

# **GW-ETH-02**

## **IQRF Ethernet Gateway**

Firmware v1.06

## **User's Guide**



## Description

GW-ETH-02 is an IQRF gateway for connection between IQRF and Ethernet networks allowing remote monitoring, data collection and control of IQRF network.

Main components are: 32b microcontroller, Ethernet controller, EEPROM memory, serial Flash memory, SD memory card, IQRF transceiver module with antenna and RTCC.

The gateway can communicate outside the IQRF network via Ethernet. Configuration is possible via internal www server or from PC software *GW-ETH Tool* which can serve as a test tool and allows remote upgrade of GW firmware and upgrade of the application in internal TR module. A PC program written by the user can also be used. The application protocol is fully documented.



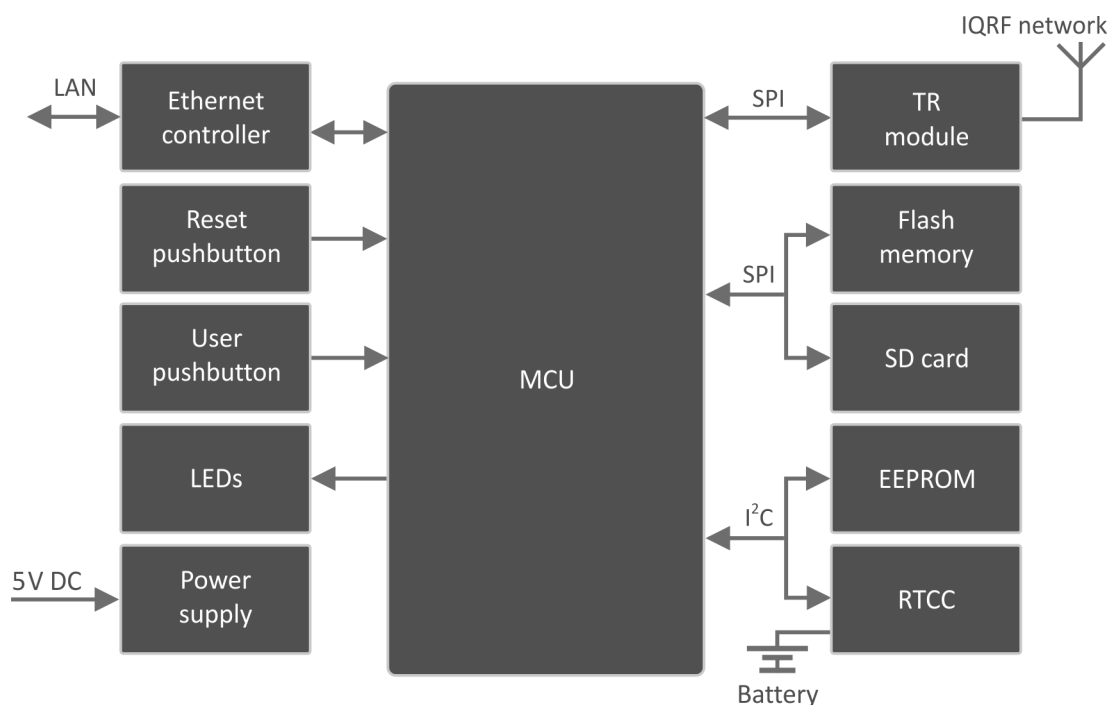
## Applications

- Remote monitoring and control
- Data acquisition
- Datalogger
- Interface to building/home automation
- Connection of more IQRF networks to single PC
- Acces to IQRF network from more locations
- Time synchronization from time servers

## Key features

- UDP server/client for communication with IQRF network
- HTTP/HTTPS server for remote management
- DDNS support
- SNTP client for getting date and time from Internet
- ICMP server allowing „ping“ to the equipment
- DHCP client for automated getting of IP address
- NBNS server for using names instead of IP addresses
- RTCC (real time clock/calendar)
- Documented application protocol
- Firmware upgrade via SD card
- Upgrade of application in internal TR module via Internet

## Block schematics



---

**Electrical specifications** (Typical values unless otherwise stated)

---

Power supply	5.0 ± 0.35 V DC
RTCC battery backup	CR2032 3 V, 215 mAh
Supply current from external source	200 mA
Temperature range	0 °C to +60 °C
Frequency range	868 MHz or 916 MHz (SW selectable)
RF output power	Up to 3.5 mW
Supported TR modules	TR54DA
Antenna	Built in TR module
EEPROM memory	16 Kb, serial I2C interface, 1 000 000 erase/write cycles typ.
Flash memory	8 Mb, serial SPI interface, 100 000 erase/write cycles typ.
SD card	Up to 2 GB
Ethernet	100BASE-TX
Dimensions	86 x 62 x 28 mm
Weight	70 g

---

**Absolute maximum ratings**

---

Stresses above those values may cause permanent damage to the device. Exposure to maximum rating conditions for extended periods may affect device reliability.

