

# nRF52 Power Management

## nRF52 Power Management

### Overview

The nRF52 Power Management module provides battery protection features to prevent over-discharge, minimise likelihood of brownout and flash corruption conditions existing, and enable safe voltage-based recovery.

### Features

#### Boot Voltage Protection

- Checks battery voltage immediately after boot and before mesh operations commence
- If voltage is below a configurable threshold (e.g., 3300mV), the device configures voltage wake (LPCOMP + VBUS) and enters protective shutdown (SYSTEMOFF)
- Prevents boot loops when battery is critically low
- Skipped when external power (USB VBUS) is detected

#### Voltage Wake (LPCOMP + VBUS)

- Configures the nRF52's Low Power Comparator (LPCOMP) before entering SYSTEMOFF
- Enables USB VBUS detection so external power can wake the device
- Device automatically wakes when battery voltage rises above recovery threshold or when VBUS is detected

#### Early Boot Register Capture

- Captures RESETREAS (reset reason) and GPREGRET2 (shutdown reason) before SystemInit() clears them
- Allows firmware to determine why it booted (cold boot, watchdog, LPCOMP wake, etc.)
- Allows firmware to determine why it last shut down (user request, low voltage, boot protection)

#### Shutdown Reason Tracking

Shutdown reason codes (stored in GPREGRET2):

Code	Name	Description
0x00	NONE	Normal boot / no previous shutdown
0x4C	LOW_VOLTAGE	Runtime low voltage threshold reached
0x55	USER	User requested powerOff()
0x42	BOOT_PROTECT	Boot voltage protection triggered

## Supported Boards

Board	Implemented	LPCOMP wake	VBUS wake
Seeed Studio XIAO nRF52840 ( <code>xiao_nrf52</code> )	Yes	Yes	Yes
RAK4631 ( <code>rak4631</code> )	Yes	Yes	Yes
Heltec T114 ( <code>heltec_t114</code> )	Yes	Yes	Yes
GAT562 Mesh Watch13	Yes	Yes	Yes
Promicro nRF52840	No	No	No
RAK WisMesh Tag	No	No	No
Heltec Mesh Solar	No	No	No
LilyGo T-Echo / T-Echo Lite	No	No	No
SenseCAP Solar	Yes	Yes	Yes
WIO Tracker L1 / L1 E-Ink	No	No	No
WIO WM1110	No	No	No
Mesh Pocket	No	No	No
Nano G2 Ultra	No	No	No
ThinkNode M1/M3/M6	No	No	No
T1000-E	No	No	No
Ikoka Nano/Stick/Handheld (nRF)	No	No	No
Keepteen LT1	No	No	No
Minewsemi ME25LS01	No	No	No

### Notes:

- "Implemented" reflects Phase 1 (boot lockout + shutdown reason capture).
- User power-off on Heltec T114 does not enable LPCOMP wake.

- VBUS detection is used to skip boot lockout on external power, and VBUS wake is configured alongside LPCOMP when supported hardware exposes VBUS to the nRF52.

# Technical Details

## Architecture

The power management functionality is integrated into the `NRF52Board` base class in `src/helpers/NRF52Board.cpp`. Board variants provide hardware-specific configuration via a `PowerMgtConfig` struct and override `initiateShutdown(uint8_t reason)` to perform board-specific power-down work and conditionally enable voltage wake (LPCOMP + VBUS).

## Early Boot Capture

A static constructor with priority 101 in `NRF52Board.cpp` captures the RESETRAS and GPREGRET2 registers before:

- `SystemInit()` (priority 102) - which clears RESETRAS
- Static C++ constructors (default priority 65535)

This ensures we capture the true reset reason before any initialisation code runs.

## Board Implementation

To enable power management on a board variant:

### 1. Enable in `platformio.ini`:

```
```ini
-D NRF52_POWER_MANAGEMENT
```
```

### 1. Define configuration in `variant.h`:

```
```c
#define PWRMGT_VOLTAGE_BOOTLOCK 3300 // Won't boot below this voltage (mV)

#define PWRMGT_LPCOMP_AIN 7 // AIN channel for voltage sensing

#define PWRMGT_LPCOMP_REFSEL 2 // REFSEL (0-6=1/8..7/8, 7=ARef, 8-15=1/16..15/16)
```
```

### 1. Implement in board `.cpp` file:

```
```cpp

#ifdef NRF52_POWER_MANAGEMENT

const PowerMgtConfig power_config = {

.lpcomp_ain_channel = PWRMGT_LPCOMP_AIN,

.lpcomp_refsel = PWRMGT_LPCOMP_REFSEL,

.voltage_bootlock = PWRMGT_VOLTAGE_BOOTLOCK

};

void MyBoard::initiateShutdown(uint8_t reason) {

// Board-specific shutdown preparation (e.g., disable peripherals)

bool enable_lpcomp = (reason == SHUTDOWN_REASON_LOW_VOLTAGE ||

reason == SHUTDOWN_REASON_BOOT_PROTECT);

if (enable_lpcomp) {

configureVoltageWake(power_config.lpcomp_ain_channel, power_config.lpcomp_refsel);

}

enterSystemOff(reason);

}

#endif

void MyBoard::begin() {

NRF52Board::begin(); // or NRF52BoardDCDC::begin()

// ... board setup ...

