

Contents

- [1 Principles of Controlling Equipment via IR](#)
- [2 Setting up Global Cache](#)
 - [2.1 Types of Global Cache Modules with IR Support](#)
- [3 Setting up of Global Cache Modules in iRidium](#)
 - [3.1 Setting up IR outputs of Global Cache Modules in iRidium](#)
 - [3.2 Driver Tokens](#)
- [4 Forming Lists of IR Commands](#)
 - [4.1 Ready IR Commands from GC Cloud \(IR Database\)](#)
 - [4.2 Learning IR Commands with GC-IRL IR Learner](#)
 - [4.3 Creating Databases of IR Commands on the Basis of Equipment Documentation](#)
 - [4.4 Settings of IR Commands](#)
- [5 Sending IR commands from Graphic Items](#)
- [6 Switching Between the Local and Internet Connection](#)

Global Cache (GC-100-06, GC-100-12, iTach IP2IR, iTach WF2IR) is a family of converters for controlling audio-video equipment. They receive commands via TCP/IP and send them to the controlled equipment as IR signals or via the COM-port. Converters can send commands in the local network or via the Internet.

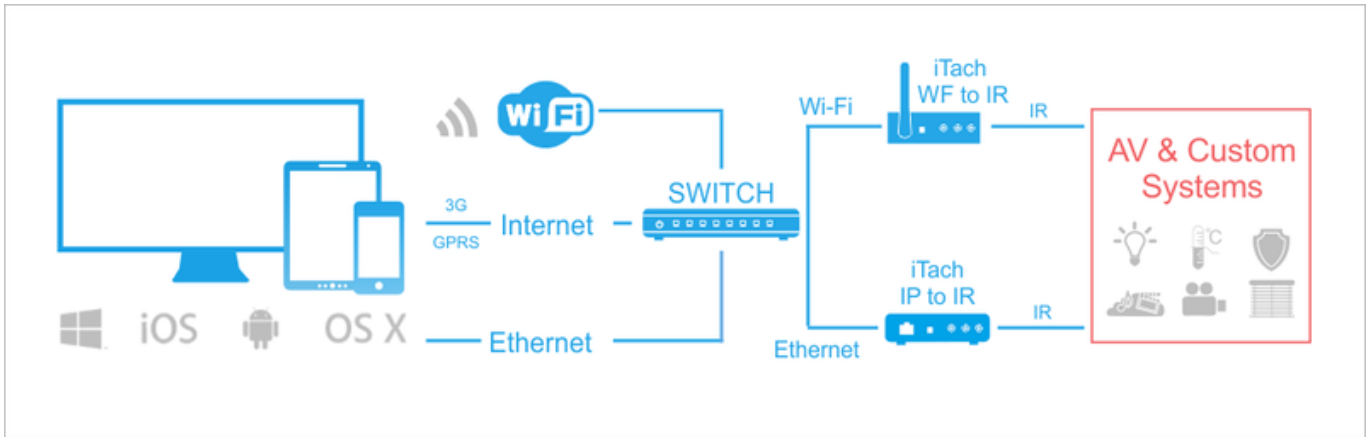
Global Cache converters are recommended for use with iRidium: they have the base of IR commands, preset drivers and the possibility of learning IR commands from remote controls.

Description of work:

iRidium for [Global Cache](#) supports connection with Global Cache converters via TCP/IP. When the connection is established, you can send commands to the converter - the list of commands is formed by the user in the process of creating a project. Commands have a particular format (syntax). Feedback from Global Cache converters includes information about the state of converters. There is no feedback from the equipment controlled via IR.

Principles of Controlling Equipment via IR

You can use "Global Cache" modules for controlling your audio-video and other equipment via IR in iRidium. The scheme of communication between iRidium and controlled equipment looks as follows:



Connection to Global Cache: TCP/IP.

Global Cache outputs: IR emitters or blasters

Format of IR commands: ASCII using Global Cache syntax

Feedback from equipment: none via the IR channel

Required license: Device License Pro for AV & Custom Systems (see [Licensing](#)). It works with any Global Cache converter. It enables use of any scripts in your projects.

[↑ Back](#)

Setting up Global Cache

A Global Cache (GC) converter should be connected to the local network and set up to enable iRidium to connect to it. Information on setting up GC:

[Video-lessons on setting up Global Cache converters](#)

[Documentation and a set of commands for Global Cache converters](#)

Types of Global Cache Modules with IR Support



[iTach WiFi to IR \(Infrared Control, WF2IR\)](#)

- Wi-Fi module (the mode of access point or infrastructure)
- 220W power supply
- independent IR sensor inputs/outputs (3 pcs)



[iTach TCP/IP to IR \(Infrared Control, IP2IR\)](#)

- Ethernet or 220W power supply
- independent IR sensor inputs/outputs (3 pcs)