Supply voltage (VCC)	5.5 V
Storage temperature	-20 °C to +60 °C

---

**Hardware**

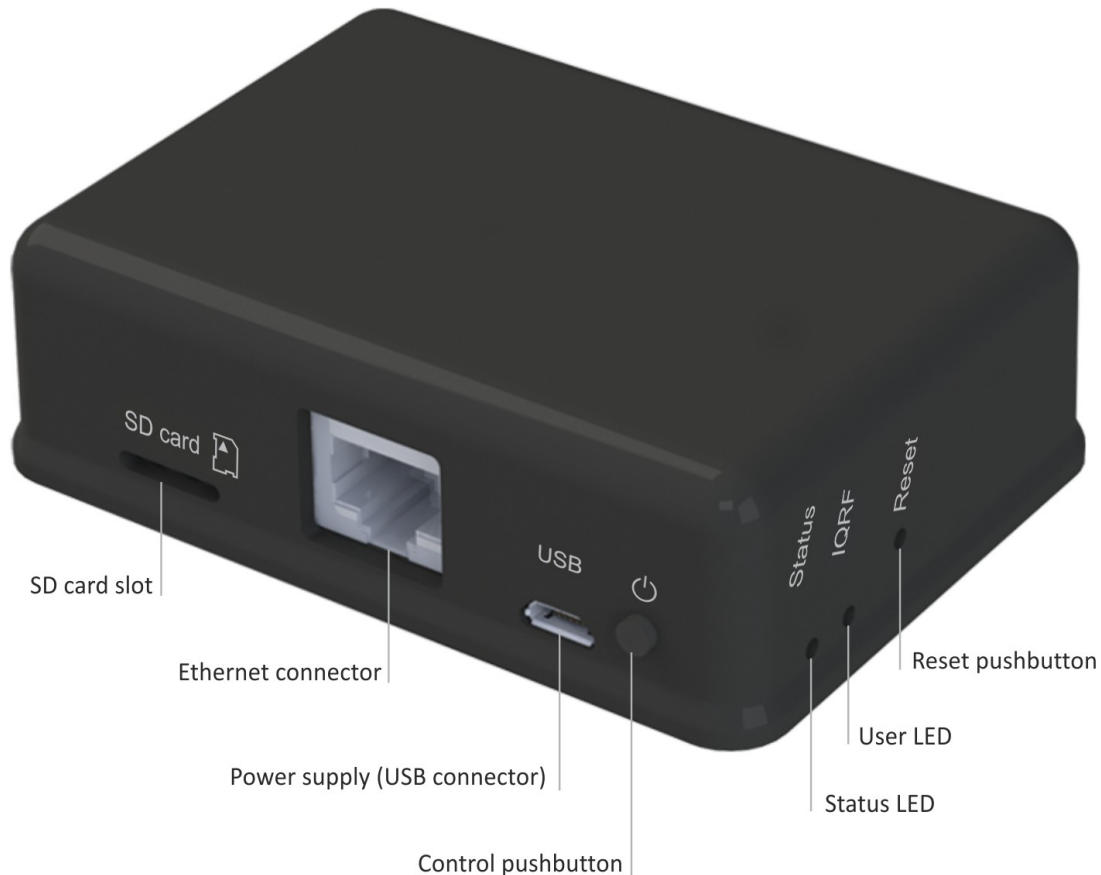
---

**Power supply**

GW-ETH-02 should be supplied by external stabilized 5 V DC via standard micro USB connector.

**Ethernet**

LAN should be connected via standard 8-pin RJ-45 connector.

**Control****Pushbuttons**

Both Reset and Control pushbutton functionality is fixed and can not be changed in application program.

**Reset**

Reset button can be pressed by a pin through a hole in the case. It is intended to completely initialize the gateway.

**Control**

- Short press: no effect.
- Long press > 10 s: configuration setup to factory values (even in the Bootloader mode, see below).

**LED indication**

- **Status LED** (green) indicates the operation of the gateway. See below (Bootloader flow chart).
- **User LED** (green) is connected to MCU (pin RC2) in TR module. It is fully under control of application program in TR module. The LED shines when the pin is driven high. It is not used in current Demo SW version.
- Two LEDs (red and green) inside the case are fully under control of application program in TR module and intended for development and service purpose. They are connected to TR pins dedicated to LEDs and supported by IQRF OS LED control functions.

---

## Operation

---

GW-ETH-02 performs the functionality given by firmware (FW) loaded in main MCU and by application program uploaded in TR module inside the GW.

The FW is fixed but can be upgraded by the user with a new version released by the IQRF manufacturer. See chapter *Bootloader* below.

Application program for the TR module can be written by the user oneself to implement specific requirements. See chapter *Demo software*.

## Services

- UDP server /client for communication with IQRF network and for remote GW management
- HTTP/HTTPS server offering web pages for remote management via a common browser
- HTTP client for getting and updating of public address from DDNS
- SNTP client for getting date and time from time server
- ICMP server allowing “ping” to GW from a remote host
- DHCP client for automated getting of IP address from the DHCP server
- NBNS server for using names instead of IP addresses within the LAN

## MAC address

Every device has a unique MAC address in the format 00 1F D5 xx xx xx, where:

- 00 1F D5 is the OUI dedicated to MICRORISC s.r.o.
- xx xx xx is a device serial number

Actual MAC address is printed on the label at the bottom of the GW-ETH-02 case.

