

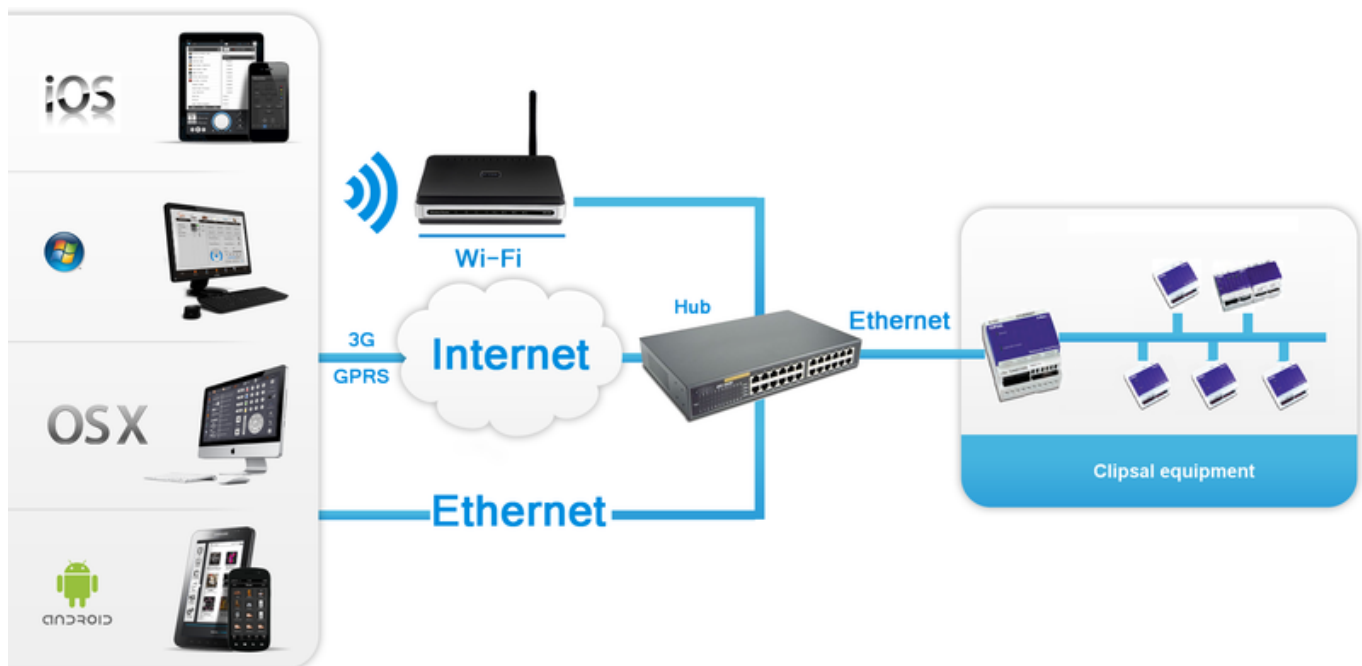
Contents

- [1 How It Works](#)
- [2 Peculiar Features of Clipsal and Clipsal Gate Drivers](#)
 - [2.1 Setting up Access of Control Panels to Clipsal Gate](#)
- [3 Data Import from C-Bus Toolkit](#)
 - [3.1 1. Preparation of the project in C-Bus Toolkit](#)
 - [3.2 2. Import Tools](#)
 - [3.3 3. Import](#)
 - [3.4 4. Setting up Connection Properties](#)
- [4 Adding and Setting up Clipsal Devices](#)
 - [4.1 Clipsal Driver](#)
 - [4.2 Clipsal Gate Driver](#)
 - [4.3 Switching Between the Local and Internet Connection](#)
- [5 Principles of Sending Commands to Clipsal Equipment](#)
 - [5.1 Creating Control Commands](#)
 - [5.2 Creating Channels for Receiving and Displaying the status of Clipsal group Addresses](#)
- [6 Emulation of Project Work](#)
- [7 Launching Projects on Control Panels](#)

How It Works

The C-Bus Serial protocol (IEEE 802.3) (TCP) is used for connection of iRidium App to Clipsal equipment.

Data sending and receiving is possible only if the panel and controller are in one IP subnetwork. Topology of connecting iRidium App to controllers:



Peculiar Features of Clipsal and Clipsal Gate Drivers

iRidium for Clipsal is a package of tools for creating interfaces and controlling automation systems

on the basis of Clipsal controllers. iRidium enables sending and receiving data about the state of device groups using the C-Bus Serial protocol.

In order to set up connection to Clipsal equipment you need to create a device in GUI Editor. The device will define the way of communication and its properties. iRidium database for Clipsal contains the following drivers:

1. **Clipsal** - work via the CNI Ethernet interface.
2. **Clipsal Gate** - work via the C-Gate software. The PC with running C-Gate should always be on.

The difference in setting up of the drivers in GUI Editor is inconsiderable:

1. In the panel settings of the Clipsal Gate driver it is required to indicate the IP-address of the PC with the installed C-Gate service and [the project name from C-Bus Toolkit](#).
2. In the command settings it is required to indicate [the device Network ID](#).
3. It is required to enable access of the control panel to [the C-Gate service](#).

The rest settings are similar to the Clipsal driver.

Setting up Access of Control Panels to Clipsal Gate

The first thing you are required to do when controlling Clipsal equipment with the help of the Clipsal Gate driver is to enable access of all control panels to C-Gate. In order to do that you need to add the control panel IP-address in the **access.txt** file which is located in the directory with the installed Clipsal software: **Clipsal\C-Gate2\config**. It should be done as follows:

```
interface 192.168.0.1 Program
```

After making changes you need to restart the C-Gate service.

