IQRF

Quick Start Guide

For IQRF OS v4.03D

Smarter Wireless. Simply.

Content

1 Components required 3
2 Development hardware 3
2.1 Development set 3
2.2 TR transceivers 3
2.3 Programmer 4
2.4 End devices 4
2.5 USB device 4
3 Typical usage of kits 5
3.1 First demonstration 5
3.2 Point-to-point 5
3.3 Network 6
4 Two application approaches 6
4.1 Without programming 6
4.2 Programming in C 6
5 Development software 7
5.1 PC software 8
IQRF IDE 8
USB drivers 8
CC5X 8
Text editor 8
5.2 Installation 8
5.3 Other PC programs operating with USB 8
5.4 Test 8
5.5 IQRF IDE usage 9
Programming and uploading procedure 9
Other development, test and service utilities 9
6 C-programming 10
6.1 Design steps 10
6.2 Basic TR modes 11
6.3 Debug 11
6.4 Application examples 11
7 Quick start with DPA without programming 12
DS-DPA-02 development set 12
1. Preparation 12
2. Upload 13
3. Bonding 15
4. Discovery 15
5. Examples 15
6. Unbonding 16
8 Recommended documentation 17
9 Document revision 17
10 Sales and Service 18
10.1 Corporate office 18
10.2 Technology and development 18
10.3 Partners and distribution 18
10.4 Quality management 18
10.5 Trademarks 18
10.6 Legal 18
10.7 On-line support 18

How to start the first IQRF design

IQRF start-up is also illustrated on IQRF Video tutorial set.

1 Components required

Just the following tools are needed:

- HW: IQRF **development set**
- SW: IQRF **Startup package** (free) including IQRF IDE development environment
- PC: Windows Vista, Windows 7, Windows 8, Windows 8.1 or Windows 10

2 Development hardware

2.1 Development set

IQRF development set **DS-START-04** (see the picture below) or **DS-DPA-02** (see chapter *DS-DPA-02 development set*) contains all HW and SW needed for effective start-up.

DS-START-04:

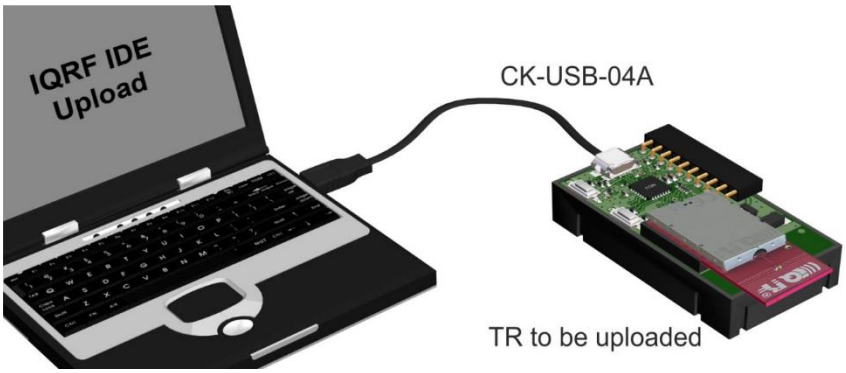
• **TR-72DAT**	IQRF transceiver	3 pcs
• **CK-USB-04A**	IQRF programmer and debugger	1 pcs
• **DK-EVAL-04A**	Universal portable development kit for TR transceivers	2 pcs
• CAB-USBABMICRO	Micro USB cable 18.5 cm	1 pcs
• USB flash drive	Software and documentation	1 pc

2.2 TR transceivers

RF transceivers **TR-72DA** fitting to SIM connector. They are fully programmable under IQRF OS operating system and allow to utilize the DPA approach (see chapter Quick start with DPA without programming) for applications without programming.

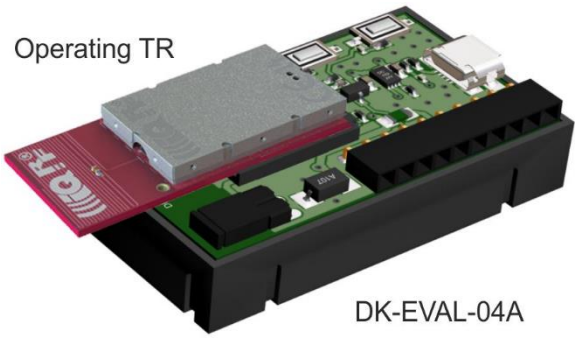
2.3 Programmer

To upload application codes into TRs and configure TR parameters, the **CK-USB-04A** kit is intended.



2.4 End devices

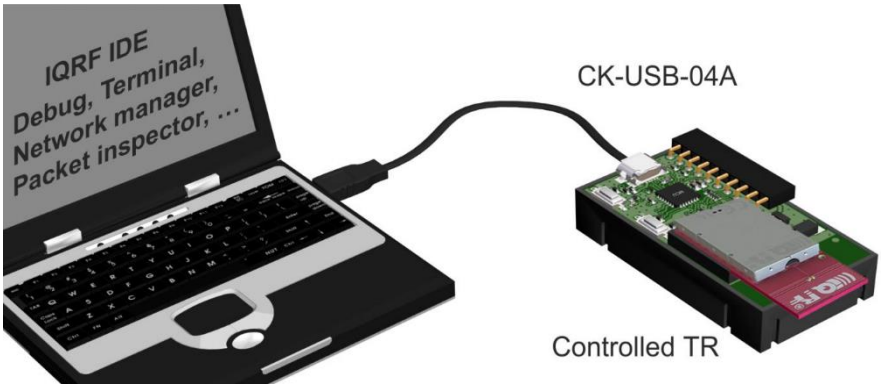
TRs (with application functionality codes uploaded using the programmer in advance) should be hosted in portable **DK-EVAL-04A** kits.