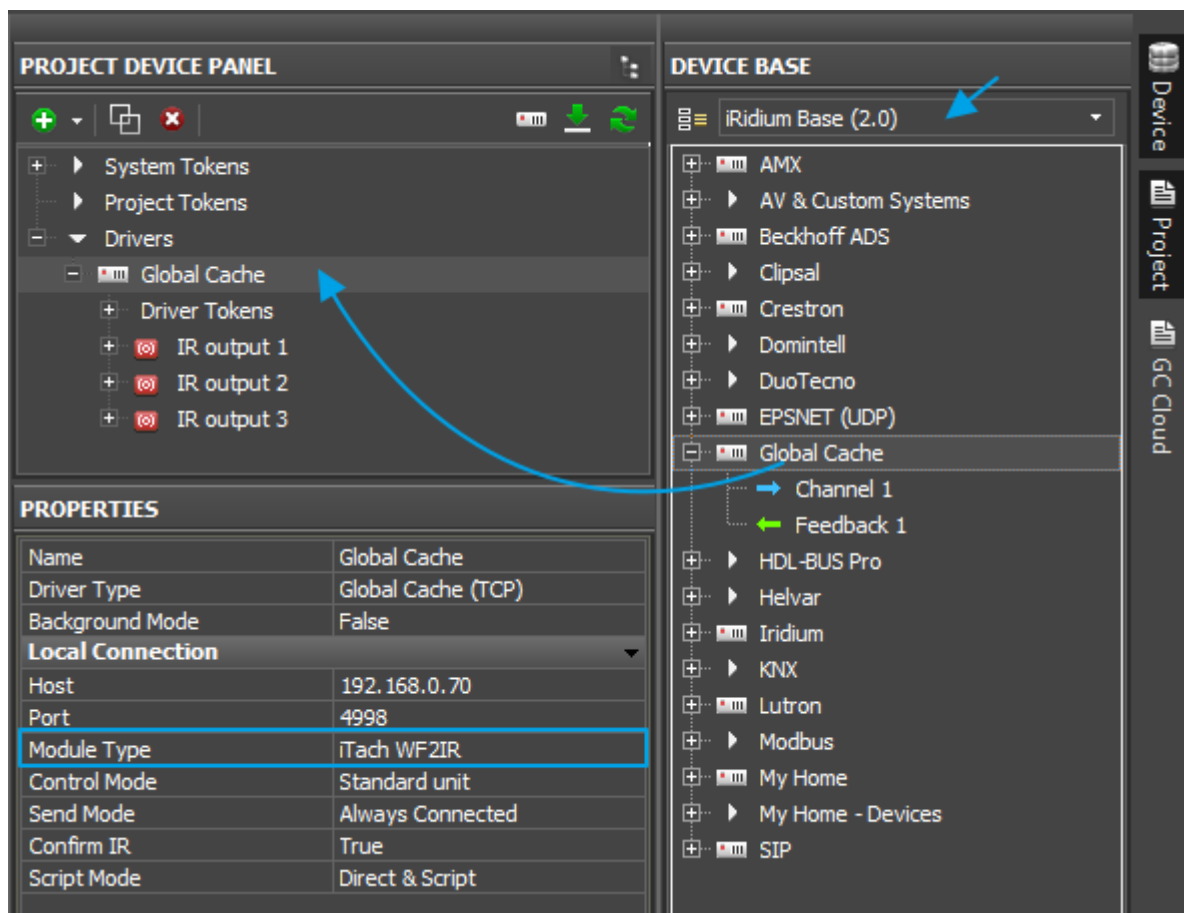
[GC-100-06, GC-100-12, GC-100-18, GC-100-18R](#)

- 220W power supply
- Ethernet
- COM-port (1 or 2 pcs)
- independent IR sensor inputs/outputs (3 or 6 pcs)
- relay outputs (0 or 3 pcs)

[↑ Back](#)

Setting up of Global Cache Modules in iRidium

Select the Global Cache driver in the DEVICE BASE list of [iRidium GUI Editor](#) and drag it into the PROJECT DEVICE PANEL window:



Properties of connection to Global Cache:

- **Host** - the IP-address of the converter (local, external or the domain name)
- **Port** - the main port of connection with GC (4998). It is used for sending IR commands, setting up and receiving the Global Cache status.
- **Module Type** - the type of the Global Cache converter: GC-100-06, GC-100-12(18,18R), iTach IP2IR (WF2IR), iTach IP2SL (WF2SL), iTach IP2CC (WF2CC). It defines the number of converter outputs and their type.
- **Control Mode** - the mode of device work
 - Standart Unit - control of equipment via IR with the help of connected IR emitters
 - Single LED / Dual LED - it enables setting up of LED light control using 3,5mm iTach slots, set up as control contacts of the light controller (see [iTach API](#))

- **Send Mode** - the mode of connection to the converter
 - Always Connected - constant connection (standard mode)
 - Connect when Sending - connection only for command sending. It is meant for GC-100-06(12,18,18R) to which you need to connect several control panels
- **Confirm IR** - to send the IR command only after confirming the sending of the previous one. It helps to avoid overlapping of IR commands.
- **Script Mode** - it defines where to send commands
 - Direct and Script - to send the command to the converter right away and duplicate it in the script (data can be processed in iRidium Script)
 - Script Only - to send the command to the script only (when clicking the button the command will not be sent to the converter). It enables sending of command components to the scripts to drivers work. *Do not activate this mode if you do not use scripts.*

To connect to Global Cache via the Internet:

1. Indicate the public IP-address (or the domain name) of your Internet router in the **Host** field
3. Set up [Port Forwarding](#) on your router to enable remote control of your equipment .

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

To switch between local and Internet connection:

Use the example presented [below](#).

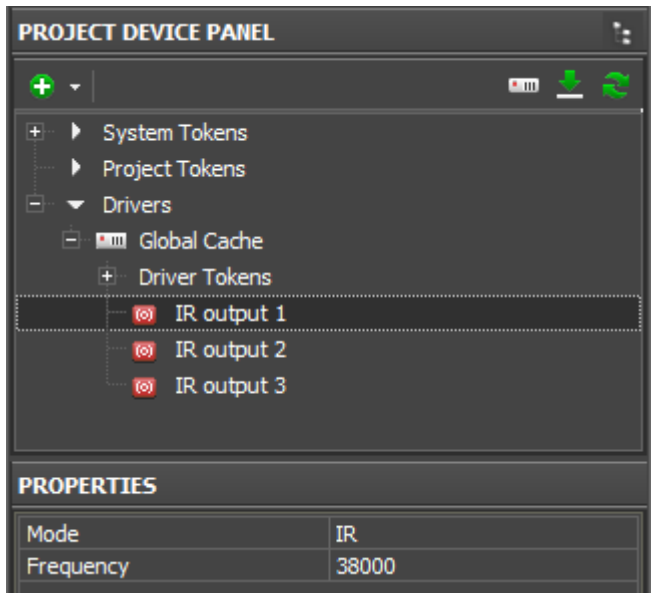
[↑ Back](#)

Setting up IR outputs of Global Cache Modules in iRidium

3.5mm outputs of Global Cache converters can operate in one of the following modes:

- **IR** - the mode of sending IR commands, when the IR emitter or IR blaster is connected to the output
- **Sensor Notify** - the mode of sensor, when the output informs about the status of the connected sensors.

To send an IR command from the Global Cache output, set it in the **IR** mode.