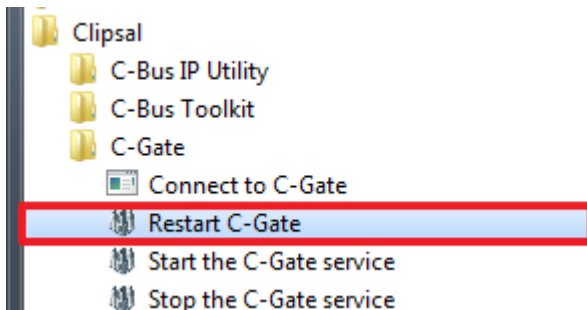
For example, you have 3 devices you want to control Clipsal from:

1. PC with installed Emulator with IP-address: 192.168.0.93
2. iPad with IP-address: 192.168.0.118
3. Android based device with IP-address: 192.168.0.104

Your access.txt file will look as follows:

```
##C-Gate Server Access Control File
## This file was written automatically by a command
issued to the server
## Created:Tue Oct 05 16:22:26 CST 2004
## File name: C:\clipsal\c-gate\config\access.txt
interface 0:0:0:0:0:0:1 Program
interface 127.0.0.1 Program
interface 192.168.0.93 Program
interface 192.168.0.118 Program
interface 192.168.0.104 Program
interface localhost Program
## End of access control file
```

After making changes, restart the C-Gate service via the Start menu:



[↑ Back](#)

Data Import from C-Bus Toolkit

iRidium for Clipsal allows you to create interfaces for controlling Clipsal equipment. The source of data for import is a project created in **C-Bus Toolkit**.

1. Preparation of the project in C-Bus Toolkit

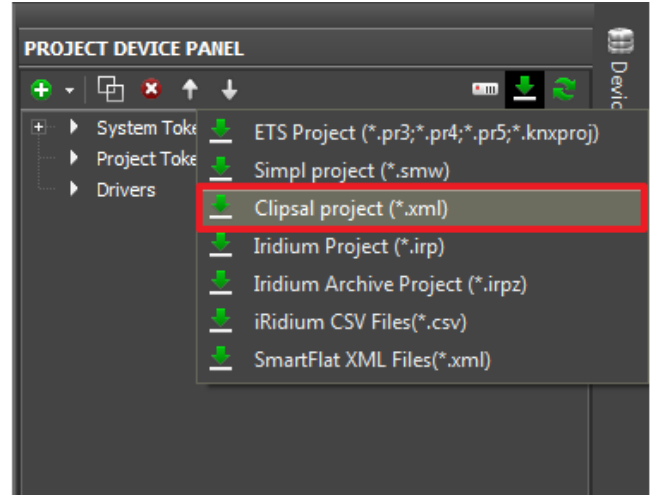
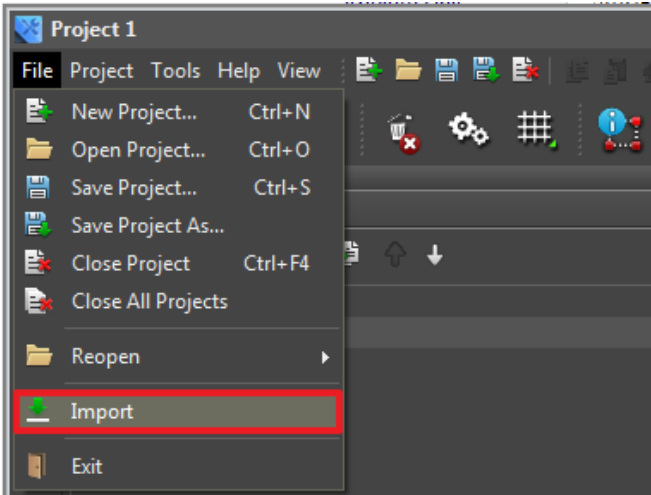
Create and set up a project for controlling Clipsal equipment in C-Bus Toolkit.

By default, the names of commands in GUI Editor are formed as follows:

ApplicationName_Group Name -> Lighting_Group 1

2. Import Tools

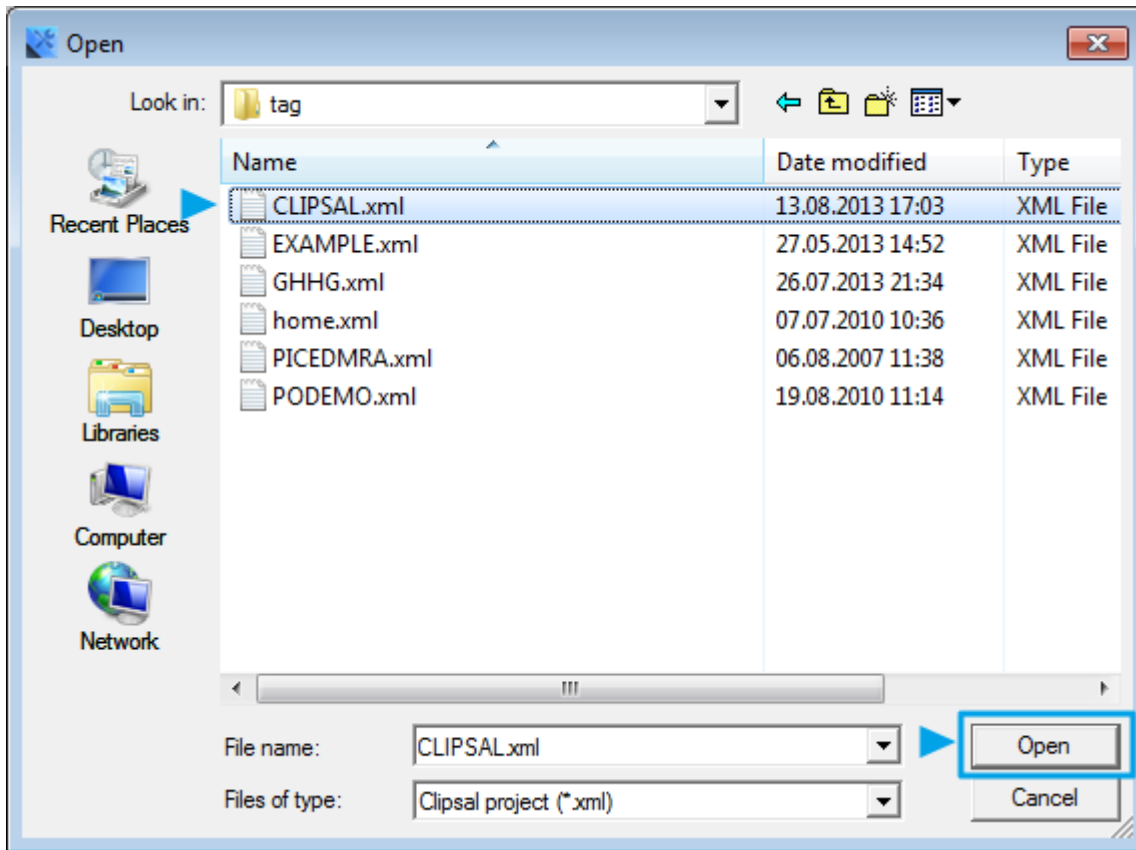
Import to GUI Editor is performed with the help of the «Import» tool in the «File» menu or on Project Device Panel:



3. Import

Select the project created in **C-Bus Toolkit**.

