

[<< back \(AV & Custom Systems: Working with Equipment via UDP\)](#)

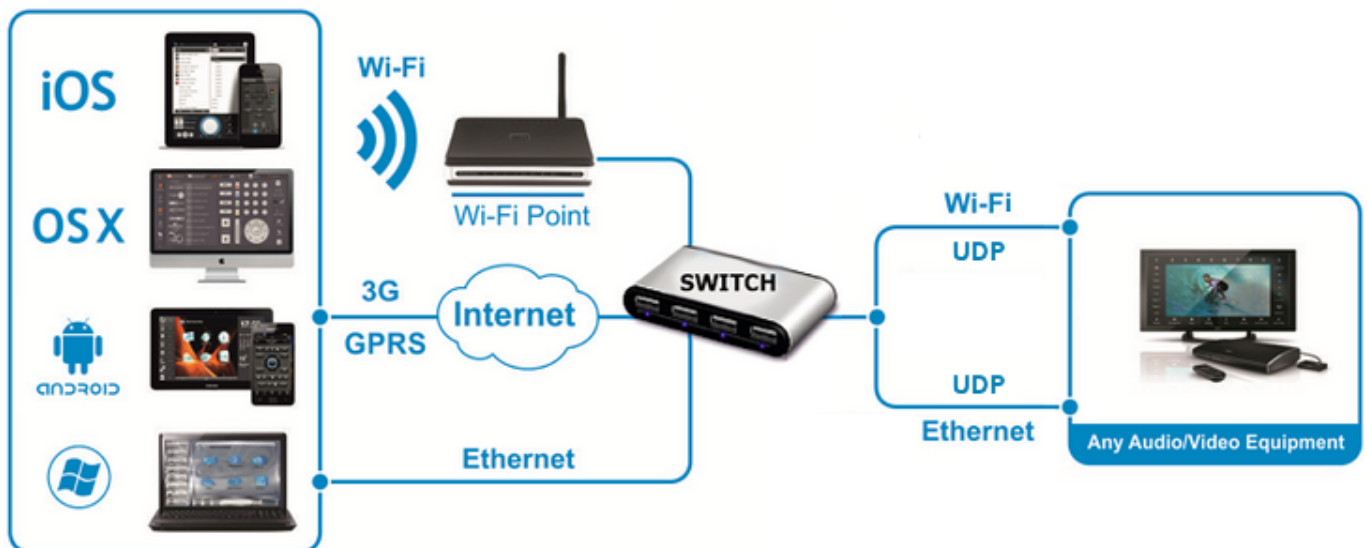
This article presents control of AV equipment or other systems via UDP through the example of the Kramer device with protocol 3000.

Contents

- [1 Principles of Controlling Equipment via UDP](#)
- [2 Adding UDP Commands to iRidium Data Base](#)
 - [2.1 Creating New Files of iRidium Data Base](#)
 - [2.2 Creating New UDP Devices](#)
 - [2.3 Setting Up UDP Devices](#)
 - [2.4 Syntax of UDP Commands](#)
 - [2.4.1 Sending Commands in the ASCII \(Strings\) Format](#)
 - [2.4.2 Sending Commands in the HEX \(Hexadecimal\) Format](#)
 - [2.4.3 Sending Commands in the DEC \(Decimal\) Format](#)
- [3 Adding UDP Devices to iRidium Projects](#)
- [4 Downloads](#)

Principles of Controlling Equipment via UDP

To control Smart Home systems via UDP commands (Ethernet or Wi-Fi) iRidium uses the freely customizable "AV & Custom Systems" (UDP) which enables iRidium to send commands via UDP. Thus the scheme of communication between an iRidium control panel and controlled equipment looks as follows:



Commands can be sent to UDP devices in the ASCII, HEX or DEC formats. When sending commands use special iRidium syntax to define the data format.

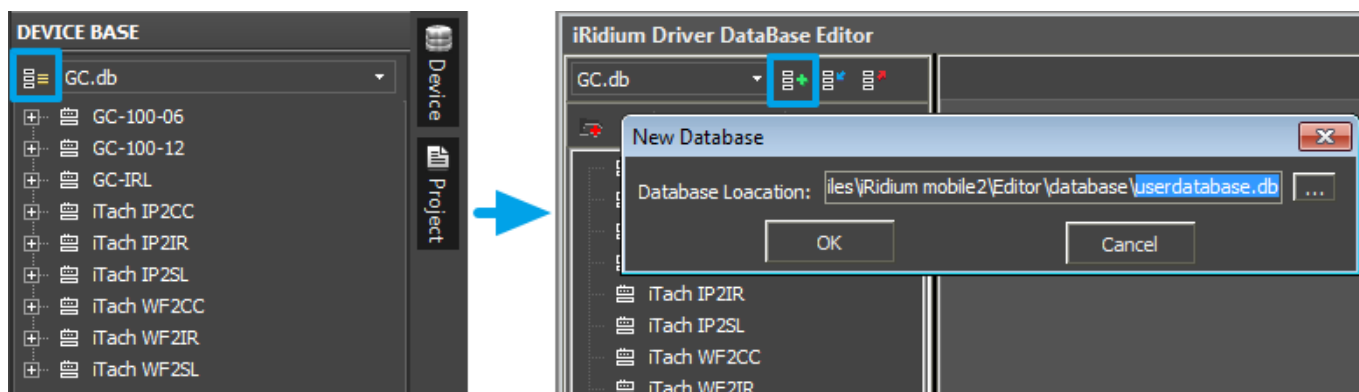
[↑ Back](#)

Adding UDP Commands to iRidium Data Base

To store UDP controlled devices with commands it is required to create their data base in [iRidium GUI Editor](#). It will enable you to use the devices in any iRidium project.

