The BlueNRG and BlueNRG-MS information register (IFR)

Introduction

This user manual describes the information register (IFR) of the BlueNRG and BlueNRG-MS devices and provides related programming instructions.

1 Information register sector

The BlueNRG and BlueNRG-MS firmware stacks use a table of configurable parameters which allows some key parameters of their devices to be properly configured. Such key parameters include high-speed crystal time, low-speed crystal type, frequency and period, and stack mode. The configurable parameters table resides in a sector of the Flash called the information register (IFR).

Figure 1. BlueNRG Flash layout and Figure 2. BlueNRG-MS Flash layout below show the Flash layouts of the BlueNRG and BlueNRG-MS devices. Specific software such as the Updater and the BlueNRG and BlueNRG-MS stack are indicated in addition to the area where the IFR data are stored. The IFR sector comprises 2 kbytes of Flash memory and normally contains device configuration parameters. However, since the configuration parameters use up only 192 bytes, the remainder of the IFR sector contains some code for the BlueNRG while it is left empty (0xFFFFFFFF) for the BlueNRG-MS. The address of IFR sector is 0x10020000.

Figure 1. BlueNRG Flash layout

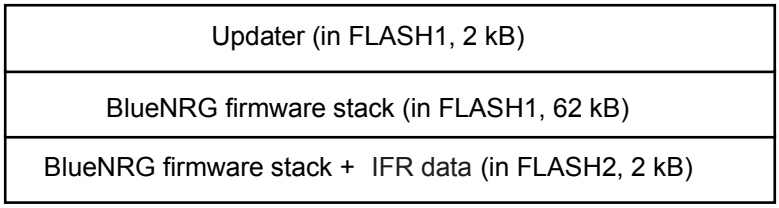
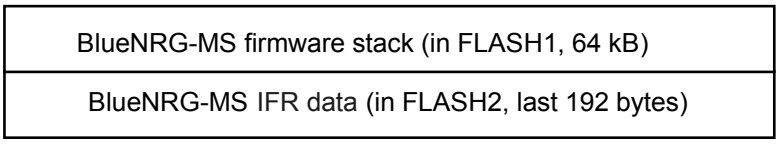


Figure 2. BlueNRG-MS Flash layout



The firmware image contains both stack and IFR data. The image file can be customized with a different IFR data before downloading it into the device. In fact, last 192 bytes can be overwritten with the new IFR data. In this way there is no need to update the IFR sector with the correct data after having downloaded the firmware image.

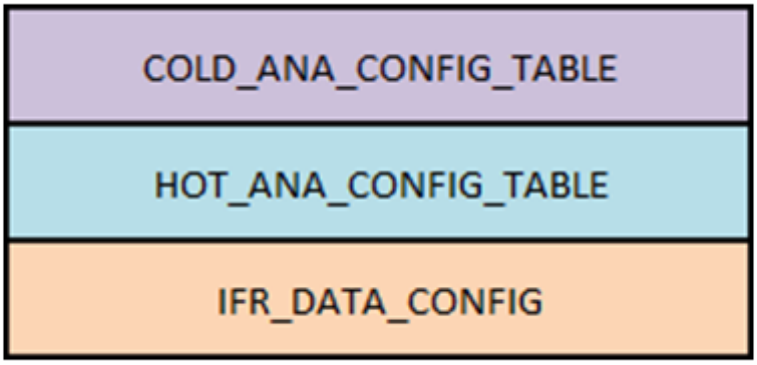
Note: Only IFR data can be carefully configured. The user must not change the content of all the other Flash regions outside IFR data (except when a new valid firmware stack is loaded).

