Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!


## Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832
Email & Skype: info@chipsmall.com Web: www.chipsmall.com
Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China

# BlueNRG-2

Datasheet

## Bluetooth® low energy wireless system-on-chip


QFN48

WLCSP34


QFN32

| Product status link |
| --- |
| BlueNRG-2 |

## Features

- Bluetooth low energy single mode system-on-chip compliant with Bluetooth 5.0 specifications:
  - master, slave and multiple simultaneous roles
  - LE data packet length extension
- Operating supply voltage: from 1.7 to 3.6 V
- Integrated linear regulator and DC-DC step-down converter
- Operating temperature range: -40 °C to 105 °C
- High performance, ultra-low power Cortex-M0 32-bit based architecture core
- Programmable 256 kB Flash
- 24 kB RAM with retention (two 12 kB banks)
- 1 x UART interface
- 1 x SPI interface
- 2 x I²C interface
- 14, 15 or 26 GPIOs
- 2 x multifunction timer
- 10-bit ADC
- Watchdog and RTC
- DMA controller
- PDM stream processor
- 16 or 32 MHz crystal oscillator
- 32 kHz crystal oscillator
- 32 kHz ring oscillator
- Battery voltage and temperature sensors
- Up to +8 dBm available output power (at antenna connector)
- Excellent RF link budget (up to 96 dB)
- Accurate RSSI to allow power control
- 8.3 mA TX current (@ -2 dBm, 3.0 V)
- Down to 1 µA current consumption with active BLE stack (sleep mode)
- ST companion integrated balun/filter chips are available
- Average advertisement current consumption 15.34 µA (advertisement interval 1000 ms) – 1 year, 8 months, 19 days with 230 mAh battery (CR2032)
- Average connection current consumption 7.059 µA (connection interval 1000 ms) – 3 years, 10 months, 12 days with 230 mAh battery (CR2032)
- Suitable for building applications compliant with the following radio frequency regulations: ETSI EN 300 328, EN 300 440, FCC CFR47 part 15, ARIB STD-T66
- Pre-programmed bootloader via UART
- QFN32, QFN48 and WCSP34 package options

## Applications

- Watches

- Fitness, wellness and sports
- Consumer medical
- Security/proximity
- Remote control
- Home and industrial automation
- Assisted living
- Mobile phone peripherals
- Lighting
- PC peripherals

# 1 Description

The BlueNRG-2 is a very low power Bluetooth low energy (BLE) single-mode system-on-chip, compliant with Bluetooth specifications.

The BlueNRG-2 extends the features of award-winning BlueNRG network processor, enabling the usage of the embedded Cortex M0 to run the user application code.

The BlueNRG-2 includes 256 kB of programming Flash memory, 24 kB of static RAM memory with retention (two 12 kB banks) and SPI, UART, I²C standard communication interface peripherals. It also features multifunction timers, watchdog, RTC and DMA controller.

An ADC is available to interface with analog sensors, and to read the measurement of the integrated battery voltage sensor. A digital filter is available to process a PDM stream.

The BlueNRG-2 offers the same excellent RF performance of the BlueNRG radio, and the integrated high efficiency DC-DC converter keeps the same ultra-low power characteristics, but the BlueNRG-2 improves the BlueNRG sleep mode current consumption allowing a further increase in the battery lifetime of the applications.

**Figure 1. BlueNRG-2 architecture**

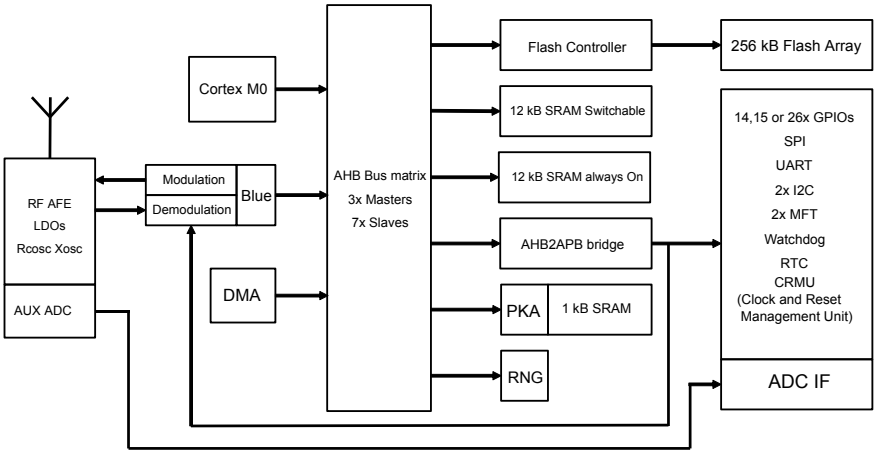
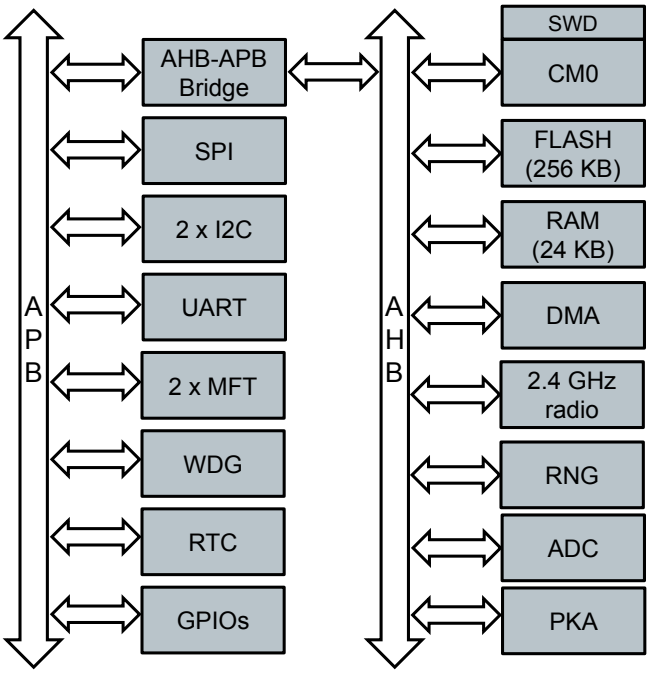


Figure 2. **BlueNRG-2 bus architecture**



# 2 BlueNRG-2 Bluetooth low energy stack

The BlueNRG-2 is complemented with a Bluetooth low energy stack C library that provides:

- Master, slave role support
- GAP: central, peripheral, observer or broadcaster roles
- ATT/GATT: client and server
- SM: privacy, authentication and authorization
- L2CAP
- Link layer: AES-128 encryption and decryption

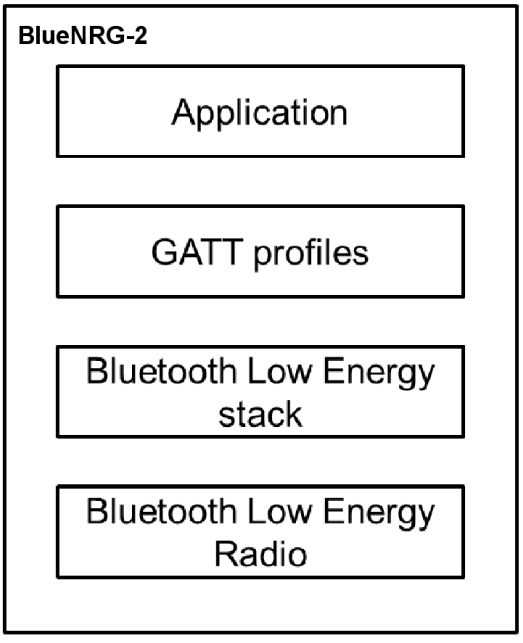
The BlueNRG-2 can be configured to support single chip or network processor applications.

The BlueNRG-2 supports LE data packet length extension, in compliance with Bluetooth smart v5.0.

In the first configuration, the BlueNRG-2 operates as single device in the application for managing both the application code and the Bluetooth low energy stack. The whole Bluetooth low energy stack is provided as object code in a single library file whereas the GATT low energy profiles are provided as object codes in separate libraries.