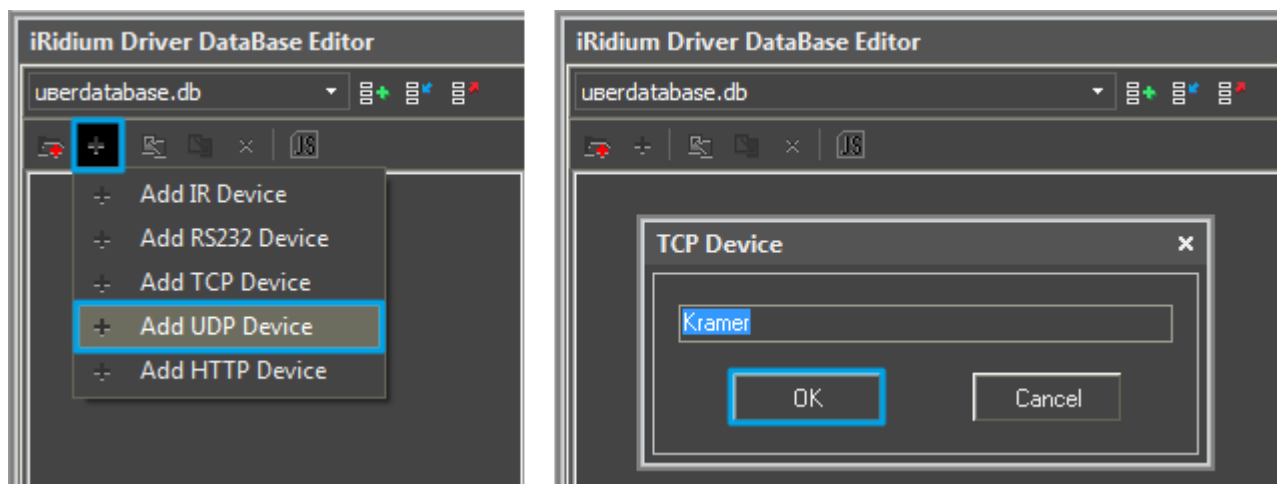
DB Editor is used for adding devices controlled via UDP and their commands to iRidium base. New devices can be added only to custom data bases as standard bases contain ready preconfigured modules of devices which cannot be edited (you can only take modules from the ready bases).

Creating New Files of iRidium Data Base



Use your own data bases (created manually) for storing your devices not standard data bases which can be updated with releases of new versions of iRidium GUI Editor. The names of the base file and folder for storing are indicated when creating the file.

