to enable the connection of iOS iRidiumHD Client to TECO 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

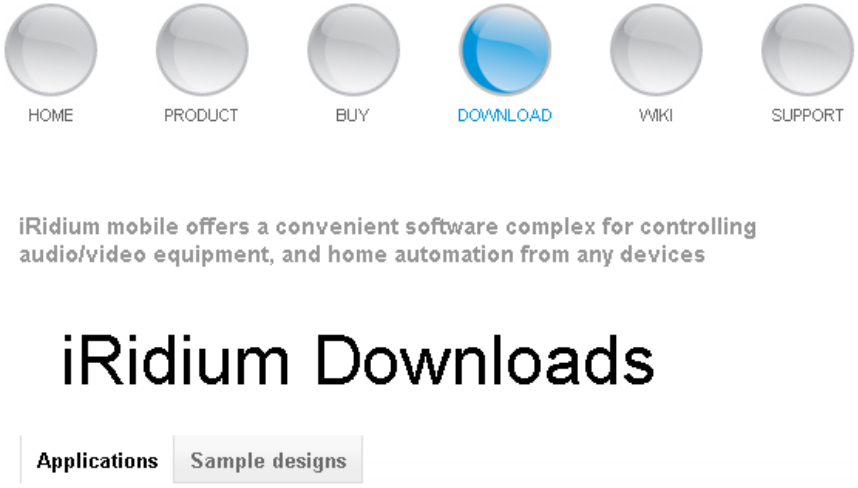
- PC under Microsoft Windows XP / 7
- iRidiumHD application installed on your iOS device from the AppStore
- Wi-Fi network for uploading and updating GUIs on iOS devices
- Internet connection for purchasing and activation of iRidium license Keys
- iRidium License file for enabling the connection to the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

Content

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



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

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

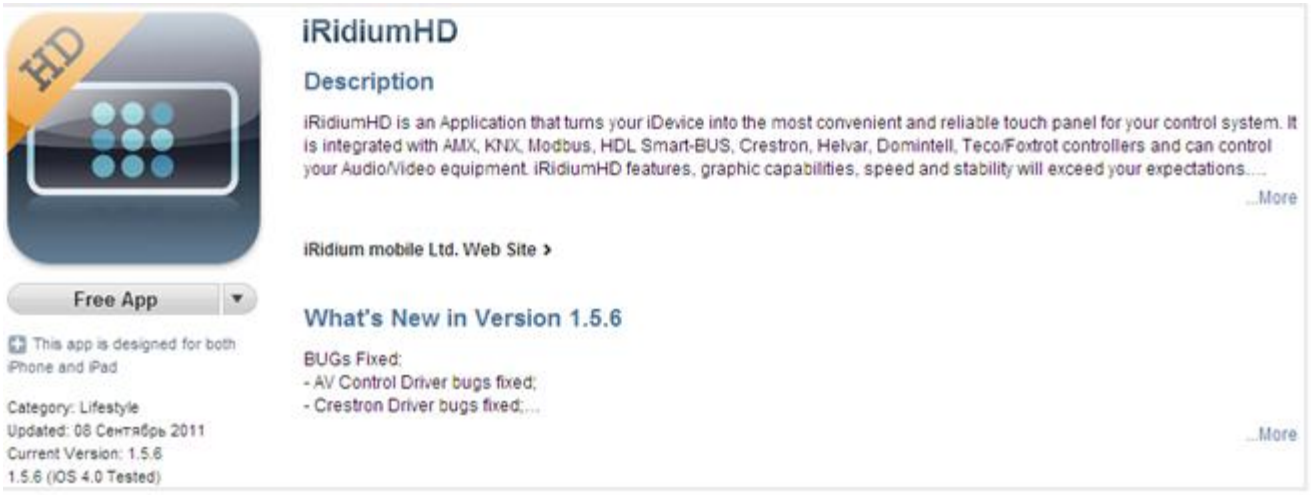
iRidium Transfer – an application for uploading graphic interfaces created in iRidium GUI Editor on iOS devices. iRidium Transfer allows you to convert interfaces into the format for their launching on Windows XP/7 devices. It can also check iRidium license availability for iOS devices.

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.

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 ***iRidiumHD*** and can be downloaded from the official *AppStore*.

iTunes Link: http://itunes.apple.com/us/app/iridiumhd/id369102319?mt=8

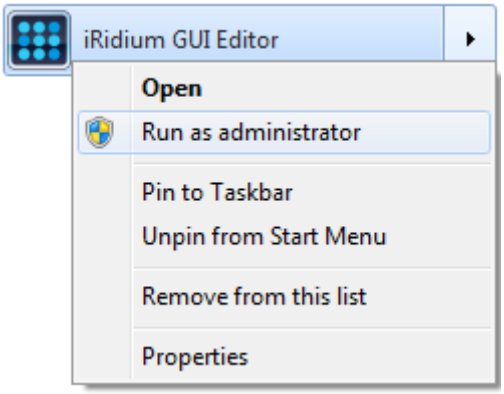
The iRidiumHD Client application is a software component of iRidium that is in charge of launching and using graphic interfaces, forming and sending commands to the controlled equipment.



Creating a GUI in iRidium GUI Editor

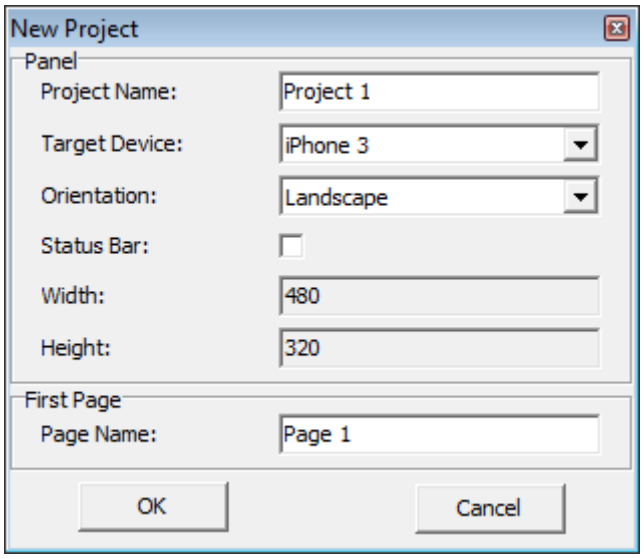
After the iRidium package for your PC and the iRidiumHD Client are installed, you can start creating a graphic interface to control the TECO equipment. Start iRidium GUI Editor on your PC.

! 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.



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.

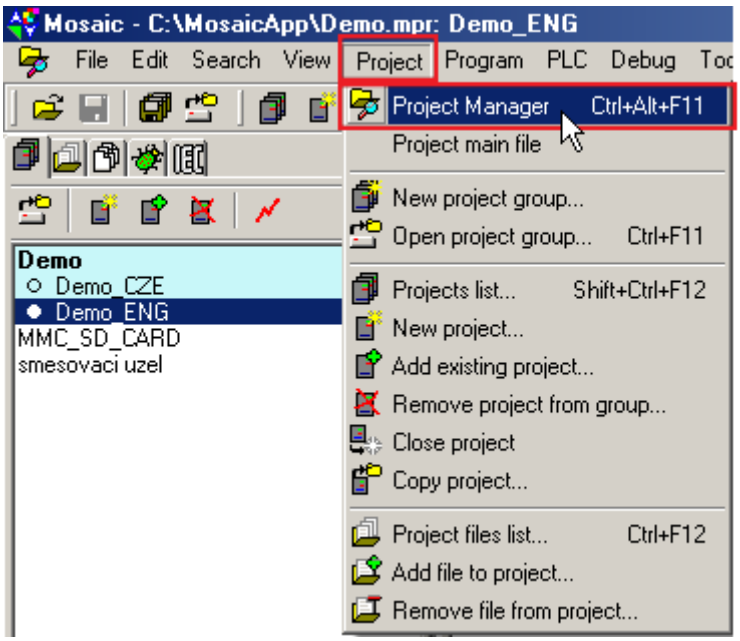


Preparation of TECO for operation with iRidium

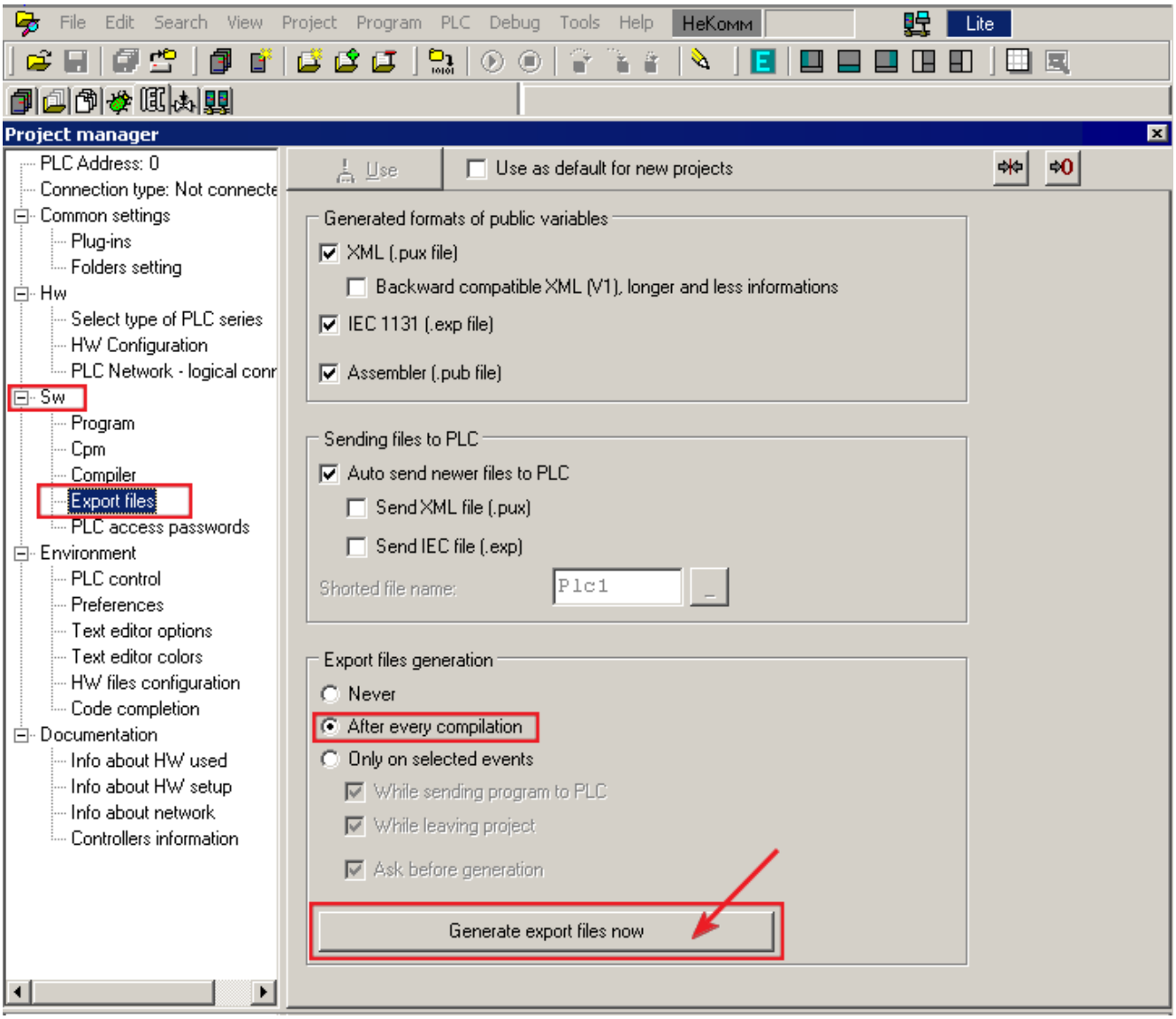
Setting of operation with TECO consists in creation and uploading of a project on the controller from the Mosaic development environment. To import Mosaic channels into iRidium GUI Editor you need to generate a project into a *.pub file in the Mosaic application. This file enables the import of channels into iRidium GUI Editor.