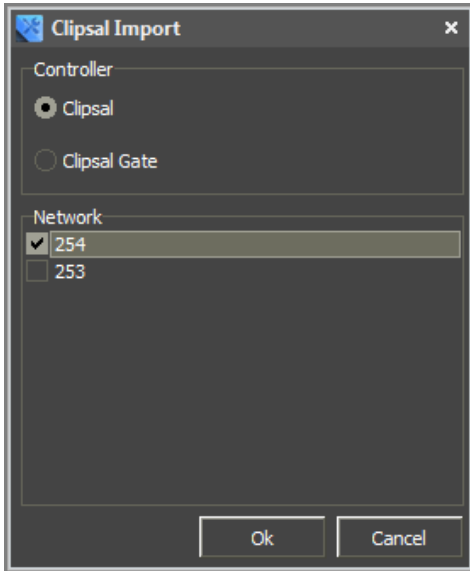
The file is located in the directory with the installed Clipsal software: **Clipsal\C-Gate2>tag** and has the **.xml** format.



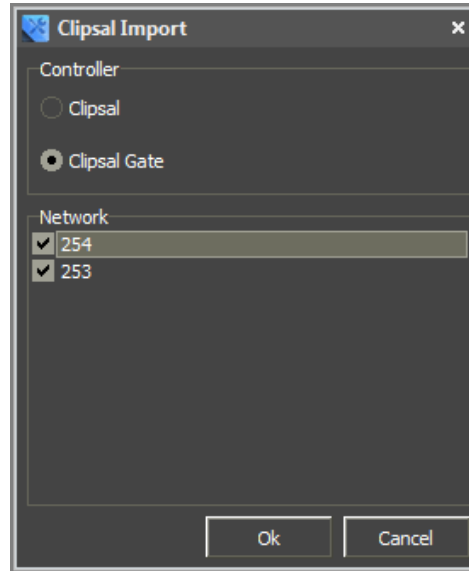
Select the type of the driver you will use in the project in the appeared window.

Clipsal

Clipsal Gate

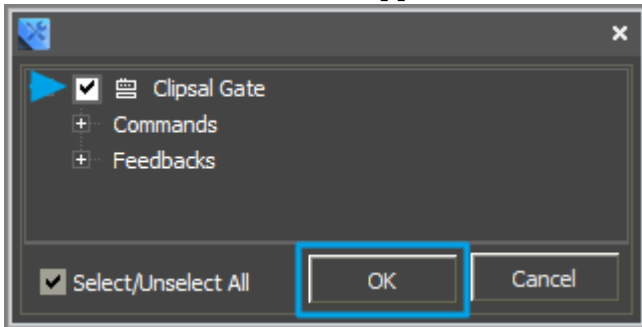


you can select only one network at a time



you can select the required number of networks you plan to use in your project

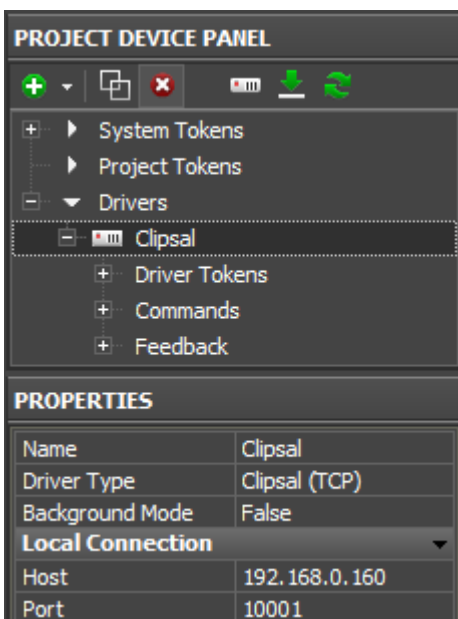
Mark the check-box in the appeared window with commands and feedback channels and press OK.



4. Setting up Connection Properties

Set up the properties of connection to the equipment (Project Device Panel):

Clipsal



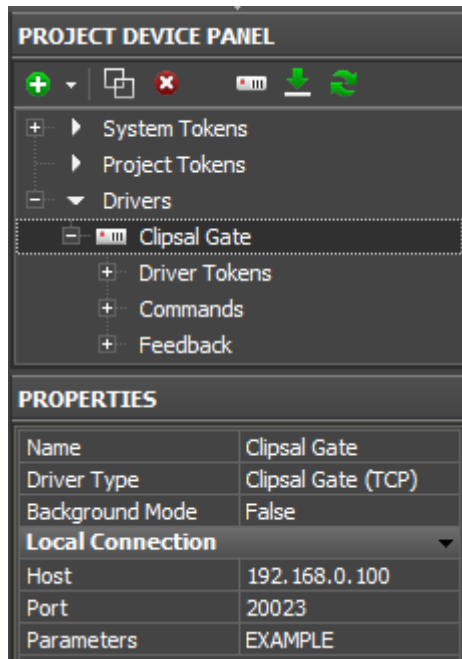
It enables iRidium App work on iOS in the background mode.

Background Mode Drivers and scripts continue to receive data from the controlled equipment. It makes displaying of notifications about the system events possible, even if the application is minimized.

Host the controller IP-address (private or public)

Port the TCP port for connection to the controller (**10001** by default)

Clipsal Gate



It enables iRidium App work on iOS in the background mode.

Background Mode Drivers and scripts continue to receive data from the controlled equipment. It makes displaying of notifications about the system events possible, even if the application is minimized.

