
How to program and debug BlueNRG-1 and BlueNRG-2 devices

Main components	
BlueNRG-1	Bluetooth Low Energy wireless system-on-chip
BlueNRG-2	Bluetooth Low Energy wireless system-on-chip

Purpose and benefits

The BlueNRG-1 and BlueNRG-2 are very low power Bluetooth low energy (BLE) single-mode system-on-chip (SoC), compliant with Bluetooth specification. They extend the features of BlueNRG network processor, enabling the usage of the embedded Cortex M0 for running the user application code.

The BlueNRG-1 and BlueNRG-2 have 160 and 256 kB of embedded Flash memory respectively. Developers can reprogram them multiple times.

The purpose of this design tip is to provide customers with an overview of the tools available for programming the device Flash memory and how to debug the devices.

The entire content of this design tip applies both to BlueNRG-1 and to BlueNRG-2 devices with no modifications. For simplicity, we will refer to the BlueNRG-2 device in the following sections of the document.

Description

In order to program the device 3 options are available:

- Serial Wire Debug (SWD) interface
- Pre-programmed bootloader (over UART transport layer)
- Over-The-Air (OTA) device Firmware upgrade over the Bluetooth Low Energy radio link

SWD interface

The BlueNRG-2 device embeds a standard ARM SWD interface, that is available on the device pins DIO9 and DIO10 (refer to the device Datasheet).

The BlueNRG-1 ST-LINK Utility software is available on st.com via ST-LINK and ST-LINK/V2 tools for programming the device using the SWD interface. The BlueNRG-1 ST-LINK utility software package is delivered in two versions: the GUI (graphical user interface) version and the CLI (command line interface) version. These are the key feature of this tools:

- Supports Motorola S19, Intel® HEX and binary formats
- Load, Edit and Save executable and/or data files generated by the Assembler/Linker or C compilers
- Erase, Program, View and Verify device Flash memory contents
- Automates programming of BlueNRG-1 and BlueNRG-2 devices (Erase, Verify, Programming)
- Compares file with target memory
- Supports memory and core status view in Live update mode

The User Manual UM2109 describes how to use the BlueNRG-1 ST-LINK Utility software.

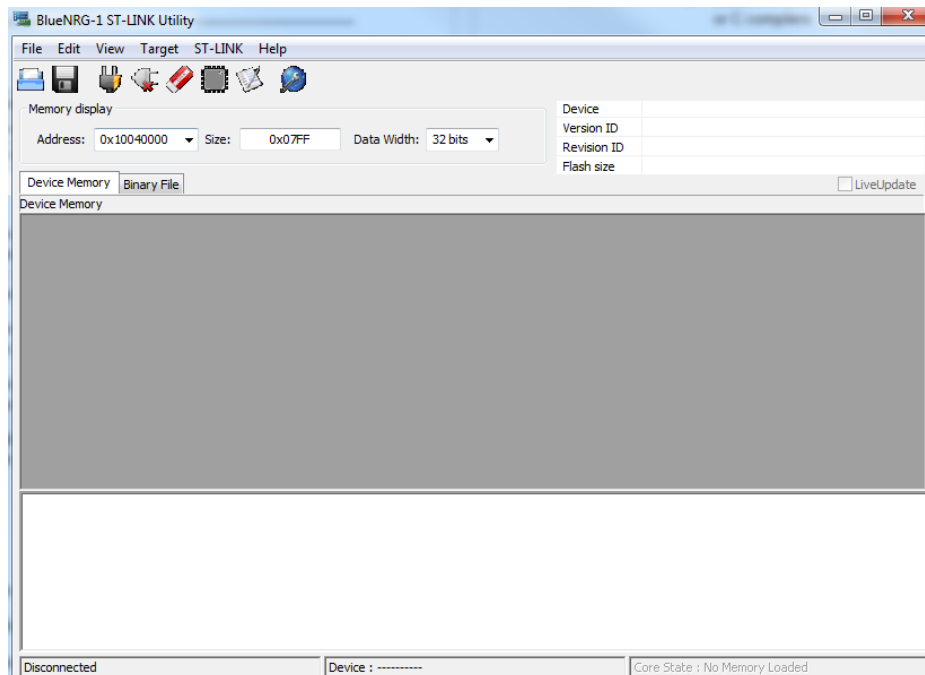


Figure 1: BlueNRG-1 ST-LINK Utility GUI interface

Pre-programmed bootloader

Additionally, the BlueNRG-2 device embeds a pre-programmed bootloader that is an application which is stored on the BlueNRG-2 internal ROM at manufacturing time by STMicroelectronics.