1.1 IFR data regions

The IFR sector is divided into three regions (see Figure 3. IFR regions):

1. Cold boot table region (COLD_ANA_CONFIG_TABLE)
2. Hot boot table region (HOT_ANA_CONFIG_TABLE)
3. Configuration parameter region (IFR_DATA_CONFIG)

Figure 3. **IFR regions**



Note: All IFR parameters that are configurable can be changed with the BlueNRG IFR utility "View/Edit" view (see Section 2 BlueNRG GUI IFR utility)

1.1.1 Cold boot region

The cold boot region (COLD_ANA_CONFIG_TABLE) contains analog configurations that are loaded on hardware reset and power-on-reset (POR). This region configures registers that maintain analog configurations while in sleep mode (e.g. clock configuration). The information to do this is provided through default IFR data configuration files or it can automatically be generated by the BlueNRG GUI IFR utility according to the hardware configuration (e.g. crystals on the board).

1.1.2 Hot boot region

The hot boot region (HOT_ANA_CONFIG_TABLE) contains analog configurations that are loaded every time the device comes out of sleep mode. This information is provided by STMicroelectronics.

1.1.3 Configuration parameter region

The configuration parameter region (IFR_DATA_CONFIG) contains the information and configuration parameters shown in Figure 4. IFR configuration parameter region.

Figure 4. **IFR configuration parameter region**

Address	31 30 29 28	27 26 25 24	23 22 21 20	19 18 17 16	15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0
100207FC	FREE							
100207F8	FREE							
100207F4	FREE							
100207F0	FREE							
100207EC	FREE							
100207E8	Day		Month		Year		RESERVED	
100207E4	UID							
100207E0	RESERVED				HS startup time			
100207DC	RESERVED		Master SCA		Slave SCA			
100207D8	LS Crystal Freq							
100207D4	LS Crystal Period							
100207D0	Max connection event time							
100207CC	RESERVED							
100207C8	RESERVED							
100207C4	RESERVED							
100207C0	RESERVED				GPIO Config		Stack Mode	

Used by the Stack
Information
Reserved

2 BlueNRG GUI IFR utility

The BlueNRG GUI IFR utility is a tool that allows customers to define the IFR data in a controlled way. Using this utility is the only way to define IFR data according to a customers needs. The utility supports both BlueNRG and BlueNRG-MS devices and it provides the following windows:

- View/Edit view: displays the IFR regions with related fields and description. The user can modify some of these fields according to his needs.
- Memory view: displays the IFR fields with memory addresses and related values that are generated by the BlueNRG GUI according to specified values on View/Edit view.
- C view: displays the C language structure related to the IFR configuration data region matching the View/Edit and Memory views.

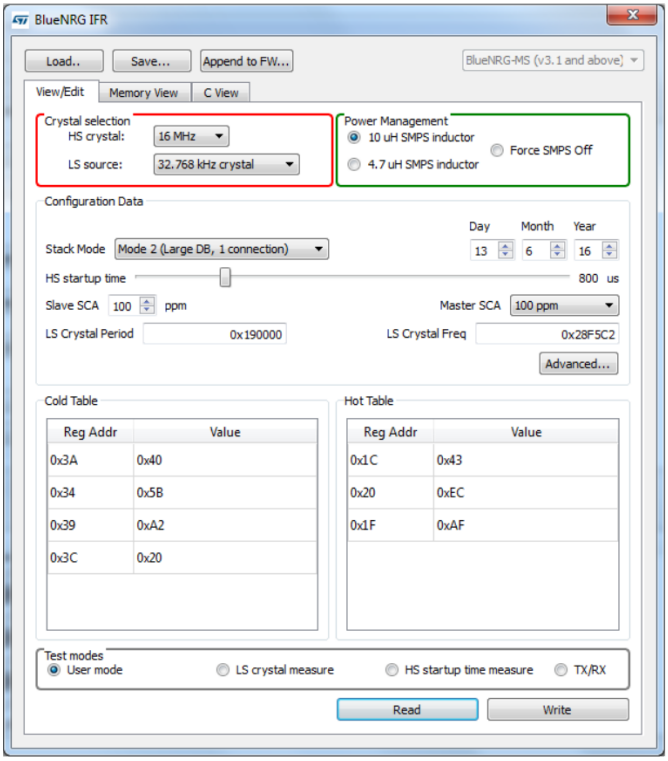
Some of the following general utilities are also available:

- A tab for selecting either the BlueNRG or the BlueNRG-MS device
- A load button to allow a configuration file to be loaded
- A save button to allow the current parameters to be saved into a configuration file
- A "Append to FW" button to generate a new file containing both a firmware image and the current IFR settings
- A read button to allow IFR content from the device to be read
- A write button to allow the displayed IFR configuration to be written into the device IFR

2.1 IFR view/edit view

The BlueNRG GUI IFR utility view/edit view allows the register fields that are written into the device IFR to be displayed and allows the different versions of the device IFR configurations to be distinguished. The user can carefully modify the editable values and retrieve these changes in both the memory and C views. The BlueNRG GUI IFR view/edit view contains the tabs shown in Figure 5. BlueNRG GUI IFR view/edit tab for the BlueNRG/BlueNRG-MS below.

Figure 5. BlueNRG GUI IFR view/edit tab for the BlueNRG/BlueNRG-MS



The tables below show the available IFR view/edit window fields.

Table 1. Crystal selection

Field name	Description	Supported value
High-speed crystal	Select which crystal to be used	16 MHz 32 MHz
Low-speed source	Select which oscillator to be used	32 kHz crystal oscillator Internal ring oscillator (without crystal)

Table 2. Power management

Field name	Description
10 µH SMPS inductor	Select this field if using SMPS with 10 µH inductor
4.7 µH SMPS inductor	Select this field if using SMPS with 4.7 µH inductor (not valid on the BlueNRG)
Force SMPS off	Forces SMPS pins off. It may be used to disable the SMPS pin on the WLCSP package or even on the QFN package when NO_SMPS pin is connected to GND.

Table 3. Configuration data

Field name	Description	Supported value
Stack mode	Indicates stack mode. This value (if valid) overrides the STACK default (0x02)	Mode 1: slave/master, one connection only, small GATT database (RAM2 off during sleep) Mode 2: slave/master, one connection only, large GATT database (RAM2 on during sleep) Mode 3: up to eight connections, master only (BlueNRG), master/slave (BlueNRG-MS), small GATT database (RAM2 on during sleep)
Day	Allows a date (day) to be associated with a specific BlueNRG IFR configuration	[1-31]
Month	Allows a date (month) to be associated with a specific BlueNRG IFR configuration	[1-12]
Year	Allows a date (year) to be associated with a specific BlueNRG IFR configuration	[0-99]
High-speed startup time	Start-up time for the high-speed crystal (time unit = 2.4414 µs)	Ranges from 512 µs to 1953 µs. It can be measured using the XTAL_startup_TEST described in AN4494
Slave SCA	Slave sleep clock accuracy, depends on low-speed oscillator	[10-1500] ppm
Master SCA	Master sleep clock accuracy, depends on low-speed oscillator	20, 30, 50, 75, 100, 150, 250, 500 ppm
Low-speed crystal period	Low-speed crystal period (read-only value defined in the BlueNRG GUI)	
Low-speed crystal frequency	Low-speed crystal frequency (read-only value defined in the BlueNRG GUI)	If 0xFFFFFFFF autocalibration is enabled
Maximum connection event time	Maximum duration of a connection event in the slave (time unit = 2.4414 µs) (1)	[0-4] s
GPIO configuration	Sets GPIOs in a given configuration for special functions (1)	Disabled: GPIOs in high impedance Active state on TEST1: high when device is active because of SPI or radio activity
UID	Allows existing UID with given 32-bit number to be overwritten. (1)	
Free	Allows IFR free addresses with user-defined values to be configured (1)	

1. *Available by selecting "Advanced" button*

Table 4. Test modes

Field name	Description	Notes
User mode	Normal mode	
Low-speed crystal measure	Output signal from 32 kHz crystal oscillator on TEST9 pin	Refer to LSOSC_center_TEST described in AN4494
High-speed start-up time measure	Output signals on TEST8 and TEST9 pins for XTAL start-up measurement	Refer to XTAL_center_TEST described in AN4494