To generate the *.pub file it is necessary to do the following:

Launch the Mosaic application and open the project you need to make a GUI for. Go to the menu: *Project → Project Manager*:



Select *Sw → Export files* and indicate an event which the file to be generated at. In this case select *After every compilation*. Compile the project and press the *Generate export files now* button for the immediate generating of the **.pub* file.

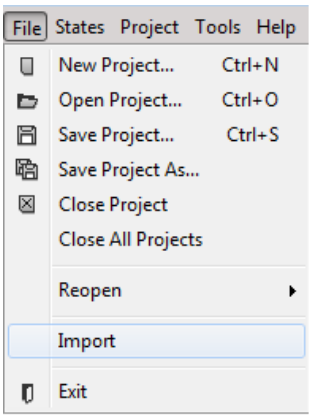


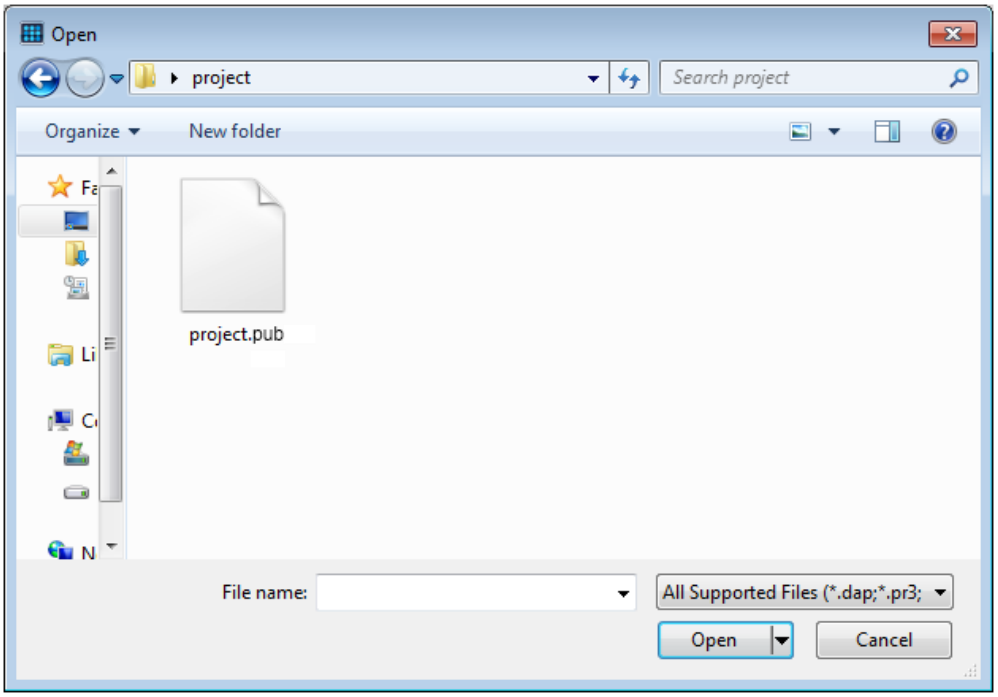
After the file is generated it can be imported into iRidium GUI Editor

Important: ***If you already generated the *.pub file for this project before, delete the old file and create a new one to avoid channel duplication at the import.***

Import of TECO Channels

It is necessary to import channel data from a project file (**.pub*) to bind channels to GUI items. You can use a special tool for importing channels 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 a TECO project file which contains channel data (**.pub*)
4. Import channels into iRidium GUI Editor
5. Check and edit (if it is necessary) channel settings in iRidium GUI Editor according to channels description.

After the import is completed, indicate the settings of connection to the controller and drag channels to graphic items.

Setting of Connection to TECO

Connection to TECO through UDP

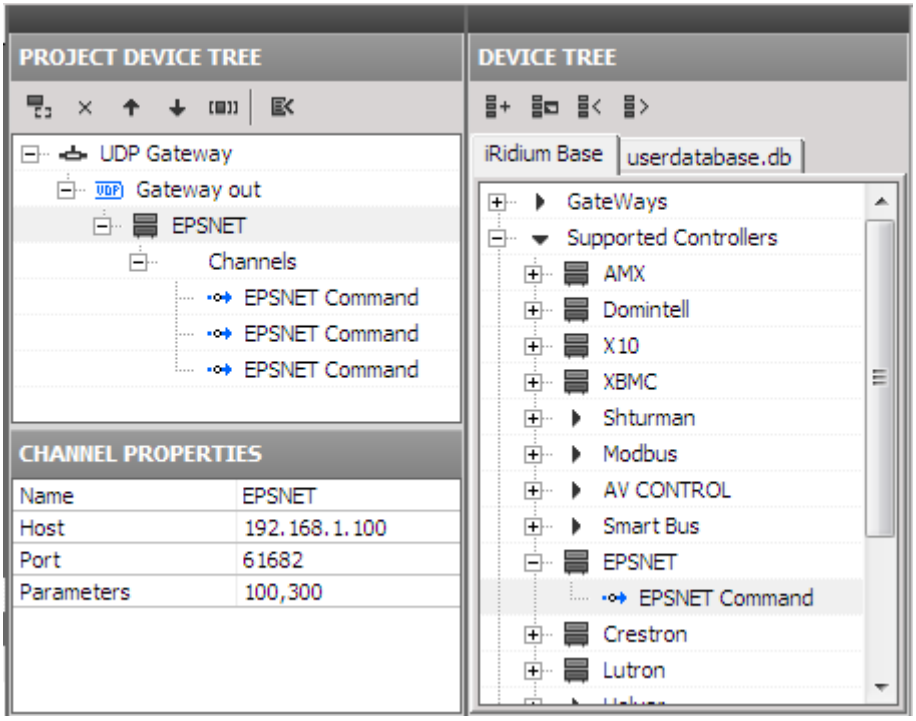
Setting of connection to the TECO controller through UDP protocol:

Import a device controlled by the EPSNET protocol from iRidium Base to Project Device Tree by the Drag&Drop method. Then drag the required number of channels to it.

Connection properties:

- *UDP Gateway* - a transport type, UDP
- *EPSNET* - a protocol of the TECO controller
- *EPSNET* Command – a TECO channel (command)
-

Project Device Tree looks as follows:



Properties of connection to the controller:

- *Name* – a device name
- *Host* - an IP-address of the TECO controller
- *Port* – a port number(it should correspond to the port in the Mosaic project, by default - 61682)
- *Parameters* – identifier (device ID – a number in the range from o to 255) and polling frequency (in ms) separated by comma.

Setting of Channels for Communication with the TECO Controller

After finishing connection setting you can proceed to setting properties for project channels. The TECO controller 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	off
Type	Boolean
Property	Value
Direction	Write
Action	Press
Register	R
Address	18354
Bit	2
Value	1

- ***Name*** – a name of the memory cell which should be communicated.
- ***Type*** – a signal type.
  - *Boolean* (1/0) - for Buttons
  - *Float 32-bit* (information) – for Levels
- ***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* – it is not used for sending data. It receives and displays data (receives all changes of a channel state and displays them).
  - *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.
- ***Register*** – an array address in the register (it is formed while setting the controller).
- ***Address*** – a number of the start byte of the selected EPSNET channel (it should be unique).
- ***Bit*** – a number of the controlled bit in the start byte (0, 1 or 2).
- ***Value*** – a set variable value (0 or 1).

Principles of Controlling Dimmers in the TECO Controller

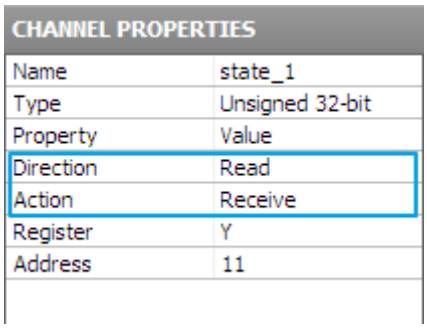
Admissible values for a dimmer are in the range from 0 to 100. Dimmers can be controlled and uncontrolled. Dimmer variables can be controlled in a few ways:

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

Uncontrolled Dimmers:

***IMPORTANT:* Such dimmers can only read data - «Read» (value output on graphic items). Other directions of data transfer («Write» and «Read/Write») cannot be used!**

Indicate *Direction: Read*. The rest settings stay the same after import:



CHANNEL PROPERTIES	
Name	state_1
Type	Unsigned 32-bit
Property	Value
Direction	Read
Action	Receive
Register	Y
Address	11

Name - a name of the memory cell which should be communicated.

Type – a variable type (it is set according to the controller manual (paperwork)). Select *«Float 32-bit»* (information) for Levels (sliders).

Direction - a direction of data transfer. Select *Read* to display the dimmer output state without possibility of changing it.

