

ICWP™

In-Circuit wireless programming for IQRF OS v3.00

The Lite version

Application Note

AN009



Simple way to smarter wireless solutions

IQRF TR modules with operating system v2.11 or higher can be uploaded even in a wireless way (RF PGM, ICWP – In-Circuit Wireless Programming). During RF upload the programmed TR is not inserted in the programmer but it is connected to the programmer in a wireless way by an auxiliary TR module. This Application note describes the Lite RF PGM version for IQRF OS v3.00 and IQRF IDE v2.07.

RF PGM allows:

- Upload TRs also in final application boards (housed, soldered etc.)
- More TRs uploaded simultaneously (e.g. all IQMESH Nodes in one stroke)

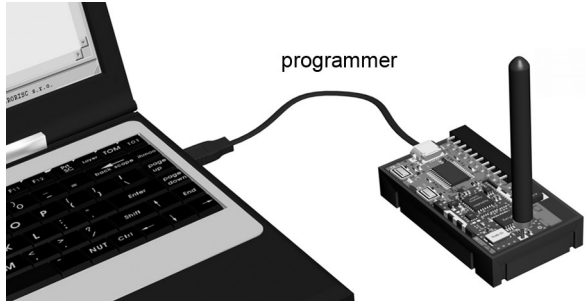


Fig. 1: Standard (wired) upload in a programmer



Fig. 2: Wireless upload (RF PGM)

Both TR modules have to have the respective plug-ins (SW options extending the OS) uploaded in advance:

- Plug-in for the auxiliary TR module (`RFPGMCK`): Must be uploaded by the user with respect to RF band (868 or 916 MHz).
- Plug-in for programmed TR module (`RFPGMTR`): The Lite version is already uploaded from the factory.

To utilize RF PGM, the *RF Programming* checkbox must be active in IQRF IDE. During wireless upload the programmed TR module is not in the standard programming mode but in the RF PGM one. The standard mode is emulated by the auxiliary TR module in the programmer. From the IQRF IDE point of view all is the same as for standard programming except of the fact that The *Module Info* window and the *Reset TR Module* button in IQRF IDE relates to the auxiliary TR module.

There are the following plug-in functions available in the user application (see the Appendix):

- `runRFPGM()` – switch TR to the RF PGM mode
- `enableRFPGM()` – request to enable switching TR to the RF PGM mode after reset
- `disableRFPGM()` – request to disable switching TR to the RF PGM mode after reset (factory default)
- `setupRFPGM(x)` – setup RF PGM parameters

To utilize these functions in user program it is needed to make the `RFPGMTR-LITE.h` header file accessible to the compiler (by the `#include` directive with the path to the file).

RF PGM mode can be entered by reset (if enabled) or by the `runRFPGM()` function (unconditionally) and abandoned by low level on the C5 pin for at least ~0.7 s (if enabled) or by the *End RF PGM* button (unconditionally).

- If RF PGM was invoked by reset based on the `enableRFPGM()` function the original application is run from beginning.
- If RF PGM was invoked by the `runRFPGM()` function the application continues program executing.

RF PGM features

- Fixed channel (default one, e.i. 52 for 868 MHz or 104 for 916 MHz bands) and fixed RF bit rate (19.2 kb/s).
- "Non-networking" communication, without addressing and broadcast.

LED indication

Operation	Auxiliary TR		Programmed TR	
	Green	Red	Green	Red
RF PGM mode	short flash in 1 s period	–	continuous	short flashes in 2 s period
Uploading	short flash in 1 s period	fast blinking	continuous	fast flashing
RF PGM mode finished	–	–	–	single flash for 1.5 s

RF PGM usage

Typical requirement used to be a different behavior during development (when modified program is uploaded repeatedly) and after finishing the development (when following-up reprogramming is not desired at all or is required for service purposes on special request only).

The `enableRFPGM()` is intended typically for development, also for applications with TR modules inaccessible for wired upload. E.g. an error in user program under development can cause that the `runRFPGM()` function can not be accessed any more. If the TR can not be reprogrammed externally the only way how to force programming mode is having a possibility to switch to RF PGM automatically after reset. Thus, main purposes of `enableRFPGM()` are:

- To enable entering RF PGM while routine development in a fast and reliable way.
- A loop-hole against user's errors during development.