USB connector is intended just for accumulator charging but not for USB connectivity.

2.5 USB device

TRs can also be hosted by **CK-USB-04A** connected to PC via USB. It enables to control the TR application from PC, first of all by the IQRF IDE powerful tools for communication, testing, network management and visualization. This is advantageous especially for the network Coordinator.



3 Typical usage of kits

3.1 First demonstration

RF link check

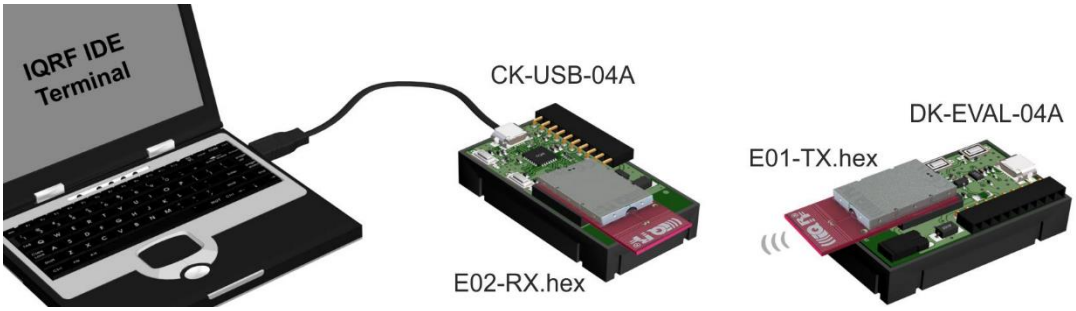
- Unpack two TR transceivers from the development set. Do not make any changes in SW inside.
- Plug the first TR into one DK-EVAL-04A kit and switch the power on by the jumper. Every transmitted RF packet is indicated by a green LED flash.
- Plug the second TR into another DK-EVAL-04A kit and switch the power on by the jumper. Every transmitted RF packet is indicated by a green LED flash.
- Moreover, transmitted packets are started to be received by the counterparts. Every received RF packet is indicated by the red LED flash.
- Thus, a bidirectional RF link is automatically established between the kits in this way. The red LED flashing indicates RF connection.
- You can move the kits and check RF link with respect to the given environment.

Note: This link works plug-and-play thanks to the E09-LINK program which is uploaded in all TR transceivers in development sets delivered from the factory. It is one of the *Basic examples* from the Startup package and can be uploaded whenever needed later on.

3.2 Point-to-point

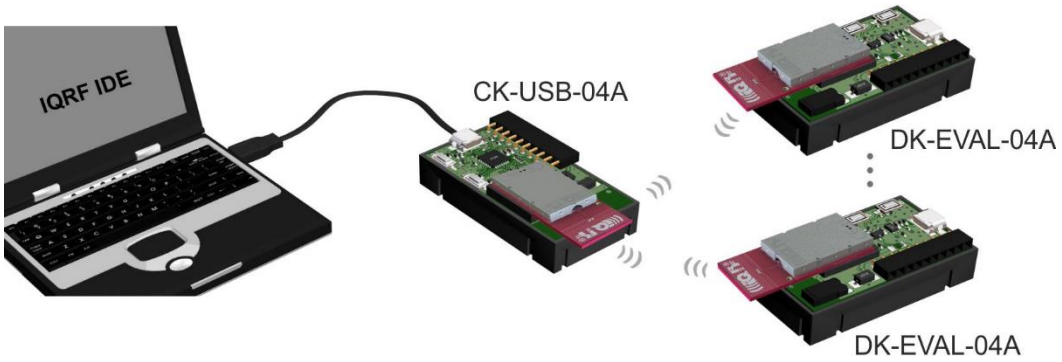
The arrangement with two DK-EVAL-04A kits see above.

The arrangement with CK-USB-04A:



See Basic examples E01-TX, E02-RX, … in IQRF Startup Package.

3.3 Network



The only way to implement an IQMESH network is the DPA approach. For a quick startup, it is recommended to apply the ready-to-use DPA plug-ins without Custom DPA Handlers. See chapter *Quick start with DPA without programming*.

4 Two application approaches

Specific functionality can be implemented into TR in one of the following two application approaches.

4.1 Without programming

Higher level, running **under DPA** and utilizing DPA plug-ins, which are ready-to-use SW plug-ins enabling communication in Mesh networks with no need for programming. TR is not controlled by an application program but from the control system via SPI or UART by the DPA protocol. All resources of the addressed device are accessed via sending requests and receiving responses. Requested functionality is achieved without programming. However, even this approach allows user-specific adapting by optional Custom DPA Handler programmable in C language.

The DPA framework solves the networking transparently. Just the addressees must be specified and then the packets are delivered automatically. This, together with the approach with no programming needed makes TRs very easy to implement. The entire network traffic is based on simple commands only specifying where and what to perform.

4.2 Programming in C

Lower level, running directly under IQRF OS. The functionality is given by the application program in C language.

5 Development software

All software and documentation is available in a single **IQRF Startup package**. It is also delivered on the USB Flash drive inside development sets.