Field name	Description	Notes
TX/RX	Output signal on TEST8 and TEST9 pins for TX/RX for BlueNRG, BlueNRG-MS devices	TX/RX on TEST9, TX on TEST 8

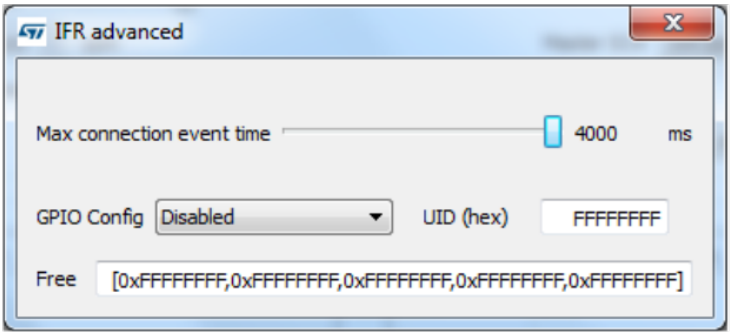
In addition, the cold and hot tables show the following:

- Cold table: shows the cold boot region (COLD_ANA_CONFIG_TABLE) register values. This information cannot be directly changed by the user. Correct data is automatically generated upon selection of correct configuration values, like the high-speed crystal, low-speed crystal, or SMSP configuration.
- Hot table: shows the hot boot region (HOT_ANA_CONFIG_TABLE) register values. This information cannot be changed by the user.

The IFR advanced tab allows:

1. The maximum connection event time, up to 4000 ms (default value), to be set
2. A device UID on IFR UID field to be specified. It can be used by the application to identify the device. Default: 0xFFFFFFFF (field available only on the BlueNRG and BlueNRG-MS)
3. Some specific values on unused section of IFR to be set for the user application purposes (field available only on the BlueNRG and BlueNRG-MS)
4. ACTIVE state to be enabled on TEST1 pin. TEST1 pin goes high when the device is not in sleep mode (field available only on the BlueNRG and BlueNRG-MS)