## Modes of communication with internal IQRF module:

Three communication modes are available:

- **Whole packet**

According to the IQRF SPI protocol, the whole packet intended for the IQRF module must be created by the host. The GW transfers this to the module without changes. The same goes for a packet received by the host.

- **Only user data**

The host sends only its own data to the IQRF module in this mode. Packet creation according to the IQRF SPI protocol is up to the GW. The same goes for a packet received by the host from the GW.

- **Datalogger**

It is not possible to communicate externally with the TR module inside the GW in the Datalogger mode. GW communication with the IQRF network depends on the application in internal TR module. All data sent via SPI from the TR module to the GW are stored in circular buffer in the GW. Every packet is equipped with numeric code and time stamp. Data can be freely read from this buffer via the HTTPS interface or user data can be sent via HTTPS to the TR module. Refer to the Datalogger User's guide for more information.

The buffer is implemented in RAM inside the main MCU with no battery backup. FW v1.07 will support an additional non-volatile buffer implemented in serial Flash memory.

## Date and time

The GW contains the real time RTCC module which can be configured in several ways:

- From Internet time server via the SNTP protocol. Updated every 60 s.
- Manually from the GW web interface.
- Using the GW-ETH Tool PC software.

## Safe Mode

If this mode is activated the GW does not respond to several commands via the UDP channel (exact list see the Application protocol below). Full communication and GW configuration as well as repeated disabling the Safe Mode is possible via HTTPS only.

## Factory setup

This initial setup can be restored whenever (in both GW and BL modes). This can be useful if it is not possible to establish communication due to wrong configuration.

- NBNS name „iqrf“
- Get IP address from DHCP server automatically on
- IP address 192.168.0.254
- Subnet mask 255.255.255.0
- Gateway 192.168.0.1
- Primary DNS server: 192.168.0.1
- Secondary DNS server 0.0.0.0
- Authorization when enter the internal www page
  - username „iqrf“
  - password „iqrf“
- DDNS off
- Getting date and time from the time server on
- Time server europe.pool.ntp.org
- Summer / winter time distinguishing on
- Time zone (GMT+01:00) Prague
- Type of communication with the TR module Whole Packet
- GW port for the application protocol 55300
- Host port for the application protocol 55000

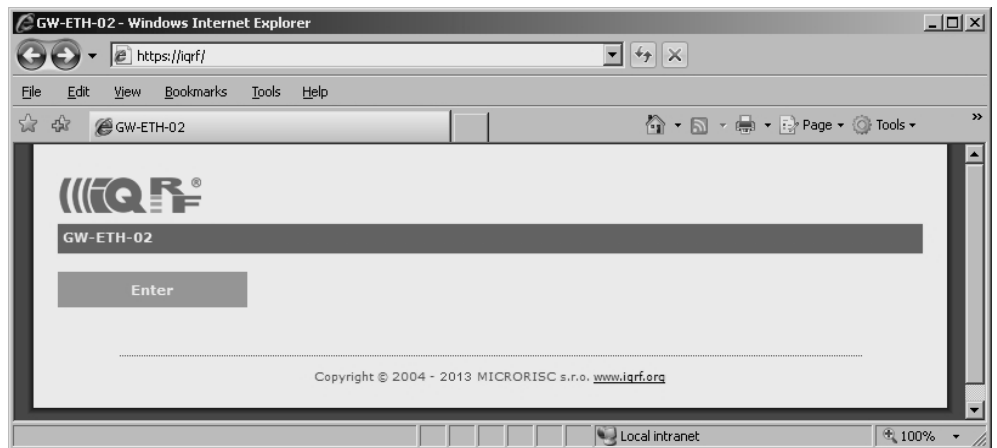
## Firmware upgrade

The bootloader is a part of the GW software allowing to upload new firmware versions (delivered by the GW manufacturer) to the GW by the user. The BL mode is entered at the device start up (see *chapter LED Indication*). If the SD card has a new firmware file in root directory, firmware will be upgraded. After upgrading the GW mode will be entered automatically. No setup changes are allowed in the BL mode. In need, factory values can be setup manually (see *Factory setup*). MAC address of given device is included in the BL. Thus, the device has the same MAC address after the upgrade.

## Connection to internal www pages in local network

Internal web interface is intended for configuration and monitoring of the GW state. Changes should be done by the skilled person only. Possible problems in wrong setup can be solved by restoring the factory values.

The GW must be switched on and connected to local LAN network. The following text assumes the factory setup. Enter „<https://iqrf/>“ or „[https://<ip\\_address>](https://<ip_address>)“ respectively. The page depicted below will appear. Click the Enter button and enter login data (login name as well as the password are „iqrf“). Working with the pages are intuitive, the state of the device can be monitored and all its parameters can be setup there.