The Startup package directory tree:

```
IQRF_OS403_7xD
├──Development
│   ├──DPA
│   │   ├──DPA-Node-interfaces
│   │   └──OTA_upgrade
│   └──include
│       ├──DPA
│       └──IQRF_OS
├──Documentation
│   ├──HW
│   └──SW
│       └──IQRF_Standard
├──Examples
│   ├──DPA
│   │   ├──CustomDpaHandlerExamples
│   │   │   ├──hex
│   │   │   │   └──IQRF_Standard
│   │   │   └──IQRF_Standard
│   │   ├──IoT-StarterKit-01
│   │   └──StartUp
│   ├──IQRF_OS
│   │   ├──Advanced_examples
│   │   │   ├──Comparator
│   │   │   │   ├──doc
│   │   │   │   └──hex
│   │   │   ├──DAC
│   │   │   │   └──hex
│   │   │   ├──PWM
│   │   │   │   ├──doc
│   │   │   │   └──hex
│   │   │   ├──RF authentication
│   │   │   │   └──hex
│   │   │   ├──RF scanner
│   │   │   │   └──hex
│   │   │   ├──Servo
│   │   │   │   ├──doc
│   │   │   │   └──hex
│   │   │   ├──SPI master
│   │   │   │   └──hex
│   │   │   ├──UART link
│   │   │   │   ├──doc
│   │   │   │   └──hex
│   │   │   ├──Watchdog
│   │   │   │   ├──doc
│   │   │   │   └──hex
│   │   ├──Basic_examples
│   │   │   └──hex
│   │   ├──DDC_examples
│   │   │   └──hex
│   │   ├──DK-SW2-01
│   │   │   └──hex
│   │   └──StartUp
│   └──Miscellaneous
│       ├──ExternalSPImaster
│       │   └──Demo-example
│       └──USBconnectivity
│           ├──Class_CDC
│           └──Class_CustomDevice
└──IQRF_IDE
    └──Notepad++
```

A new Startup package is not always generated whenever some minor changes in documentation are released. Thus, if you need to be absolutely up-to-date, check IQRF downloads for new versions of all documentation in the package.

New project

When starting a new project, always use the **newest parts** as well as the **newest software versions** including the IQRF OS (which is upgradeable).

Older project

If an older project should be maintained, go to the Archive to select the **proper Startup package version** according to the **IQRF OS version** inside your TR transceivers.

5.1 PC software

IQRF IDE

Integrated development environment by IQRF Tech to create, debug and control IQRF applications and manage the IQRF network. Required minimal IQRF IDE version depends on IQRF OS version of given TR transceiver and possibly also on the DPA version. See IQRF IDE Release notes. The best way is to **always use the latest IDE** version.

USB drivers

IQRF USB devices (e.g. CK-USB-04A) primarily utilize Custom class (with VID / PID by IQRF Tech when used with IQRF devices). Several IQRF USB devices (e.g. GW-USB-06) can also (optionally) use the CDC class. (CDC is not necessary for IQRF startup.)

USB drivers implemented in current IQRF devices (e.g. CK-USB-04A or GW-USB-06 with up-to-date firmware) are by the **verified publisher** based on the **WinUSB** by Microsoft.

CC5X

C compiler for the PIC microcontrollers (by B Knudsen Data). The evaluation version (included in IQRF Startup package) is free. The compiler is not necessary when using DPA plug-ins without Custom DPA Handler creation or modification.

Text editor

For source code creation and modification, any external editor being able to save a plain ASCII text, e.g. Windows Notepad can be used. **Notepad++** (a great source code editor, free by GPL License) is very recommended. Download it from www.notepad-plus-plus.org.

5.2 Installation

1. **Do not have any IQRF USB device connected.**
2. **Unzip** the **Startup package** (the `.ZIP` file). All you need is automatically unpacked into the specified directory.
3. **Install** the **IQRF IDE** by invoking the installation file `iqrf-ide-4xx….exe` in subdirectory `IQRF_IDE`. USB drivers and free edition of the C compiler are also installed. Follow the instructions during installation. See the *IQRF USB drivers Installation guide* (in subdirectory `Documentation\SW`) for details depending on the Windows version.
4. **Connect** the **CK-USB-04A** to PC.
5. **Invoke** the **IQRF IDE**.
6. To edit source C files, Windows native Notepad is set as default. You can change it for your favorite editor (e.g. Notepad++) in the *Tools / Options / Editors* menu.

Up to 3 instances of IQRF IDE v4.xx are allowed running at the same time. But only one IQRF IDE can be installed.

5.3 Other PC programs operating with USB

Other PC programs which utilize the same USB driver, for example, some 3rd-party SW (e.g. by Microchip) must not be running at the same time with IQRF IDE.

5.4 Test

Open the default IDE project `E00-START` in the `Examples\IQRF_OS\StartUp` directory and press the *F11* key (*USB Device / Indicate USB Device*). LED1 on the CK-USB-04A should flash 3x which indicates the correct connection between CK-USB-04A and PC.