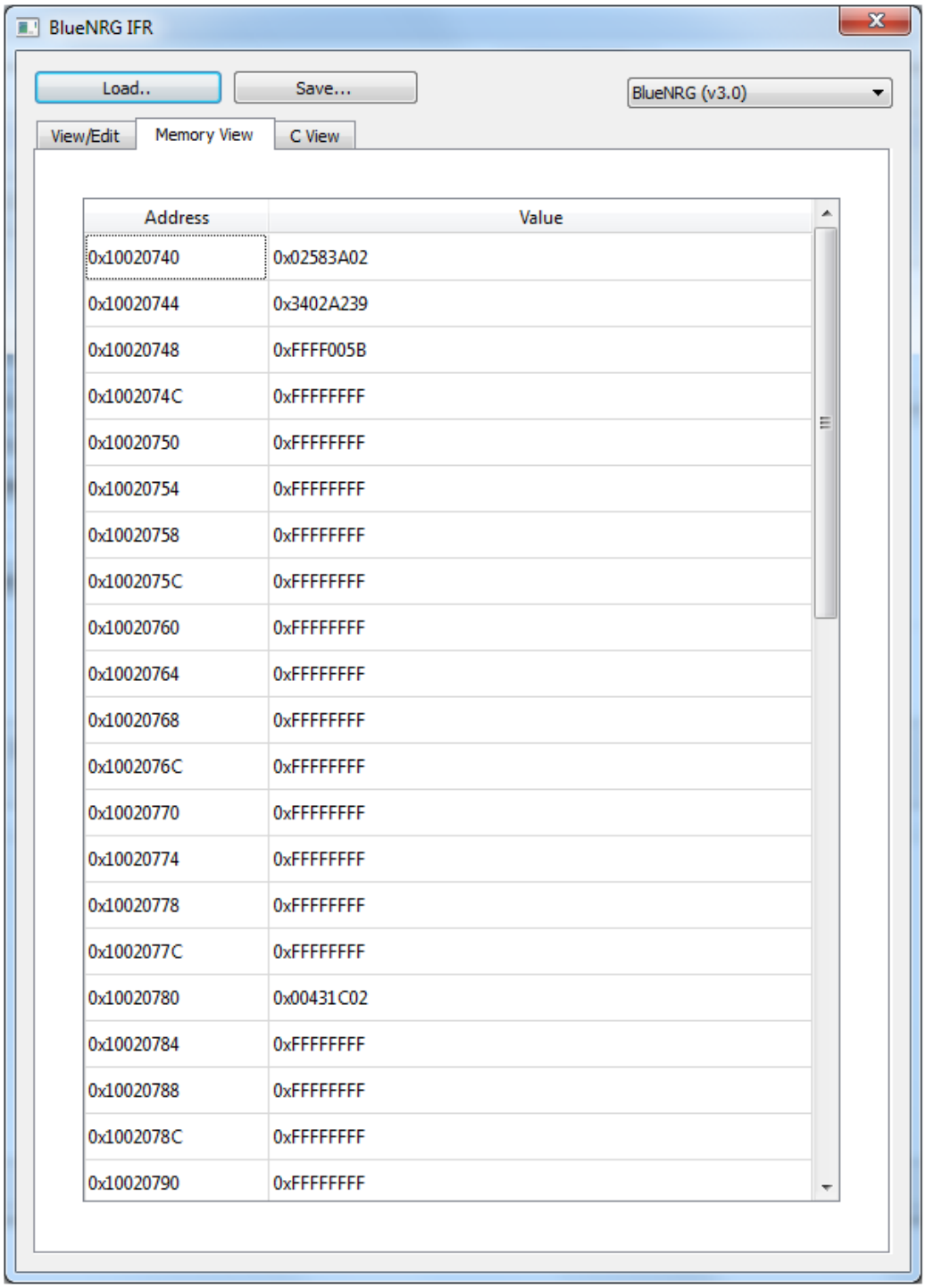
Figure 6. IFR advanced



2.2 IFR memory view

The BlueNRG GUI IFR utility memory view allows the IFR addresses, with related values as defined on the IFR regions view, to be dispayed.