## GW-ETH Tool

The GW-ETH Tool program is free to download from <http://www.iqrf.org/downloads>. The handling is intuitive, additionally the Help is available. This tool offers the same possibilities for monitoring and configuration of the equipment like internal www server and additionally the following functions are available:

- The *Terminal* tab enables communication with the TR module and then with the IQRF network via a terminal window (like in the IQRF IDE).
- The *Upload* tab enables to change the application in the TR module (file .HEX or coded file .IQRF).
- The *Communication* tab displays all communication with the equipment and offers a useful tool for development and debugging of the user software communicating with the GW.
- The program allows communication using text files. It can send a packet from and save received packet to a text file. This can be useful for users not being able to program the UDP communication but familiar with opening/saving files. See the GW-ETH Tool Help for more information.

## IQRF DNS

IQRF DNS server is provided by the MICRORISC s.r.o. and is intended as a replacement of the The Dynamic DNS server. It allows users a remote connection to an equipment (within Internet) without knowing current public IP address of the equipment. If the equipment is configured and local redirection is set properly it is possible to use the IQRF DNS names instead of IP addresses of the equipment in web browsers - see the example below. The only thing what the user should know is the MAC address of the equipment. The same approach can be applied also in case of the GW-ETH-Tool program. For details see the GW-ETH Tool Help. In following explanation the MAC address 001FD5010203 and IP address 10.100.20.200 are used.

Registering the equipment in the IQRF DNS server database <http://www.iqrfdns.org/?IDIP=001FD5010203>

The GW-ETH-02 client sends this command automatically and regularly (if this is activated in the GW setup) which keeps a record of MAC address of the equipment and corresponding IP address (for remote GW connecting) in the server database. The server responds with public IP address which the command was sent from. Thus, the equipment finds out its public IP address in the format:

*Current IP Address: 10.100.20.200*

Request for IP address <http://www.iqrfdns.org/?IP=001FD5010203>

The http client of the GW-ETH Tool program sends this command to find out the IP address assigned to given MAC address in the server database. The server returns the result in the format:

*Requested IP Address: 10.100.20.200*

If the MAC address is not found in the database the server returns:

*IQRFDNS Message: Requested device is not registered.*

Connection to the equipment - redirection <http://www.iqrfdns.org/?ID=001FD5010203>

The command is intended to be entered to the www browser for connection to www interface of the GW-ETH-02. The server finds requested ID (MAC address) in the database and redirects it to assigned IP address. The GW must have set and activated the IQRF DNS server as the DDNS.

If requested ID is not found the server returns:

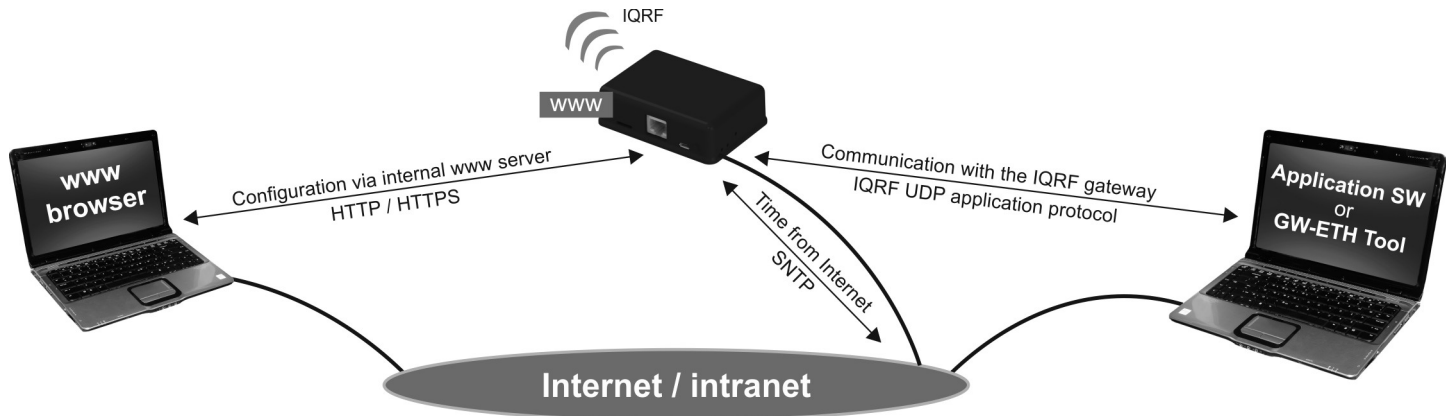
*IQRFDNS Message: Requested device is not registered.*

If there is no answer from the GW after redirection the server returns:

*IQRFDNS Message: Device is not responding. It is either offline or its IP address has changed. Try it again after xx:xx min.*

## GW usage

GW-ETH-02 can communicate outside the IQRF network via Ethernet using internal www server, the GW-ETH Tool or a PC program written by the user based on the open application protocol using the UDP transport level.





## Application protocol

The communication is realized with the application protocol described below. It uses the transport protocol UDP according to the OSI reference model. For establishing a connection it is necessary to know the UDP port where the GW is listening and its IP address or the NetBIOS name (Host Name).

GW answers and response to commands to any asking device. Asynchronous messages are sent to broadcast address until the communication is established. After communication establishing the messages are sent to address of the last communicating host. Thus, more hosts (e.g. servicing programs on more PCs) can communicate with the GW.