#ifdef NRF52_POWER_MANAGEMENT

checkBootVoltage(&power_config);

#endif

}

}
```

```
...
```

For user-initiated shutdowns, `power0ff()` remains board-specific. Power management only arms LPCOMP for automated shutdown reasons (boot protection/low voltage).

### 1. Declare override in board .h file:

```
```cpp
#ifdef NRF52_POWER_MANAGEMENT

void initiateShutdown(uint8_t reason) override;

#endif
...
```
```

## Voltage Wake Configuration

The LPCOMP (Low Power Comparator) is configured to:

- Monitor the specified AIN channel (0-7 corresponding to P0.02-P0.05, P0.28-P0.31)
- Compare against VDD fraction reference (REFSEL: 0-6=1/8..7/8, 7=ARef, 8-15=1/16..15/16)
- Detect UP events (voltage rising above threshold)
- Use 50mV hysteresis for noise immunity
- Wake the device from SYSTEMOFF when triggered

VBUS wake is enabled via the POWER peripheral USBDETECTED event whenever `configureVoltageWake()` is used. This requires USB VBUS to be routed to the nRF52 (typical on nRF52840 boards with native USB).

### LPCOMP Reference Selection (PWRMGT\_LPCOMP\_REFSEL):

| REFSEL | Fraction | VBAT @ 1M/1M divider (VDD=3.0-3.3) | VBAT @ 1.5M/1M divider (VDD=3.0-3.3) |
|--------|----------|------------------------------------|--------------------------------------|
| 0      | 1/8      | 0.75-0.82 V                        | 0.94-1.03 V                          |
| 1      | 2/8      | 1.50-1.65 V                        | 1.88-2.06 V                          |
| 2      | 3/8      | 2.25-2.47 V                        | 2.81-3.09 V                          |
| 3      | 4/8      | 3.00-3.30 V                        | 3.75-4.12 V                          |
| 4      | 5/8      | 3.75-4.12 V                        | 4.69-5.16 V                          |
| 5      | 6/8      | 4.50-4.95 V                        | 5.62-6.19 V                          |
| 6      | 7/8      | 5.25-5.77 V                        | 6.56-7.22 V                          |
| 7      | ARef     | -                                  | -                                    |

| REFSEL | Fraction | VBAT @ 1M/1M divider<br>(VDD=3.0-3.3) | VBAT @ 1.5M/1M divider<br>(VDD=3.0-3.3) |
|--------|----------|---------------------------------------|-----------------------------------------|
| 8      | 1/16     | 0.38-0.41 V                           | 0.47-0.52 V                             |
| 9      | 3/16     | 1.12-1.24 V                           | 1.41-1.55 V                             |
| 10     | 5/16     | 1.88-2.06 V                           | 2.34-2.58 V                             |
| 11     | 7/16     | 2.62-2.89 V                           | 3.28-3.61 V                             |
| 12     | 9/16     | 3.38-3.71 V                           | 4.22-4.64 V                             |
| 13     | 11/16    | 4.12-4.54 V                           | 5.16-5.67 V                             |
| 14     | 13/16    | 4.88-5.36 V                           | 6.09-6.70 V                             |
| 15     | 15/16    | 5.62-6.19 V                           | 7.03-7.73 V                             |

**Important:** For boards with a voltage divider on the battery sense pin, LPCOMP measures the divided voltage. Use:

$VBAT\_threshold \approx (VDD \text{ fraction}) \text{ divider\_scale}$ , where  $divider\_scale = (R_{top} + R_{bottom}) / R_{bottom}$  (e.g., 2.0 for 1M/1M, 2.5 for 1.5M/1M, 3.0 for XIAO).

## SoftDevice Compatibility

The power management code checks whether SoftDevice is enabled and uses the appropriate API:

- When SD enabled: `sd_power_*` functions
- When SD disabled: Direct register access (`NRF_POWER->*`)

This ensures compatibility regardless of BLE stack state.

## CLI Commands

Power management status can be queried via the CLI:

| Command                             | Description                                                           |
|-------------------------------------|-----------------------------------------------------------------------|
| <code>get pwrmgmt.support</code>    | Returns "supported" or "unsupported"                                  |
| <code>get pwrmgmt.source</code>     | Returns current power source - "battery" or "external" (5V/USB power) |
| <code>get pwrmgmt.bootreason</code> | Returns reset and shutdown reason strings                             |
| <code>get pwrmgmt.bootmv</code>     | Returns boot voltage in millivolts                                    |

On boards without power management enabled, all commands except `get pwrmgmt.support` return:

```
ERROR: Power management not supported
```

# Debug Output

When `MESH_DEBUG=1` is enabled, the power management module outputs:

```
DEBUG: PWRMGT: Reset = Wake from LPCOMP (0x20000); Shutdown = Low Voltage (0x4C)
DEBUG: PWRMGT: Boot voltage = 3450 mV (threshold = 3300 mV)
DEBUG: PWRMGT: LPCOMP wake configured (AIN7, ref=3/8 VDD)
```

## Phase 2 (Planned)

- Runtime voltage monitoring
- Voltage state machine (Normal -> Warning -> Critical -> Shutdown)
- Configurable thresholds
- Load shedding callbacks for power reduction
- Deep sleep integration
- Scheduled wake-up
- Extended sleep with periodic monitoring

## References

- [nRF52840 Product Specification - POWER](#)
- [nRF52840 Product Specification - LPCOMP](#)
- [SoftDevice S140 API - Power Management](#)

Revision #2

Created 2026-05-03 05:05:11 UTC by Mesh America Admin

Updated 2026-05-03 12:58:39 UTC by Mesh America Admin