AN14464

Low Power Checklist for RW61x Family

Rev. 1.0 — 3 September 2025 Application note

Document information

Information	Content
Keywords	AN14464, Power Management, Low Power, FRDM-RW612
Abstract	This document provides an overview on how to use the low power consumption features of the RW61x.

1 Introduction

This document provides an overview on how to use the low power consumption features of the RW61x. The document explains the different power modes available, how to use them, and how to measure power consumption on the FRDM-RW612 board for reference. By understanding and using these modes, developers can significantly extend battery life, reduce thermal output, and enhance the overall efficiency of their designs.

The RW612 SoC features three cores: one dedicated to Wi-Fi (CPU1), another for narrowband communication (CPU2), and the third for the main application (CPU3). This document focuses exclusively on power management for the Application CPU (CPU3). The Wi-Fi and narrowband subsystems are outside the scope of this application note.

The Power Management Unit (PMU) manages the power features. For details and more explanations, see **Section 8 Power Management** in the *RW61x User Manual* (document UM11865).

2 RW61x architecture overview

The NXP RW612 is a highly integrated, low-power, and tri-radio wireless microcontroller unit (MCU) designed to meet the growing demands of connected smart devices across consumer, industrial, and enterprise applications. The RW612 MCU subsystem includes a 260 MHz Arm Cortex-M33 core with TrustZone-M, 1.2 MB on-chip SRAM. Featuring support for Wi-Fi 6, Bluetooth Low Energy (LE) 5.4, and 802.15.4 (for protocols like Thread and Matter), the RW612 delivers robust wireless connectivity with advanced security and energy efficiency.