If the IP address or NetBIOS of the GW and the port number are known it is just possible to establish the connection. Otherwise the request for a GW identification can be sent to broadcast address (in a single LAN segment only – routers filter packets with broadcast addresses.) All GWs receiving this request will answer which allows to get actual IP address of given GW.

### Packet description

The packet structure is the same for both communication directions:

HEADER (H = 9 B)	DATA (D = 0 – 497 B)	CRC (2 B)
------------------	----------------------	-----------

For receiving, GW checks:

- GW\_ADR validity – see the packet header
- packet length must not be lower than HEADER + CRC
- packet length must not be higher than HEADER + DATA MAX + CRC
- CRC

### Header

Packet header has a fixed length:

0	1	2	3	4	5	6	7	8
GW_ADR	CMD	SUBCMD	RES	RES	PACID_H	PACID_L	DLEN_H	DLEN_L

GW_ADR	identification address of the equipment of the GW-ETH-02 type (value: 0x22)
CMD	command of a packet (values: 0x01 ... 0x7F)
SUBCMD	answer to given command: CMD = CMD   0x80 (bit 7 of given CMD set)
RES	auxiliary information of a command (values: see description below)
RES	reserved
PACID_H	packet identification – upper byte (values: 0x00 ... 0xFF)
PACID_L	packet identification – lower byte (values: 0x00 ... 0xFF)
DLEN_H	data length – upper byte (values: 0x00, 0x01)
DLEN_L	data length – lower byte (values: 0x00 ... 0xFF)

The packet identification (PACID) can be any number 0x0000 – 0xFFFF according to the host specification. The GW always copies this number to the answer to given packet.

### CRC

The check part has a fixed length:

H+D	H+D+1
CRC_H	CRC_L

CRC type	CRC-16-CCITT
Polynomial	0x1021 ( $x^{16} + x^{12} + x^5 + 1$ )
Initial value	0x0000

CRC is calculated from the HEADER and DATA parts.

## Communication initiated by the host (e.g. GW-ETH Tool)

### Get GW identification

Getting a GW identification. It is intended to identification of the GW which the connection is established with.

#### *host -> GW direction*

CMD	0x01	returns the GW identification
SUBCMD	0x00	no information
DLEN	0x0000	no data

#### *GW -> host direction*

CMD	0x81	answer to the 0x01 command
SUBCMD	0x00	no information
	0x10	bootloader
DATA		text with GW identification. See Data identification

### Get GW status

The GW returns the information about its state.

#### *host -> GW direction*

CMD	0x02	returns the GW status
SUBCMD	0x00	no information
DLEN	0x0000	no data

#### *GW -> host direction*

CMD	0x82	answer to the 0x02 command
SUBCMD	0x00	no information
DATA		see Status data

### Write data to TR module

Data according the IQRF SPI protocol is written to the TR module. The answer is an acknowledge of writing to the module. If the answer is requested from the module it is sent with the actual command – see Communication initiated by the GW. According to GW configuration either just actual data or the whole packet is written.

#### *host -> GW direction*

CMD	0x03	write data to the TR module
SUBCMD	0x00	no information
DATA		data for the TR module

#### *GW -> host direction*

CMD	0x83	answer to the 0x03 command
SUBCMD	0x50	write OK
	0x60	write Error (number of data = 0 or more than TR buffer COM length)
	0x61	write Error (SPI bus busy)
	0x62	write Error (IQRF - CRCM Error)
DLEN	0x0000	no data

### Read GW configuration

Reading of configuration parameters from EEPROM in GW.

#### *host -> GW direction*

CMD	0x06	reading of GW configuration
SUBCMD	0x00	no information
DLEN	0x0000	no data

#### *GW -> host direction*

CMD	0x86	answer to the 0x06 command
SUBCMD	0x00	no information
DATA		see Configuration data

**Write GW configuration**

Writing of configuration parameters to EEPROM in GW.

```
host -> GW direction
CMD      0x07      write GW configuration
SUBCMD   0x00      no information
DATA     see Configuration data

GW -> host direction
CMD      0x87      answer to the 0x07 command
SUBCMD   0x50      write OK
          0x60      write Error (invalid number of configuration bytes)
DLEN     0x0000    no data
```

---

**Write to RTCC in GW**

Writing of time and date to the RTCC in GW

```
host -> GW direction
CMD      0x08      write to RTCC
SUBCMD   0x00      no information
DB1      GW time – seconds (values see Time and data coding)
DB2      GW time – minutes
DB3      GW time – hours
DB4      GW data – day of week
DB5      GW data – day
DB6      GW data – month
DB7      GW data – year

GW -> host direction
CMD      0x88      answer to the 0x08 command
SUBCMD   0x50      write OK
          0x60      write Error (invalid values)
DLEN     0x0000    no data
```

---

**Change autentization**

Changing of username and password for access to GW via www.

```
host -> GW direction
CMD      0x09      change autentization
SUBCMD   0x00      no information
DB1      username: text
to       (max. 15 characters, not used ones replace with zeroes)
DB15

DB16     old password: text
to       (max. 15 characters, not used ones replace with zeroes)
DB30

DB31     new password: text
to       (max. 15 characters, not used ones replace with zeroes)
DB45

GW -> host direction
CMD      0x89      answer to the 0x09 command
SUBCMD   0x50      write OK
          0x60      write Error (invalid old password)
DLEN     0x0000    no data
```

---

**Read TR Module Info in GW**

Getting information about the module in the GW

*host -> GW direction*

CMD	0x11	read the TR Module Info from the GW
SUBCMD	0x00	no information
DLEN	0x0000	no data

*GW -> host direction*

CMD	0x91	answer to the 0x11 command
SUBCMD	0x00	no information
DATA		see Module Info

---

**Reset GW (GW as well as the bootloader)**

Remote GW reset (initialization).