Host the IP-address of the PC with the running C-Gate service (private or public)

Port the TCP port for connection to the controller (**20023** by default)

Parameters the name of the project used in C-Bus Toolkit (considering the register)

[↑ Back](#)

Adding and Setting up Clipsal Devices

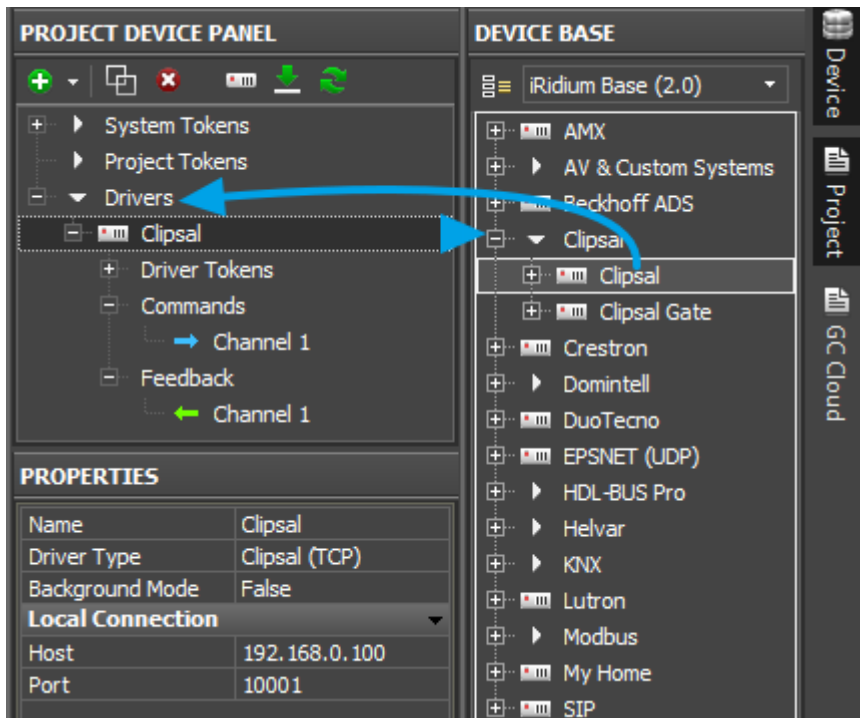
Clipsal Driver

The Clipsal driver in the GUI Editor base of devices enables work with Clipsal equipment

via the **CNI Ethernet interface**.

To connect to Clipsal equipment, create a new iRidium project and add the Clipsal driver from the iRidium data base in it. After that set up properties of connection to the controller using TCP/IP. In order to do that indicate the controller IP-address (private or public), the TCP port for connection to

the controller and parameters of connection:



It enables iRidium App work on iOS in the background mode.

Background Mode Drivers and scripts continue to receive data from the controlled equipment. It makes displaying of notifications about the system events possible, even if the application is minimized. div>

Host the controller IP-address (private or public)

Port the TCP port for connection to the controller (**10001** by default)

To enable work with the controller via the Internet indicate the IP-address of the router the controller is connected to in the connection settings. Set up the [Port Forwarding service](#) to enable remote control - referring to the local (private) address of the controlled controller from the Internet.

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

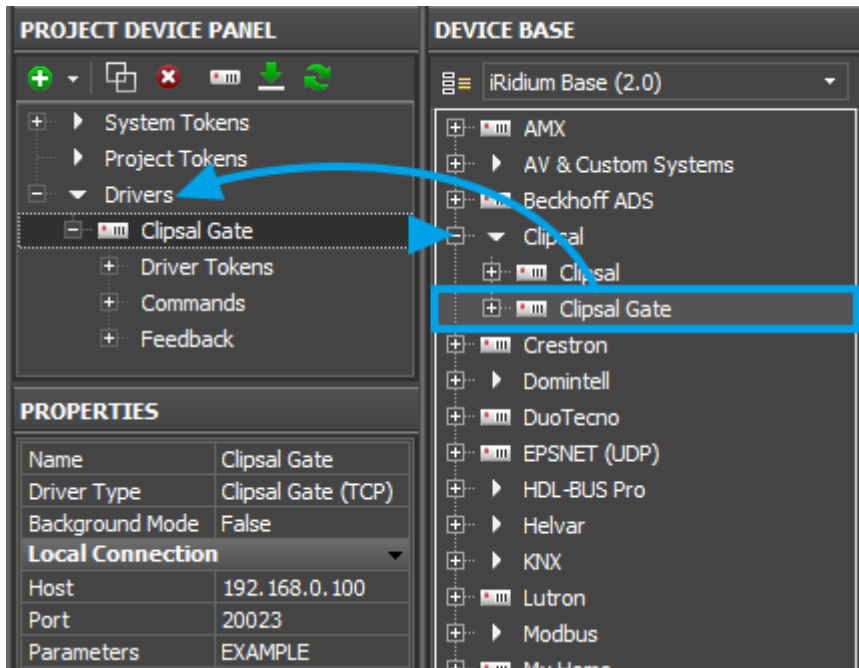
Clipsal Gate Driver

The Clipsal Gate driver in the GUI Editor base of devices enables work with Clipsal equipment

via the C-Gate application.

First of all it is required to [enable connection of control panels to C-Gate](#)

To connect to Clipsal equipment via C-Gate, create a new iRidium project and add the Clipsal Gate driver from the iRidium data base in it. After that set up properties of connection to the controller using TCP/IP. In order to do that indicate the controller IP-address (private or public), the TCP port for connection to the controller and parameters of connection:



It enables iRidium App work on iOS in the background mode.

Background Mode Drivers and scripts continue to receive data from the controlled equipment. It makes displaying of notifications about the system events possible, even if the application is minimized.

Host the IP-address of the PC with the running C-Gate service (private or public)

Port the TCP port for connection to the controller (**20023** by default)

Parameters the name of the project used in C-Bus Toolkit (considering the register)

To enable work with the controller via the Internet indicate the IP-address of the router the controller is connected to in the connection settings. Set up the [Port Forwarding service](#) to enable remote control - referring to the local (private) address of the controlled controller from the Internet.

You can learn the external (public) IP-address of your router with the help of external resources, for example [\[2\]](#)

[↑ Back](#)

Switching Between the Local and Internet Connection

If the control panel should hold the connection with the system when the panel is out of the limits of the Wi-Fi network of the system, you need to set up the switch between the Internet and the local network.

The remote mode suggests connection to the system via the Internet. At that the external IP-address or the domain name of the system to which you need to connect have to be used.

⚠ In iRidium Wi-Fi/3G CANNOT be switched automatically. For switching between the Internet and the local network you need buttons with special settings. See the settings below.