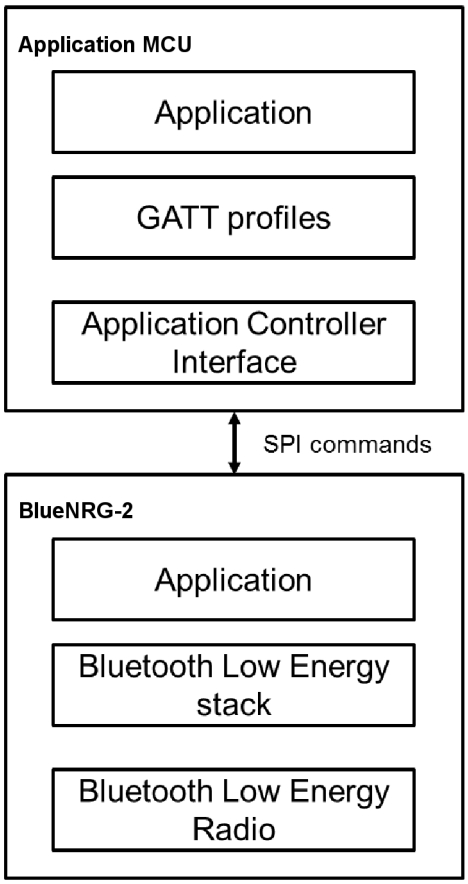
The figure below shows the single chip RF software layers.

**Figure 3. BlueNRG-2 single-chip RF software layers**




The BlueNRG-2 can be configured to operate as a network coprocessor. In this case, dedicated firmware is provided to support the interface with an external application processor. The whole Bluetooth low energy stack runs in the BlueNRG-2; the GATT profiles are provided to run in the application processor together with the application code. The figure below shows the network processor RF software layers.

Figure 4. **BlueNRG-2 network processor RF software layers**



# 3 Functional details

The BlueNRG-2 integrates:

- ARM Cortex-M0 core
- Interrupts management
- 256 kB Flash memory
- 24 kB of RAM with two retention options (12 kB or 24 kB)
- Power management
- Clocks
- Bluetooth low energy radio
- Random number generator (RNG) (reserved for Bluetooth low energy protocol stack, but user applications can read it)
- Public key cryptography (PKA) (reserved for Bluetooth low energy protocol stack)
- Peripherals:
  - SPI interface
  - UART interface
  - I²C bus interface
  - GPIO
  - Multifunction timer
  - DMA controller
  - Watchdog
  - RTC
  - ADC with battery voltage sensor and temperature sensor
  - PDM stream processor

## 3.1 Core

The ARM® Cortex®-M0 processor has been developed to provide a low-cost platform that meets the needs of MCU implementation, with a reduced pin count and low-power consumption, while delivering outstanding computational performance and an advanced system response to interrupts.

The ARM® Cortex®-M0 32-bit RISC processor features exceptional code-efficiency, delivering the high-performance expected from an ARM core in the memory size usually associated with 8-bit and 16-bit devices.

The BlueNRG-2 has an embedded ARM core and is therefore compatible with all ARM tools and software.

## 3.2 Interrupts

The Cortex-M0 nested vector interrupt controller (NVIC) handles interrupts. The NVIC controls specific Cortex-M0 interrupts (address 0x00 to 0x3C) as well as 32-user interrupts (address 0x40 to 0xBC). In the BlueNRG-2 device, the user interrupts are connected to the interrupt signals of the different peripherals.

**Table 1. BlueNRG-2 interrupt vectors**

| Position | Priority | Priority type | Description | Address |
|---|---|---|---|---|
| | | | Initial main SP | 0x0000_0000 |
| | -3 | Fixed | Reset handler | 0x0000_0004 |
| | -2 | Fixed | NMI handler | 0x0000_0008 |
| | -1 | Fixed | HardFault handler | 0x0000_000C |
| | | | RESERVED | 0x0000_000C – 0x0000_0028 |

| Position | Priority | Priority type | Description | Address |
|---|---|---|---|---|
| | 3 | Settable | SVC handler | 0x0000_002C |
| | | | RESERVED | 0x0000_0030 - 0x0000_0034 |
| | 5 | Settable | PendSV handler | 0x0000_0038 |
| | 6 | Settable | SystemTick handler | 0x0000_003C |
| 0 | Init 0 | Settable | GPIO interrupt | 0x0000_0040 |
| 1 | Init 0 | Settable | FLASH controller interrupt | 0x0000_0044 |
| 2 | Init 0 | Settable | RESERVED | 0x0000_0048 |
| 3 | Init 0 | Settable | RESERVED | 0x0000_004C |
| 4 | Init 0 | Settable | UART interrupt | 0x0000_0050 |
| 5 | Init 0 | Settable | SPI interrupt | 0x0000_0054 |
| 6 | Init 0 | CRITICAL | BLE controller interrupt | 0x0000_0058 |
| 7 | Init 0 | Settable | Watchdog interrupt | 0x0000_005C |
| 8 | Init 0 | Settable | RESERVED | 0x0000_0060 |
| 9 | Init 0 | Settable | RESERVED | 0x0000_0064 |
| 10 | Init 0 | Settable | RESERVED | 0x0000_0068 |
| 11 | Init 0 | Settable | RESERVED | 0x0000_006C |
| 12 | Init 0 | Settable | RESERVED | 0x0000_0070 |
| 13 | Init 0 | Settable | ADC interrupt | 0x0000_0074 |
| 14 | Init 0 | Settable | I2C 2 interrupt | 0x0000_0078 |
| 15 | Init 0 | Settable | I2C 1 interrupt | 0x0000_007C |
| 16 | Init 0 | Settable | RESERVED | 0x0000_0080 |
| 17 | Init 0 | Settable | MFT1 A interrupt | 0x0000_0084 |
| 18 | Init 0 | Settable | MFT1 B interrupt | 0x0000_0088 |
| 19 | Init 0 | Settable | MFT2 A interrupt | 0x0000_008C |
| 20 | Init 0 | Settable | MFT2 B interrupt | 0x0000_0090 |
| 21 | Init 0 | Settable | RTC interrupt | 0x0000_0094 |
| 22 | Init 0 | Settable | PKA interrupt | 0x0000_0098 |
| 23 | Init 0 | Settable | DMA interrupt | 0x0000_009C |
| 24 – 31 | Init 0 | Settable | RESERVED | 0x0000_00A0 – 0x0000_00BC |

## 3.3 Memories

The memory subsystem consists 256 kB Flash memory and two banks of 12 kB ultra-low leakage static RAM blocks.

The 256 kB Flash memory is available to the user and can be accessed per 32-bit for read access and per 32-bit for write access (with 4x32-bit FIFO).

The access to the static RAM can be bytes, half words (16 bits) or words (32 bits).

A 12 kB RAM block is always in retention mode, whereas the second 12 kB RAM block is switchable and can be put in retention mode according to the user needs.

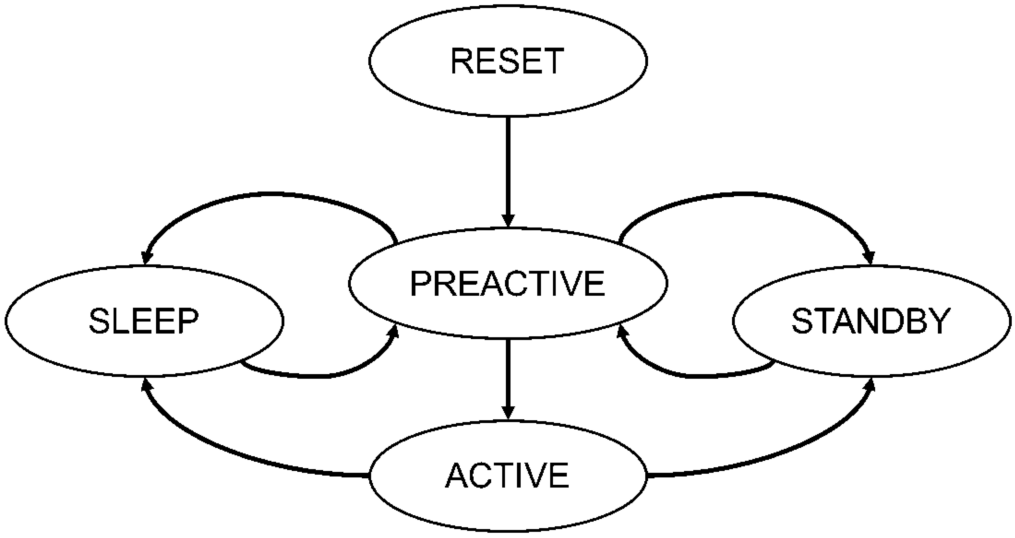
## 3.4 Power management

The BlueNRG-2 integrates both a low dropout voltage regulator (LDO) and a step-down DC-DC converter to supply the internal BlueNRG-2 circuitry.