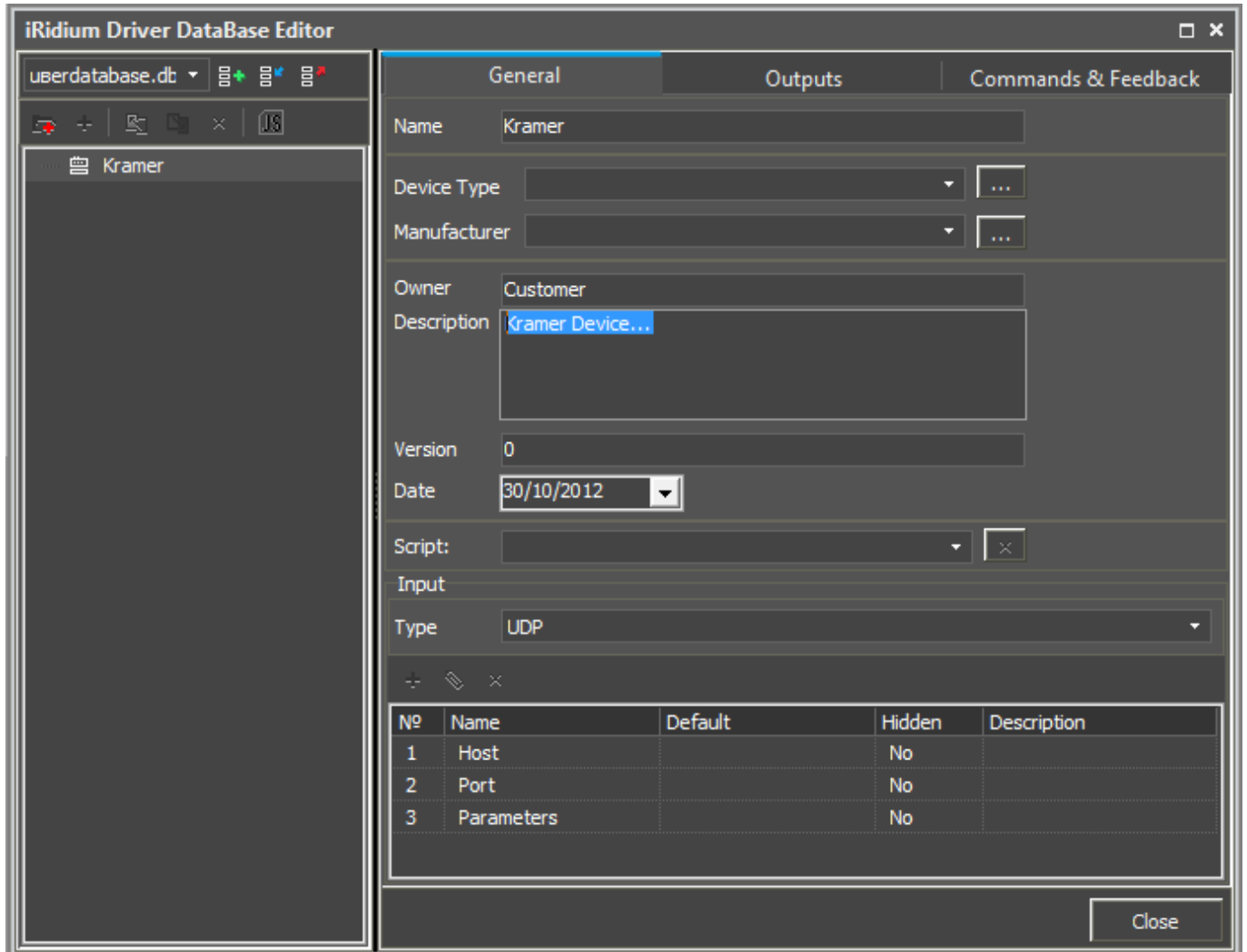
Creating New UDP Devices



Tools for creating new devices in iRidium data base are available in the right-click menu. The type of the created device - UDP (UDP Device) - defines its characteristics. For UDP devices you can indicate information about the manufacturer, type of the controlled equipment, comments and commands.

Setting Up UDP Devices

Creation of description for new devices is not compulsory but it can help with their further identification



- **Device Type** - a type of the controlled device (TV, DVD, Blu Ray, ...)
- **Manufacturer** - the name of the device manufacturer
- **Description** - description of the device, it can include hyper links
- **Date** - the date of creating the driver

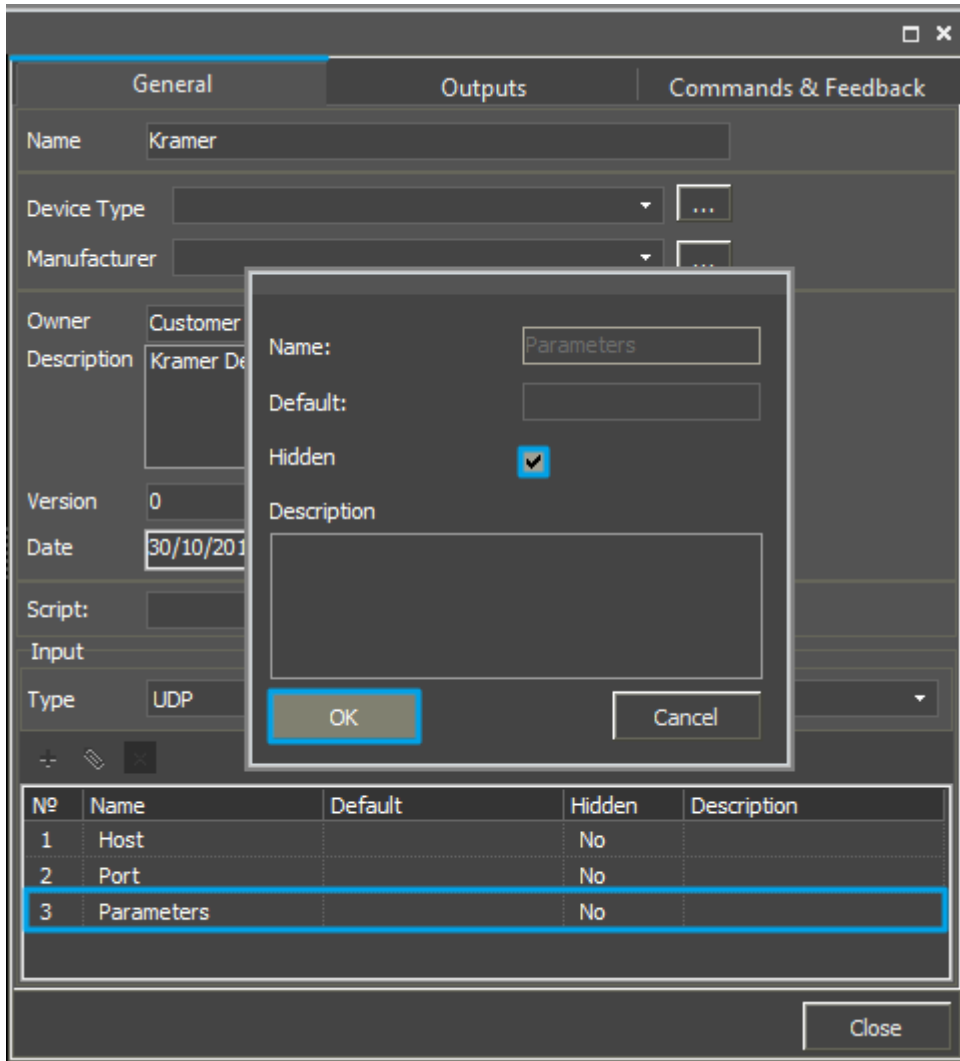
UDP devices have the following properties for connection:

- **Host** - an IP-address of the controlled Kramer device in your network.
- **Port** - a port the device works through.

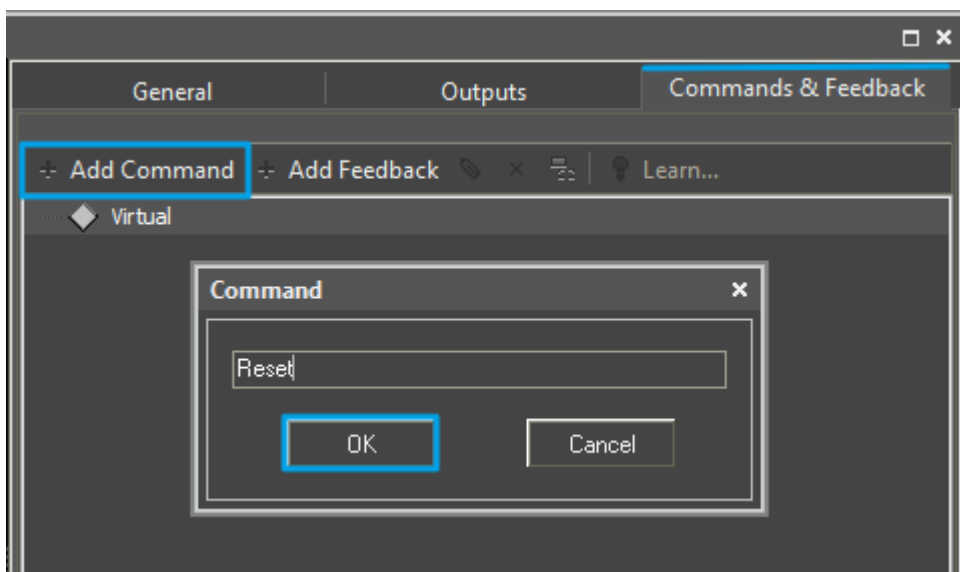
Optional properties:

- **Parameters** - additional properties for connection

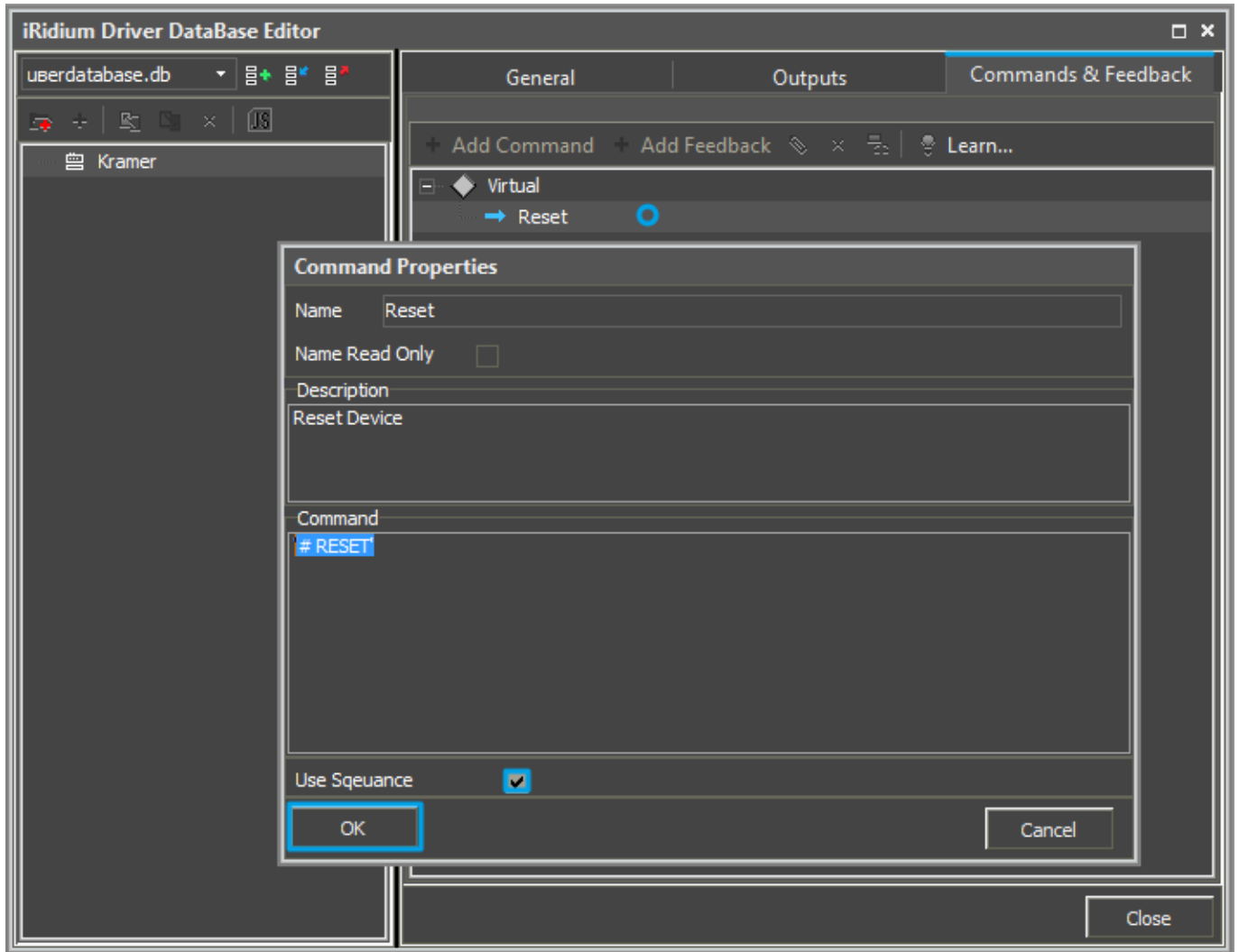
The optional properties are used seldom and if you are not going to use them it is recommended to hide them: click two times on each property to open the window for settings and select **Hidden**.