But it is recommended to keep a way of how to invoke RF PGM mode even in completed and debugged designs on request (at least to allow services and future upgrades). That is why `disableRFPGM()` should not be used unless there is a user's defined way of how to invoke back the RF PGM mode via `runRFPGM()` or `enableRFPGM()`. It is up to the user to define the entry point into RF PGM, e.g. using a jumper, pushbutton or a special RF packet. If being omitted there is no any other way of getting back into RF PGM unless removing the module from a device and placing it into the programmer.

Thus, RF upload is fully under the user's control. Invoking the RF PGM mode, its termination, behavior after reset – everything can be adapted according to the user's needs.

Example

Wireless upload procedure can be tested using the demo program `RF-PGM-TST`:

1. **Preparatory phase:** Upload the `RFPGMCK` plug-in to the auxiliary TR module (not in a wireless way).
2. In case of the 916 MHz band select this in the `RF-PGM-TST` demo program and recompile it.
3. Insert the programmed TR to the programmer and upload (not in a wireless way) the `RF-PGM-TST` demo program. Then:
 - The application just starts up which is indicated by red LED flashing.
 - But thanks to the `enableRFPGM()` function used in the demo program the programmed TR is configured so that it will be switched to RF PGM after reset from this time.
4. **Development phase:** Arrange both TR modules according to Fig. 2 for RF PGM upload and activate the *RF PGM* checkbox.
5. For test purpose change LED flashing from the red LED to the green one in user program and compile this.
6. Set the programmed TR module in RF PGM mode (by the TR module reset) and invoke upload in the IDE environment by the *Upload* button (or by the *F5* key). After RF PGM finish the red LED flashes for 1.5 s and then the application starts up automatically. (RF PGM mode can be terminated without RF uploading by the pushbutton on the kit by pressing for ~0.7 s.)
7. Repeat steps 5 and 6 until the application is debugged. Respective LED flashing indicates success.
8. **Final upload:** Upload final application with `disableRFPGM()`. After finishing the RF PGM mode is invoked for the last time. Press the pushbutton for more than 0.7 s to terminate this. The application starts to run, executes the `disableRFPGM()` function and automatical entering RF PGM after following resets is disabled.

Since now RF PGM is accessible via the pushbutton only. It makes **RF PGM upgrades** possible by `runRFPGM()` later on.

Document history

- 110304 Bug in Appendix, `runRFPGM` function (Preconditions and Example 2-3) fixed
- 110121 Updated for 916 MHz
- 110107 Updated for OS v3.00
- 090629 First release

Appendix – Plug-in functions**enableRFPGM**

Function	Request to configure OS for switching to RF PGM mode after TR module reset
Purpose	Enable switching to RF PGM mode after reset
Syntax	<code>void enableRFPGM()</code>
Parameters	–
Return value	–
Output values	OS configured
Preconditions	–
Remarks	This function must be executed first to modify OS and just the following reset will switch to RF PGM.
Side effects	–
See also	<code>disableRFPGM</code> , <code>runRFPGM</code>
Example	See <code>disableRFPGM</code>

disableRFPGM

Function	Request to configure OS for not switching to RF PGM mode after TR module reset
Purpose	Disable switching to RF PGM mode after reset
Syntax	<code>void disableRFPGM()</code>
Parameters	–
Return value	–
Output values	OS configured
Preconditions	–
Remarks	This function must be executed first to modify OS and just the following reset will not switch to RF PGM.
Side effects	–
See also	<code>enableRFPGM</code>
Example	<pre>enableRFPGM(); // During development // disableRFPGM(); // For final application</pre>