*host -> GW direction*

CMD	0x12	GW reset
SUBCMD	0x00	no information
DLEN	0x0000	no data

*GW -> host direction*See Communication initiated by the GW – Sending the GW status message.

---

**TR module reset**

Remote reset of the TR module.

*host -> GW direction*

CMD	0x13	TR module reset
SUBCMD	0x00	no information
DLEN	0x0000	no data

*GW -> host direction*

CMD	0x93	answer to the 0x13 command
SUBCMD	0x50	OK command
	0x60	Error command
DLEN	0x0000	no data

---

**Sending a command not implemented in GW***host -> GW direction*

CMD	?	unknown command
SUBCMD	?	arbitrary information
DATA		arbitrary data

*GW -> host direction*

CMD	?   0x80	answer to an unknown command
SUBCMD	0x60	packet error
DLEN	0x0000	no data

---

## Communication initiated by the GW (asynchronous message)

### Send date from TR module

Sends data from the TR module according to the IQRF SPI protocol. According to the settings either just actual data or the whole packet is send.

<i>GW -&gt; host direction</i>		
CMD	0x04	send data from TR module
SUBCMD	0x00	no information
DATA		data from TR module

### Send GW status message

It is a message from the GW about its state. The information is indicated by the SUBCMD byte.

<i>GW -&gt; host direction</i>		
CMD	0x05	GW status message
SUBCMD	0x01	GW reset, sent after switching on
	0x02	change its own IP address (only if the DHCP is active)
	0x03	power supply: external
	0x04	power supply: acumulator
	0x10	reserved for upgrade (sent by the bootloader only)
DATA		text with GW identification. See Identification data.

### Identification data

Identification data consists of several texts separated by 0x0D 0x0A.

1. - GW name e.g.: „GW-ETH-02“
2. - GW version e.g.: „v0.30b“
3. - MAC address e.g.: „00 11 22 33 44 55“
4. - TCP/IP Stack version e.g.: „v4.51“
5. - IP address of GW e.g.: „192.168.2.100“
6. - Net BIOS Name e.g.: „iqr“ “ 15 characters
7. - IQRF module OS version e.g.: „v2.08“
8. - Public IP address e.g.: „213.214.215.120“

### Status data

GW status data for the *Get GW status* command.

DB1		TR module status (see the IQRF SPI protocol)
DB2		no used
DB3	0x01	supplied from external source
DB4		GW time – seconds (see Time and date coding)
DB5		GW time – minutes
DB6		GW time – hours
DB7		GW date – day of the week
DB8		GW date – day
DB9		GW date – month
DB10		GW date – year
DB11		no used
DB12		no used

**Configuration data**

GW configuration data for reading and writing to GW.

DB1		... M byte of GW IP address
DB2		... .
DB3		... .
DB4		... L byte of GW IP address
DB5		... M byte of Subnet Mask IP address
DB6		... .
DB7		... .
DB8		... L byte of Subnet Mask IP address
DB9		... M byte of Gateway IP address
DB10		... .
DB11		... .
DB12		... L byte Gateway IP address
DB13		... M byte of Primary DNS Server IP address
DB14		... .
DB15		... .
DB16		... L byte of Primary DNS Server IP address
DB17		... M byte of Secondary DNS Server IP address
DB18		... .
DB19		... .
DB20		... L byte of Secondary DNS Server IP address
DB21	0x00	... DHCP Disabled
	0x01	... DHCP Enabled
DB22		... NetBIOS name: text
to		... (max. 15 characters, not used ones replace with zeroes)
DB36		...
DB37	0x00	... communication with TR module – whole packets according to the IQRF SPI protocol
	0x01	... communication with TR module – user data only
	0x02	... communication with TR module – Datalogger Mode
DB38		... local UDP of GW port – H byte
DB39		... local UDP of GW port – L byte
DB40		... remote UDP of GW port – H byte
DB41		... remote UDP of GW port – L byte
DB42	0x00	... DDNS Client off
	?	... DDNS Client - polling period 1 – 255 min
DB43		... DDNS domain name: text
to		... (max. 30 characters, not used ones replace with zeroes)
DB72		...
DB73		... DDNS user name: text
to		... (max. 15 characters, not used ones replace with zeroes)
DB87		...
DB88		... DDNS password: text
to		... (max. 15 characters, not used ones replace with zeroes)
DB102		...