5.5 IQRF IDE usage

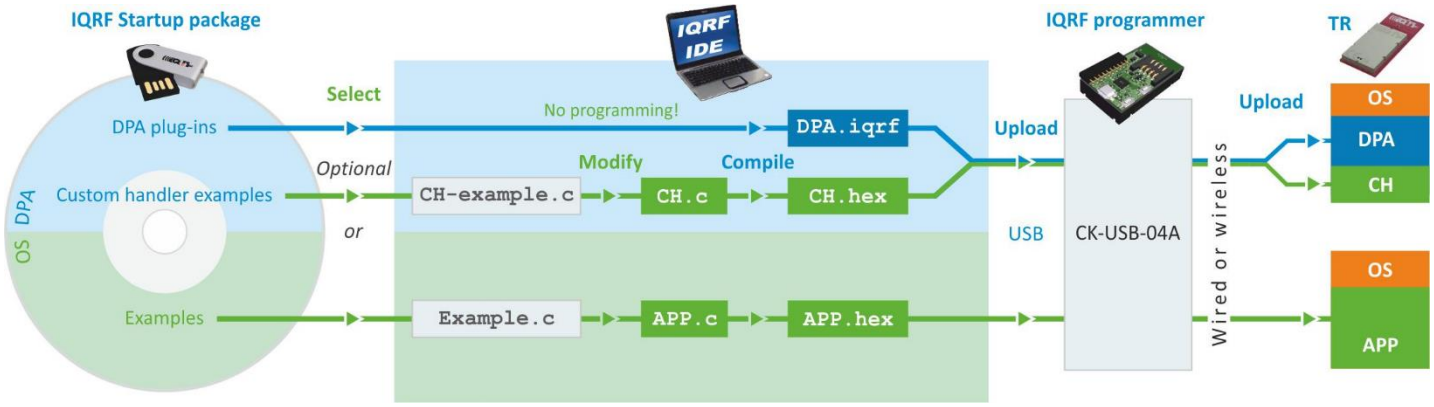
IQRF IDE environment integrates all SW tools needed for application development (and a lot of utilities for the following control, maintenance and service). It is project oriented. Thus, a Project containing necessary definitions and files must be specified first. Default Projects are available for immediate start. Project definition files have the `.IQRFPRJ` extension. See IQRF IDE Help for details.

Programming and uploading procedure

Programming – a creation of a user-specific IQRF application program.

- **Editing** – creation / modification of source code in C language.
- **Compilation** – compiling the source program from C language to `.HEX` machine code.
- **Upload** – uploading the code into the TR:
  - TR Configuration
  - Application code
    - Either a DPA plug-in, optionally with a Custom DPA Handler
    - Or a user-specific application.

  Wireless upload (RFPGM) is also possible – see *IQRF OS User's guide*, Appendix *RFPGM* and *IQRF OS Reference guide*, function *RFPGM*.



Other development, test and service utilities

- **Debug** – allows to stop program execution and watch internal variables (break, watch and continue).
- **Terminal** – utility to control serial communication.
- **Packet inspector** decoding and interpreting packets logged in Terminal Log window.
- **Network management**, **visualization**, **testing**, **scanner**, **QR Code generator**, …

6 C-programming

Programming in C language is needed:

- When DPA is used and given ready-to-use DPA plug-in has to be extended by a user-specific Custom DPA Handler.
- When the application is written under IQRF OS, without DPA.

Programming in C language is not needed when DPA is used and some existing Custom DPA Handler is to be used without modification or no Custom DPA Handler is required at all.

If DPA is used, once the DPA plug-in is uploaded, it is not necessary to upload it anymore when just the Custom DPA Handler is changed.

The Startup Package contains:

- Demo Projects:
  - Under DPA: `HWP-demo.iqrfprj`
  - Under OS: `E00-START. iqrfprj`
- A series of examples in the `Examples/` subdirectories to be utilized as needed:
  1. **Using a ready-made example** (without modifications): Add a `.HEX` file to the Project and upload it into the TR plugged in CK-USB-04A by the *F5* key (*Programming/Upload* menu).
  2. **Using an example with possible modification**: Add a source program (with the `.C` extension) to the Project, make modifications by editing (if desired) and compile it from the C language to the HEX format by the *F10* key (*Project/Build Target* menu). Upload resulting `.HEX` file similarly as in the case 1 above.
  3. **Creation of a new program**: Use the `CustomDpaHandler-Template.c` or `Template.c` empty source file and continue similarly as in case 2 above.

Refer to the IQRF IDE Help to learn about configuring the Project.

6.1 Design steps

1. Plug the TR into CK-USB-04A kit and run IQRF IDE.
2. Open a new Project (*Project/New Project* menu and specify TR type (*Project/Properties/TR Module* menu).
3. If you do not intend to modify or display a source code but to upload an existing code only, add required `.IQRF` or `.HEX` file to the Project (*Project/Add Existing Item* menu) and go to step 7.
4. Add required source file (one of the examples, e.g. `CustomDpaHandler-Timer.c`, `E09-LINK.c` or empty template `CustomDpaHandler-Template.c` or `Template.c` to the Project (*Project/Add Existing Item* menu).
5. Make possible modifications and save them then (in the text editor).
6. Compile the program by the *F10* key (*Project/Build Target* menu). If any error is reported by the compiler go to step 5 to correct it. The result is a compiled code in the `.HEX` format stored in the Project.
7. Program the code into TR by the *F5* key (*Programming/Upload* menu).
8. After uploading, the application is just automatically invoked. It is possible to debug it by the IQRF IDE. See chapter *Debug* below.
9. Then the transceiver can be carried e.g. to another development kit or to an end user equipment.

Typical repeated “loop” during development is **Edit**, **Save** (*Ctrl+S*), **Compile** (*F10*), **Upload** (*F5*) and **Debug**.

Refer to the IQRF Video tutorial set and IQRF IDE training video.

Caution: The TR module can be plugged/unplugged into/from the SIM connector while powered off only.

Tip: In case of CK-USB-04A, the SW2 pushbutton (the one closer to SIM connector) can be used for this. TR is not powered while the SW2 pushbutton is held.

Note: The IQRF IDE reports the TR state at the bottom bar of the main IDE window. The *SPI not active* message need not mean a missing or damaged TR. It is issued if SPI communication is not activated which depends on the application running in TR. SPI communication can be forced by switching the TR in the programming mode by *F6* key (*Programming/Enter Programming Mode* menu). The application in the TR is not running in the programming mode.