The BlueNRG-2 most efficient power management configuration is with DC-DC converter active where best power consumption is obtained without compromising performances. Nevertheless, a configuration based on LDO can also be used, if needed.

A simplified version of the state machine is shown below.

**Figure 5. BlueNRG-2 power management state machine**




### 3.4.1 State description

#### 3.4.1.1 *Preactive state*

The preactive state is the default state after a POR event.

In this state:

- All the digital power supplies are stable.
- The high frequency clock runs on internal fast clock RC oscillator (16 MHz).
- The low frequency clock runs on internal RC oscillator (32.768 kHz).

#### 3.4.1.2 *Active state*

In this state:

- The high frequency runs on the accurate clock (16 MHz ±50 ppm) provided by the external XO. The internal fast clock RO oscillator is switched off.

*Note: If the external XO is at 32 MHz, some specific programming are needed, see Switching to external clock.*

#### 3.4.1.3 *Standby state*

In this state:

- Only the digital power supplies necessary to keep the RAM in retention are used.

The wake-up from this low power state is driven by the following sources:

- IO9
- IO10
- IO11
- IO12
- IO13(1)

1. *Not available on WCSP34.*

If they have been programmed as wake-up source in the system controller registers.

#### 3.4.1.4 *Sleep state*

In this state:

- Only the digital power supplies necessary to keep the RAM in retention are used.
- The low frequency oscillator is switched on.

The wake-up from this low power state is driven by the following sources:

- IO9
- IO10
- IO11
- IO12
- IO13 (1)

1. *Not available on WCSP34.*

If they have been programmed as wake-up source in the system controller registers and from the internal timers of the BLE radio.

#### 3.4.1.5 *IO wake-up sources*

The IOs programmed to be wake-up sources need an external drive according to the selected level sensitivity.

If the wake-up level is high level, a pull-down drive should be used. If the wake-up level is low level, a pull-up drive should be used.

If no external drive is applied, IO9, IO10 and IO11 are only sensitive to low level as they have an internal pull-up (activated by default). IO12 and IO13 do not have an internal pull and therefore require an external drive.

#### 3.4.1.6 *Wake-up time*

The wake-up time is typically 200 µs at 3.0 V and a temperature of 25 °C.

#### 3.4.1.7 *GPIO special settings in low power modes*

The GPIO9, GPIO10 and GPIO11 can be configured as output GPIO during sleep and standby mode. In addition, they can have enabled their internal pull. Their configuration is done in specific registers:

- SLEEPIO_OEN: has the same functionality of the register OEN (GPIO peripheral) and it is used to configure in output mode or input mode (default)
- SLEEPIO_PE: has the same functionality of the register PE (GPIO peripheral) and it is used to enable the internal pull. This register allows the internal pull of these IOs to be enabled or disabled also if they are not configured in output state.
- SLEEPIO_DS: has the same functionality of the register DS (GPIO peripheral) and it is used to configure the drive strength
- SLEEPIO_OUT: has the same functionality of the register DATA (GPIO peripheral) and it is used to set the state of the GPIO (high state or low state)

*Note: If the GPIO9, GPIO10 or GPIO11 are used as wake-up source, then the SLEEPIO_OEN has no effect, but it is always possible to enable or disable the internal pull.*

### 3.4.2 Power saving strategy

The application power saving strategy is based on clock stopping, dynamic clock gating,

digital power supply switch off and analog current consumption minimization.

A summary of functional blocks versus the BlueNRG-2 states is provided below.

**Table 2. Relationship between the BlueNRG-2 states and functional blocks**

| Functional blocks | RESET | STANDBY | SLEEP | Preactive | Active | LOCK RX/ LOCK TX | RX | TX |
|---|---|---|---|---|---|---|---|---|
| LDO_SOFT_1V2 or LDO_SOFT_0V9 | OFF | ON | ON | ON | ON | ON | ON | ON |

| Functional blocks | RESET | STANDBY | SLEEP | Preactive | Active | LOCK RX/ LOCK TX | RX | TX |
|---|---|---|---|---|---|---|---|---|
| LDO_STRONG_1V2 | OFF | OFF | OFF | ON | ON | ON | ON | ON |
| LDO_DIG_1V8 | OFF | OFF | OFF | ON | ON | ON | ON | ON |
| SMPS | OFF | OFF | OFF | ON | ON | ON | ON | ON |
| LDO_DIG_1V2 | OFF | OFF | OFF | ON | ON | ON | ON | ON |
| BOR | OFF | OFF | OFF | ON | ON | ON | ON | ON |
| 16 MHz RO | OFF | OFF | OFF | ON | OFF | OFF | OFF | OFF |
| 16 MHz XO | OFF | OFF | OFF | OFF | ON | ON | ON | ON |
| 32 kHz RO or XO | OFF | OFF | ON | ON | ON | ON | ON | ON |

### 3.4.3 System controller registers

SYSTEM_CTRL peripheral base address (SYSTEM_CTRL_BASE_ADDR) 0x40200000.

**Table 3. SYSTEM_CTRL registers**

| Address offset | Name | RW | Reset | Description |
|---|---|---|---|---|
| 0x00 | WKP_IO_IS | RW | 0x00000000 | Level selection for wake-up IO (1 bit for IO).<br>0: The system wakes up when IO is low.<br>1: The system wakes up when IO is high. |
| 0x04 | WKP_IO_IE | RW | 0x00000007 | Enables the IO that wakes up the device (1 bit for IO).<br>0: The wake-up feature on the IO is disabled.<br>1: The wake-up feature on the IO is enabled. |
| 0x08 | CTRL | RW | 0x00000000 | XO frequency indication is provided by the application. Refer to the detailed description below. |
| 0x0C | SLEEPIO_OEN | RW | 0x00000007 | GPIO output enable register for low power modes. |
| 0x10 | SLEEPIO_OUT | RW | 0x00000000 | GPIO output value register for low power modes. |
| 0x14 | SLEEPIO_DS | RW | 0x00000000 | GPIO drive strength control register for low power modes. |
| 0x18 | SLEEPIO_PE | RW | 0x00000000 | GPIO pull enable register for low power modes. |

**Table 4. SYSTEM_CTRL - WKP_IO_IS register description: address offset SYSTEM_CTRL_BASE_ADDR+0x00**

| Bit | Field name | Reset | RW | Description |
|---|---|---|---|---|
| 4:0 | LEVEL_SEL | 0x00 | RW | Selects the active wake-up level for the five IOs.0: The system wakes up when IO is low.<br>1: The system wakes up when IO is high.<br>One bit by IO:<br>Bit0: IO9<br>Bit1: IO10<br>Bit2: IO11<br>Bit3: IO12<br>Bit4: IO13 |
| 31:5 | RESERVED | 0x00 | RW | RESERVED |

**Table 5. SYSTEM_CTRL - WKP_IO_IE register description: address offset SYSTEM_CTRL_BASE_ADDR+0x04**

| Bit | Field name | Reset | RW | Description |
|---|---|---|---|---|
| 4:0 | IO_WAKEUP_EN | 0x07 | RW | Enables the IOs to be wake-up source.<br>0: Wake-up on the IO disabled.<br>1: Wake-up on the IO enabled.<br>One bit by IO:<br>Bit0: IO9<br>Bit1: IO10<br>Bit2: IO11<br>Bit3: IO12<br>Bit4: IO13 |
| 31:5 | RESERVED | 0x00 | RW | RESERVED |