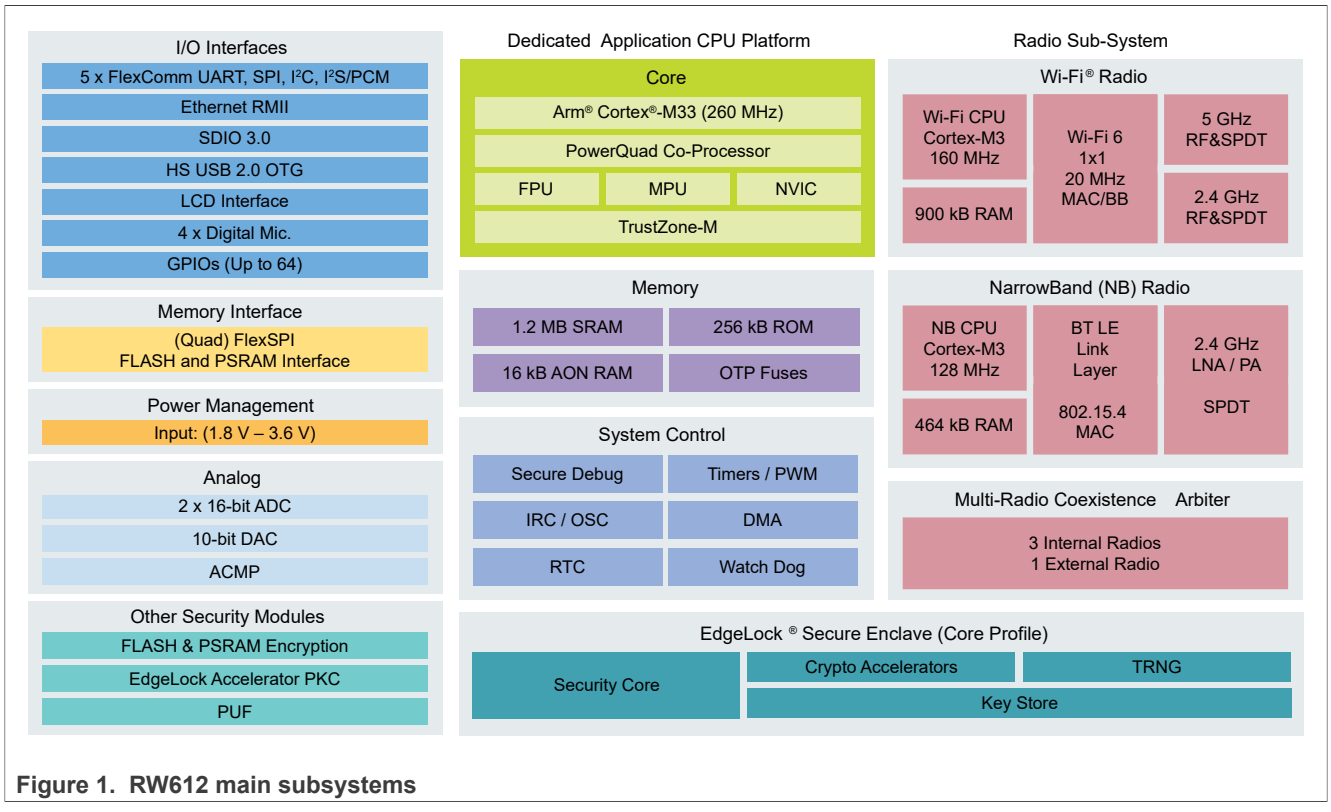


Figure 1. RW612 main subsystems

Power management is a critical aspect of embedded system design, especially in battery-powered or energy-constrained environments. The RW612 addresses this need with a comprehensive set of power modes that allow developers to finely tune energy consumption based on application requirements. These modes range

from full-performance active states to deep sleep and power-down configurations, enabling optimal trade-offs between performance and power savings.

3 RW61x power modes description

The RW61x family supports five distinct power modes, each progressively reducing system activity and power consumption. As the device transitions through these modes, non-essential components are systematically powered down, retaining only the critical circuitry required for wake-up and resume program execution.

- PM0 (active) is the MCU with no power restrictions so it can run at full capacity, perfect for High-performance tasks.
- PM1 (idle) is similar to PM0 with few power restrictions on not needed components, which works for wait-for-interrupt (WFI) scenarios, short idle periods.
- PM2 (standby) gates the CPU clock and puts it to sleep with its state retained, peripherals change to retention state, the AON domain remains active, ideal for Low-power standby with quick wake-up.
- PM3 (sleep) puts the core in to power off mode, still able to retain internal SRAM but losing peripheral configurations, which works well for Long idle periods, low power consumption scenarios.
- PM4 (deep sleep) is similar to PM3 but powers off 1.216 MB SRAM, retaining 8 kB AON SRAM0 with the option to retain or turn off 8 kB AON SRAM1 if needed, good for Battery-critical scenarios, infrequent wake-up events.

Table 1. Power mode transitions

Power Mode	PM0	PM1	PM2	PM3	PM4
Name	Active mode	Idle mode	Standby mode	Sleep mode	Deep sleep mode
Description	Full-speed operation with all clocks and peripherals enabled	Core enters sleep mode and waits for the interrupt instructions. Peripherals and memory remain active	Core is in sleep with state retained. Most clocks off, only essential peripherals and memory retained	The core turns off. All peripherals are power removed except PMU/ RTC/capture pulse, which is active, context and configurations lost. 1.216 MB SRAM + 16 kB AON SRAM0/1 are retained.	Same as PM3, except only AON SRAM0 is retained and AON SRAM1 can be turned off or retained.
CPU state	On	Sleep mode	Sleep mode	Off	Off

Figure 2 shows all power modes transition only from PM0.