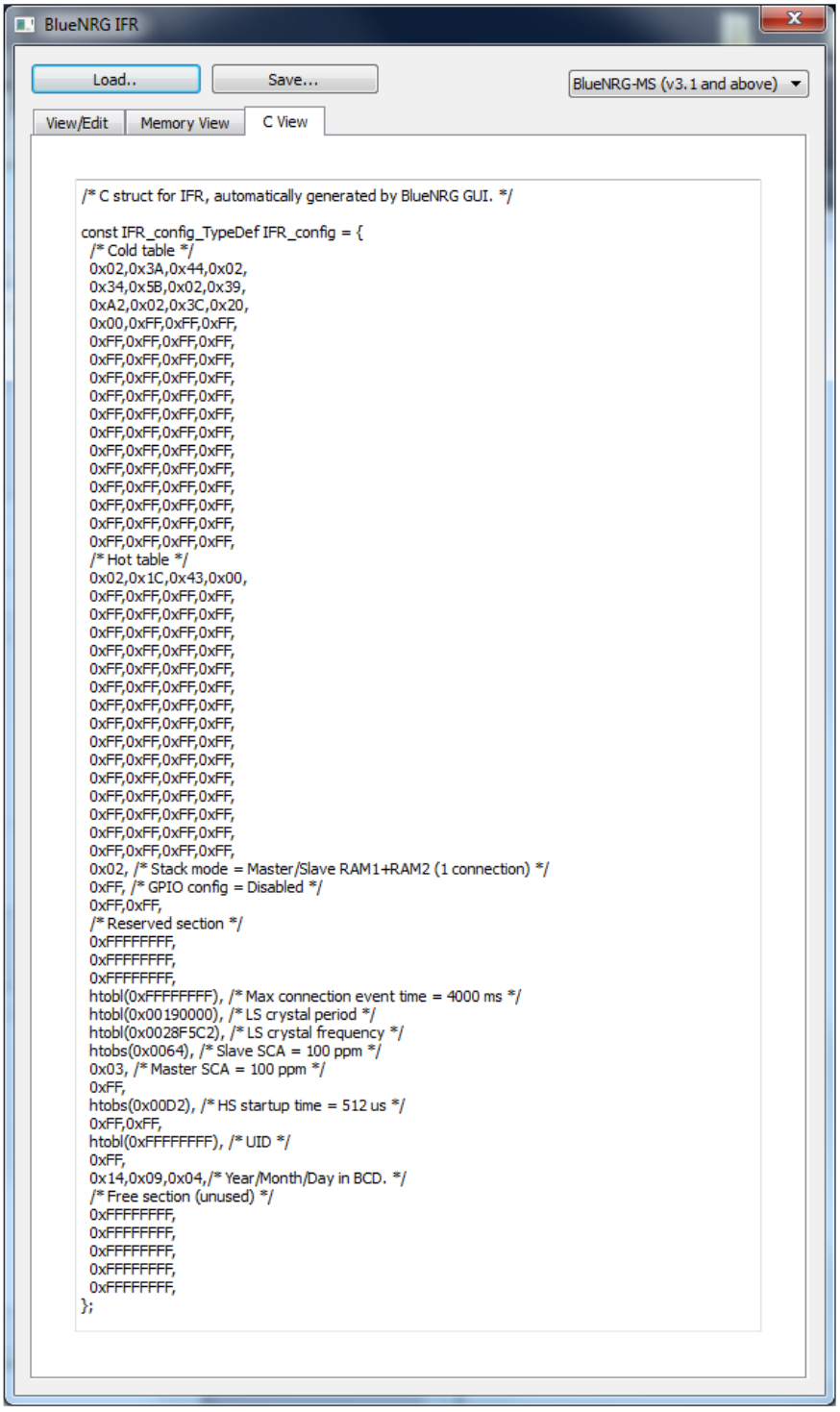
Figure 7. **BlueNRG GUI IFR memory view**



2.3 IFR C view

The BlueNRG GUI C-source view provides the C language structure, related to the IFR configuration data region and matching the IFR view/edit view. The BlueNRG GUI C-source view is automatically updated with the values loaded/configured on the BlueNRG GUI IFR utility view/edit view. It can be used to generate a C-language structure that can be used within the BlueNRG IFR updater demonstration firmware application with the option BLUENRG_CONFIG = BLUENRG_CUSTOM_CONFIG. This application can be customized and used by the customer during PCB manufacturing to customize the IFR data.

Figure 8. **BlueNRG GUI IFR C-source view**



3 BlueNRG IFR programming steps

Regarding IFR programming, two procedures are available:

- The BlueNRG GUI IFR utility, which is useful for bench testing and brings up application PCBs.
- Dedicated external microcontroller firmware based on the BlueNRG IFR ACI utility APIs, available within the latest available BlueNRG DK software package. This example firmware can be used during customer PCB manufacturing

Some reference IFR data configuration files (*.dat) are available with the BlueNRG DK software package. These files can be used as a starting point to customize the IFR based on customer needs.

The "Append to FW" button allows a new firmware, containing the IFR configuration currently set in the IFR utility, to be generated. If this new firmware containing the correct IFR configuration is downloaded into the device to upgrade current stack (e.g. during manufacturing), IFR settings do not need updating in a second step.

3.1 BlueNRG GUI IFR programming

The BlueNRG IFR can be programmed by following the steps below:

- Open the BlueNRG GUI available on the BlueNRG GUI SW package (STSW-BNRGUI)
- Connect the BlueNRG or BlueNRG-MS platform to a PC USB port
- Load the prebuilt Virtual COM image BlueNRG_VCOM_1_x.hex available on the Firmware \STM32L1_prebuilt_images using the GUI, tools, and Flash motherboard firmware provided that the BlueNRG or BlueNRG-MS platform is in DFU mode (on the Projects\Project\Virtual_COM_Port folder, an IAR project is also available for building and downloading the Virtual COM image through JTAG).
- On the BlueNRG GUI, select the associated COM port and click on the "Open" button
- Select the Tools, BlueNRG IFR... utility and the View/Edit tab
- Customize the IFR fields according to the user's needs or click on the “Load” button to load a saved IFR configuration file (*.dat file).
- Click on the "Write" button to write the IFR file into the BlueNRG or BlueNRG-MS device IFR