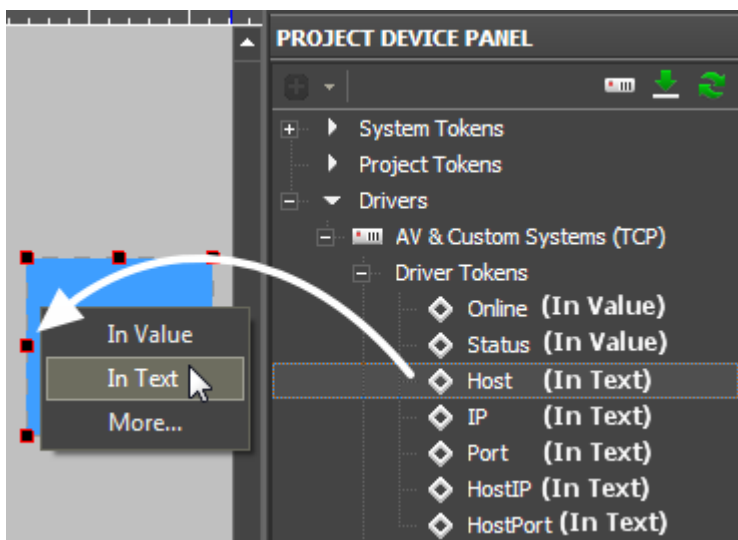


- **Frequency** (Hz) - the frequency of sending IR commands from the selected IR output. It can be changed for the folder with IR commands or for a separate command. The frequency depends on requirements of controlled equipment. You can learn it with the help of IR remote control or during the process of transformation of IR commands to the GC format (if you couldn't find the set up commands in the GC database).

[↑ Back](#)

Driver Tokens

Driver variables are available for reading only. They help to monitor connection to equipment.



- Online** the state of connection to the controlled system (Online/Offline = 1/0)
- Status** the status of connection to the system (Offline/Connect/Online/Waiting = 0...3)
- Host** the domain name of the remote system
- HostPort** the port of the remote system to which [iRidium App](#) connects
- IP** the IP-address of the control panel
- HostIP** the IP-address of the remote system to which iRidium App is connected
- Port** the client local port through which connection to the remote device is performed

Forming Lists of IR Commands

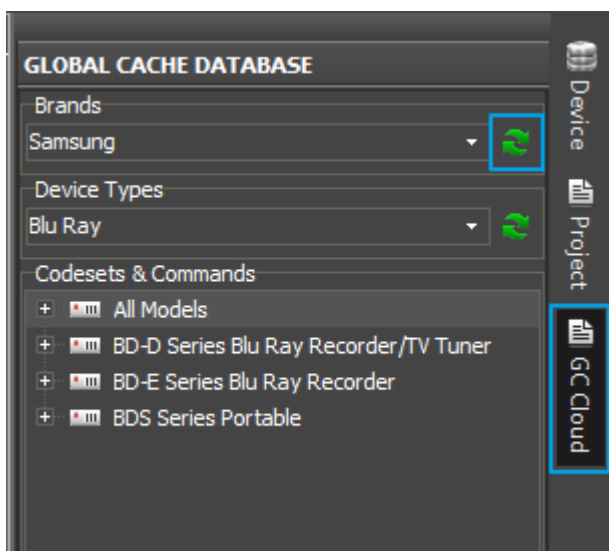
The set up Global Cache outputs have to be filled in with IR commands. They can be received in several ways:

- Add ready commands from Global Cache Cloud (IR Database)
- Learn commands from the native IR remote control with the help of [GC-IRL IR Learner](#)
- Transform commands received from documentation to equipment into the Global Cache format

Ready IR Commands from GC Cloud (IR Database)

The online database of IR commands is available in [iRidium GUI Editor](#), GC Cloud.

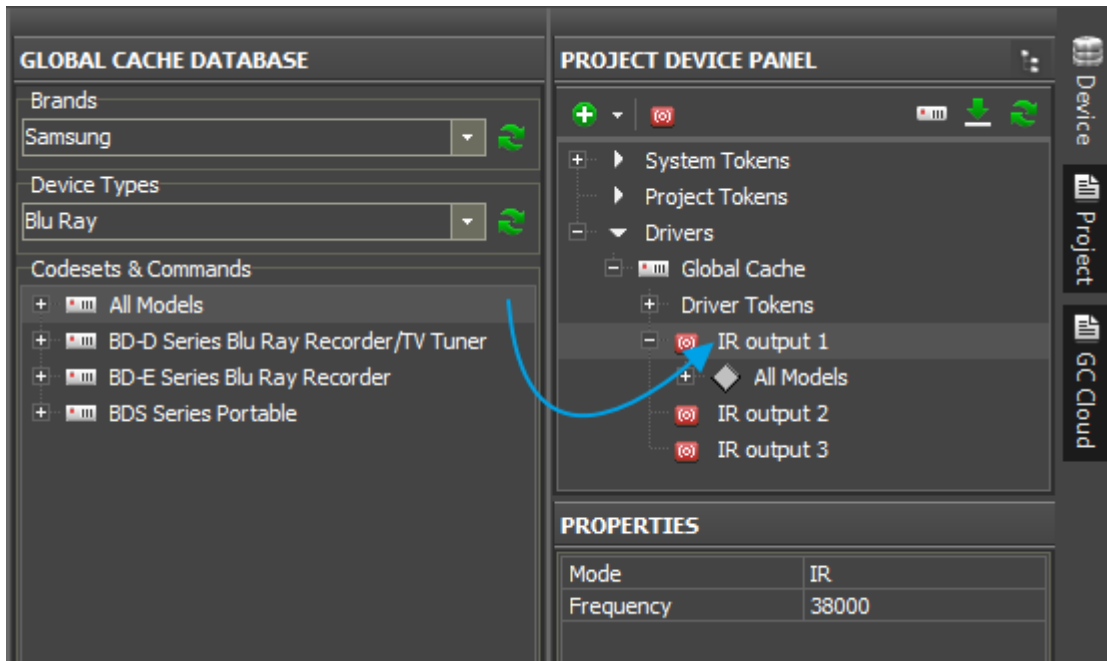
Click "**Update**" (connection to the Internet is required) to upload the list of manufacturers and commands.



The database structure:

- Brands - the equipment manufacturers
- Device Types - the equipment categories of manufacturers
- Codesets & Commands - the models and the corresponding IR commands

Drag the folder with commands to the Global Cache IR output in your project.
The commands are preset, you don't need to edit them:



Commands and folders can be moved to IR modules of GC modules in your project by dragging.

[↑ Back](#)

Learning IR Commands with GC-IRL IR Learner

[GC-IRL IR Learner](#) is a reader of IR commands for Global Cache.