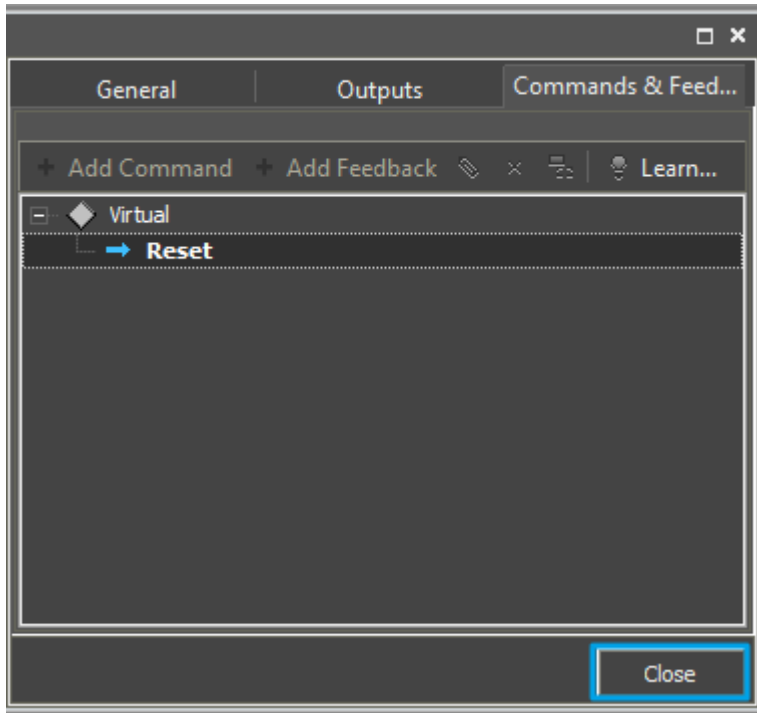
To add the command in the list of the created device go to the Commands & Feedback tab and use tools of the Editor base:



The command is created empty by default. The instruction the command should send is indicated in the **Command Properties** windows. Click two times on the created command to open the **Command Properties** window.



- Write the instruction for the command in the **Command** field of the opened window. In our case for the **Reset** command (device resetting) there is the **# RESET** instruction (the command is sent in the ASCII format).
- Mark the **Name Read Only** field if you do not want to change the command name in the future (it is used when the driver works with scripts which refer to the command identifying it by its name).
- In the **Descriptions** field indicate the command description.



- Now you can close the **DB Editor** window.

[↑ Back](#)

Syntax of UDP Commands

Commands for sending via the UDP protocol cannot be added to DB Editor without checking them for compliance with the syntax accepted in iRidium. It allows you to identify the command data format and form the command correctly so it could be read by the controlled equipment.

The syntax of data sending via UDP (Ethernet or Wi-Fi) defines the rules for sending data in one of the formats selected by the user - ASCII, HEX, DEC.

Sending Commands in the ASCII (Strings) Format

To indicate a string in the ASCII format in the "Data" window write the string in 'single quotes'. Unlike for RS232 and TCP the symbol of «string end» is not required for UDP:

Syntax	'string'
--------	----------

In the Data window the string looks as follows:

'string'

Sending Commands in the HEX (Hexadecimal) Format

To send a string in the HEX format write \$ or **0x** before each HEX symbol in the "Data" window.