The bootloader is accessible from a UART interface ONLY on the device pins DIO8 and DIO11 (refer to the device Datasheet).

The UART bootloader is activated by hardware forcing high DIO7 pin at device reset. Once the bootloader is activated, the code starts a procedure to auto-detect the host UART baud rate and begins to scan the USART RX line pin.

The Application Note AN4872 contains the specifications of the BlueNRG-1 and BlueNRG-2 UART bootloader protocol.

Moreover, in the BlueNRG-2 device software development kit (SDK) a PC interface is available for programming the device Flash memory using this protocol. The interface is named BlueNRG-1 Flasher Utility.

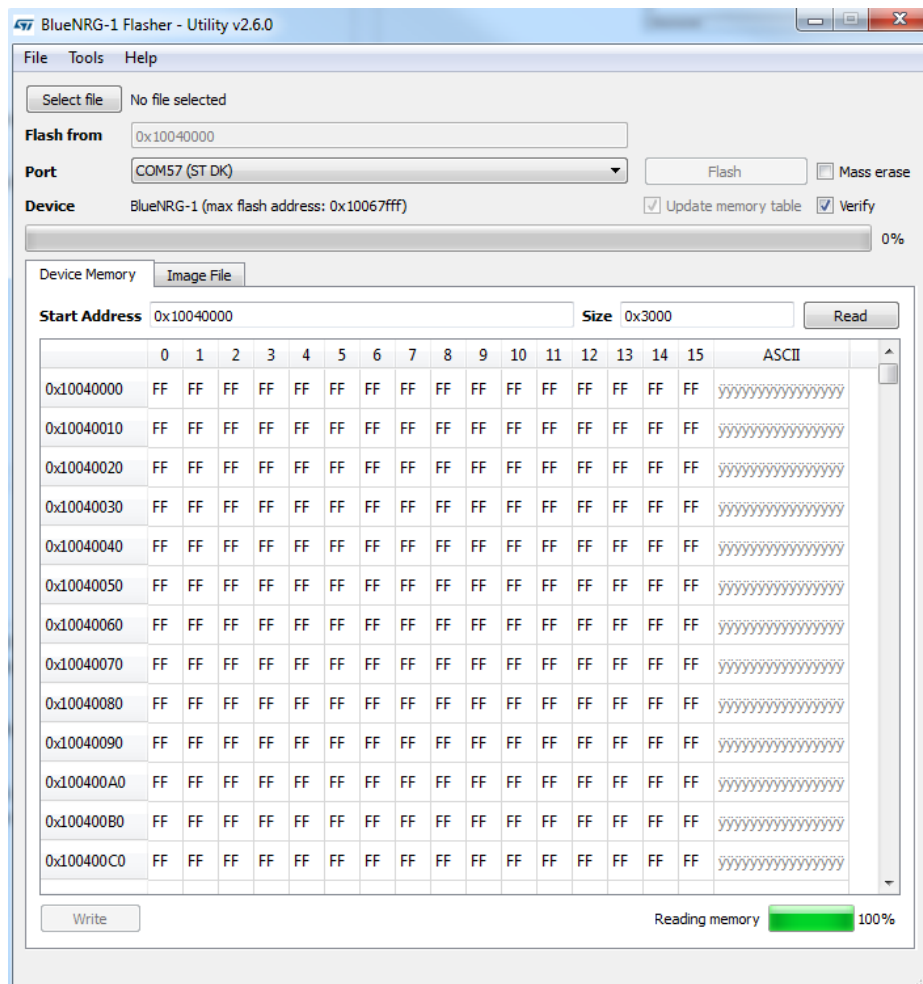


Figure 2: BlueNRG-1 Flasher Utility GUI interface

Note that the customer application must ensure that IO7 is forced low during power up in order not to boot the device in updater mode.

OTA FW upgrade

Finally, the last option for programming the BlueNRG-2 Flash memory is the “over-the-air” (OTA) firmware upgrade procedure running on top of Bluetooth low energy (BLE) stack provided with the BlueNRG-1, BlueNRG-2 systems-on-chip.

OTA firmware upgrade is a protocol that allows a Bluetooth low energy slave device to receive a firmware image over-the-air from a Bluetooth low energy master device and write it into Flash memory.

The Application Note AN4869 describes the SW framework for BlueNRG-1, BlueNRG-2 BLE OTA (over-the-air) firmware upgrade. Additionally, reference application example in source code are available in the device SDK.