6.2 Basic TR modes

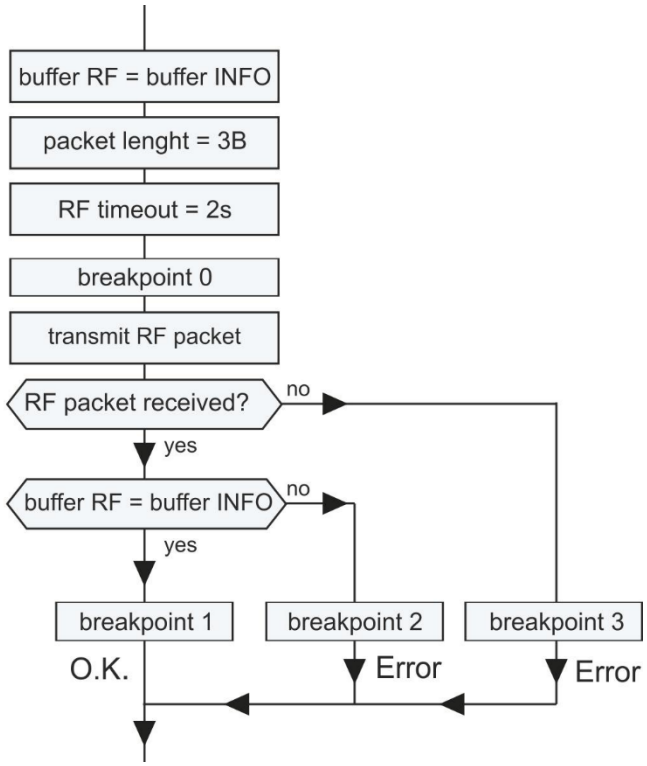
- **Programming** The mode allowing to upload the application into TR, e.g. via IQRF IDE. See *IQRF SPI Technical guide*, chapter *TR upload*.
- **Run** The application is running (executing the application program or DPA plug-in).
- **Debug** The application execution is stopped at the selected breakpoint for testing. See chapter *Debug* below.

6.3 Debug

The operating system offers the `breakpoint(n)` macro to be placed in the source code wherever you wish to stop and watch results (e.g. display internal variables and registers). After selecting the *Skip Breakpoint* button, the application continues running until the next `breakpoint(n)` macro is encountered and so on. See `IQRF-macros.h` header file and basic example E06-RAM for an explanation.

Example

```
copyBufferINFO2RF();

DLEN=3;

toutRF=200;

breakpoint(0);

PIN=0;
RFTXpacket();

if (RFRXpacket())


 if (compareBufferINFO2RF(4))
   breakpoint(1);
 else
   breakpoint(2);
 else
   breakpoint(3);
```



6.4 Application examples

The Startup Package contains the demo programs:

- **For DPA**
  - **Custom DPA handler examples**
- **For IQRF OS**
  - **Basic** To getting familiar with simple TR operation and OS functions
  - **Advanced** More complex
  - **DDC** For IQRF DDC development kits

7 Quick start with DPA without programming

This chapter is intended as a step-by-step guide to get familiar with the DPA approach.

DS-DPA-02 development set

To learn the DPA usage, development set **DS-DPA-02** can be used.

• **TR-72DAT**	IQRF transceiver	6 pcs
• **CK-USB-04A**	IQRF programmer and debugger	2 pcs
• **DK-EVAL-04A**	Universal portable development kit for TR transceivers	4 pcs
• CAB-USBABMICRO	Micro USB cable 18.5 cm	2 pcs
• USB flash drive	Software and documentation	1 pc

1. Preparation

All SW, plug-ins, documentation etc. are available on the flash disk and on www.iqrf.org/support/download.

- Prepare TR-72DATs with OS 4.03D as follows: 5 pcs as the Nodes and 1 pc as the Coordinator.
- Prepare 4 (5) pieces of DK-EVAL-04A and 1 (2) piece(s) of CK-USB-04A.
- Install the latest IQRF IDE. Refer to chapter *Installation* above.
- Launch IQRF IDE and open Project `DPA-demo.iqrfprj` from IQRF Startup package. All necessary files and macros are included in the Project.

2. Upload

Creating Nodes

A. Plug a TR transceiver in CK-USB-04A, select the `DPA-Node-STD-7xD*.iqrf` file in *Project* window and click the *Upload Plug-ins* button on the Toolbar (or use the *F5* key). The plug-in should be uploaded then.

B. Double-click on the configuration file `DPA-config.xml` in IDE *Project* window to open the *TR Configuration* window. Make the following settings:
- Select the desired RF channel (*RF Channel A*) in the *OS* tab. For the Coordinator only, see below. Nodes will inherit the value from the Coordinator during bonding.
- Select the desired *RX filter* in the *DPA* tab.
  - For short range testing (within the room) select 15.
  - For operating in real conditions select 5.
- Select the desired *Access Password* in the *Security* tab. See *IQRF OS User's guide*, chapter *Access encryption*.