⚠ To control the system remotely you have to open the system for external access - to set up [the Port Forwarding Service](#).

⚠ To secure equipment from unauthorized access we recommend using secure connection with the remote system (VPN).

Setting up of the switch Wi-Fi/3G in iRidium projects:



1. Open the script editor in iRidium GUI Editor.
2. Download and add into your project the template of the Wi-Fi/3G switch (Add Script from file): [download the template of the Wi-Fi/3G switch](#)

The Wi-Fi/3G switch is performed with the help of the script function [SetParameters](#)

Setting up of parameters of the Wi-Fi/3G switch:

```
function Internal_1() // Function name
{
    IR.GetDevice('Clipsal Gate').SetParameters({Host: '192.168.0.101', Port: '20023',
    Parameters: 'ToolkitProject'}); // Driver Name + Parameters
}

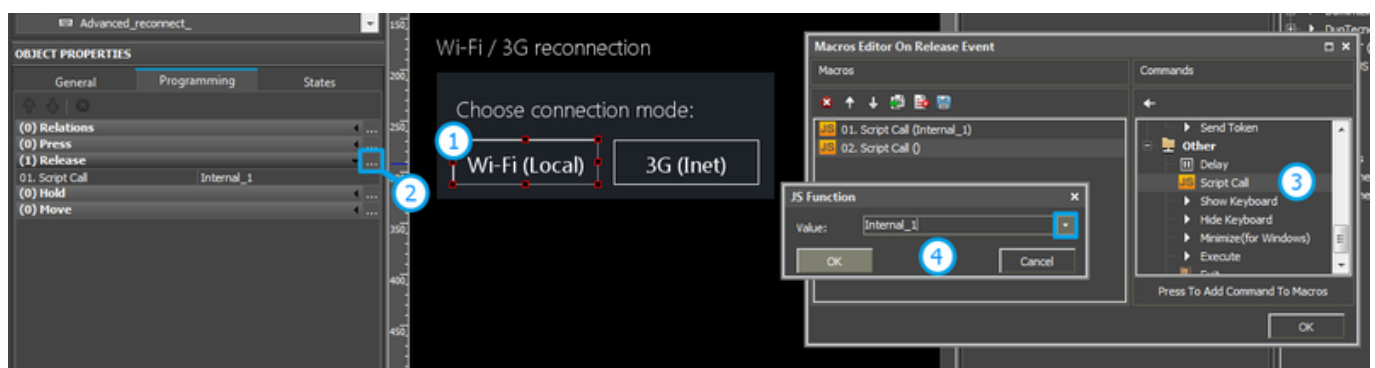
function External_1()
{
    IR.GetDevice('Clipsal Gate').SetParameters({Host: '220.115.10.10', Port: '20023',
    Parameters: 'ToolkitProject'});
}
```

Indicate in the command settings:

- Function name - the name of the switch function (command). Two functions cannot have the same name in a project.
- Driver Name - the name of the driver which parameters are changed
- Parameters - the set of the switch parameters which you need to apply to the driver

Assign commands to buttons:

1. Select the button which will be responsible for the Wi-Fi/3G switch.
Open the properties of the button: Object Properties > Programming
2. Open [Macros Editor](#) of the button for the Press or Release events
3. Select the **Script Call** command and add it by double-clicking on it
4. Select the name of the function you want to activate in the drop-down list. Create the command.



Set up access to the equipment from the Internet:

In order to do that [open the equipment ports for remote access](#).

[Download the example of the Wi-Fi/3G switch \(project\) >>](#)

[↑ Back](#)

Principles of Sending Commands to Clipsal Equipment

To send commands and receive data about the state of variables the driver has 2 tabs:

- **Commands** - commands for setting up values of controlled group addresses.
- **Feedbacks** - channels receiving feedback from controlled group addresses.

Setting up commands and feedback channels is identical for both drivers. But there is an additional field in Clipsal Gate commands called **Network ID**. There you should write the number of network to which the group address in C-Bus Toolkit belongs.

In iRidium you can control the following types of applications:

*Lighting - it is used mainly for controlling light but also for other electrical load (shutters, ventilators, etc.)

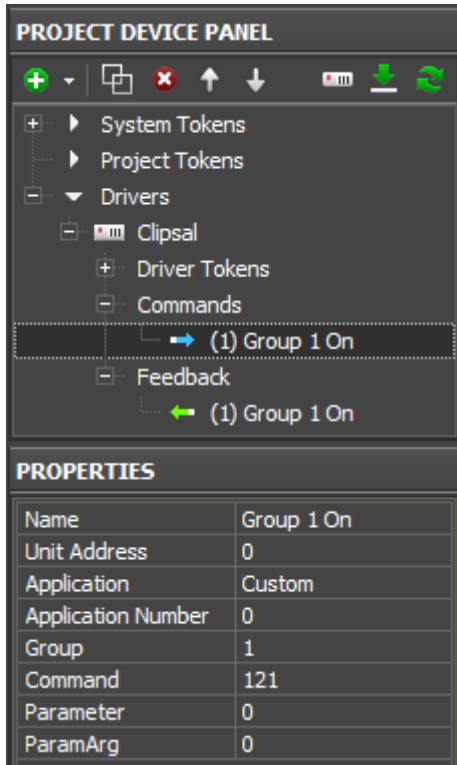
- Heating
- Room Control
- Security
- Metering
- Trigger Control - activate several commands (scenes) by pressing on one button
- Enable Control
- Temperature Broadcast
- Ventilation
- Access Control
- Clock
- Telephony
- Air Conditioning
- Measurement
- Irrigation Control
- Audio / Video
- Pool
- Media Transport
- Hvac Actuator 1
- Hvac Actuator 2
- Custom (for custom applications) - if you create your own applications they will be defined as Custom when you import it in GUI Editor. In the field below you will be able to indicate the application number.