It is available in two ways:

- as a separate device for connecting to the COM-port of PC or the converter GC-100-06(12,18,18R).
- as a reader of IR commands built-in in iTach IP2IR (WF2IR).

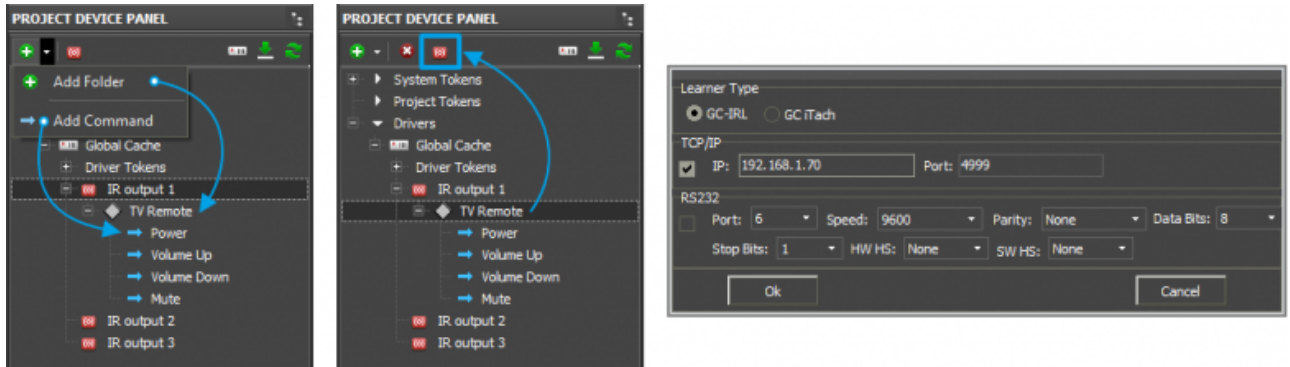
The reader allows you to create the list of IR commands which can be used in all iRidium projects for Global Cache.

Learning of IR commands is possible in two ways:

- To write commands directly on the IR output of the Global Cache driver in the iRidium project
- To create a custom database of IR commands in DEVICE BASE and fill it

The fast way of learning IR commands - inside the opened iRidium project:

1. Place the cursor on the Global Cache IR output
2. Click "(+) > add a folder" of IR commands (for convenience)
3. Click "(+) > add a command" several times to match the number of buttons of the remote control you are reading
4. Name all added commands according to the use of the buttons
5. Select the first command in the folder and click "Learning..."



6. Select the type of the learning device (Learner Type)

- GC-IRL - if you use the device GC-IRL IR Learner
- GC iTach - if you read commands through iTach IP2IR(WF2IR)

7. Select the learning interface (TCP/IP or RS232)

- TCP/IP - if you use iTach IP2IR or GC-IRL through GC-100

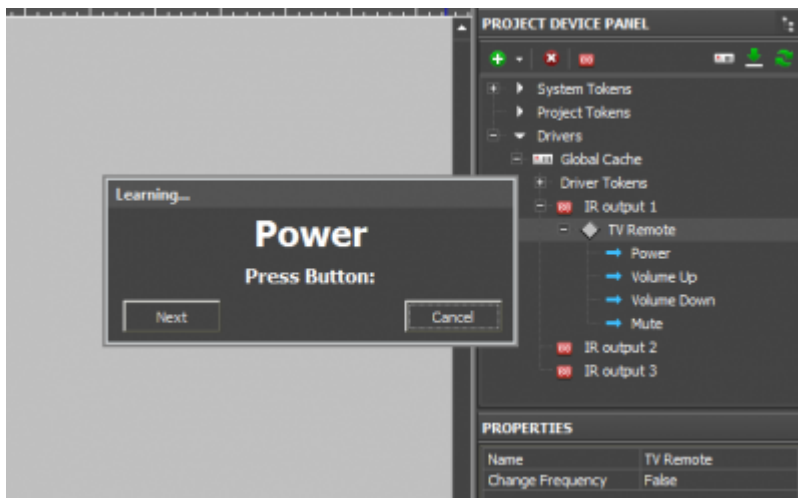
Port: 4999 - COM1 of GC-100-06(12,18,18R)

Port: 5000 - COM2 of GC-100-06(12,18,18R)

* Learning with the help of GC-IRL is not supported through iTach IP2SL(WF2SL)

- RS232 - if GC-IRL is connected to the COM-port of your PC

8. During learning click the button on the IR remote control which corresponds to the current name in the window. The remote control has to be directed to the device being read from the minimum distance:

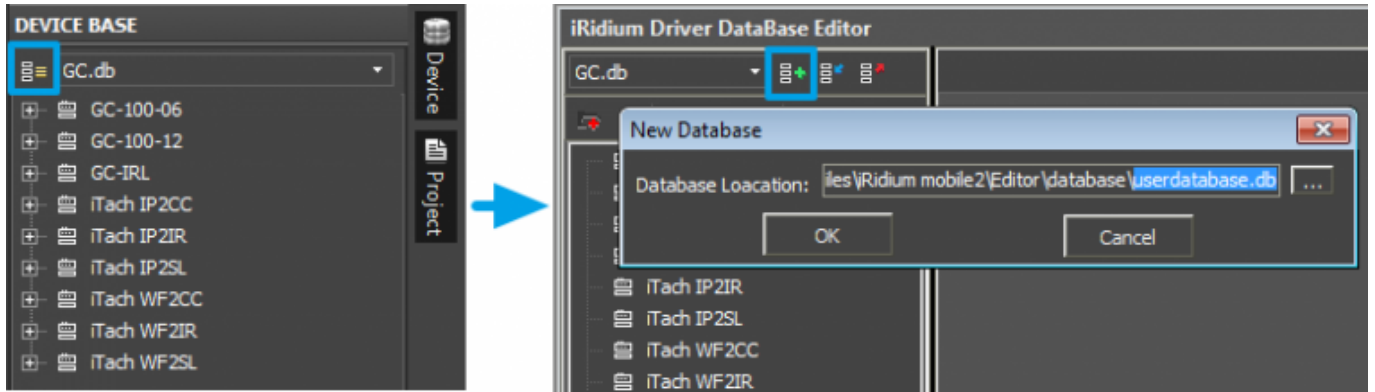


* If the command name in the window doesn't change when you click the button on the IR remote control - the learning failed! To find the reason of the problem, use the program iLearn from Global Cache. It will show error at connection. Check the connection with the converter.