Syntax \$HEX

In the Data window the string looks as follows:

\$01,\$02,\$03
0x01,0x02,0x03

Sending Commands in the DEC (Decimal) Format

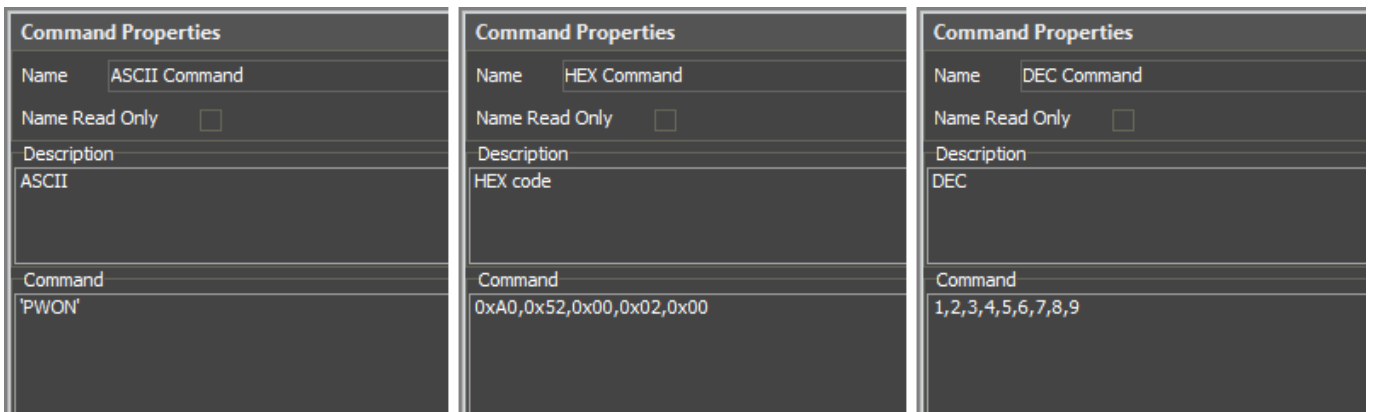
To indicate a string in the DEC format in the "Data" window write data in the decimal format separated by commas.

Syntax DEC,DEC,DEC

In the Data window the string looks as follows:

1,2,3

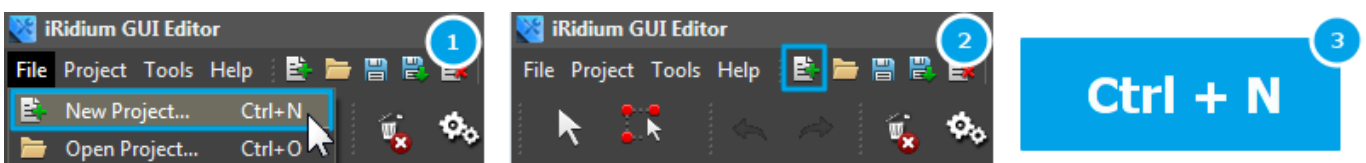
Examples of using the syntax (ASCII, HEX, DEC):



[↑ Back](#)

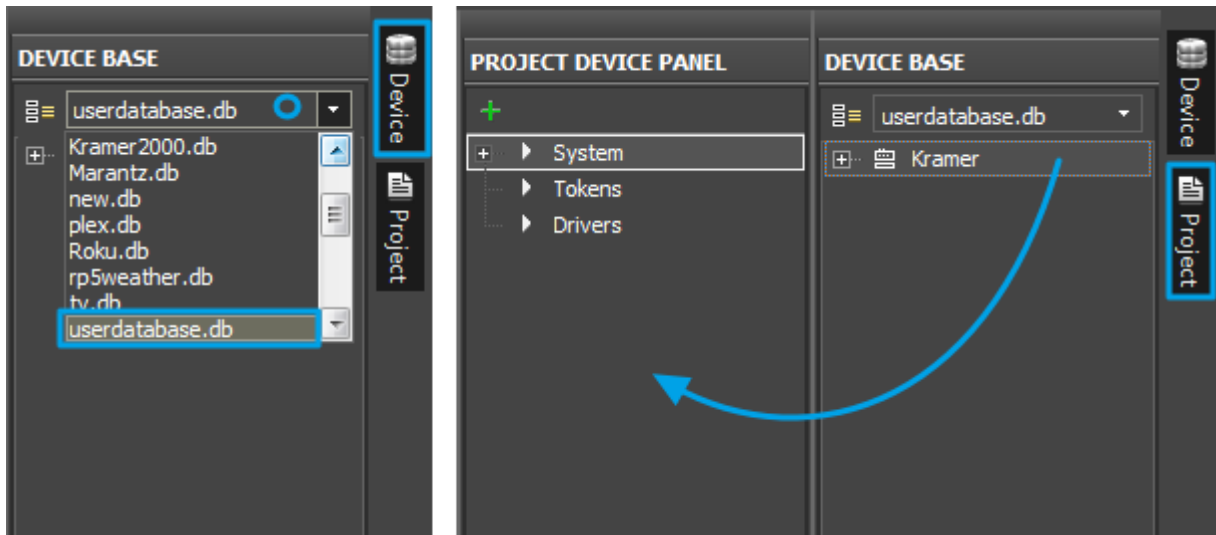
Adding UDP Devices to iRidium Projects

Create a new project. New projects can be created with the help of the button on Tool Box or through the menu **File > New Project**.