Creating Control Commands

1. Create a command in the project device tree and indicate its properties:

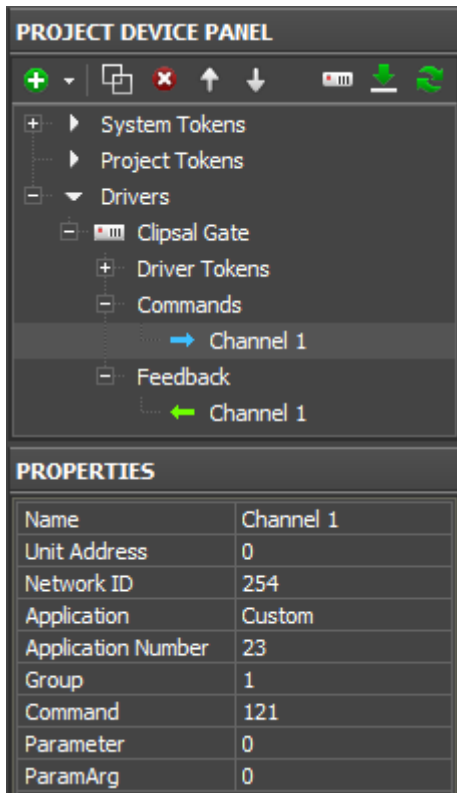
(different Applications have a different sets of properties)

Clipsal driver



- Name** the command name (at random)
- Unit Address** the number of the module the command addresses (Network/Units/Address in C-Bus Toolkit)
- Application** the application from C-Bus Toolkit
- Application Number** the number of the application from C-Bus Toolkit (if you use non-default application)
- Group** the group address from C-Bus Toolkit
- Command** the command sent to the group address
- Parameter** the additional condition for the command (used with some applications)
- ParamArg** a parameter argument (used with some applications)

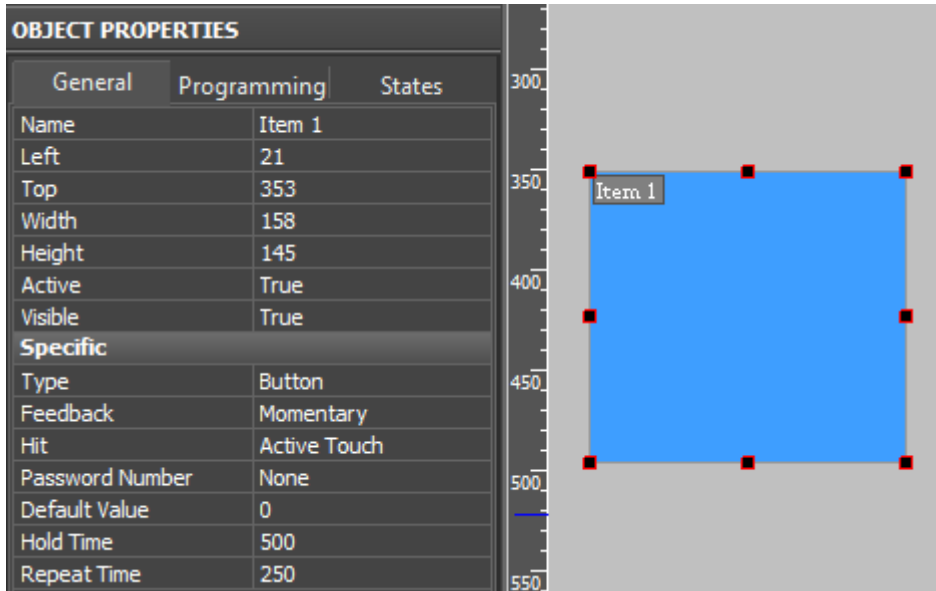
Clipsal Gate driver



- Name** the command name (at random)
- Unit Address** the number of the module the command addresses (Network/Units/Address in C-Bus Toolkit)
- Application** the application from C-Bus Toolkit
- Application Number** the number of the application from C-Bus Toolkit (if you use non-default application)
- Group** the group address from C-Bus Toolkit
- Command** the command sent to the group address
- Parameter** the additional condition for the command (used with some applications)
- ParamArg** a parameter argument (used with some applications)

2. Create a graphic item for command sending

(item properties depend on the command type and control task):

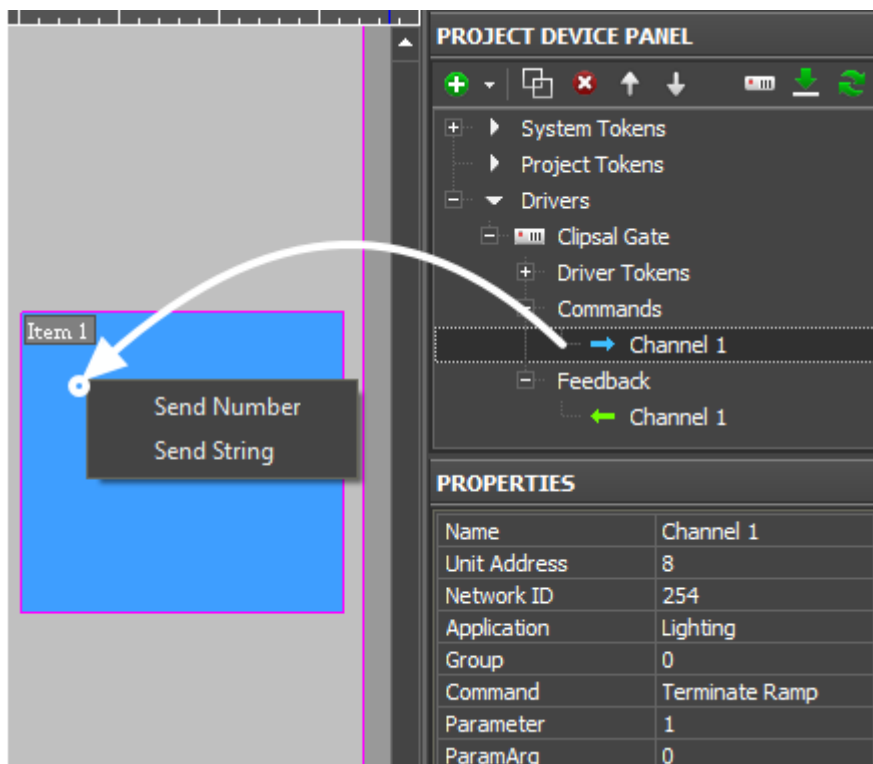


The selection of the item type depends on the behavior you need from it. The item feedback type affects processing and displaying of the data which are received from the controller. The main feedback types:

- **Momentary** -not to display data from the channel but change the item state when pressing on it (go to the initial state when releasing the item)
- **Channel** - to display data received from the channel. The data can be output in the item text field, affect the item state (switch the states depending on the received values) or affect the selected item properties.