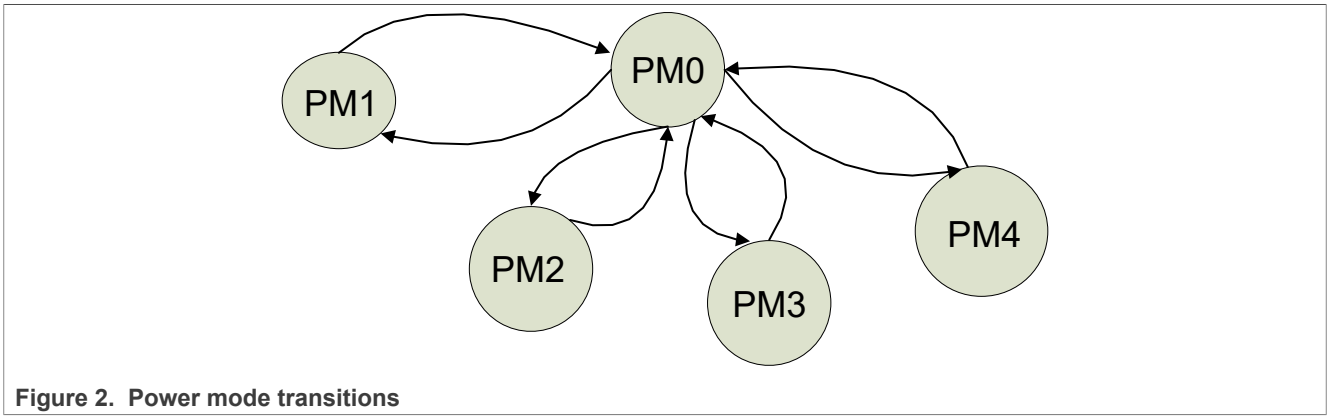


Figure 2. Power mode transitions

For more information, see **Section 8.2 Power mode** in the *RW61x User Manual* (document UM11865).

4 Peripheral behavior in low power modes

Peripheral functionality varies depending on the active power mode. As the system transitions into lower power states, more peripherals are disabled or placed into low-power configurations, reducing overall functionality in favor of energy savings. Consider these changes in the peripheral configurations when exiting from low power modes and returning to a normal execution flow to ensure that the application will continue to work properly after changing power modes.

Table 2. Peripheral behavior in low power modes

Power mode	Peripheral behavior	Wakeup sources	Special considerations
PM0 / PM1	No significant behavioral changes in peripherals.	N/A	None
PM2	• Analog modules enter power-down mode by default (can be kept on if needed). • Peripherals enter the retention state. • AON domain peripherals remain active. • MCU peripherals stay configured.	Most MCU peripherals	MCU peripherals can act as wakeup sources.
PM3	Most peripherals are completely powered off. Peripherals configuration lost.	PMU, RTC, GPIO[24]-[26], BOD1	GPIO pin states can be configured to retain/set values upon entering/exiting PM3.
PM4	Same as PM3	Same as PM3	Power off all the GPIO pads except for six AON IO pads GPIO[22]-[27].

5 Memory operation in low power modes

5.1 Internal SRAM operation in low power modes

In PM0/PM1, there are no changes to memory operations.

For PM2, the internal SRAM is set to the standby state by default with the option of keeping it in the active state if needed by software programming. FlexSPI can remain configured for external memory operations.

For PM3, the 1.216 MB SRAM goes to programmable retention state, which can be set to off if not needed, while 16 kB AON SRAM0/1 are set to retention state. FlexSPI peripheral is now disabled, cutting out any communication with external memories.

For PM4, the 1.216 MB SRAM is now off, 8 kB AON SRAM0 is set to retention mode, and 8 kB AON SRAM1 is programmed to retention state or turned off. FlexSPI peripheral is now disabled, cutting out any communication with external memories.

To turn off unnecessary internal SRAM, you can do so by writing to the `MEM_CFG` register for PM3/4, or `MEM_PD_CFG` and `MEM_PD_CTRL` registers on SYSCTL2 for PM2. Internal SRAM retention/off mode control can only be done for PM3/4. For PM2, the state can be on/standby and for PM0/1, it remains on with no option to turn it off.

Use `MEM_PD_CTRL` to enable the control signals, and with `MEM_PD_CFG`, you can select which SRAM or AON RAM bank, for example:

```
SYSCTL2->MEM_PD_CTRL = 0x1FFFFFFFU //enable all control signals
SYSCTL2->MEM_PD_CFG = 0x1FU; // standby mode for banks 0-5 of 1.2MB SRAM
```

Use bits 0 to 5 of `MEM_CFG` to turn of the different SRAM banks in PM3: 0 keeps retention and 1 shuts down:

Table 3.

Bit0	ram0-5 384 kB
Bit1	ram6 64 kB
Bit2	ram7 64 kB
Bit3	ram8-9 128 kB
Bit4	ram10-13 256 kB
Bit5	ram14-18 320 kB

Use bit 8 of `MEM_CFG` to control AON memory retention in PM4 mode: 0 for retention and 1 for power down.

```
PMU->MEM_CFG = 0x11FU //turn of all SRAM banks in PM3 and 8KB of AON RAM in PM4
```

It is also possible to set these configurations using the SDK APIs. Use a variable type **power_sleep_config_t** to set the needed configurations that are used to set the power mode:

```
power_sleep_config_t sleepConfig;
/* Set your custom config on "sleepConfig" */
sleepConfig.memPdCfg = 0x11FU; //turn of all SRAM banks in PM3 and 8KB of AON
 RAM in PM4
sleepConfig.clkGate = kPOWER_ClkGateAll;
/* end of custom configurations */
LPM_SetPowerMode(pm_number, &sleepConfig);
```

SDK APIs may change names over different SDK versions.