**Table 6. SYSTEM_CTRL - CTRL register description: address offset SYSTEM_CTRL_BASE_ADDR+0x08**

| Bit | Field name | Reset | RW | Description |
|---|---|---|---|---|
| 0 | MHZ32_SEL | 0x0 | RW | Indicates the crystal frequency used in the application.<br>0: The 16 MHz is selected.<br>1: The 32 MHz is selected. |
| 31:1 | RESERVED | 0x0 | RW | RESERVED |

**Table 7. SYSTEM_CTRL - SLEEPIO_OEN register description: address offset SYSTEM_CTRL_BASE_ADDR+0x0C**

| Bit | Field name | Reset | RW | Description |
|---|---|---|---|---|
| 2:0 | SLEEPIO_OEN | 0x07 | RW | Enables the IOs to act as output during low power modes.<br>0: output mode.<br>1: input mode.<br>One bit by IO:<br>Bit0: IO9<br>Bit1: IO10<br>Bit2: IO11 |
| 31:4 | RESERVED | 0x00 | RW | RESERVED |

**Table 8. SYSTEM_CTRL – SLEEPIO_OUT register description: address offset SYSTEM_CTRL_BASE_ADDR+0x10**

| Bit | Field name | Reset | RW | Description |
|---|---|---|---|---|
| 2:0 | SLEEPIO_OUT | 0x00 | RW | Writing to a bit drives the written value on the corresponding IO when it is configured in output direction in SLEEPIO_OEN register. Reading a bit in this register returns the last written value. One bit by IO:<br>Bit0: IO9<br>Bit1: IO10<br>Bit2: IO11 |
| 31:4 | RESERVED | 0x00 | RW | RESERVED |

**Table 9. SYSTEM_CTRL - SLEEPIO_DS register description: address offset SYSTEM_CTRL_BASE_ADDR+0x14**

| Bit | Field name | Reset | RW | Description |
|---|---|---|---|---|
| 2:0 | SLEEPIO_DS | 0x00 | RW | Configure the drive strength during low power modes for the IO9, IO10 and IO11.<br>0: low drive strength.<br>1: high drive strength<br>One bit by IO:<br>Bit0: IO9<br>Bit1: IO10<br>Bit2: IO11 |
| 31:4 | RESERVED | 0x00 | RW | RESERVED |

**Table 10. SYSTEM_CTRL - SLEEPIO_PE register description: address offset SYSTEM_CTRL_BASE_ADDR+0x18**

| Bit | Field name | Reset | RW | Description |
|---|---|---|---|---|
| 2:0 | SLEEPIO_PE | 0x00 | RW | Enable/disable the internal pull during low power modes for the IO9, IO10 and IO11.<br>0: pull disabled.<br>1: pull enabled.<br>One bit by IO:<br>Bit0: IO9<br>Bit1: IO10<br>Bit2: IO11 |
| 31:4 | RESERVED | 0x00 | RW | RESERVED |

AHBUPCONV peripheral base address (AHBUPCONV_BASE_ADDR) 0x40C00000.

**Table 11. AHBUPCONV registers**

| Address offset | Name | RW | Reset | Description |
|---|---|---|---|---|
| 0x00 | COMMAND | RW | 0x00000000 | AHB up/down converter command register |

BLUE_CTRL peripheral base address (BLUE_CTRL_BASE_ADDR) 0x48000000.

**Table 12. BLUE_CTRL registers**

| Address offset | Name | RW | Reset | Description |
|---|---|---|---|---|
| 0x04 | TIMEOUT | RW | 0x00000000 | Timeout programming register |
| 0x0C | RADIO_CONFIG | RW | 0x00000000 | Radio configuration register |

*Note: All RESERVED fields inside registers must always be written with their default value.*

## 3.5 Clocks and reset management

The BlueNRG-2 embeds an RC low-speed frequency oscillator at 32 kHz and an RO high-speed frequency oscillator at 16 MHz.

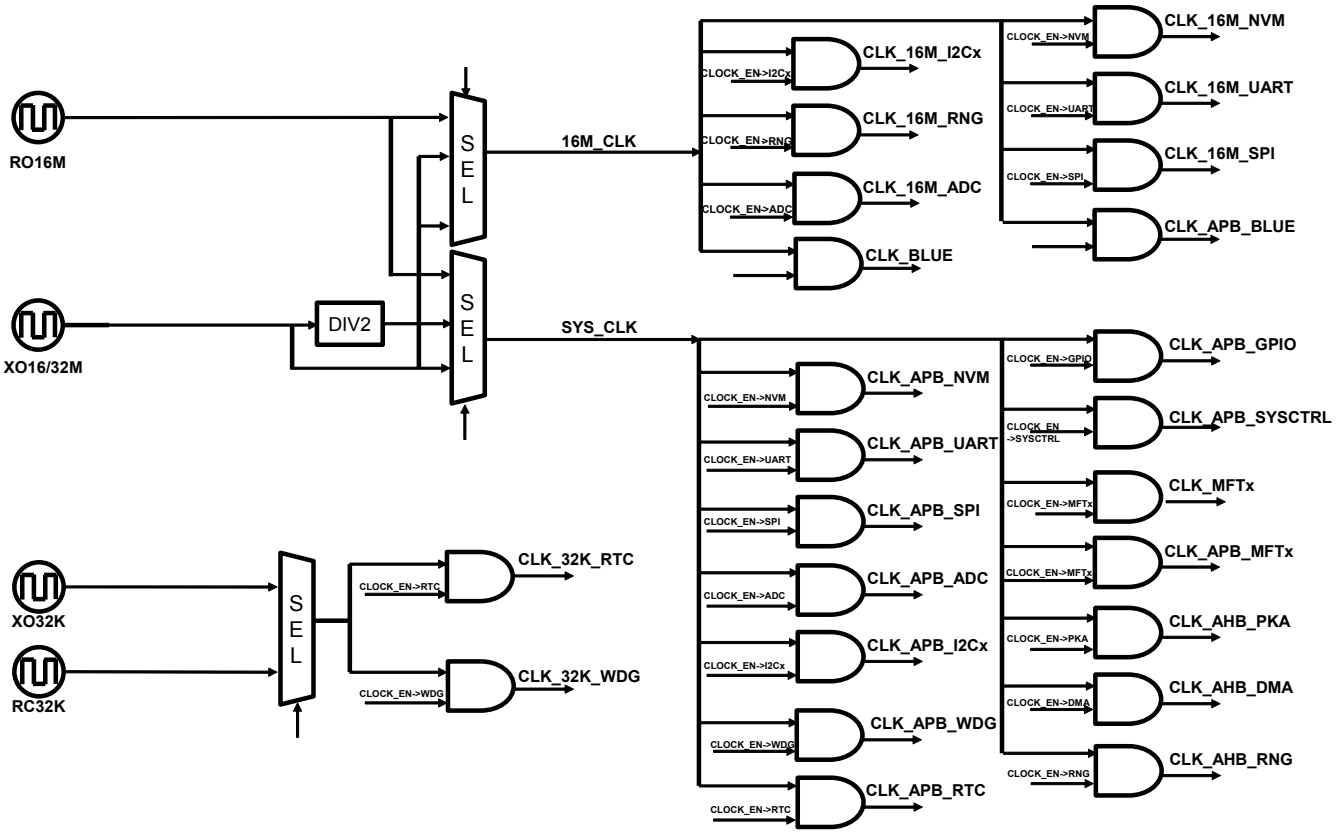
The low-frequency clock is used in low power mode and can be supplied either by a 32.7 kHz oscillator that uses an external crystal and guarantees up to ±50 ppm frequency tolerance, or by a ring oscillator, which does not require any external components.

The primary high-speed frequency clock is a 16 MHz or 32 MHz crystal oscillator. A fast-starting 16 MHz ring oscillator provides the clock while the crystal oscillator is starting up. Frequency tolerance of high-speed crystal oscillator is ±50 ppm.

The usage of the 16 MHz (for constraints related to the 16 MHz high-speed crystal usage, refer to the BlueNRG-1 DK SW release notes) (or 32 MHz) crystal is strictly necessary for RF communications.