The rest feedback types have specific applications and are used seldom.

3. Drag the command on the graphic item

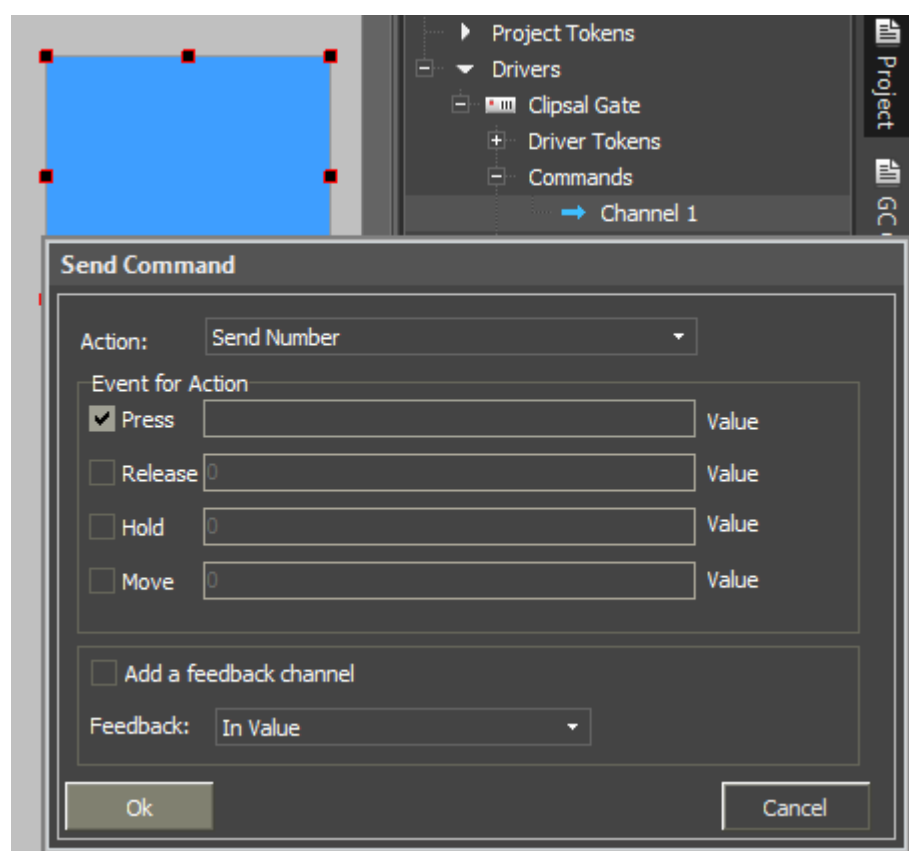


When dragging the command you will see the window for selecting the type and source of the data sent to the controller:

- **Send Number** - send the number in the decimal format (used when sending whole numbers). The outgoing value should be fixed. It is indicated when dragging the command on the graphic item)
- **Send String** - sent the data string in the UTF-8 format (it is not used when working with Clipsal).
- **Send Token** - send the value taken by one of the item properties or token (for example, you can select to send the current slider position. It is used for regulating the value and at work with Triggers and Up/Down Buttons).

4. Input data for sending and event which will activate the sending

If you need to read data from the same group address, mark the "Create Feedback Channel" event, indicate the item property which will be affected by the feedback channel:



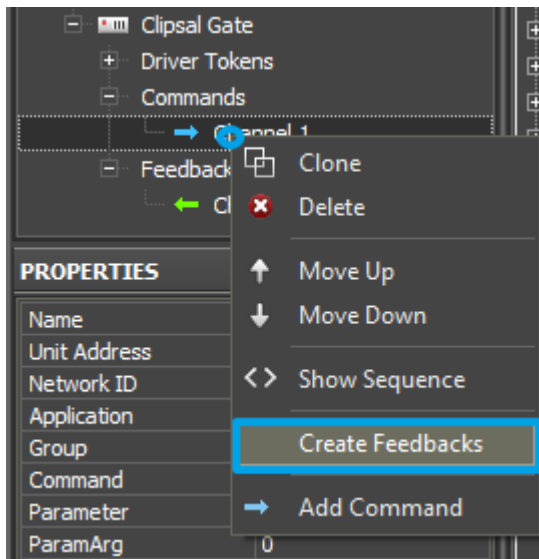
In the dialog window of command adding you should indicate the value you want to send and the event at which the value should be sent:

- **Press** - pressing on the item (sending a telegram at pressing)
- **Release** - releasing the item
- **Hold** - cyclic value sending when holding the item (delay between data sending should be set up additionally)
- **Move** - sending data at each slider movement (all values taken by Level). It is used for Level only.

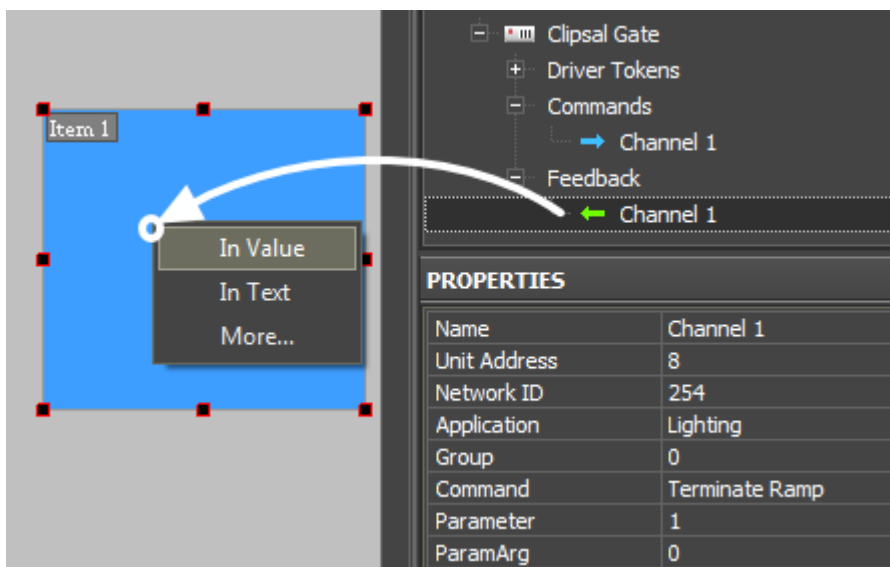
Creating Channels for Receiving and Displaying the status of Clipsal group Addresses