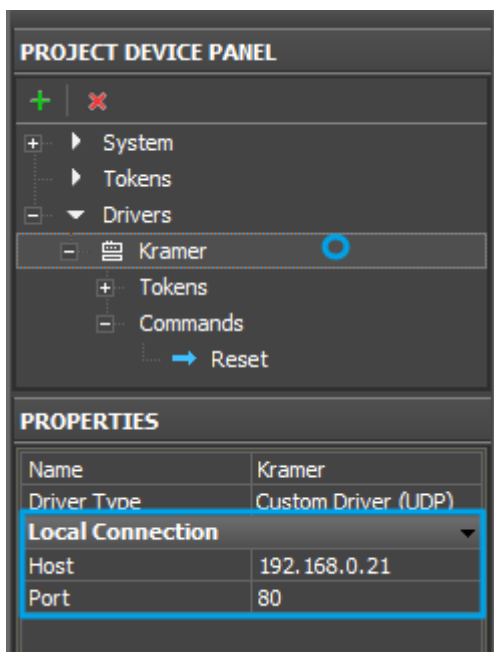


You can find more information about project creating in the section [Creating the Graphic Part of iRidium Projects](#)

Select the **kramer.db** data base in the **Device Base** panel and add the **Kramer** device you have in **Project Device Panel** of your project using the **Drag&Drop** method.



To connect iRidium to Kramer, indicate the local IP-address in general settings of the device in the **PROPERTIES** window of the Project Device Panel:

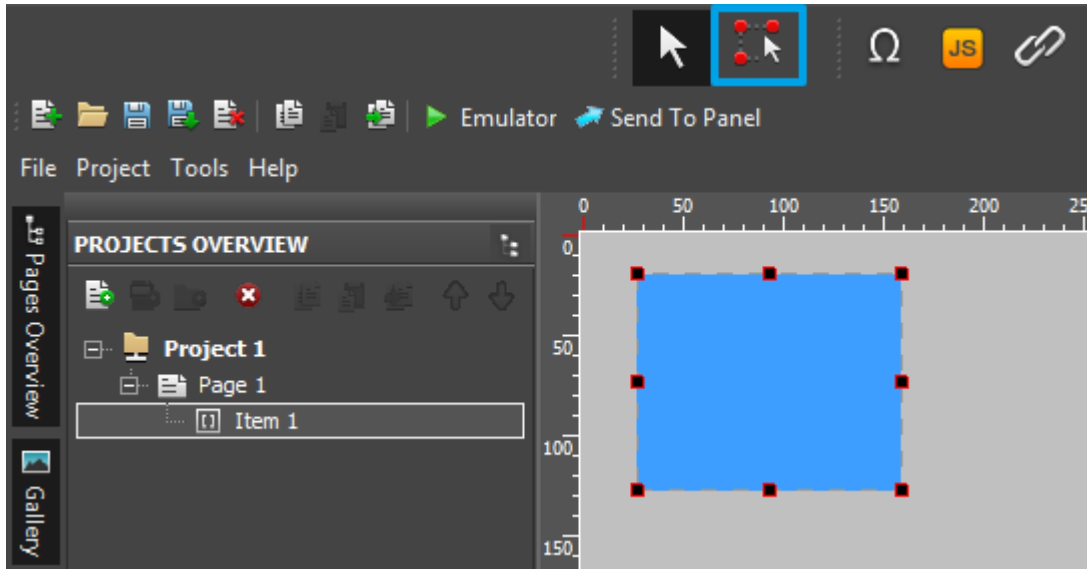


- **Host** - an IP-address of the controlled Kramer device.
- **Port** - a port the device works through.

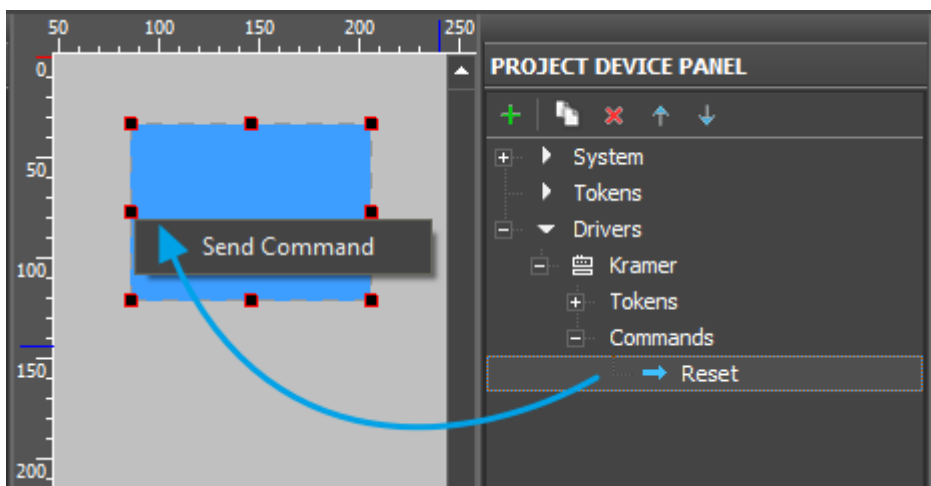
To work with the Kramer device via the Internet indicate the public IP-address of the router the converter is connected to. Set up the [Port Forwarding Service](#) for the router which enables possibility of referring to the local address of the controlled converter from the Internet (remote control).