Debug tips

There are certain situations where debug access is disabled and the chip cannot be accessed, including:

- User application that disables debug pins
- User application that sets the device in sleep or standby state. In these states the debug port is not powered.

These cases are common during application development and device can end up in a state where debug access is no longer possible. To recover this situation, it is recommended to force IO7 pin high and hardware reset the device in order to force execution of the updater code (see Section 3.23: "Pre-programmed bootloader"). The user can then connect with SWD interface and erase the device Flash memory.

Another possibility, in case the application sets the device in sleep or standby state, is to delay entering in such states for the time necessary to access the debug port. The following pseudo-code implements this routine making use of a control variable and a virtual timer, set to 10 seconds.

The variable *wait_SWD* will allow entering in sleep or standby mode, allowing the library function *BlueNRG_Sleep* to be executed.

```
volatile uint8_t wait_SWD = 0;
```

The virtual timer 0 is set at the device boot to expire in 10 seconds (10 seconds is just a reference number that can be modified accordingly to user needs).

```
HAL_VTimerStart_ms(0, 10000);
```

In the main loop the execution of the *BlueNRG_Sleep* function is conditioned to the value of the variable *wait_SWD*.

```
if(wait_SWD)
```

```
    ret = BlueNRG_Sleep(SLEEPMODE_NOTIMER, wakeup_source, wakeup_level, 0);
```

In the library callback raised at a virtual timer expiration, the variable *wait_SWD* changes is set to 1 upon expiration of the virtual timer 0. This will enable the execution of the *BlueNRG_Sleep* function, which will let the BlueNRG-2 core entering sleep or standby state.

```
void HAL_VTimerTimeoutCallback(uint8_t timerNum)
```

```
{  
  
    if(timerNum==0) {  
  
        wait_SWD =1; //Set bit for sleep  
  
    }  
  
}
```

BlueNRG-2 ST Eval kits

The official evaluation kits for the BlueNRG-2 device are named STEVAL-IDB008V2 or STEVAL-IDB008V1, while for the BlueNRG-1 device the official evaluation kit is named STEVAL-IDB007V1.

All of them are pre-programmed with a reference FW application named "Sensor Demo" (refer to the device SDK for more details). This reference FW application sets the device in sleep mode thus disabling the SWD interface. In order to re-enable the debug port, the recommended procedure is to first erase the device Flash using the pre-programmed bootloader. By simply using the USB connection from the PC to the STEVAL board, users can run the BlueNRG-1 Flasher utility. The STEVAL-IDB008V2 (or the IDB008V1/IDB007V1) is recognized as a COM port from the PC. From the BlueNRG-1 Flasher utility users need to select the corresponding COM port: this will start the reading of the device Flash memory content. At this point, user can select the tab "Tools->Mass Erase" for erasing the memory. The SWD interface will be then active and accessible from the ST-LINK and ST-LINK/V2 debugger.

Support material

Documentation
Datasheets: BlueNRG-1 BlueNRG-2 Programming Manual: PM0257: BLE stack programming guidelines Application Notes: AN4869: BlueNRG-1, BlueNRG-2 BLE OTA (over-the-air) firmware upgrade AN4872: BlueNRG-1 and BlueNRG-2 UART bootloader protocol User Manuals: UM2109: BlueNRG-1 ST-LINK Utility software description
Embedded Software: STSW-BLUENRG1-DK: BlueNRG-1, BlueNRG-2 DK SW package STSW-BNRG_V1-DK: BlueNRG-1 DK SW package for BLE stack family v1.x STSW-BNRG1STLINK: BlueNRG-1 ST-LINK utility for BlueNRG-1, BlueNRG-2 MCU

Revision history

Date	Version	Changes
27-Sep-2018	1	Initial release

IMPORTANT NOTICE – PLEASE READ CAREFULLY

STMicroelectronics NV and its subsidiaries (“ST”) reserve the right to make changes, corrections, enhancements, modifications, and improvements to ST products and/or to this document at any time without notice. Purchasers should obtain the latest relevant information on ST products before placing orders. ST products are sold pursuant to ST’s terms and conditions of sale in place at the time of order acknowledgement.

Purchasers are solely responsible for the choice, selection, and use of ST products and ST assumes no liability for application assistance or the design of Purchasers’ products.

No license, express or implied, to any intellectual property right is granted by ST herein.

Resale of ST products with provisions different from the information set forth herein shall void any warranty granted by ST for such product.

ST and the ST logo are trademarks of ST. All other product or service names are the property of their respective owners.

Information in this document supersedes and replaces information previously supplied in any prior versions of this document.

© 2018 STMicroelectronics – All rights reserved