1. Create a feedback channel for receiving the group address status

(the channel can be created manually or generated on the basis of the command in the device tree with the help of the «CreateFeedbacks» button in the right-click menu)



2. Drag the feedback channel on the graphic item



Indicate the item property which will be affected by the feedback channel:

- **In Text** - to output values received from the channel as numbers in the text field
- **In Value** - values received from the channel affect the item state (change the slider position, switch on/off states of Button, activate animation, etc.)
- **More...** - you can select more complicated communication between the channel and item properties (for example, to change item coordinates when the channels value changes)

Emulation of Project Work

Emulator

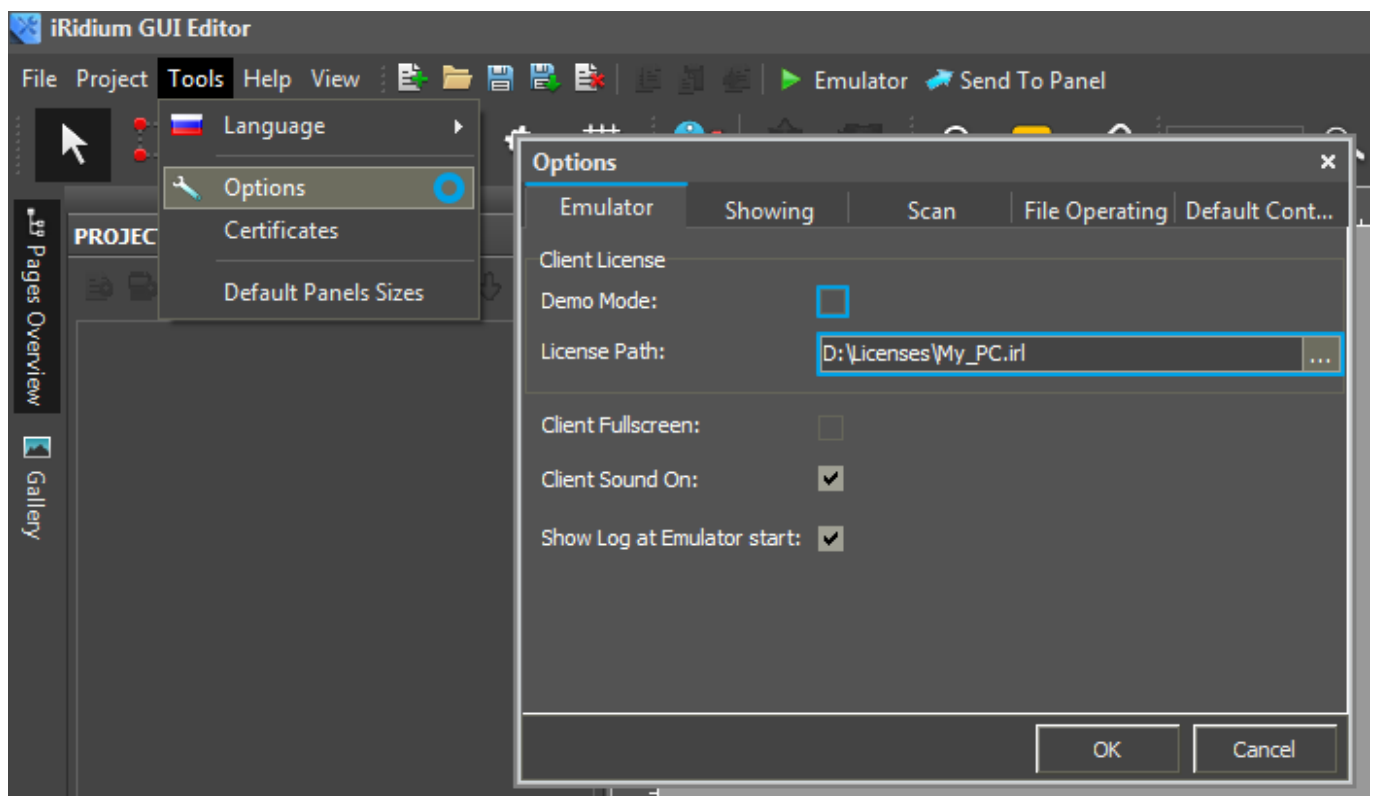
- is an iRidium application for Windows which can be launched from GUI Editor for testing your projects. Emulator can work both with a license (with connection to the equipment) and without it (when only the project graphic part is functional).

Operation modes for Emulator (see GUI Editor > Tools > Options > Emulator):

Without the license (Demo Mode: on) - **no connection with the controlled equipment.**

With the license (License Path: [...]) - when the [license file](#) is selected, all Emulator functions work and there is connection with the controlled equipment.

Indicate the path to the [license file](#) for Emulator (GUI Editor > Tools > Options > Emulator):



* When **Demo Mode** is activated Gui Editor won't show the message about starting without the connection to the controlled equipment (without a license). Check if there is a license when setting up your project.

Emulator hot keys

Click **F5** to start Emulator.

Click **F8** to open [the Emulator settings](#) (password: **2007**)

Click **F4** to open the Emulator log.



For **fully functional work** of iRidium on your PC it is required [to get](#) an iRidium license and [activate](#) it for [your PC](#).
Activation of licenses for iPad/iPhone/Mac/Android based or other devices does NOT lead to the automatic licensing of the PC with the installed iRidium Environment. A license for your PC is required for iRidium client on your PC (including Emulator) to work in the fully functional mode. The license should be purchased separately or you can use free For testing purposes you can use free [licenses](#)).

Logging in Windows



iRidium Log is a window where information about iRidium work, error messages and iRidium Script logs (IR.Log) are output in the text format.

To open the iRidium log in Windows, click **F4**.

[↑ Back](#)

Launching Projects on Control Panels

Uploading and launching of iRidium projects on control panels are performed with the help of the [iRidium Transfer](#) application installed on your PC. You can also upload your project on the panel from GUI Editor with the help of Transfer.

- Instructions for setting up properties of the iRidium project launch on control panels: [Properties for Launching iRidium Projects](#)
- Instructions for uploading iRidium projects on control panels: [iRidium Transfer](#).
- Instructions for setting up iRidium projects on control panel: [iRidium App](#)

[↑ Back](#)