3.2 BlueNRG IFR updater demonstration application

On the BlueNRG DK software package, some IFR utility APIs are available to allow the device IFR to be updated through an application running on the external microcontroller. A reference demonstration application with the IAR project is already provided within the package on Projects\Project\BlueNRG_Stack_IFR_Updater folder. This example can be used to develop a customer-specific application to program the IFR during PCB manufacturing.

Example instructions

1. Open the IAR project EWARM\BlueNRG_Stack_IFR_Updater.eww for a BlueNRG device or EWARM_BlueNRG-MS\BlueNRG_Stack_IFR_Updater.eww for a BlueNRG-MS device.
2. Select the IFR_updater workspace
3. In the IAR preprocessor option, add one of the available define values for selecting the proper IFR data to be used. The following IFR options are available:
   a. **BLUENRG_CONFIG=BLUENRG_32_MHZ** (32 MHz high-speed crystal and external 32 kHz low-speed crystal configuration).
   b. **BLUENRG_CONFIG=BLUENRG_32_MHZ_RO** (32 MHz high-speed crystal and internal low-speed ring oscillator configuration).
   c. **BLUENRG_CONFIG=BLUENRG_16_MHZ** (16 MHz high-speed crystal and external 32 kHz low-speed crystal configuration).
   d. **BLUENRG_CONFIG=BLUENRG_16_MHZ_RO** (16 MHz high-speed crystal and internal low-speed ring oscillator configuration).
   e. **BLUENRG_CONFIG = BLUENRG_CUSTOM_CONFIG** (it allows custom IFR configuration data built with the BlueNRG GUI IFR utility to be used and made available on the IFR utility C View. The user has simply to copy the custom IFR_config structure on the IFR utility C view on the STM32L\Bluetooth LE \SimpleBlueNRG_HCI\hci\controller\bluenrg_IFR.c file as follows:

```
#elif BLUENRG_CONFIG == BLUENRG_CUSTOM_CONFIG

/* Copy and paste here your custom IFR_config structure.

It can be generated with BlueNRG GUI.*/

#endif
```

4. Build and download the built image to the BlueNRG or BlueNRG-MS platform. IFR data are programmed accordingly. If everything is ok, the LED D1 blinks.

4 List of acronyms

Table 5. **List of acronyms**

Term	Description
ACI	Application command interface
API	Application programming interface
DK	Development kit
GATT	Generic attribute profile
GPIO	General-purpose input/output
HS	High-speed
IFR	Information register
LS	Low-speed
SCA	Sleep clock accuracy
SMPS	Switched-mode power supply
SW	Software
UID	Universal identifier
USB	Universal serial bus
XTAL	Crystal

5 Related documentation and references

Table 6. **Related documentation and references**

Term	Description
AN4494	Bringing up the BlueNRG, BlueNRG-MS application note
STSW-BLUENRG-DK	BlueNRG, BlueNRG-MS development kit software package
UM1686	BlueNRG development kits user manual
UM1870	BlueNRG-MS development kits user manual
STSW-BNRGUI	BlueNRG GUI SW package

Revision history

Table 7. **Document revision history**

Date	Revision	Changes
05-Mar-2015	1	Initial release
19-Jul-2018	2	Updated Section 1 Information register sector, Section 2 BlueNRG GUI IFR utility. Updated Figure 5. BlueNRG GUI IFR view/edit tab for the BlueNRG/ BlueNRG-MS. Added Figure 6. IFR advanced.
18-Jan-2019	3	Updated Section 3.1 BlueNRG GUI IFR programming and Section 5 Related documentation and references.

Contents