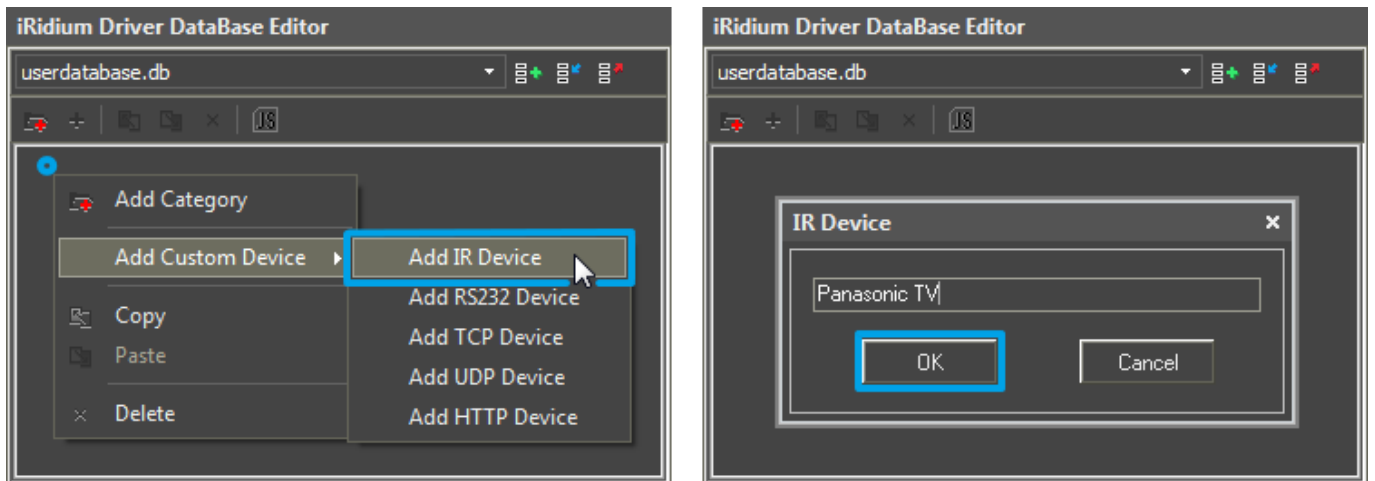
Creation of a custom database of IR commands (for quick adding to new projects)

The database of IR commands is located in DEVICE BASE as the standard iRidium database, but it can be formed manually. From the database you can easily add the IR device on the output of the Global Cache module (it can also be transferred from another project using File > Import).

DB Editor - the editor for creating custom databases:

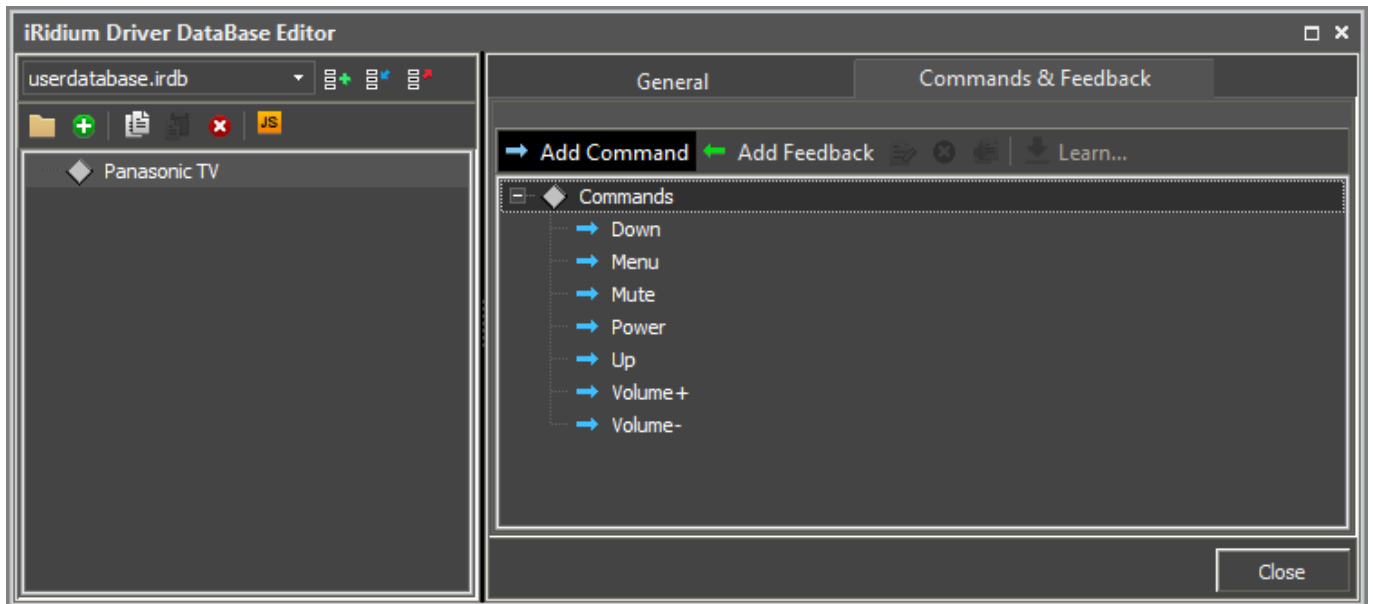


You can create an IR device only in a custom database as the standard database is not available for editing:



There is a minimal number of settings for a new IR device as it is a folder of commands for Global Cache.

In the Commands & Feedbacks tab form the list of IR commands which will be filled with data at start of learning (Learn...) through GC-IRL IR Learner:



It is not recommended to use special symbols in commands names.

Start of learning:

1. Connect [GC-IRL IR Learner](#) to the or GC-100-06(12,18,18R). You can also use the built-in Learner iTach IP2IR(WF2IR).
2. Place the cursor on the first command of the list created for the IR device.
3. Click "Learn ..." and select the learning settings as it is indicated in the beginning of the section.
4. When the learning window opens, click the button on the IR remote control corresponding to the current name in the window. The remote control has to be directed to the device being read from the minimum distance.