The clock tree for the peripherals is as follows:

**Figure 6. Clock tree**




*Note: When 32 MHz XO is used, the Cortex-M0, the DMA and the APB tree (except for BLE radio access) run at 32 MHz. The rest of the clock tree is divided by two and is at 16 MHz.*

The following clocks can be enabled/disabled by software to implement optimal power consumption:

- DMA
- BLE controller
- BLE clock generator
- RNG
- Flash controller
- GPIO
- System controller
- UART
- SPI
- I2C1(1)
- I2C2

- ADC
- MFT1
- MFT2
- RTC
- WDG
- PKA

1. *The $I^2C1$ is not available with WLCSP34 package.*

By default, all the peripheral APB and AHB clocks are enabled, except for the PKA peripheral. The following clocks are enabled/disabled automatically:

- Processor clock (disabled in sleep mode)
- RAM clock (disabled if processor clock, SPI clock and BLE clock are all disabled)

*Note: It is possible to provide an external 32 kHz signal to the BlueNRG-2 device through the SXTAL0 pin by sourcing a periodic waveform from 0 to 1.2 V.*

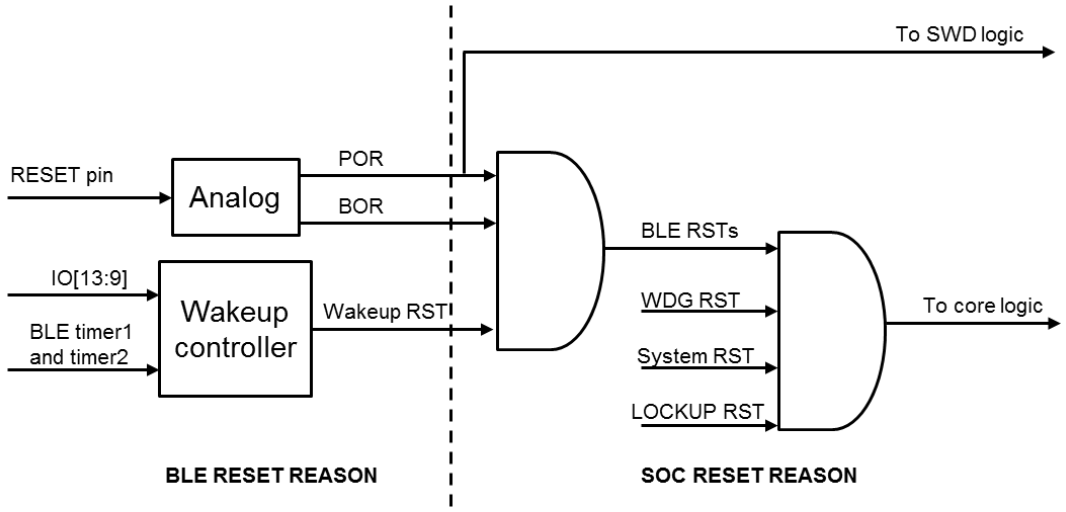
### 3.5.1 Reset management

Figure 7. Reset and wake-up generation shows the general principle of reset. Releasing the reset pin puts the chip out of shutdown state. The wake-up logic is powered and receives the POR. Each time the wake-up controller decides to exit sleep or standby modes, it will generate a reset for the core logic. The core logic can also be reset by:

- Watchdog
- Reset request from the processor (system reset)
- LOCKUP state of the Cortex-M0

The SWD logic is reset by the POR. It is important to highlight that reset pin actually power down chip, so it is not possible to perform debug access with system under reset.

**Figure 7. Reset and wake-up generation**




If, for any reason, the users would like to power off the device there are two options:

1. Force RESETN pin to ground, keeping VBAT level
2. To put VBAT pins to ground (e.g. via a transistor)

In the second option, care must be taken to ensure that no voltage is applied to any of the other pins since device can be powered and having an anomalous power consumption. ST recommendation is to use RESETN whenever it is possible.

#### 3.5.1.1 *Power-on-Reset*

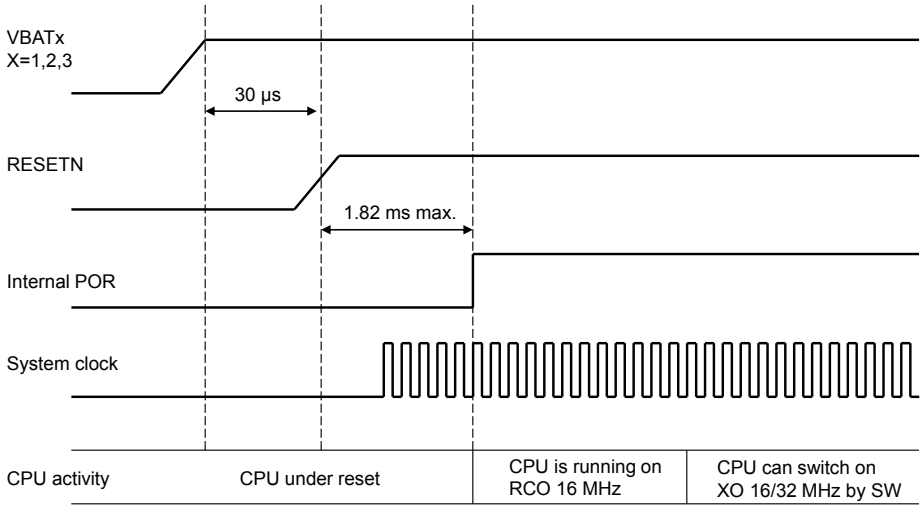
The Power-on-Reset (POR) signal is the combination of the POR signal and the BOR signal generated by the analog circuitry contained in the BlueNRG-2 device. The combination of these signals is used to generate the input to the Cortex-M0, which is used to reset the debug access port (DAP) of the processor. It is also used to generate the signal, which resets the debug logic of the Cortex-M0. The POR signal also resets the TAP controller of the BlueNRG-2 and a part of the Flash controller (managing the Flash memory boot, which does not need to be impacted by system resets).

The BOR reset is enabled by default. At software level, it can be decided to change the default values after reset.

#### 3.5.1.2 *Power-on-sequence*

The starting sequence of the BlueNRG-2 supply and reset signal is shown below.

**Figure 8. BlueNRG-2 power-up sequence**

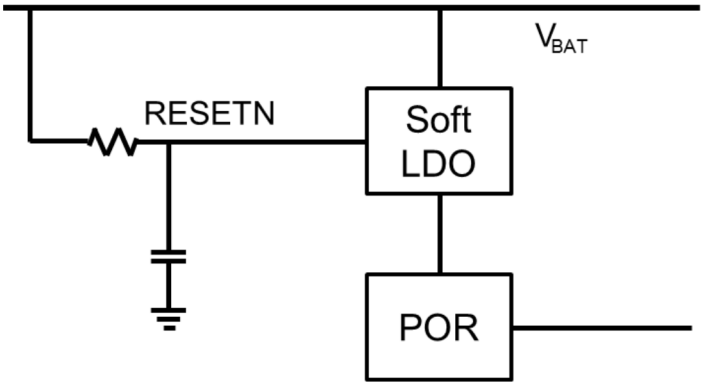



- The VBATx power must only be raised when RESETN pin is low.
- The different VBATx (x=1,2,3) power can be raised separately or together.
- Once the VBATx (x=1, 2, 3) reaches the nominal value, the RESETN pin could be driven high after a 30 us.
- The internal POR is released once internal LDOs are established and RCO clock is ready.
- The system starts on RCO 16 MHz clock system. The software is responsible for configuring the XO 16/32 MHz when necessary.

*Note: The minimum negative pulse to reset the system must be at least 30 µs.*

The POR circuit is powered by a 1.2 V regulator, which must also be powered up with the correct startup sequence. Before VBAT has reached the nominal value, RESETN line must be kept low. An external RC circuit on RESETN pin adds a delay that can prevent RESETN signal from going high before VBAT has reached the nominal value.

Figure 9. **Reset circuit**




If the above conditions are not satisfied, ST cannot guarantee the correct operation of the device.

#### 3.5.1.3 *Watchdog reset*

The BlueNRG-2 contains a watchdog timer, which may be used to recover from software crashes. The watchdog contains a 32-bit down counter, which generates an interrupt, if the interrupt is not serviced, the watchdog generates a reset. The watchdog reset resets the Flash controller, the Cortex-M0 and all its peripherals but it does not reset the debug circuitry of the Cortex-M0.