For more information on these registers, see sections **4.1.23 MEM PD Control enable register when PM2 mode (MEM_PD_CTRL)**, **4.1.24 MEM PD Configure register when PM2 mode (MEM_PD_CFG)**, and **19.1.23 mem configuration register (MEM_CFG) for PM3/4** in the *RW61x Registers reference manual* (document RM00278)

5.2 External NOR flash operation in low power modes

For the external Winbond flash memory in the FRDM-RW612, to send it to the sleep mode, add the `Power-down(0xB9)` command to the LUT, and call it before the RW612 goes to low power mode. This Power-down command is executed from RAM when it is sent. The memory shuts down and stops sending any data back. In code, it looks like below:

```
__RAMFUNC(RAM) status_t LPM_SendFlashToSleep(void)
{
    flexspi_transfer_t flashXfer;
    status_t status;

    uint32_t tempLUT[8] = {
        /* Power Down */
        [0] = FLEXSPI_LUT_SEQ(kFLEXSPI_Command_SDR, kFLEXSPI_1PAD,
 0xB9, kFLEXSPI_Command_STOP, kFLEXSPI_1PAD, 00),
        /* Release power down */
        [4] = FLEXSPI_LUT_SEQ(kFLEXSPI_Command_SDR, kFLEXSPI_1PAD,
 0xAB, kFLEXSPI_Command_STOP, kFLEXSPI_1PAD, 00),
    };
    /* Update bottom LUT table (44-59). */
    FLEXSPI_UpdateLUT(FLEXSPI, 4*12, tempLUT, ARRAY_SIZE(tempLUT));

    /* Write data */
```

```
    flashXfer.deviceAddress = 0U;
    flashXfer.port          = kFLEXSPI_PortA1;
    flashXfer.cmdType       = kFLEXSPI_Command;
    flashXfer.SeqNumber     = 1;
    flashXfer.seqIndex      = 12;

    status = FLEXSPI_TransferBlocking(FLEXSPI, &flashXfer);
    return status;
}
```

This power down mode is volatile and is cleared when the memory goes through a power-on reset. To return the memory from its sleep state into an active state without a power cycle, a fuse must be configured. You can do this with BL Host with the RW612 in the ISP mode:

```
blhost -p COMx get-property 1
blhost -p COMx efuse-program-once 16 0x3531C000
```

16 is the fuse and `0x3531C000` is the value that must be set to activate the wakeup command.

For more information, see **section 11.3.1.1 FlexSPI NOR Flash boot** in the *RW61x User Manual* (document UM11865)

Important: *If the fuse is not configured once the memory is sent to its power-down state, it will not wake up until a power cycle is done. The fuse configuration allows the RW612 to send the `Release Power-down(0xAB)` command to the memory from the bootloader, which is needed to return to normal operation after PM exiting. These sleep and wake-up commands must be standard, but it is highly recommended that if using another custom memory, check before if these commands are supported and if they are implemented in the same way as mentioned before to avoid locking or damaging your memory.*