You can learn the external (public) IP-address of your router with the help of external resources, for example [\[1\]](#)

Select the **Draw Item** tool and create a new item.



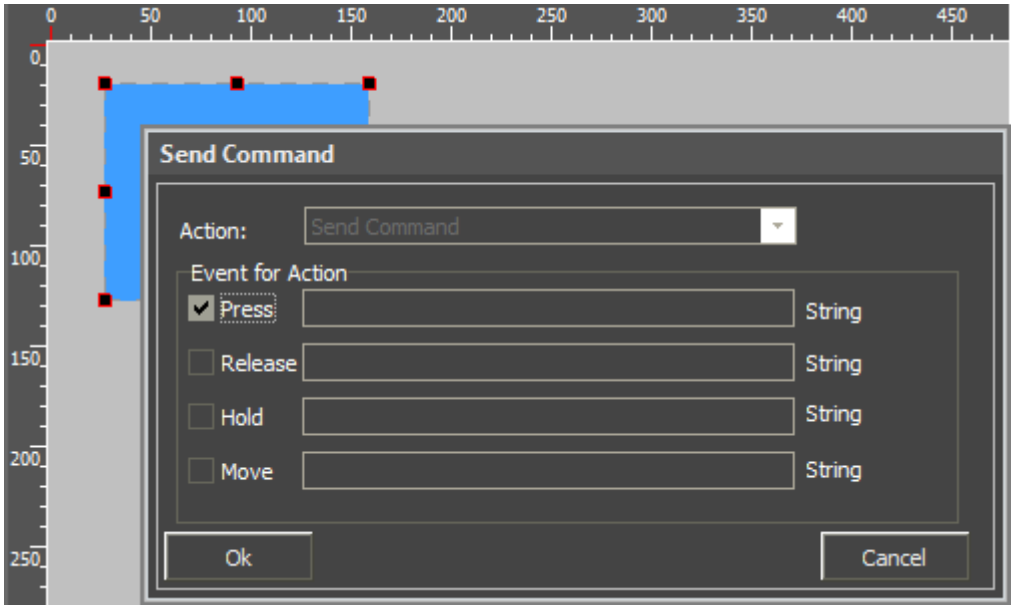
Open commands of the **Kramer** device and drag the **Reset** command on the item. Select **send command** in the appeared window.



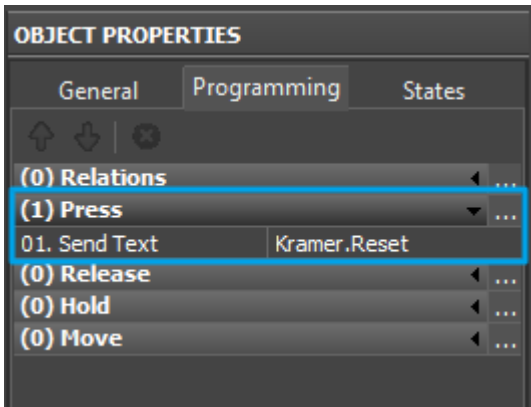
In the **Send Command** dialog window select the type of the event which will activate command sending to the device.

- **Press** - pressing on the item
- **Release** - releasing the item
- **Hold** - holding the item
- **Move** - moving on the item (it is mainly used for the **Level** item type)

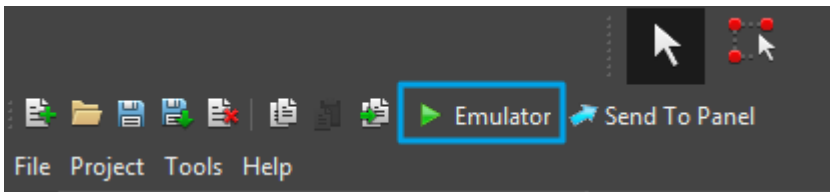
The **Press** event is indicated by default as the most commonly used.



The commands added on the item can be seen in the **Programming** tab of the **Object Properties** panel.



Launch Emulator to check how the command works.



[↑ Back](#)

Downloads

[Download: Example of a project for controlling the Kramer commutator via UDP \(1 Mb\)](#)

[<< back \(AV & Custom Systems: Working with Equipment via UDP\)](#)

[↑ Back](#)