#### 3.5.1.4 *System reset request*

The system reset request is generated by the debug circuitry of the Cortex-M0. The debugger writes to the SYSRESETREQ bit of the “application interrupt and reset control register” (AIRCR). This system reset request through AIRCR register can also be done by embedded software. The system reset request does not affect the debugger, thus allowing the debugger to remain connected during the reset sequence.

#### 3.5.1.5 *LOCKUP reset*

The Cortex-M0 generates an output LOCKUP that indicates that the core is in a deliberate lock-up state resulting from an unrecoverable exception. The LOCKUP signal is used to generate a reset in the BlueNRG-2, which affects the Cortex-M0, the Flash controller and all the peripherals.

The LOCKUP signal does not reset the Cortex-M0 debug circuitry and it is not generated if a debugger is connected.

### 3.5.2 Reset and wake-up reason decoding

The BlueNRG-2 provides a set of registers to identify the reason behind a reset and wake-up generation. Two registers are used: CKGEN_SOC->REASON_RST and CKGEN_BLE->REASON_RST. The possible reasons are listed below:

1. If the register CKGEN_SOC->REASON_RST = 0, according to the CKGEN_BLE->REASON_RST the possible reasons are:
   a. Wake-up from IO9, IO10, IO11, IO12, IO13
   b. Wake-up from internal timer: BLE timer 1 or BLE timer 2
   c. POR or BOR
2. If the register CKGEN_SOC->REASON_RST is not 0, according to its value the possible reasons are:

a. System reset
b. Watchdog reset
c. Lockup reset

### 3.5.3 Clock and reset registers

CKGEN_SOC peripheral base address (CKGEN_SOC_BASE_ADDR) 0x40900000.

**Table 13. CKGEN_SOC registers**

| Address offset | Name | RW | Reset | Description |
|---|---|---|---|---|
| 0x08 | REASON_RST | R | 0x00000000 | Indicates the reset reason from Cortex-M0. Refer to the detailed description below. |
| 0x1C | DIE_ID(1) | R | 0x00000100 | Identification information of the device. Refer to the detailed description below. |
| 0x20 | CLOCK_EN | RW | 0x0003FFFF | Enable or gates the APB clock of the peripherals. Refer to the detailed description below. |
| 0x24 | DMA_CONFIG | RW | 0x00000000 | DMA config. Refer to the detailed description below. |
| 0x28 | JTAG_IDCODE | R | 0x0200A041 | BlueNRG-2 JTAG IDCODE. |

1. *It depends on the cut version.*

**Table 14. CKGEN_SOC - REASON_RST register description: address offset CKGEN_SOC_BASE_ADDR+0x08**

| Bit | Field name | Reset | RW | Description |
|---|---|---|---|---|
| 0 | RESERVED | 0x0 | R | RESERVED. |
| 1 | SYSREQ | 0x0 | R | Reset caused by Cortex-M0 debug asserting SYSRESETREQ. |
| 2 | WDG | 0x0 | R | Reset caused by assertion of watchdog reset. |
| 3 | LOCKUP | 0x0 | R | Reset caused by Cortex-M0 asserting LOCKUP signal. |
| 31:4 | RESERVED | 0x0 | R | RESERVED. |

**Table 15. CKGEN_SOC - DIE_ID register description: address offset CKGEN_SOC_BASE_ADDR+0x1C**

| Bit | Field name | Reset | RW | Description |
|---|---|---|---|---|
| 3:0 | REV | 0x0 | R | Cut revision. |
| 7:4 | VERSION | 0x0 | R | Cut version. |
| 11:8 | PRODUCT | 0x1 | R | Product. |
| 31:12 | RESERVED | 0x0 | R | RESERVED. |

**Table 16. CKGEN_SOC - CLOCK_EN register description: address offset CKGEN_SOC_BASE_ADDR+0x20**

| Bit | Field name | Reset | RW | Description |
|---|---|---|---|---|
| 0 | GPIO | 0x1 | RW | GPIO clock |
| 1 | NVM | 0x1 | RW | Flash controller clock |
| 2 | SYSCTRL | 0x1 | RW | System controller clock |
| 3 | UART | 0x1 | RW | UART clock |
| 4 | SPI | 0x1 | RW | SPI clock |

| Bit | Field name | Reset | RW | Description |
|---|---|---|---|---|
| 6:5 | RESERVED | 0x3 | RW | RESERVED |
| 7 | WDOG | 0x1 | RW | Watchdog clock |
| 8 | ADC | 0x1 | RW | ADC clock |
| 9 | I2C1 | 0x1 | RW | I2C1 clock |
| 10 | I2C2 | 0x1 | RW | I2C2 clock |
| 11 | MFT1 | 0x1 | RW | MFT1 clock |
| 12 | MFT2 | 0x1 | RW | MFT2 clock |
| 13 | RTC | 0x1 | RW | RTC clock |
| 15:14 | RESERVED | 0x3 | RW | RESERVED |
| 16 | DMA | 0x1 | RW | DMA AHB clock |
| 17 | RNG | 0x1 | RW | RNG AHB clock |
| 18 | PKA | 0x0 | RW | PKA AHB clock |
| 19 | PKA RAM | 0x0 | RW | PKA RAM clock |
| 31:20 | RESERVED | 0x0 | RW | RESERVED |

**Table 17. CKGEN_SOC - DMA_CONFIG register description: address offset CKGEN_SOC_BASE_ADDR+0x24**

| Bit | Field name | Reset | RW | Description |
|---|---|---|---|---|
| 0 | ADC_CH0 | 0x0 | RW | Select ADC on DMA channel 0 instead of peripheral. |
| 1 | ADC_CH1 | 0x0 | RW | Select ADC on DMA channel 1 instead of peripheral. |
| 2 | ADC_CH2 | 0x0 | RW | Select ADC on DMA channel 2 instead of peripheral. |
| 3 | ADC_CH3 | 0x0 | RW | Select ADC on DMA channel 3 instead of peripheral |
| 4 | ADC_CH4 | 0x0 | RW | Select ADC on DMA channel 4 instead of peripheral. |
| 5 | ADC_CH5 | 0x0 | RW | Select ADC on DMA channel 5 instead of peripheral. |
| 6 | ADC_CH6 | 0x0 | RW | Select ADC on DMA channel 6 instead of peripheral. |
| 7 | ADC_CH7 | 0x0 | RW | Select ADC on DMA channel 7 instead of peripheral. |
| 31:8 | RESERVED | 0x0 | RW | RESERVED |

*Note: Only one DMA channel for the ADC should be selected at time. Hardware does not prevent selecting more than one DMA channel for ADC.*

**Table 18. CKGEN_SOC - JTAG_IDCODE register description: address offset CKGEN_SOC_BASE_ADDR+0x28**

| Bit | Field name | Reset | RW | Description |
|---|---|---|---|---|
| 0 | RESERVED | 0x1 | R | RESERVED |
| 11:1 | MANUF_ID | 0x020 | R | Manufacturer ID |
| 27:12 | PART_NUMBER | 0x200A | R | Part number |
| 31:28 | VERSION_NUM | 0x0 | R | Version |

CKGEN_BLE peripheral base address (CKGEN_BLE_BASE_ADDR) 0x48100000.