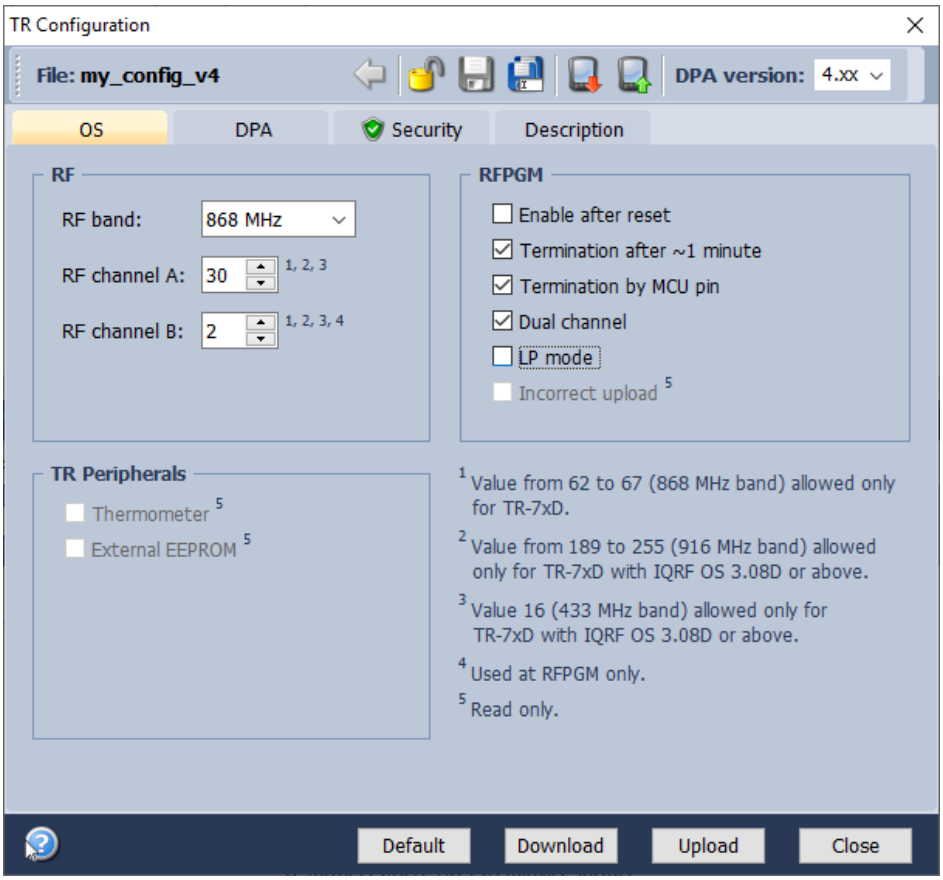
This setting must be the same in the entire network!
Do not change other parameters for now.
Save the configuration into the TR by button *Upload*.

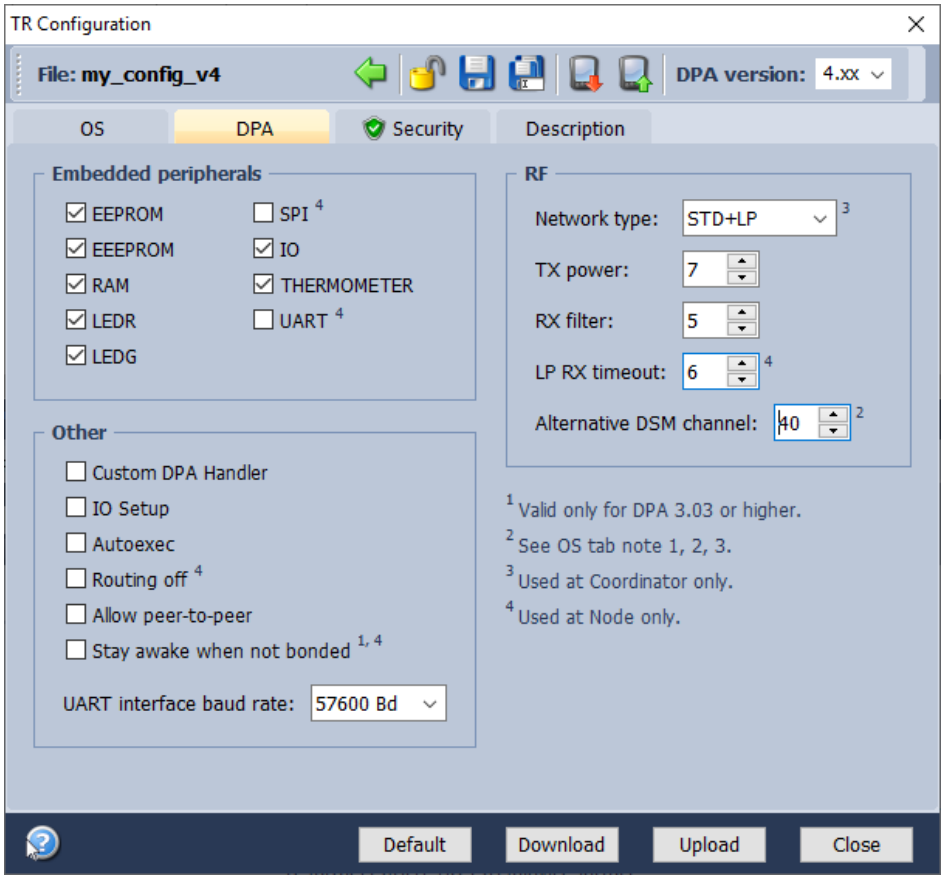
C. Plug this TR into the DK-EVAL-04A kit.