If the learning is successful, the command names will change automatically.

** Learnt IR commands are written in the database in bold.*

** If the command name in the window doesn't change when you click the button on the IR remote control - the learning failed! To find the reason of the problem, use the program iLearn from Global Cache. It will show error at connection. Check the connection with the converter.*

Close the database editor after saving the changes.

To add the received IR device in your iRidium project:

- Select your database from the Device Base drop-down list
- Drag the device from the database to the Global Cache IR output

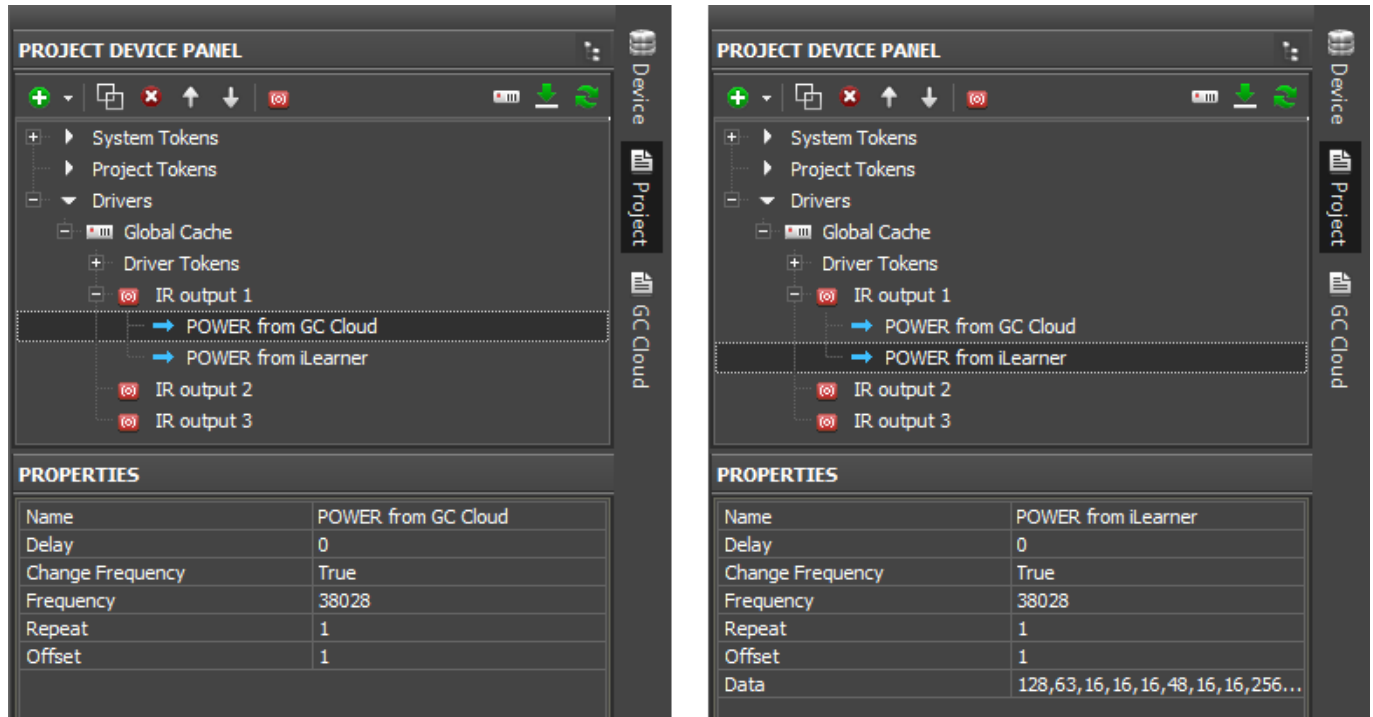
The frequency of command sending is indicated in the properties of the Global Cache IR output, when commands are added in the project.

Save the changes in the database and add the created IR device on any Global Cache IR output in your iRidium project.

[↑ Back](#)

Settings of IR Commands

IR command settings differ depending on the way of receiving IR commands (from the database or manually):



- **Delay** (ms) - the delay before sending commands. It is count from the last command sent (after receiving confirmation of command delivery, if it's set)
- **Change Frequency**: False/True - it allows you to select the frequency of command sending (Hz): by default (see IR outputs) or a unique frequency for this command
 - **Frequency** (Hz) - the frequency of command sending, if Change Frequency: True
- **Repeat** - how many times it is required to send the IR command (repetition is performed on the Global Cache level, but not because of the repeated sending)
- **Offset** - it is used only if Repeat > 1 for IR commands with preambles. A command preamble - it is a part which does not have to be repeated several times. Offset - it is preamble length which is measured in pairs of pulse-pauses comprising the IR command. It is always an odd number. See more [Global Cache API](#)
- **Data** - the IR command body. It is not displayed for all commands added from the Global Cache database. The command body contains the sequence of IR impulses which Global Cache will send to equipment.

[↑ Back](#)

Sending IR commands from Graphic Items

Create a visualization interface

IR commands are not initially linked to visualization interfaces by default. To send a command it has to be assigned to a button in your iRidium project.

Events at which commands assigned to Buttons can be sent:

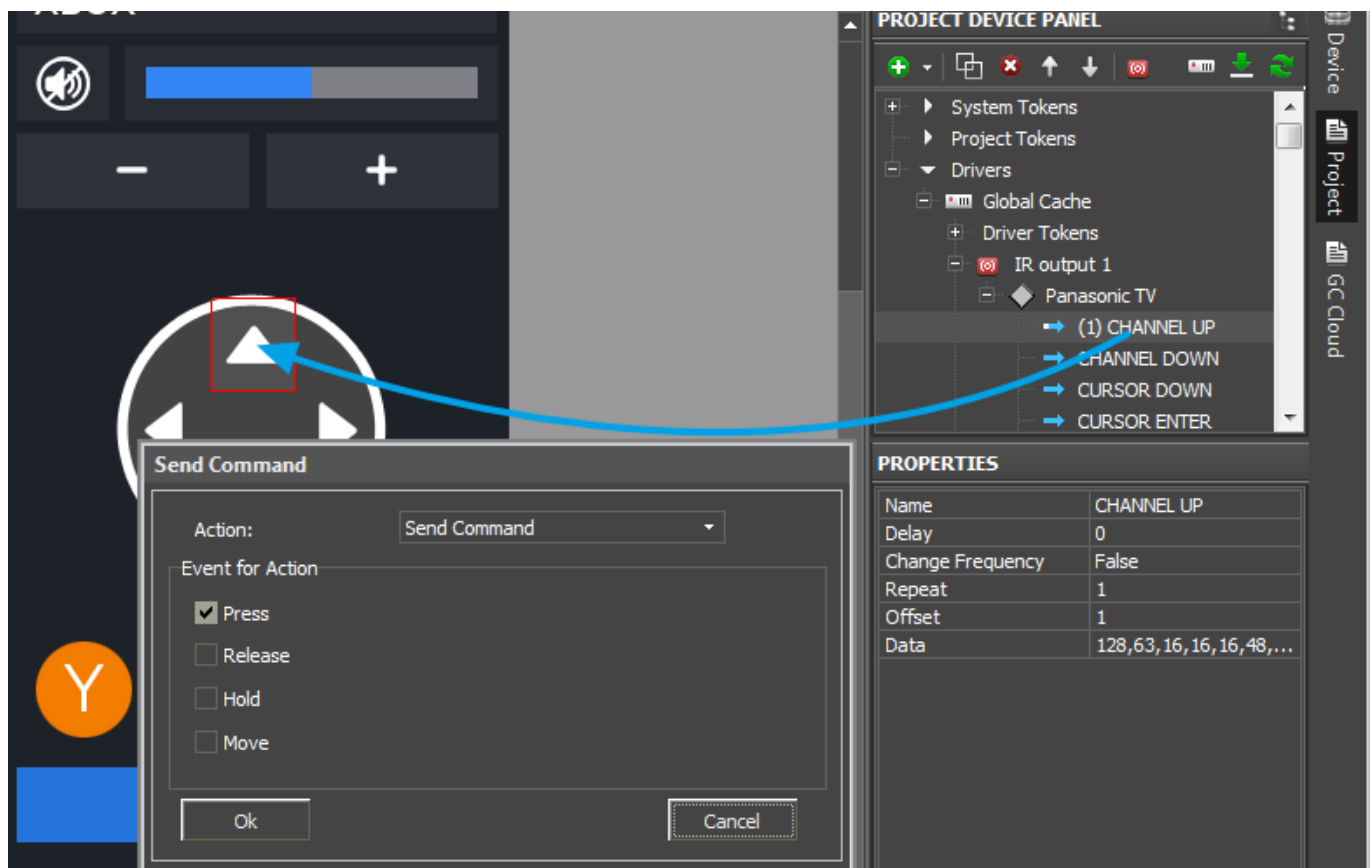
- **Press** - pressing on the graphic item
- **Release** - releasing the graphic item
- **Hold** - holding the graphic item (in cycles, until the item is released)

Events at which commands can be sent from scripts (see [iRidium Script API](#)):

- receipt of feedback with a particular value
- pressing on/releasing interface items
- timer expiration
- receipt of particular values from the panel sensors (GPS, accelerometer, gyroscope, ..)
- receipt of particular values from Levels

Sending IR commands when pressing on (releasing) Button

Drag an IR command to the graphic item and select the event of command sending in the **Send Command** dialog window:



- **Press** - pressing the item

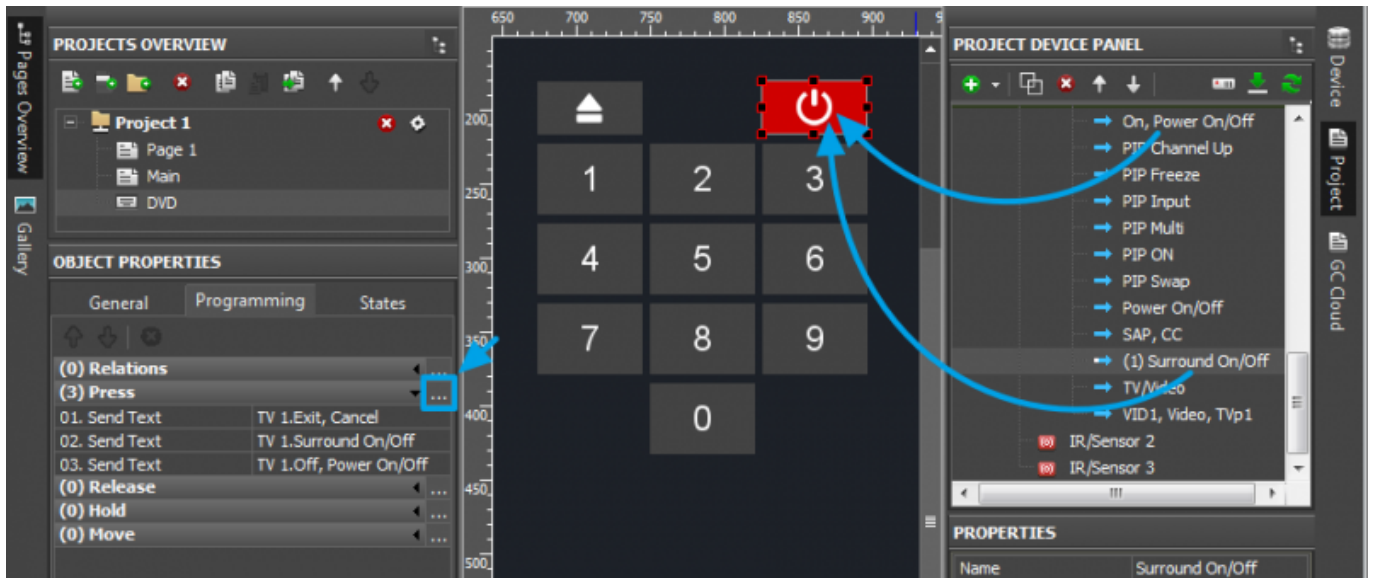
- **Release** - releasing the item

You can see added commands in Object Properties > Programming. Commands can be dragged from one event to another.