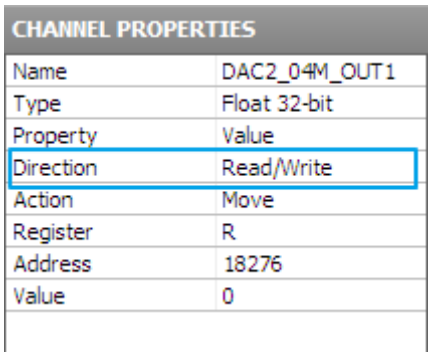
Action - a way of sending a command. Select “Receive”, as the channel is used for data sending and displaying only without sending values to the network.

Register – an array address in the register (it is formed while setting the controller).

Address – a number of the start byte of the selected EPSNET channel (it is indicated while setting the controller).

Controlled Dimmers:

Such channels are meant for creating a dimmer - changing the variable value in the given range. These channels are imported automatically. They are set and ready for operation without any additional settings.



CHANNEL PROPERTIES	
Name	DAC2_04M_OUT1
Type	Float 32-bit
Property	Value
Direction	Read/Write
Action	Move
Register	R
Address	18276
Value	0

Name - a name of the memory cell which should be communicated.

Type - a variable type (it is set according to the controller manual (paperwork)). Select *«Float 32-bit»* (information) for Levels (sliders).

Direction - a direction of 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* – sending a command when moving the slider bar.

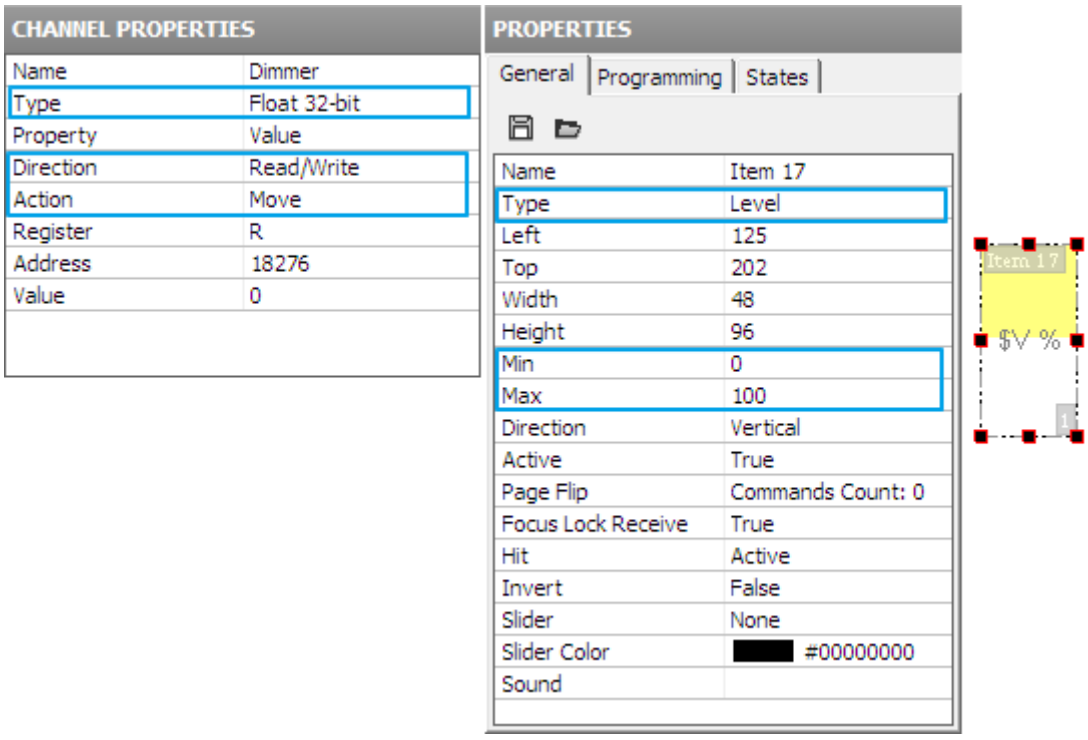
Register – an array address in the register (it is formed while setting the controller).

Address – a number of the start byte of the selected EPSNET channel (it is indicated while setting the controller).

Bit – a number of the used bit in the start byte (0, 1 or 2; it can be seen in the properties of channels imported from Mosaic).

Value - a set variable value (0 or 1).

Dimming by Level (slider)



Set Channel Properties as follows:

Type – a variable type (it is indicated at setting the controller). Select "Float *32 - 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).

Action – a way of sending a command. You can select *Move* – sending a command when moving the slider bar (it is recommended for Levels).

Register – an array address in the register (it is formed while setting the controller).

Address – a number of the start byte of the selected EPSNET channel (it can be seen in the properties of the imported channels).

Value – a dimmer level (from 0 to 100). As the value is set by Level, leave 0.

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.

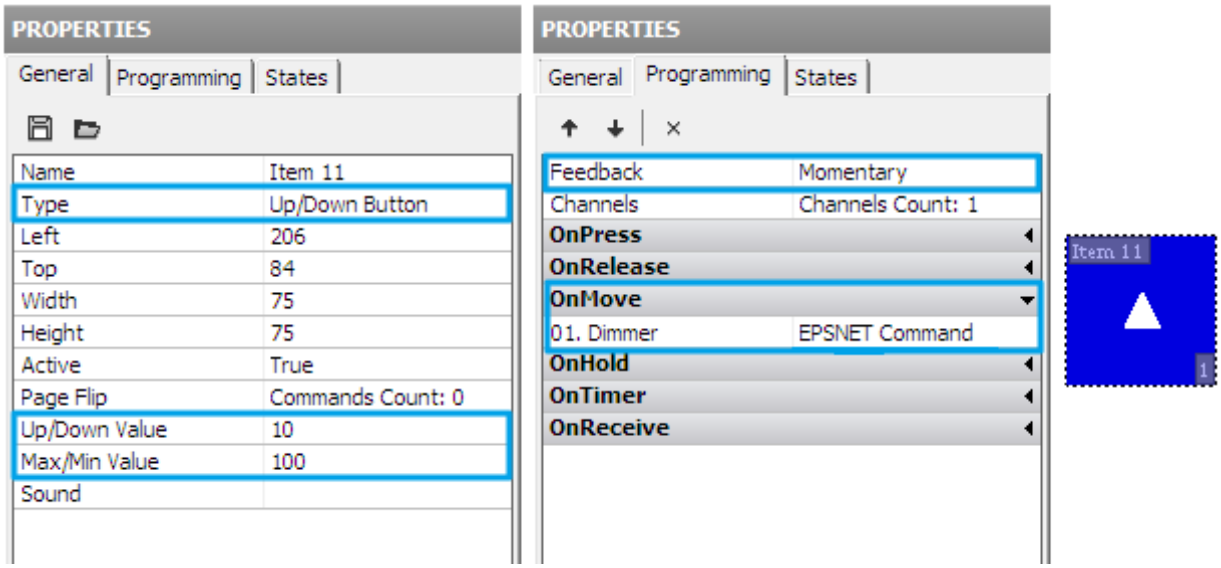
TECO dimmer values change in the range from 0 to 100. You can set a step of value change within the range and regulate it with two buttons. The same channel is bound to both buttons. Value change direction and change limits are set in the graphic item Properties where the "*Up/Down Button*" type is selected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

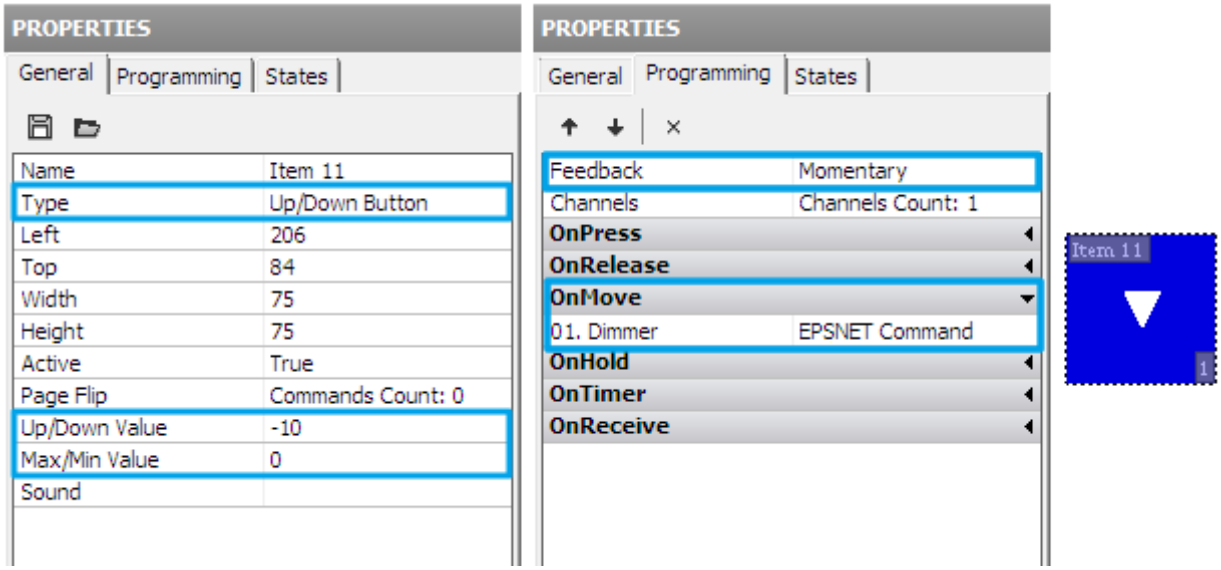
You don't need to create a separate channel for binding to the *Up/Down Button* type graphic items. You can bind to them the same channel which was set for dimming by *Level*.

CHANNEL PROPERTIES

Name	DAC2_04M_OUT4
Type	Float 32-bit
Property	Value
Direction	Read/Write
Action	Move
Register	R
Address	18525
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:





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».