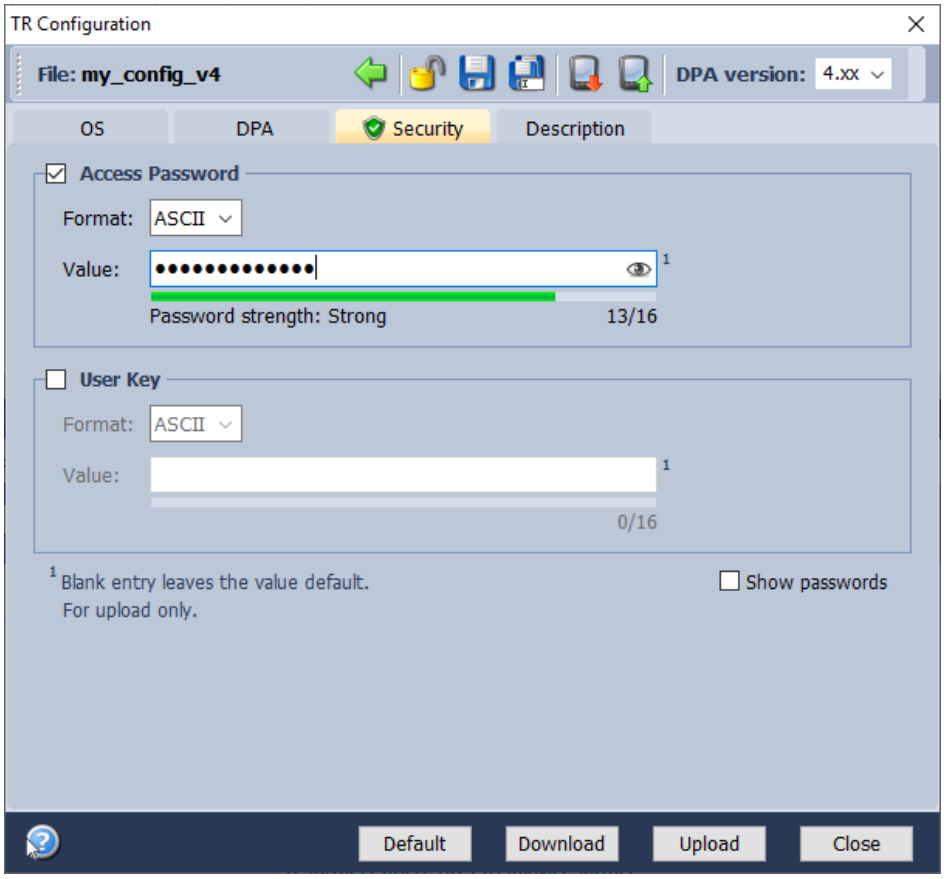
Repeat steps A to C for all Nodes.

Creating the Coordinator

Use the same procedure but with the `DPA-Coordinator-SPI-7xD*.iqrf` and `DPA-config.xml` files. Then leave the TR plugged in CK-USB-04A.







Warning: If you use a stronger RX filter during the development (e.g. on the table), do not forget to reduce it then in final application (in the real environment).

3. Bonding

See the *IQMESH Network Deployment Technical guide*, chapter *Bonding (including a Node into the network)*.

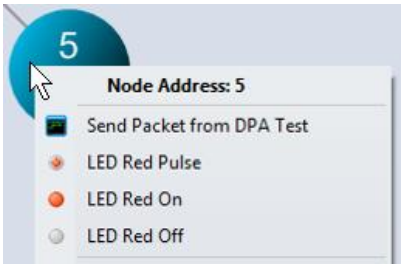
4. Discovery

See the *IQMESH Network Deployment Technical guide*, chapter *Discovery (creating the routing structure)*.

5. Examples

A quick test of the network functionality can be performed by the pop-up menu (*example A* below). For detailed testing of DPA commands the *Terminal – DPA Test* tool provides two ways how to set up a specific DPA command: using macros (*example B*) or manual arrangement of DPA packet (*example C*).

A. Pop-up menu

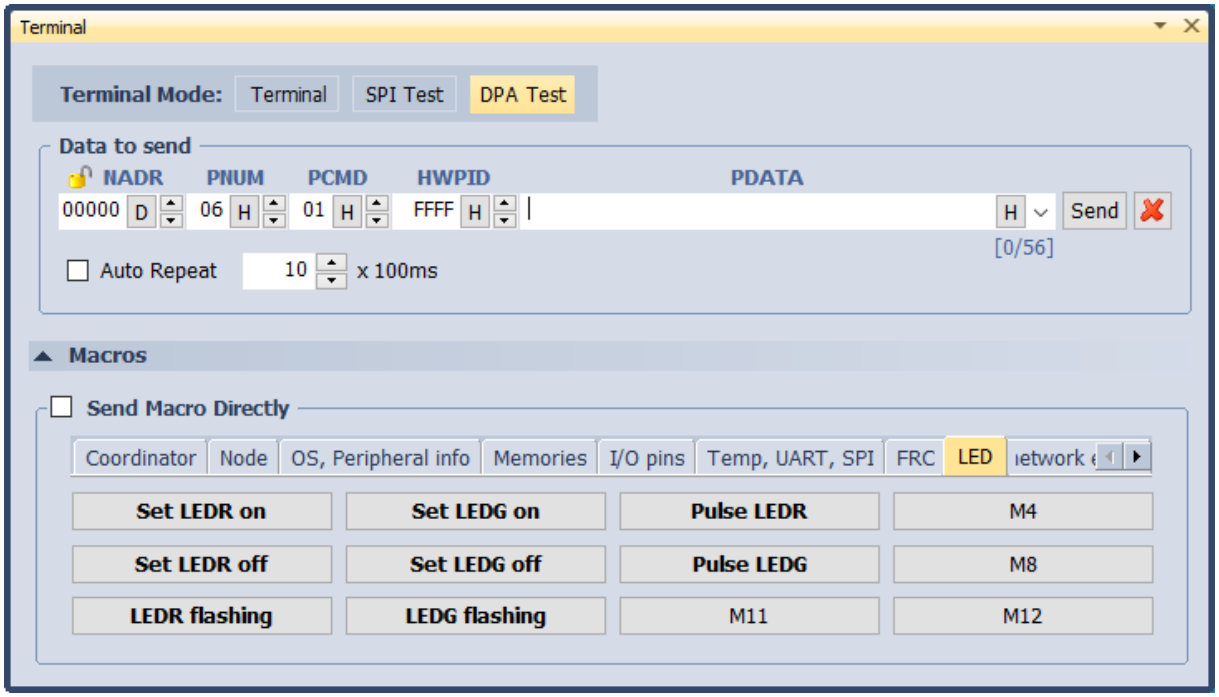


For the simplest checking, a command to switch the red LED On/Off on selected Node can be immediately sent by clicking the left mouse button on the symbol of the given Node in the map.
The command currently selected in the *DPA Test – Data to Send* window can be executed by the right mouse button click on the symbol of given Node in the map and by selecting the *Send Packet from DPA Test* item. `NADR` is set automatically according to the selected Node.