---

DB103	0x00	... DDNS Provider Index:	www.iqrfdns.org
	0x01	...	www.dyndns.org
	0x02	...	www.no-ip.com
	0x03	...	www.dnsomatic.com
DB104	0x00	... SNTP Disabled	
	0x01	... SNTP Enabled	
DB105	0x00	... Daylight Savings Disabled	
	0x01	... Daylight Savings Enabled	
DB106	0x00	... Time Server Index:	pool.ntp.org
	0x01	...	europa.pool.ntp.org
	0x02	...	asia.pool.ntp.org
	0x03	...	oceania.pool.ntp.org
	0x04	...	north-america.pool.ntp.org
	0x05	...	south-america.pool.ntp.org
	0x06	...	africa.pool.ntp.org
DB107		... Time Zone Index	see Time zone coding
DB108		... Datalogger Packet Size	(1 - 64 B)
DB109	0x00	... Safe Mode Disabled	
	0x01	... Safe Mode Enabled	

---

### Time zone coding

0x00	(GMT-12:00)	0x0B	(GMT-02:00)	0x16	(GMT+05:45)
0x01	(GMT-11:00)	0x0C	(GMT-01:00)	0x17	(GMT+06:00)
0x02	(GMT-10:00)	0x0D	(GMT)	0x18	(GMT+06:30)
0x03	(GMT-09:00)	0x0E	(GMT+01:00)	0x19	(GMT+07:00)
0x04	(GMT-08:00)	0x0F	(GMT+02:00)	0x1A	(GMT+08:00)
0x05	(GMT-07:00)	0x10	(GMT+03:00)	0x1B	(GMT+09:00)
0x06	(GMT-06:00)	0x11	(GMT+03:30)	0x1C	(GMT+09:30)
0x07	(GMT-05:00)	0x12	(GMT+04:00)	0x1D	(GMT+10:00)
0x08	(GMT-04:00)	0x13	(GMT+04:30)	0x1E	(GMT+11:00)
0x09	(GMT-03:30)	0x14	(GMT+05:00)	0x1F	(GMT+12:00)
0x0A	(GMT-03:00)	0x15	(GMT+05:30)	0x20	(GMT+13:00)

### Time and date coding

For individual values the BCD code is used. The upper nibble (UN) means tens and the lower nibble (LN) means units.

*Example (seconds):*

DB = 0x53 means 53 s

*Legal ranges:*

seconds	UN = 0 – 5	LN = 0 – 9	min. 0	max. 59
minutes	UN = 0 – 5	LN = 0 – 9	min. 0	max. 59
hours	UN = 0 – 2	LN = 0 – 9	min. 0	max. 23
day	UN = 0 – 3	LN = 0 – 9	min. 1	max. 31
month	UN = 0 – 1	LN = 0 – 9	min. 1	max. 12
year	UN = 0 – 9	LN = 0 – 9	min. 8	max. 99 (2008 - 2099)
day of week	UN = 0	LN = 0 – 6	min. 0	max. 6 (0 – Sunday, 1 – Monday, ... )

These ranges are checked during writing to the GW. If out of range the packet with SUBCMD = *write Error* is returned.

### Module Info data

Information about the TR module in the GW.

DB1	module ID – MSB
DB2	module ID
DB3	module ID
DB4	module ID – LSB
DB5	OS version (upper 4b: major version, lower 4b: minor version)
DB6	microcontroller type (2: PIC16LF88, 3: PIC16F886)
DB7	OS build: LSB
DB8	OS build: MSB

### Commands blocked in Safe Mode

The following commands and functions are blocked in the Safe Mode when communicating via the UDP channel. GW return value is 0x60 in these cases.

0x07	write GW configuration
0x08	write to RTCC
0x09	change authentication
0x12	GW reset
0x13	TR module reset