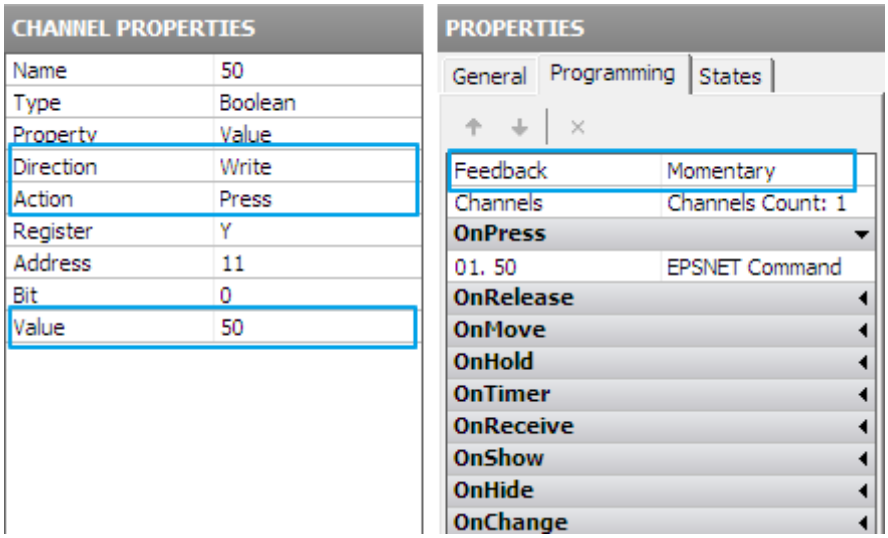
Max/Min Value – maximum or minimum values taken by a variable. For *Up Button* set dimmer maximum: 100, for *Down Button* set its minimum: 0. The limits have to correspond with the actual extreme values of the variable or can be less to make a cut-off.

The "Programming" tab:

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

Setting a Fixed Value

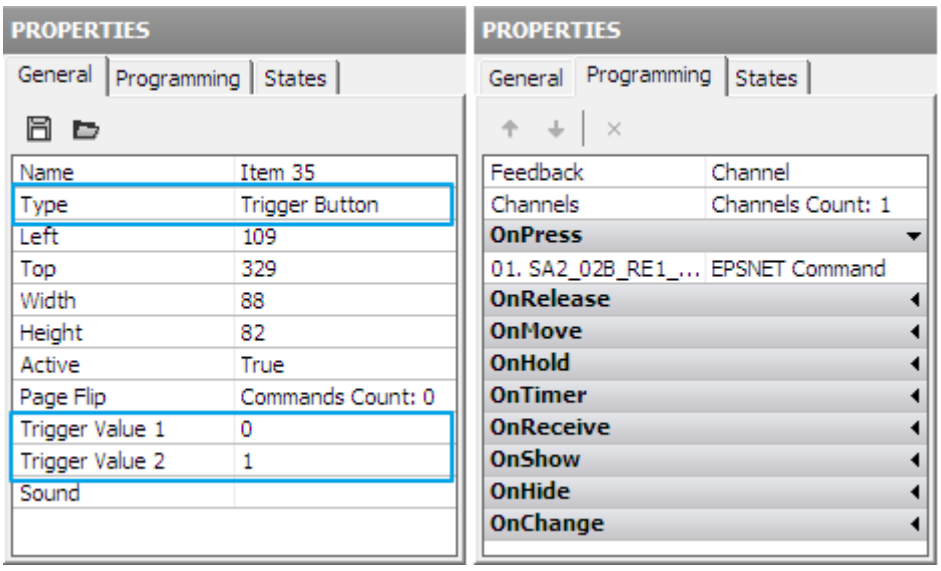
You can set the level value in the available range by *Button*. The channel, which setting is performed similarly to a dimmer or relay channel, is used for that. The only difference in setting is indication of the fixed Value (i.e. the value the variable has to take).



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

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.



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*.

Trigger Switch by TECO

A trigger switch by TECO controller means can be created in two ways depending on the number of a bit in the start byte: one trigger channel or two channels set for switching on/off with different ways of sending a command. You can see the required properties of the start byte which is used as a trigger in the controller manual (paperwork).

Example:

SA2_02B_RE1_ON R B 18731 .0 BOOL PUB_INOUT

SA2_02B_RE1_OFF R B 18731 .1 BOOL PUB_INOUT

SA2_02B_RE1_TRIG R B 18731 .2 BOOL PUB_INOUT

This extract from the paperwork presents 3 bits *(0, 1* and *2)* of one start byte *(18731). "ON", "OFF"* and *"TRIG"* indicate the command each bit is responsible for.

CHANNEL PROPERTIES	
Name	SA2_02M_RE2_Trig
Type	Boolean
Property	Value
Direction	Write
Action	Press
Register	R
Address	18731
Bit	2
Value	1

CHANNEL PROPERTIES	
Name	SA2_02M_RE2_ON
Type	Boolean
Property	Value
Direction	Write
Action	Press
Register	R
Address	18731
Bit	0
Value	1

CHANNEL PROPERTIES	
Name	SA2_02M_RE2_OFF
Type	Boolean
Property	Value
Direction	Write
Action	Release
Register	R
Address	18731
Bit	1
Value	1

When using a trigger channel the type of response (*Action*) can be *Press* or *Release*. When using two channels set for switching on/off the type of response can also be *Press* or *Release* but it should not be similar for both channels.

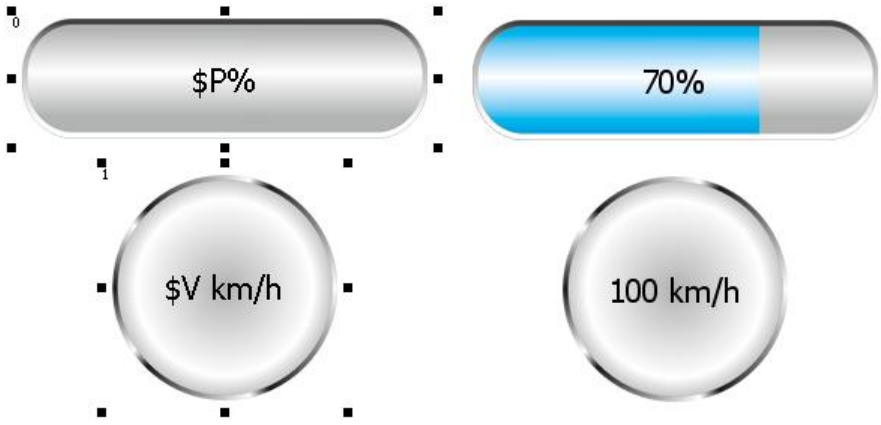
Output of temperature or channel variable values

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

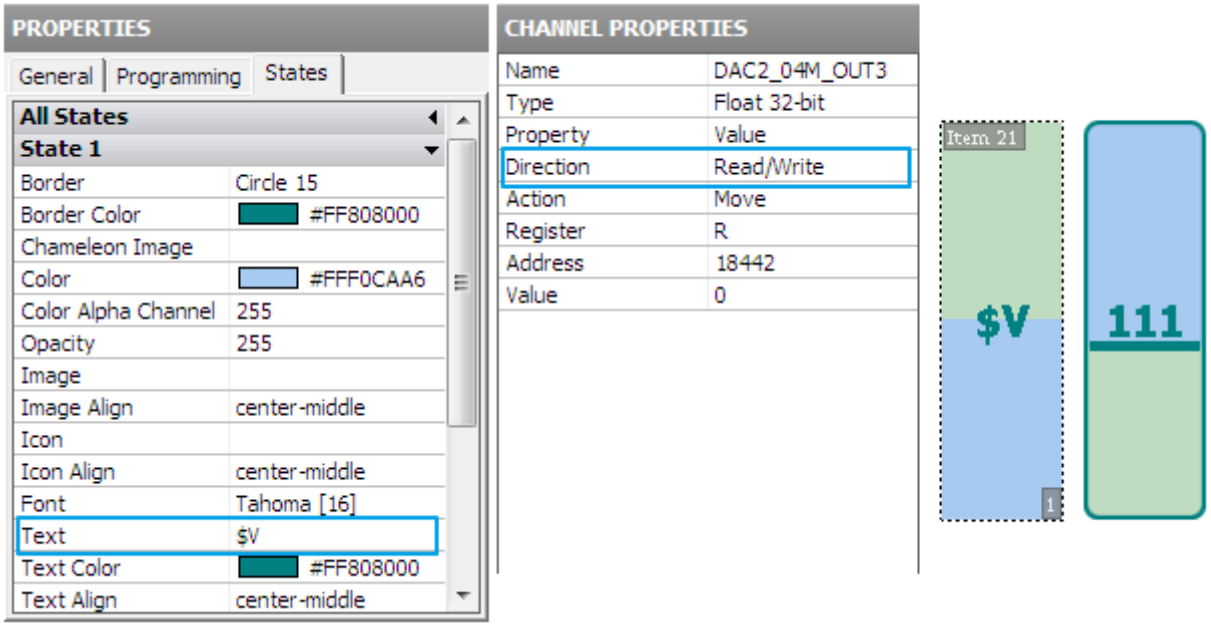
Let us give an example of one of the templates of value output to graphic items: the **$V km/h** template for a Level item with the Channel feedback, allows you to output a current speed value that is visualized by Level position and the changing number. Data on the current variable state can be received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

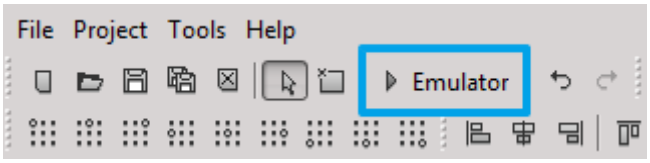


Uploading a Project on a Panel, Connection to Equipment

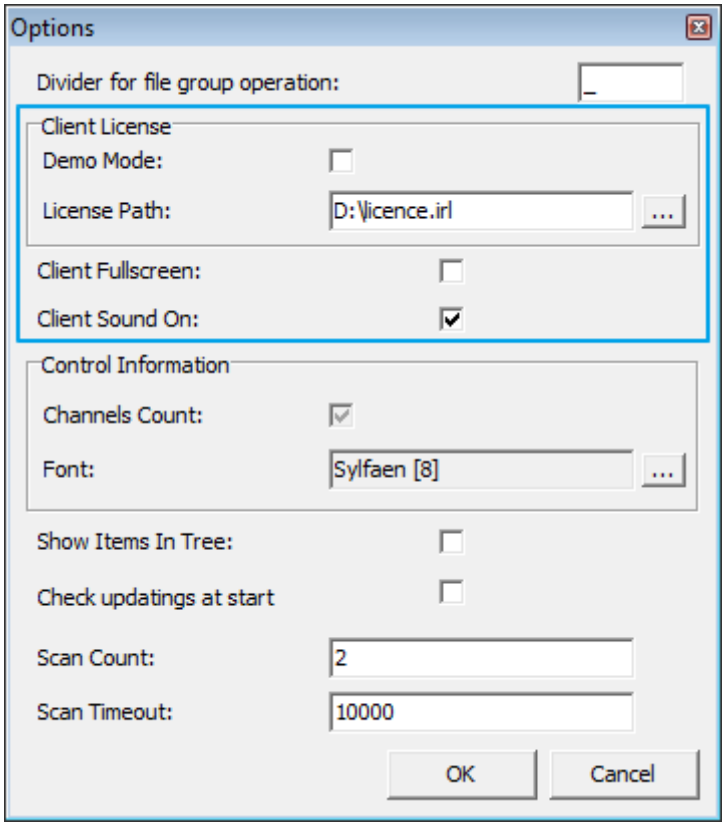
When GUI creation is completed it is necessary to check its operation and upload a ready GUI on an iOS or Windows control device.