Table 19. **CKGEN_BLE registers**

| Address offset | Name | RW | Reset | Description |
|---|---|---|---|---|
| 0x08 | REASON_RST | R | 0x00000005 | Indicates the Reset reason from BLE. Refer to the detailed description below. |
| 0x0C | CLK32K_COUNT | RW | 0x0000000F | Counter of 32 kHz clock. Refer to the detailed description below. |
| 0x10 | CLK32K_PERIOD | R | 0x00000000 | Period of 32 kHz clock. Refer to the detailed description below. |
| 0x14 | CLK32K_FREQ | R | 0x00000000 | Measurement of frequency of 32 kHz clock. Refer to the detailed description below. |
| 0x18 | CLK32K_IT | RW | 0x00000000 | Interrupt event for 32 kHz clock measurement. Refer to the detailed description below. |

Table 20. **CKGEN_BLE - REASON_RST register description: address offset CKGEN_BLE_BASE_ADDR+0x08**

| Bit | Field name | Reset | RW | Description |
|---|---|---|---|---|
| 0 | RESERVED | 0x1 | R | RESERVED |
| 1 | BOR | 0x0 | R | Reset from BOR |
| 2 | POR | 0x1 | R | Reset from POR |
| 3 | WKP_IO9 | 0x0 | R | Wake-up from external IO9 |
| 4 | WKP_IO10 | 0x0 | R | Wake-up from external IO10 |
| 5 | WKP_IO11 | 0x0 | R | Wake-up from external IO11 |
| 6 | WKP_IO12 | 0x0 | R | Wake-up from external IO12 |
| 7 | WKP_IO13 | 0x0 | R | Wake-up from external IO13 |
| 8 | WKP_BLUE | 0x0 | R | Wake-up comes from the timer 1 expiration in the wake-up control block of the BLE radio |
| 10 | WKP2_BLUE | 0x0 | R | Wake-up comes from the timer 2 expiration in the wake-up control block of the BLE radio |
| 31:11 | RESERVED | 0x0 | R | RESERVED |

Table 21. **CKGEN_BLE - CLK32K_COUNT register description: address offset CKGEN_BLE_BASE_ADDR+0x0C**

| Bit | Field name | Reset | RW | Description |
|---|---|---|---|---|
| 8:0 | SLOW_COUNT | 0xF | RW | Program the window length (in slow clock period unit) for slow clock measurement |
| 31:9 | RESERVED | 0x0 | RW | RESERVED |

Table 22. **CKGEN_BLE - CLK32K_PERIOD register description: address offset CKGEN_BLE_BASE_ADDR+0x10**

| Bit | Field name | Reset | RW | Description |
|---|---|---|---|---|
| 18:0 | SLOW_PERIOD | 0x0 | R | Indicates slow clock period information. The result provided in this field corresponds to the length of SLOW_COUNT periods of the slow clock (32 kHz) measured in 16 MHz half-period unit. The measurement is done automatically each time the device enters in active2 mode using SLOW_COUNT = 16. A new calculation can be launched by writing zero in CLK32K_PERIOD register. In this case, the time window uses the value programmed in SLOW_COUNT field. |

| Bit | Field name | Reset | RW | Description |
|---|---|---|---|---|
| 31:19 | RESERVED | 0x0 | R | RESERVED |

**Table 23. CKGEN_BLE - CLK32K_FREQ register description: address offset CKGEN_BLE_BASE_ADDR+0x14**

| Bit | Field name | Reset | RW | Description |
|---|---|---|---|---|
| 26:0 | SLOW_FREQ | 0x0 | R | Value equal to $2^{39}$ / SLOW_PERIOD |
| 31:27 | RESERVED | 0x0 | R | RESERVED |

**Table 24. CKGEN_BLE - CLK32K_IT register description: address offset CKGEN_BLE_BASE_ADDR+0x18**

| Bit | Field name | Reset | RW | Description |
|---|---|---|---|---|
| 0 | CLK32K_MEAS_IRQ | 0x0 | RW | When read, provides the status of the interrupt indicating slow clock measurement is finished:<br>0: No pending interrupt.<br>1: Pending interrupt.<br>When written, clears the interrupt:<br>0: No effect.<br>1: Clear the interrupt. |
| 31:1 | RESERVED | 0x0 | RW | RESERVED |

*Note: All RESERVED fields inside registers must always be written with their default value.*

## 3.6 ADC

### 3.6.1 Introduction

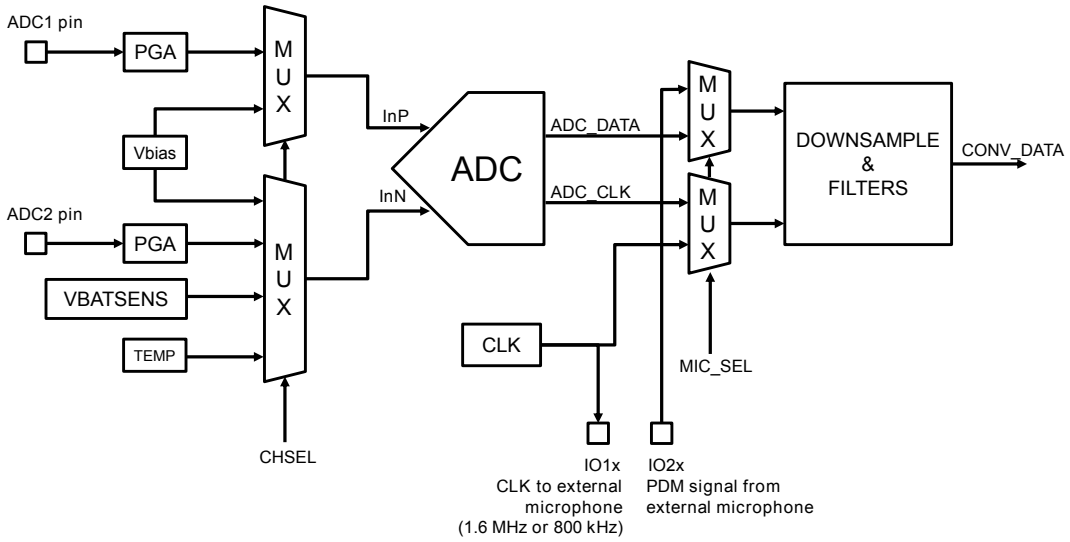
The BlueNRG-2 integrates a 10-bit analog-to-digital converter (ADC) for sampling an external signal.
Main features are:

- Sampling frequency 1 MHz
- One channel in single ended or differential input through the pins ADC1 and ADC2
- Temperature and battery voltage sensors
- The conversion are either continuous or single step acquisition
- An integrated digital filter is used to process a PDM data stream from a MEMS microphone

### 3.6.2 Functional overview

The figure below shows a top diagram of the ADC.

Figure 10. **ADC block diagram**




Several channels are available for the conversion, the CHSEL selects the channel according to Table 25. ADC channels.

Table 25. **ADC channels**

| CHSEL | Channels description |
|---|---|
| 0 | All switch open. No input |
| 1 | Single ended through ADC2 pin. InP = Vbias (internal), InN = ADC2 pin |
| 2 | Single ended through ADC1 pin. InP = ADC1 pin, InN = Vbias (internal) |
| 3 | Differential ADC1 pin – ADC2 pin. InP = ADC1 pin, InN = ADC2 pin |
| 4 | Temperature sensor. InN=TEMP, InP = 0.6 V (internal) |
| 5 | Battery voltage sensor. InN = VBATSENS, InP = 0.6 V (internal) |
| 6 | Short. InP = InN = 0.6 V (internal) |

The conversion can be single (CONT = 0) or continuous (CONT = 1). In continuous mode, the conversion runs with a pre-programmed sampling rate, the user must discard the first samples that are not valid because generated during the establishment of the internal filter. In particular, it must discards a number of samples as follows:

- 10 samples if the bitfield SKIP is 0 (COMP filter not bypassed)
- 3 samples if the bitfield SKIP is 1 (COMP filter bypassed)

In single step mode the ADC performs a conversion and then stops.

The output data rate depends on the setting of OSR according to the following table.

Table 26. **ADC data rate**

| OSR | Output data rate [Ksample/s] |
|---|---|
| 0 (200) | 5 |
| 1 (100) | 10 |
| 2 (64) | 15.625 |

| OSR | Output data rate [Ksample/s] |
|---|---|
| 3 (32) | 31.25 |

The setting of the oversampling ratio (OSR) must be done according to the frequency of the input signal (AC), while for DC signals, the best performance is with OSR = 200. In order to achieve the best performance within the selected input voltage range, the attenuation value must be set through the corresponding register PGASEL value, as in the following table.