TR module upload  
GW upload



---

***Demo software***

---

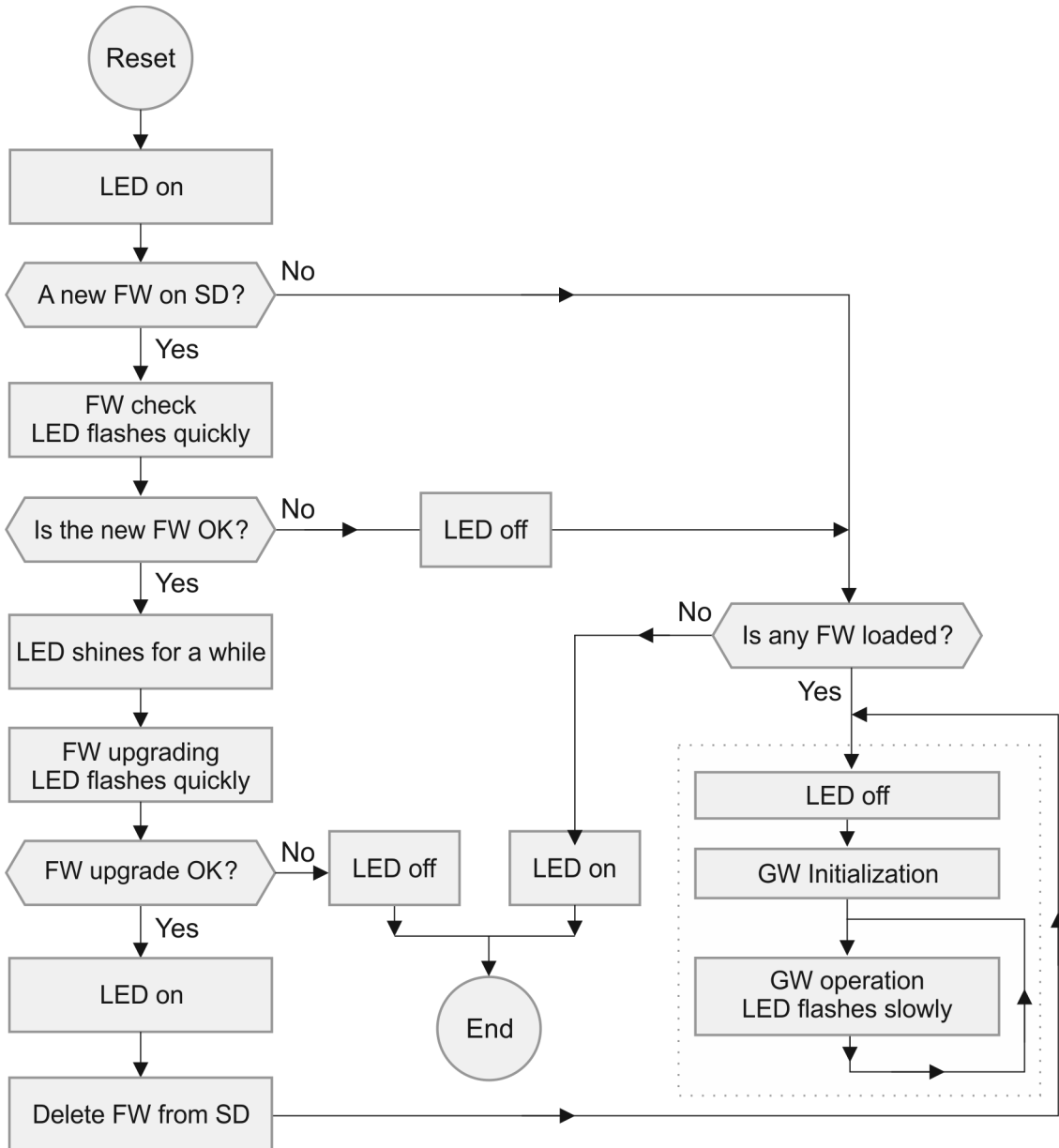
To implement specific requirements, application program for the TR module can be written by the user oneself. Refer to datasheet of the TR module, IQRF Quickstart guide, IQRF OS User's and Reference guides and IQRF examples.

The E03-TR (one of IQRF basic examples) is uploaded in internal TR module as a demo provided with this gateway. It works as a bidirectional RF–SPI communication between IQRF wireless and main MCU inside the gateway which transfers the data from/to Ethernet interface or to the datalogger.

## Bootloader

FW can be upgraded by a new version released by the IQRF manufacturer (.HEX file). To accomplish an upgrade, download the HEX file from product web page, store it to SD card (to the root directory), insert the card into the GW and press the Reset button. The new FW is automatically loaded and then the device is switched to operational mode. After successful upgrade the HEX file is deleted from SD card. It is restricted to load a FW older than the current one.

Boot procedure is indicated by Status LED and follows the flowchart:



---

**Product information**

---

**Pack list**

- GW-ETH-02
- TR module TR-54DA inside, with E03-TR demo example uploaded
- Power source TY-A6-microUSB (5V DC, 500 mA, stabilized, with micro USB connector)
- Printed Brief User's Manual

**Ordering code**

- GW-ETH-02      IQRF Ethernet gateway

**Firmware history**

- v1.06              First release.

**Document history**

- 130702              Bugs in Ethernet bit rate and temperature ranges fixed.
- 130613              Chapter *Key features* and LEDs description slightly precised.
- 130612              First release.

---

# Sales and Service

---

## Corporate office

MICRORISC s.r.o., Delnicka 222, 506 01 Jicin, Czech Republic, EU  
Tel: +420 493 538 125, Fax: +420 493 538 126, [www.microrisc.com](http://www.microrisc.com)

## Partners and distribution

Please visit [www.iqrf.org/partners](http://www.iqrf.org/partners)

---

## Quality management

*ISO 9001 : 2009 certified*

*Complies with Directive 2002/95/EC (RoHS)*



## Trademarks

*The IQRF name and logo and MICRORISC name are registered trademarks of MICRORISC s.r.o.  
PIC, SPI, Microchip, RFM and all other trademarks mentioned herein are property of their respective owners.*

## Legal

*All information contained in this publication is intended through suggestion only and may be superseded by updates without prior notice. No representation or warranty is given and no liability is assumed by MICRORISC s.r.o. with respect to the accuracy or use of such information.*

*Without written permission it is not allowed to copy or reproduce this information, even partially.*

*No licenses are conveyed, implicitly or otherwise, under any intellectual property rights.*

*The IQRF® products utilize several patents (CZ, EU, US)*

---

**On-line support: [support@iqrf.org](mailto:support@iqrf.org)**

---