The *Emulation Mode*

The Emulation mode allows you to check the proper operation of the project graphic interface and connection to the TECO 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):

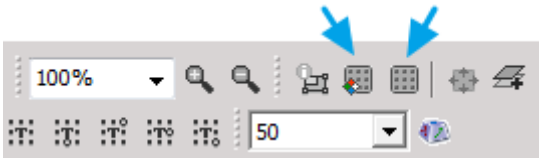


When there is a tick in the *Demo Mode* field, there is no connection to the controller and there will be no messages about operation in the demo mode at each start of Emulator. To operate in the fully functional mode indicate a path to the license which is activated for Windows PC where Emulator is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

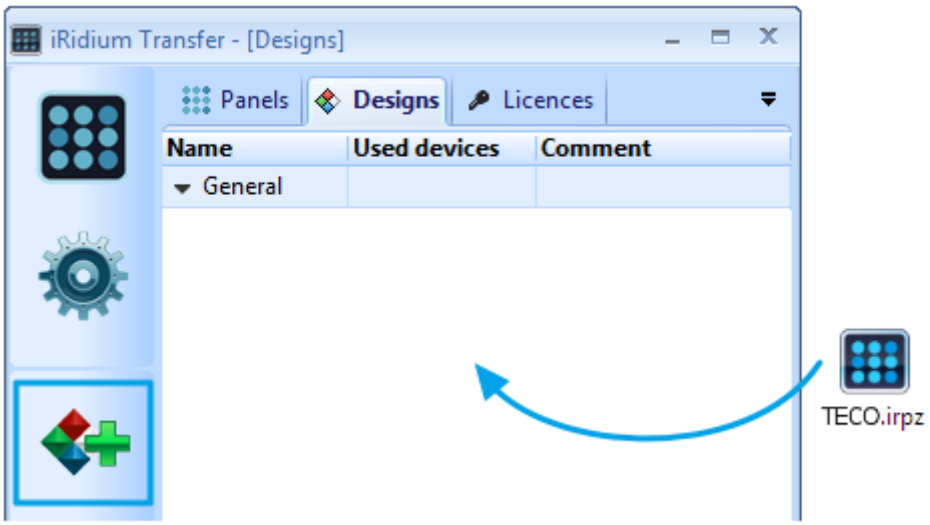


   a) The first type – the *Send to Transfer* button – opens the project in Transfer for the quick upload on an iOS device (iPad / iPhone / iPod Touch). The window for selecting the iOS device, which the design is to be bound, from the list of Panels in Transfer will open when transferring. After selecting 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 Widows target control device.

The upload of projects on iOS and Widows 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

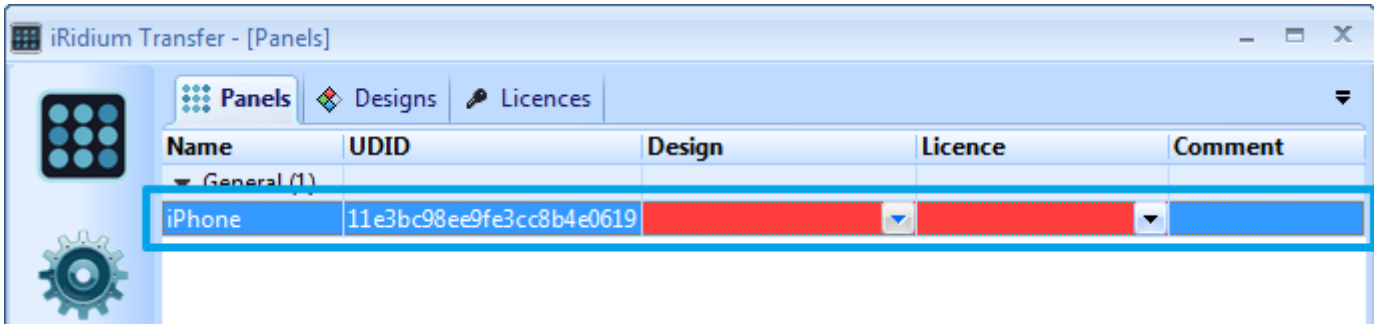
In order to upload a design on the control device based on Apple iOS operation system, it is necessary to establish connection between the iRidium Transfer application installed on your personal computer, and the Client application iRidiumHD installed on the iOS device.

Addition of iOS Devices to the Transfer Panels List

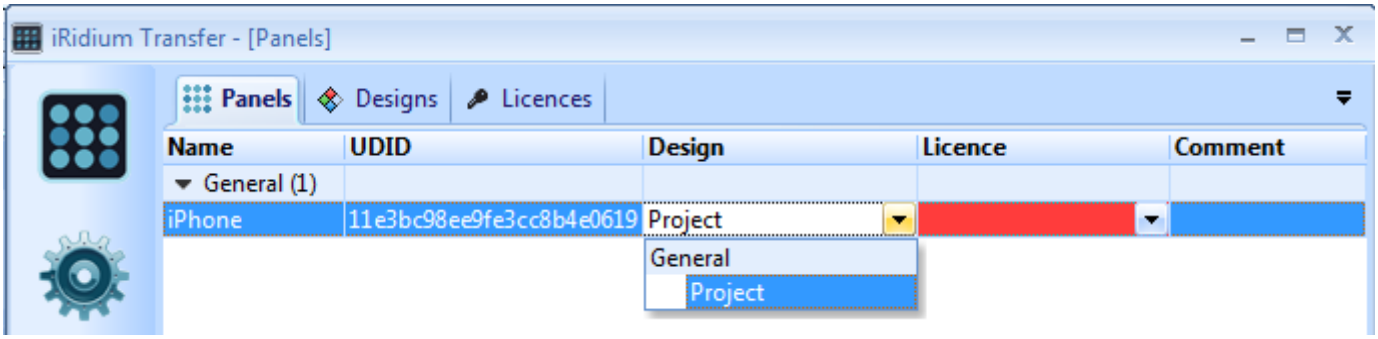
In order to establish the connection between Transfer and iRidiumHD, it is needed to add the control device with the installed iRidiumHD Client into the list of Panels in Transfer. Addition of the iOS based device into the list of Panels in Transfer is available through the Device Search option. For 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 iRidiumHD 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 iRidiumHD 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.



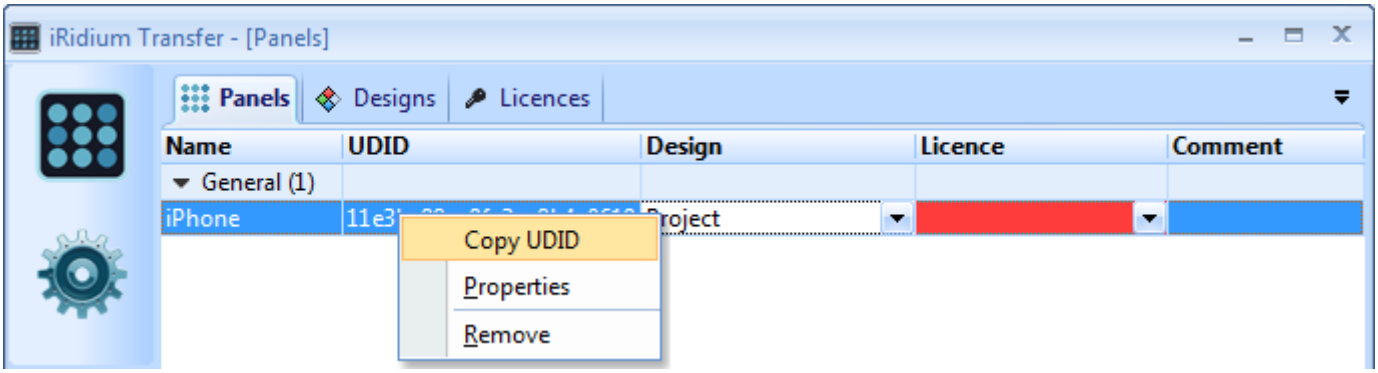
The second field is for the attaching the license file that will enable the graphic interface to connect to the controlled equipment. In case if a license file is absent, there will be no connection between iRidiumHD and the equipment, you will not be able to neither send control commands nor receive 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.

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:

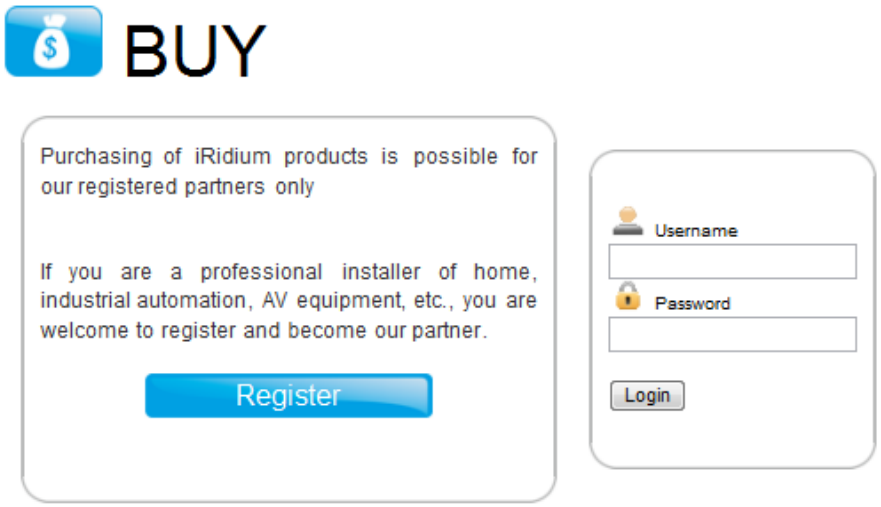


Besides, UDID of the iOS device can be copied from the iTunes application. It can be helpful when device is added manually or when you work via Internet:

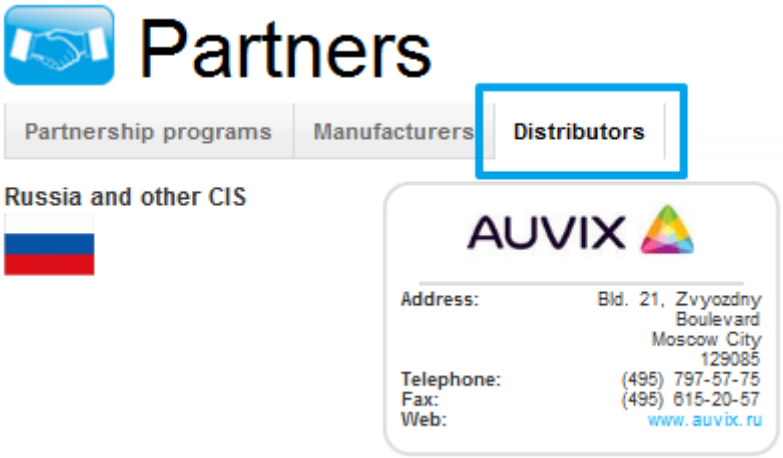
Licensing Procedure for an iOS Device

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



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



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

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** [illegible] using the following demo activation key for your:

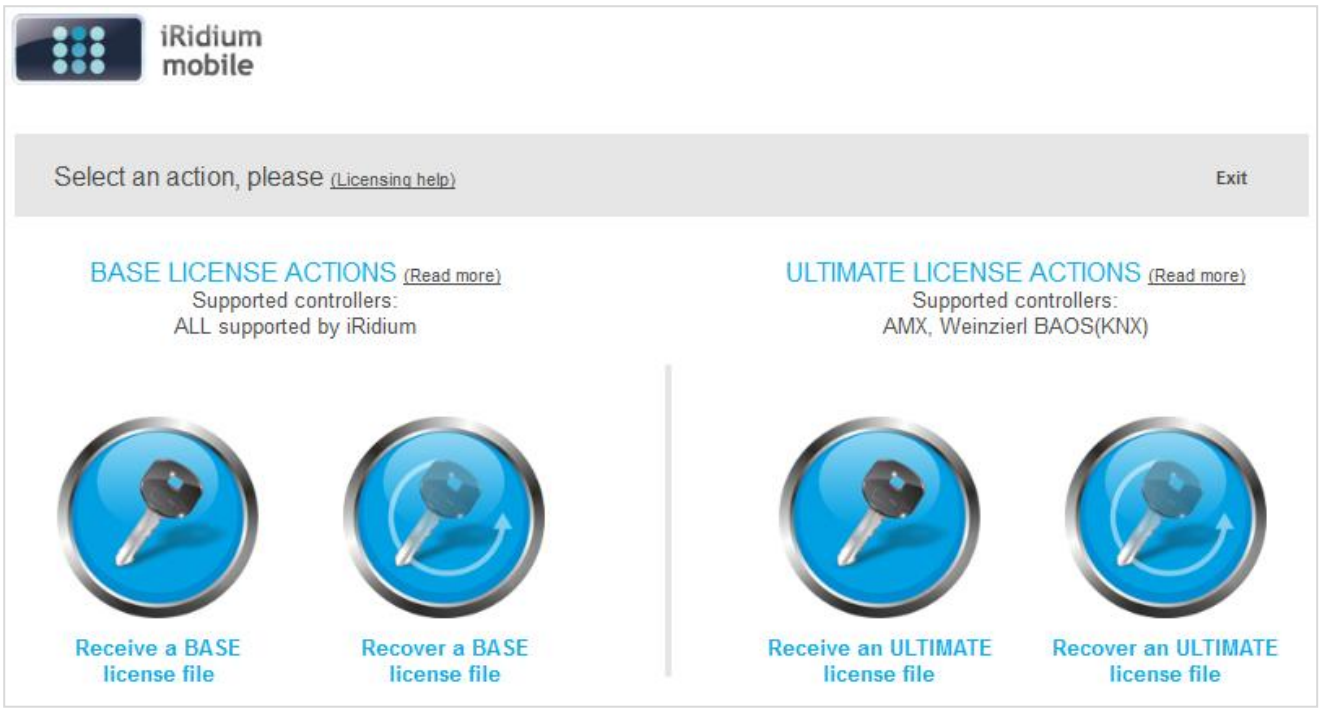
iPad - 532b38fbdf23e1392b5d0a0d0e184c7990225784cbe793b3c5b183f4395b5d0a4439060d472b453d6eae74e2eba7ee93

Activation of your Key is made at the following web site:

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

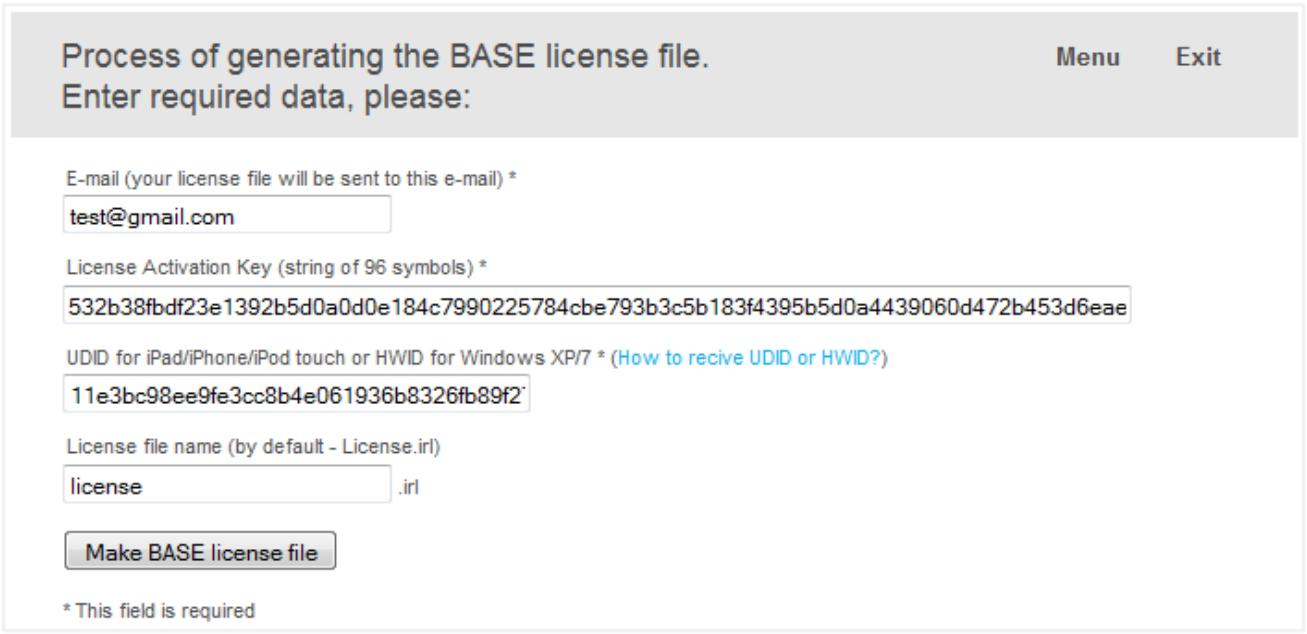
You will receive login information to access the activation web site in the e-mail.

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

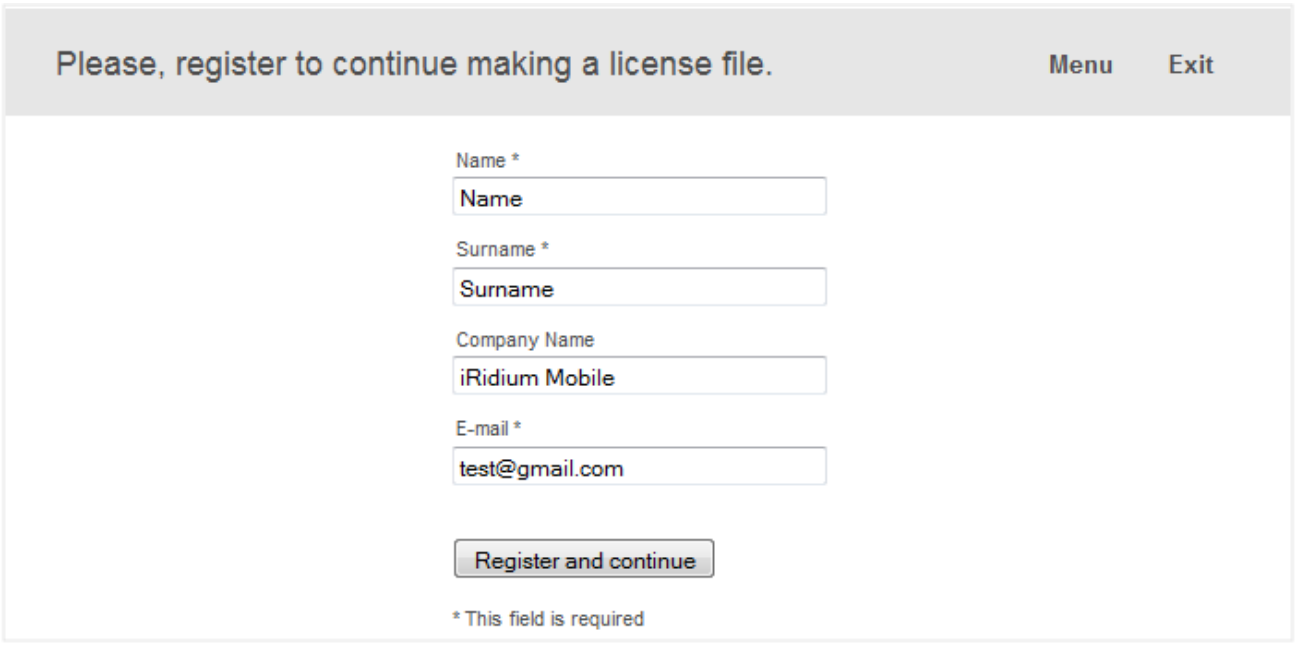


Activation of the iRidium License Key for an iOS Device

In order to activate the 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 identifier of the specific device (Device UDID) for which the license is generated:

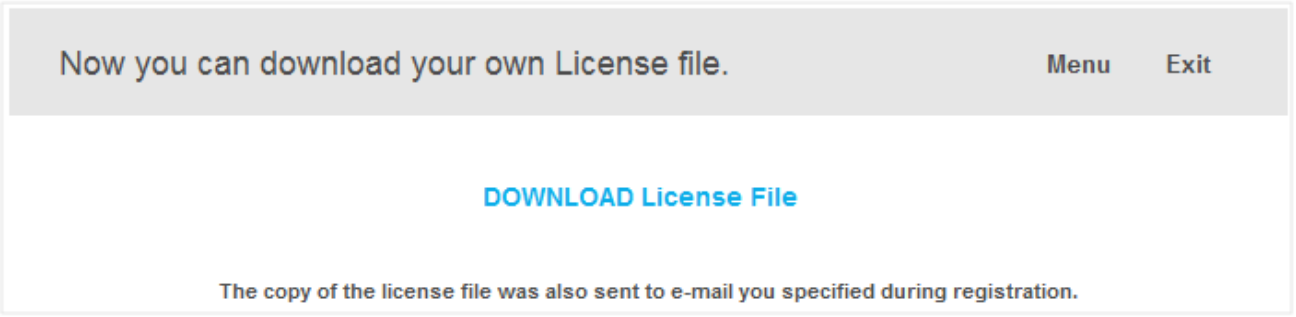


After completing all required fields in the Activation window, you will be asked to fill out a user form. This short form needs to be filled out only once and will be used in the future for activation of new licenses from the same e-mail address:

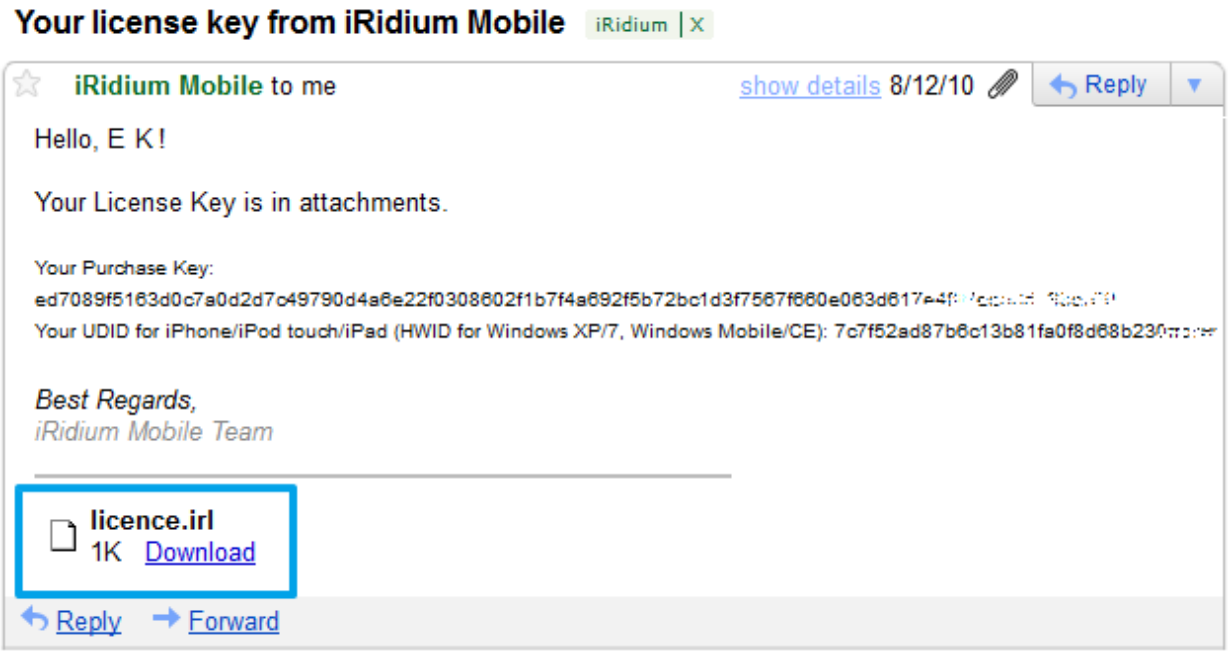


The next step will be receiving of your License File.

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

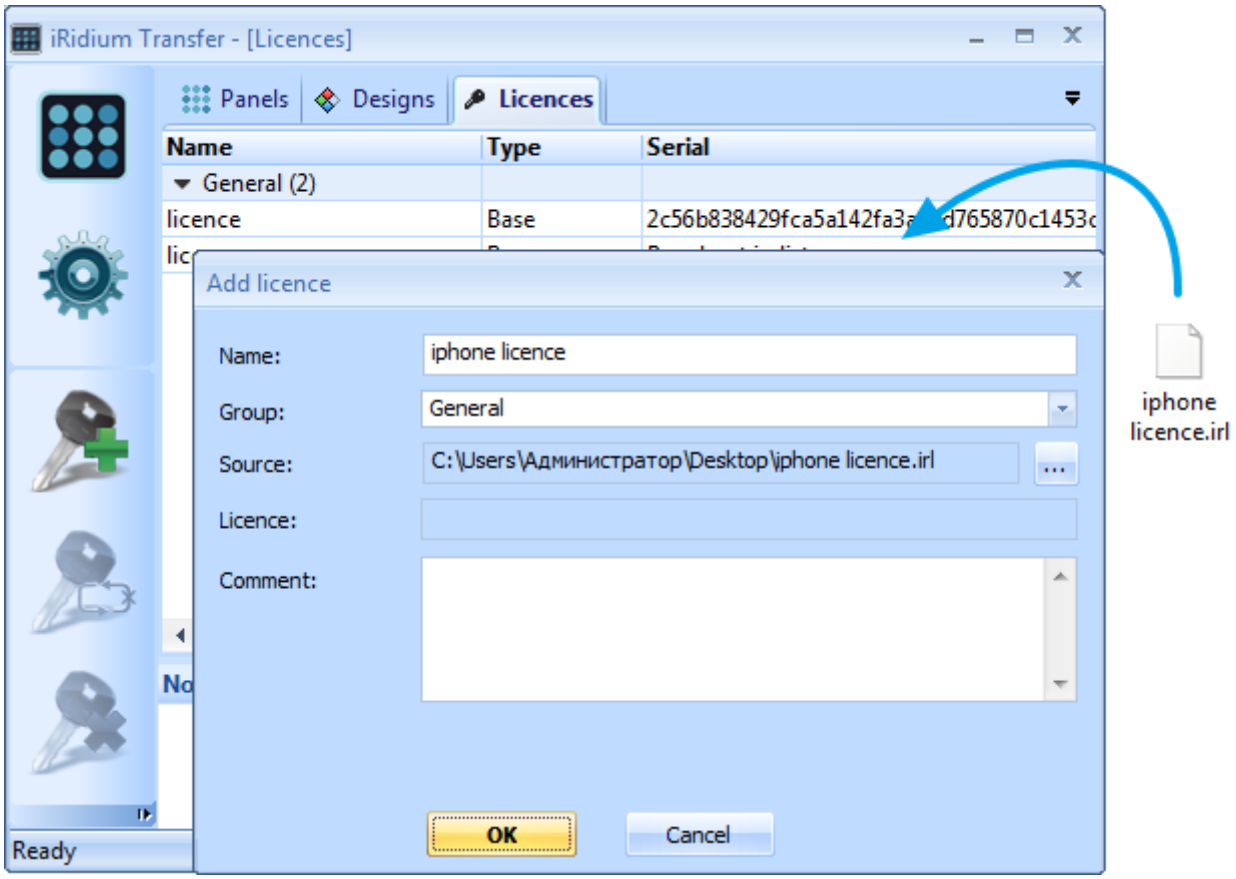


or it can be saved from the e-mail that will be automatically sent to the specified e-mail address.

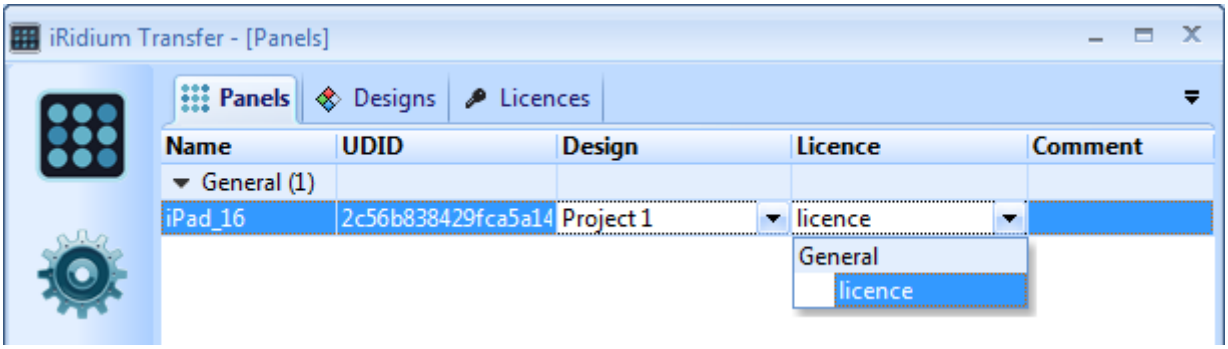


Uploading of the License File for iOS to iRidium Transfer

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:



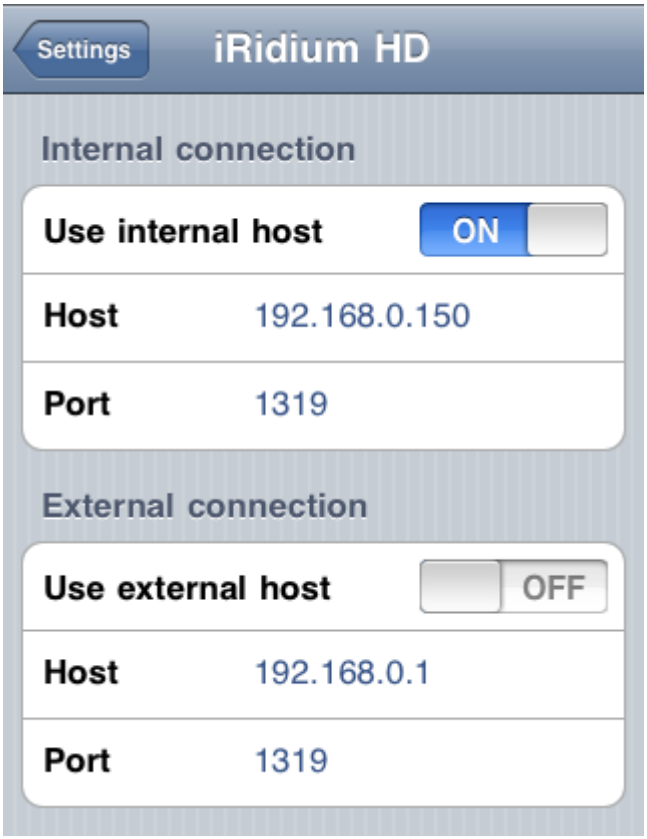
After the file has been added to it can be attached to the iOS device in the *Panels* tab. Do not forget that in the *Licences* tab you can add licenses for iOS devices only. You can bind only that license file to the device (in the *Panels* tab) which was activated for that particular panel.