runRFPGM

Function	Switch to RF PGM mode
Purpose	One-shot immediate switching to RF PGM mode
Syntax	<code>void runRFPGM()</code>
Parameters	–
Return value	–
Output values	RF PGM mode initiated
Preconditions	<ul style="list-style-type: none"> Parameters like RF band, RF channel and bit rate are fixed to OS default values in the Lite version. If the application uses different ones, they must be set to default before using <code>runRFPGM</code> and then (just for the case of RF PGM refusal) restored to desired values. Non-networking mode must be selected.
Remarks	<ul style="list-style-type: none"> RF PGM mode can be refused by low level on the C5 pin for at least ~0.7 s (if enabled) or by the <i>End RF PGM</i> button). Then the application continues without reset. After successful RF PGM finishing the application is reset.
Side effects	–
See also	<code>enableRFPGM</code> , <code>setupRFPGM</code>
Example 1	<code>if (jumperSet) runRFPGM(); // Enter RF PGM mode on special request</code>
Example 2	<pre> setNoNetMode(); // Disable networking (if enabled) setRFchannel(52); // If a different channel is used // default (52 for 868 MHz or 104 for 916 MHz) runRFPGM(); setFilteringOn(); // Restoring (for continuing without RF PGM only) setRFchannel(my_channel); // -"- </pre>
Example 3	<pre> setNoNetMode(); setRFchannel(52); runRFPGM(); reset(); // Reset (for continuing without RF PGM only) // Restoring is not necessary here </pre>

setupRFPGM

Function	Setup RF PGM parameters																																
Purpose	Configure behavior for RF PGM invoking and termination																																
Syntax	void setupRFPGM(x)																																
Parameters	uns8 x: <table border="1" style="margin-left: 20px; width: 60%;"> <thead> <tr> <th rowspan="2">x</th> <th colspan="2">RF PGM invoking</th> <th colspan="2">RF PGM termination</th> </tr> <tr> <th>by runRFPGM()</th> <th>by reset</th> <th>by C5 pin</th> <th>by IQRF IDE</th> </tr> </thead> <tbody> <tr> <td>0x80</td> <td style="text-align: center;">+</td> <td style="text-align: center;">-</td> <td style="text-align: center;">+</td> <td style="text-align: center;">+</td> </tr> <tr> <td>0x00</td> <td style="text-align: center;">+</td> <td style="text-align: center;">-</td> <td style="text-align: center;">-</td> <td style="text-align: center;">+</td> </tr> <tr> <td>0x10</td> <td style="text-align: center;">+</td> <td style="text-align: center;">+</td> <td style="text-align: center;">-</td> <td style="text-align: center;">+</td> </tr> <tr> <td>0x90</td> <td style="text-align: center;">+</td> <td style="text-align: center;">+</td> <td style="text-align: center;">+</td> <td style="text-align: center;">+</td> </tr> </tbody> </table> <div style="margin-left: 40px; text-align: right;"> Factory default: 0x80 +: enabled, -: disabled </div>				x	RF PGM invoking		RF PGM termination		by runRFPGM()	by reset	by C5 pin	by IQRF IDE	0x80	+	-	+	+	0x00	+	-	-	+	0x10	+	+	-	+	0x90	+	+	+	+
x	RF PGM invoking		RF PGM termination																														
	by runRFPGM()	by reset	by C5 pin	by IQRF IDE																													
0x80	+	-	+	+																													
0x00	+	-	-	+																													
0x10	+	+	-	+																													
0x90	+	+	+	+																													
Return value	-																																
Output values	OS is modified and setup values are applicable anytime later.																																
Preconditions	-																																
Remarks	-																																
Side effects	-																																
See also	runRFPGM, enableRFPGM																																
Example 1	<pre>setupRFPGM(0x10); // RF PGM entered: after reset or runRFPGM // RF PGM abandoned: by End RF PGM button only</pre>																																
Example 2	<pre>setupRFPGM(0x90); // RF PGM entered: after reset or runRFPGM // RF PGM abandoned: by C5 pin or End RF PGM button only</pre>																																

Sales and Service

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

Partners and distribution

Please visit www.iqrf.org/partners

Quality management

ISO 9001 : 2000 certified

Trademarks

*The IQRF name and logo are registered trademarks of MICRORISC s.r.o.
PIC, SPI, Microchip, MPLAB 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: www.iq-esupport.com



Simple way to smarter wireless solutions