B. Macros

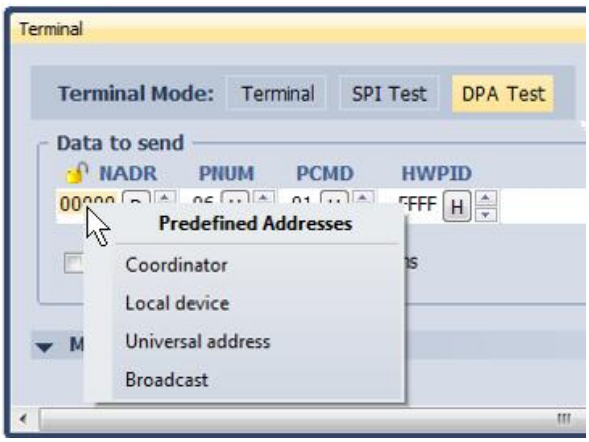
Click on the particular macro and `PNUM`, `PCMD`, `HWPID` and `PDATA` are automatically filled in. Then select *Node Address* in the `NADR` box (for Broadcast use address 0xFF) and click the *Send button*.

- Go to the LED macros and click on macro *Set LEDR on*. To switch all LEDs on, fill in `0xFF` in `NADR` and click *Send button*.
- Use various macros and Node addresses to test the functionality.



C. DPA packet arranged manually

To get a better understanding of the DPA packet structure, you can also fill in the DPA packet manually. By clicking the right mouse button to the area for selecting *NADR*, *PNUM*, *PCMD* or *HWPID*, menu *Predefined Addresses*, *Predefined Peripherals*, *Predefined Commands* or *Predefined HWPIDs* is displayed. These lists allow to select items defined by the DPA specification and directly arrange the packet.



- From menu *Predefined Addresses* select *Broadcast* or directly specify the address of given Node.
- From menu *Predefined Peripherals* select the LEDR peripheral.
- From menu *Predefined Commands* select the Set on command.
- From menu *Predefined HWP* select the *To All HWP* item.
- Click the *Send button*.

Test other peripherals and commands in the same way.

6. Unbonding

See the *IQMESH Network Deployment Technical guide*, chapter *Unbonding (removing a Node from the network)*.

Besides of *IQMESH Network Manager* and *DPA Test*, IQRF IDE provides also other tools (*Terminal Log* and *Packet Inspector*) for working with DPA especially for developers implementing or testing the DPA protocol.

See the *IQMESH Network Deployment Technical guide*, chapter *IQMESH support in IQRF IDE*.

8 Recommended documentation

- IQRF Video tutorial set
- IQMESH Network Deployment Technical guide
- IQRF DPA Framework Technical guide. For the DPA approach only.
- IQRF OS User's guide.
- IQRF OS Reference guide – description of IQRF OS functions. For non-DPA approach or for DPA with Custom DPA handler only.
- IQRF SPI User's guide – implementation in IQRF TR modules. It is needed only if you design one's own SPI master to communicate with the TR module.
- TR transceiver datasheet
- User's guides of development kits used
- IQRF USB drivers Installation guide

9 Document revision

190117	Revised and updated for DPA v4.00 and IQRF IDE v4.50. Links to the *IQMESH Network Deployment Technical guide* added.
181018	Updated for IQRF OS v4.03D. Screenshot *TR configuration* and chapter *Bonding*, steps A and C are slightly changed vs. v4.02D.
170821	Updated for IQRF OS v4.02D. No difference between TR and DCTR. HWP demo not available anymore. The word “General” removed from the HWP plug-in name. HWP file structure slightly changed. Chapter *Basic TR modes* added.
170810	Updated for IQRF OS v4.01D. Removed due to a serious bug in OS v4.01D RFPGM (wireless upload).
170310	Updated for IQRF OS v4.00D.
150805	First release for *IQRF Quick start guide* merged with *IQRF DPA Quick start guide*. Updated for (DC)TR-7xD transceivers, IQRF OS v3.07D, DPA v2.20 and IDE v4.30 or higher.

10 Sales and Service

10.1 Corporate office

IQRF Tech s.r.o., Prumyslova 1275, 506 01 Jicin, Czech Republic, EU
Tel: +420 493 538 125, Fax: +420 493 538 126, www.iqrf.tech
E-mail (commercial matters): sales@iqrf.org

10.2 Technology and development

www.iqrf.org
E-mail (technical matters): support@iqrf.org

10.3 Partners and distribution

www.iqrf.org/sales/distributors

10.4 Quality management

ISO 9001 : 2009 certified

10.5 Trademarks

The IQRF name and logo are registered trademarks of IQRF Tech s.r.o.

PIC, SPI, Microchip and all other trademarks mentioned herein are property of their respective owners.

10.6 Legal

10.7 On-line support

support@iqrf.org

Smarter Wireless. Simply.