**Table 27. ADC parameter settings**

| Vin range [V] | Vbias [V] | Attenuation [dB] | REFSEL value | PGASEL value |
|---|---|---|---|---|
| [0, 1.2] | 0.6 | 0 | 2 | 0 |
| [0, 2.4] | 0.6 | 6.02 | 2 | 1 |
| [0, 3.6] | 0.6 | 9.54 | 2 | 2 |

#### 3.6.2.1 *ADC microphone mode*

The system can work in conjunction with an external MEMS microphone. In this mode the user must configure:

- an IO as PDM_CLK (GPIO Serial2 mode) in order to provide the clock signal to an external MEMS microphone (output signal)
- an IO as PDM_DATA (GPIO Serial2 mode) in order to receive and process the PDM data stream from the external MEMS microphone (input signal). See Table 129. IO functional map for more details about how these pins can be used for this mode.
- set the MIC_SEL bitfield of the CONF register, in order to provide a clock to the MEMS microphone. The PDM_CLK signal provides a clock that can be 1.6 MHz (DIG_FILT_CLK = 0) or 0.8 MHz (DIG_FILT_CLK =1)
- set the MIC_ON bitfield of the CTRL register, in order to make the ADC start the conversion from the MEMS microphone

*Note: MIC_ON and ON bitfields must be exclusive and must not be set together.*

The output data rate changes with the OSR and according to the clock frequency provided as explained in Table 28. Output data rate with microphone.

**Table 28. Output data rate with microphone**

| DIG_FILT_CLK | OSR | Output data rate [Ksample/s] |
|---|---|---|
| 1 (clock = 0.8 MHz) | 0 (200) | 4 |
| | 1 (100) | 8 |
| | 2 (64) | 12.5 |
| | 3 (32) | 25 |
| 0 (clock = 1.6 MHz) | 0 (200) | 8 |
| | 1 (100) | 16 |
| | 2 (64) | 25 |
| | 3 (32) | 50 |

#### 3.6.2.2 *ADC start conversion*

The ADC both analog and digital sub-system are switched on by setting ADCON and SWSTART.

The conversion operation consists of four phases.

1. The wake-up phase lasts 6 µs, is present at the beginning of a single acquisition, with the goal to let the analog system to settle before to start the acquisition.

2. When CALEN bit and AUTO_OFFSET are set in ADC_CTRL register, a calibration starts. It permits to compensate the offset in the analog part. The conversion status is tracked by SR status register. At the beginning of the conversion the BUSY bit is set and masks any attempt to change CONF, up to the end of the conversion. At end of this conversion, the ENDCAL flag is generated and the OFFSET register is written with the converted offset voltage.
3. The acquisition phase is regulated by a timeout depending on the resolution. In this phase, digital filter chain processes the data coming from ADC.
4. The elaboration phase is at the end of the timeout, the data obtained at the output of the digital filter is written in the DATA register. The content of the OFFSET register is automatically used to compensate the final result. Furthermore, the ADCEOC flag is generated to warn about the end of conversion. If ENAB_COMP bit is set, the WDOG flag is generated to warn that the result of the conversion is between a high THRESHOLD_HI and low threshold THRESHOLD_LO.

*Note: It is always advisable to set the register fields CALEN and AUTO_OFFSET in order to perform automatic calibration for each measurement.*

#### 3.6.2.3 *ADC offset*

The ADC automatically corrects a potential offset error by taking into account the content of the register OFFSET.

To enable the automatic offset correction the CALEN and the AUTO_OFFSET must be both set. The result of the calibration is stored in the OFFSET register.

The correction of the offset can be also done manually, for example by performing firstly an automatic offset calibration by making an ADC conversion with both AUTO_OFFSET and CALEN bitfields set. In this way, the OFFSET register is updated with the current offset error. Then, the automatic offset calibration can be disabled by set to 0b the AUTO_OFFSET and the CALEN bitfields. And so, the offset value is applied to all the next ADC conversions.

The calibration value is a 16-bit value in the register OFFSET. It must be placed in the bitfield OFFSET_MSB if the bitfield SKIP is 0 (filter not bypassed). While, if the bitfield SKIP is 1 (filter bypassed), the calibration value must be placed in the bitfield OFFSET_LSB.

#### 3.6.2.4 *ADC conversion*

The relationship between differential input voltage and $ADC_{RAW}$ code (first 16-bit MSB of DATA_CONV register) depends on a limited set of parameters: the digital reference voltage VREF, the PGA value, and a scaling factor.

**Differential mode**

This mode enables the ADC differential conversion from the pins ADC1 and ADC2.

$$V_{ADC12(Volt)} = V_{ADC1} - V_{ADC2} = (1 + PGASEL)*\left(\frac{ADC_{RAW}}{FS\_16(OSR)}\right)*VREF \quad (1)$$

**Single-ended mode**

This mode enables the ADC conversion from the pin ADC1 or from the pin ADC2.

$$V_{ADC1(Volt)} = (1 + PGASEL)*\left(Vbias + \left(\frac{ADC_{RAW}}{FS\_16(OSR)}\right)* VREF \right) \quad (2)$$

$$V_{ADC2(Volt)} = (1 + PGASEL)*\left(Vbias - \left(\frac{ADC_{RAW}}{FS\_16(OSR)}\right)* VREF \right) \quad (3)$$

**Battery voltage sensor**

This mode enables the monitoring of the battery voltage $V_{BATT}$, through an internal resistive bridge.

$$V_{BATT(Volt)} = K_{BATT}*\left(Vbias - \left(\frac{ADC_{RAW}}{FS\_16(OSR)}\right)* VREF \right) \quad (4)$$

**Temperature sensor**

This mode enables the monitoring of the temperature by means of an internal sensor, with the following voltage to temperature conversion:

$$V_{TEMP(°C)} = K_{TC}*\left(Vbias - \left(\frac{ADC_{RAW}}{FS\_16(OSR)}\right)* VREF \right) + OFFSET_{TC} \quad (5)$$

To ensure an accurate temperature reading, average the value over several readings.

Below the values for the symbols used in the ADC conversion formulas:

- PGASEL is the input attenuation register, values: 0, 1, or 2

- FS_16(OSR) is the full scale factor for $ADC_{RAW}$ and it depends on the oversampling ratio (OSR) as shown below:
- If SKIP is 0 (filter not bypassed), then:
  - $ADC_{RAW}$ is DATA_CONV_MSB
  - FS_16(32) = FS_16(64) = 35442
  - FS_16(100) = FS_16(200) = 41260
- If SKIP is 1 (filter bypassed), then:
  - $ADC_{RAW}$ is DATA_CONV_LSB
  - FS_16(32) = FS_16(64) = 32768
  - FS_16(100) = FS_16(200) = 38147
- $V_{REF}$ represents the digital core power supply, typical $V_{REF}$ value is 2.4 V
- $V_{bias}$ is given by the register REFSEL, with a typical value of 0.6 V
- $K_{BATT}$ is 4.36
- $K_{TC}$ is 401
- $OFFSET_{TC}$ is 267 °C

### 3.6.3 ADC registers

ADC peripheral base address (ADC_BASE_ADDR) 0x40800000.

**Table 29. ADC registers**

| Address offset | Name | RW | Reset | Description |
|---|---|---|---|---|
| 0x00 | CTRL | RW | 0x00000000 | ADC control register. Refer to the detailed description below. |
| 0x04 | CONF | RW | 0x0000000C | ADC configuration register. Refer to the detailed description below. |
| 0x08 | IRQSTAT | R | 0x00000000 | IRQ masked status register. Refer to the detailed description below. |
| 0x0C | IRQMASK | RW | 0x0000000F | It sets the mask for ADC interrupt. Refer to the detailed description below. |
| 0x10 | IRQRAW | R | 0x00000000 | IRQ status register. Refer to the detailed description below. |
| 0x14 | DATA_CONV | R | 0x00000000 | Result of the conversion in two complement format. |
| 0x18 | OFFSET | RW | 0x00000000 | Offset for correction of converted data |
| 0x20 | SR_REG | RW | 0x00000000 | ADC status register. Refer to the detailed description below. |
| 0x24 | THRESHOLD_HI | RW | 0xFFFFFFFF | High threshold for window comparator. |
| 0x28 | THRESHOLD_LO | RW | 0x00000000 | Low threshold for window comparator. |

**Table 30. ADC - CTRL register description: address offset ADC_BASE_ADDR+0x00**

| Bit | Field name | Reset | RW | Description |
|---|---|---|---|---|
| 0 | ON | 0x0 | RW | Starts ADC analog subsystem. This bit must be set before starting a conversion.<br>0: ADC is OFF.<br>1: ADC is ON.<br>This bit works for all the mode except the microphone mode. |