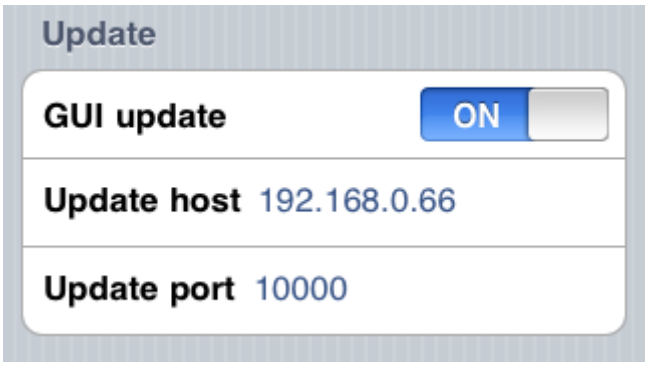


When the design and license are ready they can be uploaded on an iOS panel from Transfer.

Setting of iOS iRidiumHD Client

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





It is required to indicate in the iOS iRidiumHD Client Settings:

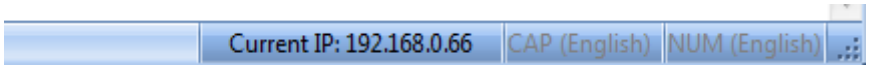
- **Type of connection to the controller** – either through the local network or through the Internet. Make one of the choices active.
- **Settings of design update on an iOS device** – indicate the required data in the Update section.

The rest connection settings are not necessary.

Further the update options are set:

Turn on the ability to process an update and indicate the **IP-address** of the PC where the Transfer application is running.

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



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

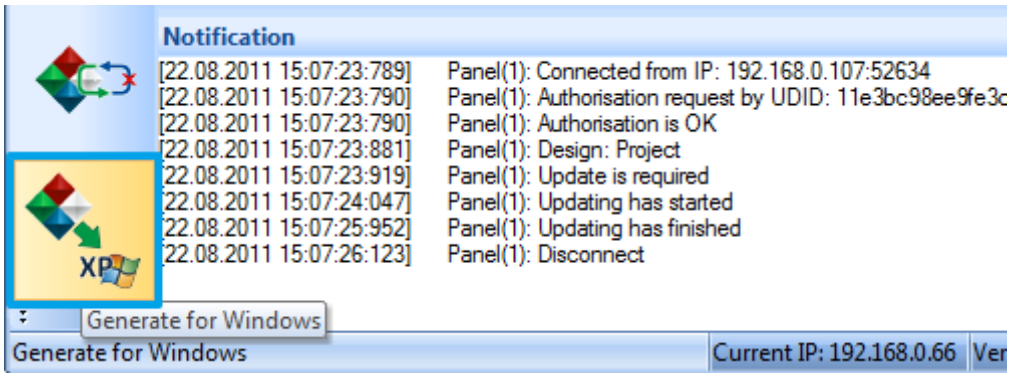
When all the settings are completed – all that is required is to restart iRidiumHD Client on your iOS device in order to start update. Please note that for devices that support multitasking, you need to fully close the software, as minimizing – maximizing of the software will not work. In order to do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

.

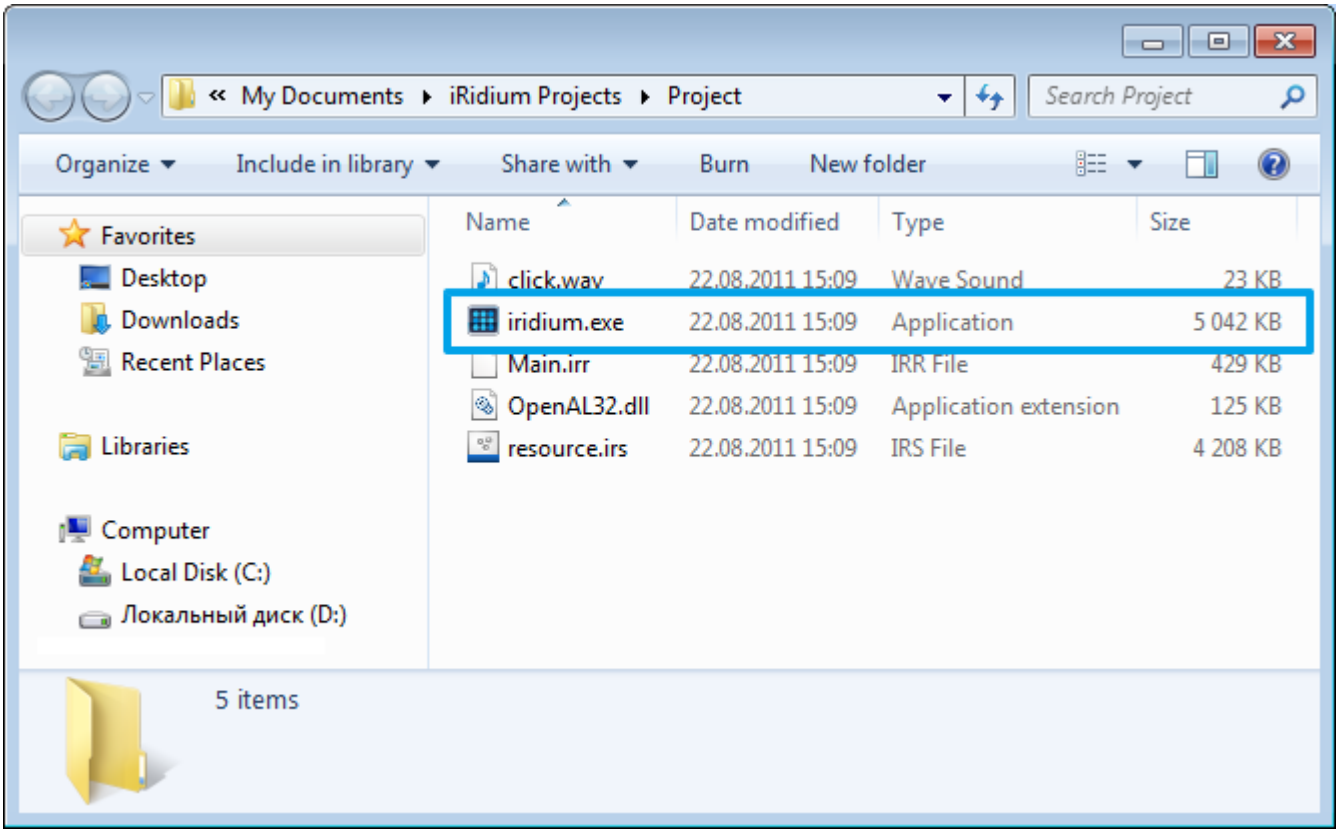
Opening of the configured application will start the update process from Transfer. After it's completed your design should be fully functional and you can disable GUI update ability in Settings.

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:



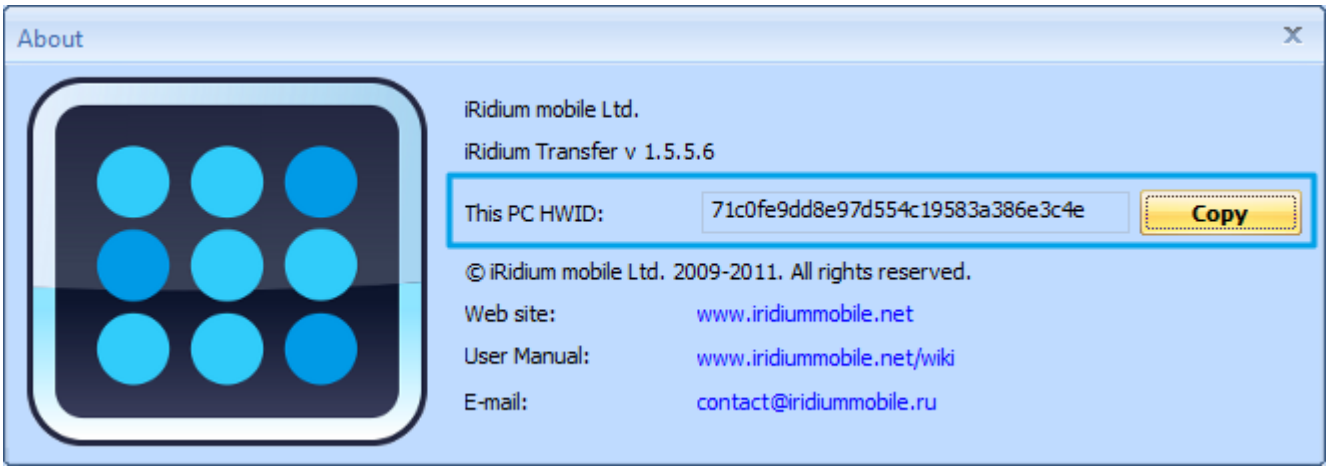
You will receive a file folder with the Windows Client application and additional files:



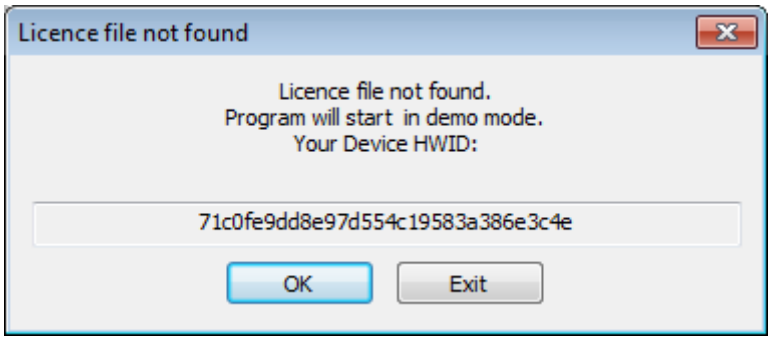
Now in order to make the Client application to be fully functional it is necessary to add a license file for Windows into the same folder. Receipt of a license file is done exactly in the same way as for iOS devices described above, except for the process of receiving the unique identifier for the panel.

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/