You can assign several several commands on one button:

Sending several IR commands from one button (macros, scenes)

To assign several commands drag them one by one on one button and assign them to one of the events for sending (Press or Release):

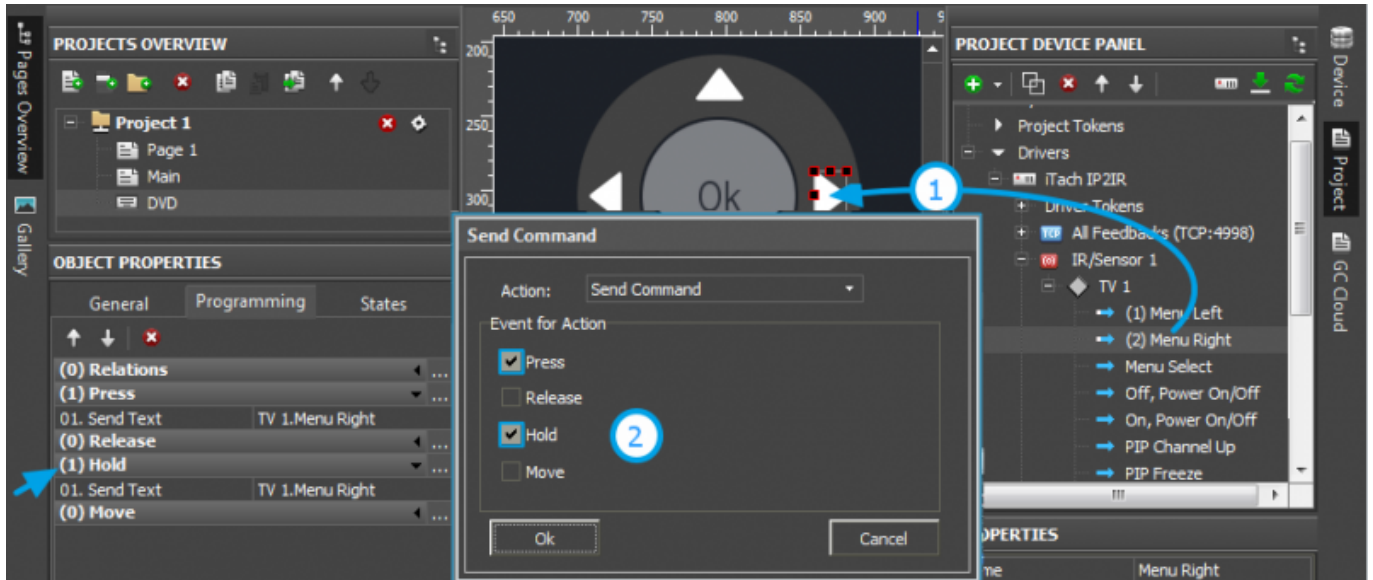


The macros editor for the Press (or Release) events will allow you to move the IR commands and separate them with delays (Delay), additional commands.

Sending IR commands on Hold

Hold - the event of "holding the item". It enables sending of one or several commands in a cycle all the time you hold the item (like the Volume + button on the IR remote control).

Activate the Press event for the item so it could react not only on holding but on pressing as well:



The Hold event settings are in Object Properties > General:

- **Hold Time** (ms) - the time after which the Hold event is activated when holding the item
- **Repeat Time** (ms) - the interval of repeating commands from the Hold macro.

The commands written in the Hold event will be sent in a cycle until the button is held with the interval indicated in Repeat Time.

Control from Levels, Trigger Buttons and other items

When working via IR it is not possible to set up sending of commands from Levels. It is due to the fact that an IR command contains a target action and you cannot change it without changing a part of the IR command.

In the case of IR commands *the control from Level is not possible*. Each Level value corresponds to a separate IR command and the majority of IR remote controls do not offer setting up of a particular volume value (they have only + and -).

In order to send two IR commands in turns or to create a complex sequence of command sending you need to use iRidium Script. It is a built-in programming language which enables setting up of required relations. See [iRidium Script API](#).

[↑ 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('Global Cache').SetParameters({
```

```
Host: "210.110.10.10", // the IP-address of the GC module  
Port: 4998, // main connection port  
Port1: 4999, // the port of connection to COM1 on GC (if it has COM-ports)  
Port2: 5000, // the port of connection to COM2 on GC (if it has COM-ports)  
SendMode: 0, // 0 - always connected, 1 - online only to send a command  
IrConfirm: 0, // 0 - IR with confirmation, 1 - IR without confirmation  
ScriptMode: 0, // 0 - send directly and to the script, 1 - send to the script only  
BackGroundMode: 0 // 1 - on, 0 - off
```

```
}); // Driver Name + Parameters
```

```
}
```

```
function External_1()
```

```
{
```

```
IR.GetDevice('Global Cache').SetParameters({
```

```
Host: "210.110.10.10", Port: 4998, Port1: 4999, Port2: 5000, SendMode: 0, IrConfirm: 0,  
ScriptMode: 0, BackGroundMode: 0});
```

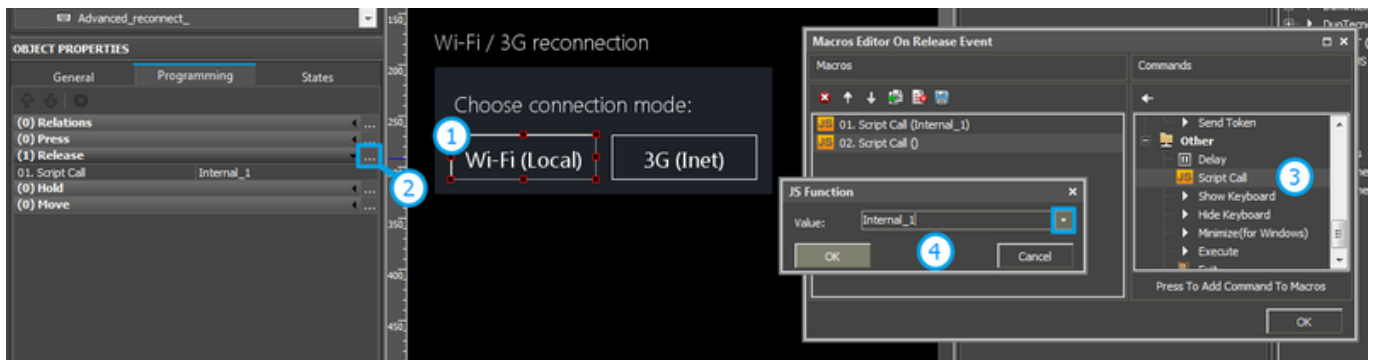
```
}
```

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 \(project\) >>](#)