5.3 External pSRAM memory operation in low power modes

For the external PSRAM memory in the FRDM-RW612, there is no low power mode command. It automatically goes to the standby mode when CS goes high, with a current of around 100 µA according to the datasheet. There is also a low power mode with data retention on the 1.8 V version of the same memory HalfSleep, which drops the current to around 20 µA. To enter this mode, issue the command `0xC0` and set CS to high. To exit this mode, set CS to low. After that, CS can be set high or kept the chip clock gated until the next instruction is sent. Check the specifics of your PSRAM memory to see if it supports a low power mode. In code, it looks like below:

```
__RAMFUNC(RAM) status_t LPM_SendPSRAMToSleep(void)
{
    flexspi_transfer_t flashXfer;
    status_t status;

    uint32_t tempLUT[8] =
    {
                /* HalfSleep enter*/
                [0] = FLEXSPI_LUT_SEQ(kFLEXSPI_Command_SDR, kFLEXSPI_1PAD,
 0xC0, kFLEXSPI_Command_STOP, kFLEXSPI_1PAD, 0x00),

                /*Dummy wake-up*/
                [4] = FLEXSPI_LUT_SEQ(kFLEXSPI_Command_DUMMY_SDR, kFLEXSPI_1PAD,
 8, kFLEXSPI_Command_STOP, kFLEXSPI_1PAD, 0x00),
    };
    /* Update bottom LUT table (44-59). */
    FLEXSPI_UpdateLUT(FLEXSPI, 44, tempLUT, ARRAY_SIZE(tempLUT));

    /* Write data */
```

```
    flashXfer.deviceAddress = 0U;
    flashXfer.port          = kFLEXSPI_PortB1;
    flashXfer.cmdType       = kFLEXSPI_Command;
    flashXfer.SeqNumber     = 1;
    flashXfer.seqIndex      = 12;

    status = FLEXSPI_TransferBlocking(FLEXSPI, &flashXfer);
    return status;
}
```

6 Wake-up sources from low power modes

In PM1, any interrupt source works as a wake-up for the core as it enters in the WFI state.

In PM2, there are many different wake-up sources: GPIO interrupts, Pin interrupts, Timers, Flexcomm interfaces, DMIC, FlexSPI, USB, DMA1, SDU, PowerQuad, GAU, Ethernet, and ITRC chip reset.

In PM3/4, the wake-up sources are a bit more limited: RTC, CAPT, GPIO[24-26], and BOD1.

For more information on wake-up sources, see **section 8.2.2.2 MCU wake-up** in the RW61x User Manual (document UM11865)

7 Exit from low power modes

Exiting from PM1 or PM2 is relatively simple as it does not require re-configurations and has many wakeup source alternatives. However, when exiting PM3 or PM4, there are important things to be considered to keep the application running after exiting said power modes.

In some RTOS, like Zephyr, power mode entering and exiting is mostly handled by the operative system and requires few application considerations to work. However, it is still a good practice to check and ensure that your peripherals are correctly configured after exiting a low power mode.

For PM3 exiting, it is important to reinitialize peripherals since their states have been lost. It means that you must call all the initialization functions relevant to your application in your PM3 exit handler (clocks, pins, device-specific configurations). After a wake-up event the bootloader runs, FlexSPI is re-initialized for communication with external memories and the saved context on the AON SRAM is recovered. As a result, the application can seamlessly recover from where it was before going to sleep.

For PM4 exiting, a wake-up reset is needed, which means that no reinitialization is needed since the bootloader is executed and the application jumps to the beginning of main and runs all your initializations again.

For wakeup time in PM3 and PM4, measuring the time it takes from the wakeup source to activate to the time it takes the core to be back into the application and ready to run. Table 4 describes the measured values.

Table 4. Wakeup time

PM3	15.5 ms
PM4	25.88 ms

8 Measuring power consumption in the FRDM-RW612

The FRDM-RW612 is a compact and scalable development board from NXP, which is designed for rapid prototyping with the RW61x series of wireless MCUs.

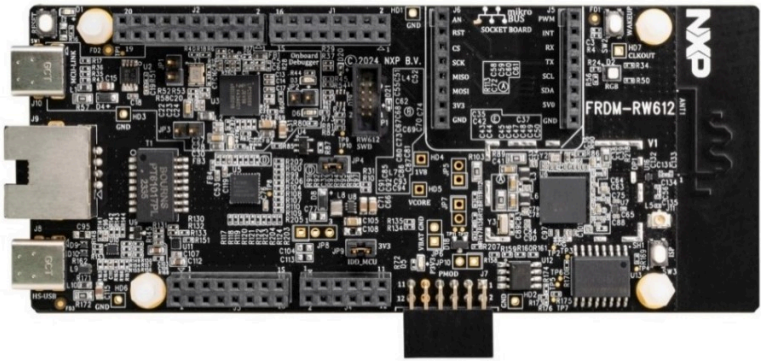


Figure 3. FRDM-RW612 board

To properly measure the chip power consumption, a small rework must be done. To measure across JP5, remove the R103. Here, only measure the VBAT for the RW612:

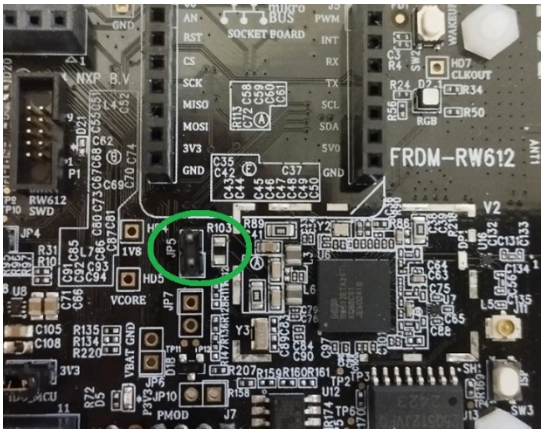


Figure 4. FRDM-RW612 Measurement rework

You can add pins in JP5 to make it easier to connect your measuring tool and to connect a jumper to close the circuit and use the board as normal.

Another place to get measurements is JP7. To enable this jumper for current measuring, remove the R128. JP7 measures VIO for the RW612.

If no specific VBAT or VIO measures of the core are needed, use JP9, labeled `IDD_MCU`, which combines the lines from JP5 (VBAT), JP7 (VIO), and VPA. This jumper is placed by default and no rework is needed.

To accurately assess the power consumption of the RW61x, it is essential to include the contribution of external memory components, such as PSRAM and Flash. These components can significantly influence overall system power usage, particularly in low-power modes. The consumption of these may change depending on the specifics of the memory itself, so understanding their individual power profiles is key to evaluating total system performance and verifying that the RW61x operates within expected parameters. For power measurements of external memories on the FRDM-RW612 evaluation board, remove and measure current across resistors R139 (Flash) and R173 (PSRAM). The FRDM board is intended for evaluation purposes and may not represent an optimized configuration. For detailed guidance, refer to the memory manufacturers' documentation and the FRDM-RW612 schematic.

9 Using SDK low power tool

The SDK comes with two examples that showcase how to use the low power mode APIs and how to handle power mode enter/exit.

- Power manager test BM
- Power mode switch

These examples show a menu through the serial interface, which allows the user to set the MCU to the different power modes. Keep in mind that these are only demo applications and can be optimized to reduce power consumption even further, for example, by disabling unused peripherals in PMs that don't directly disable them, or disabling debug pins.

Using the power mode switch demo application with default configuration and measuring the power consumption as mentioned in Section 8, we get the following results:

Table 5. JP5 measures

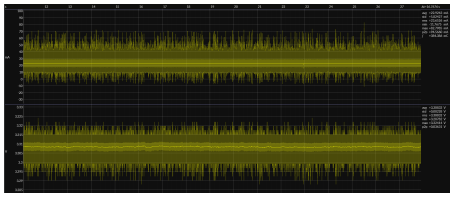
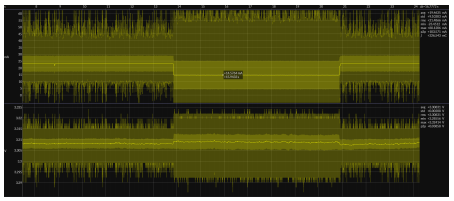
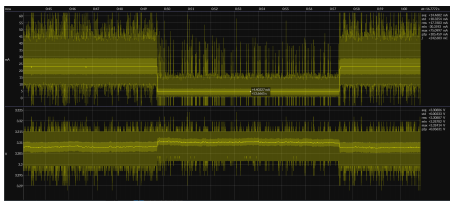
Power mode	Measured value (JP5)	Menu	Joule-scope view
PM0	22.9 mA		
PM1	14.5 mA		
PM2	4.4 mA		

Table 5. JP5 measures*...continued*

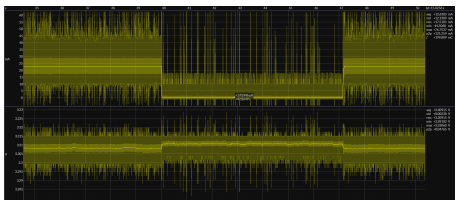
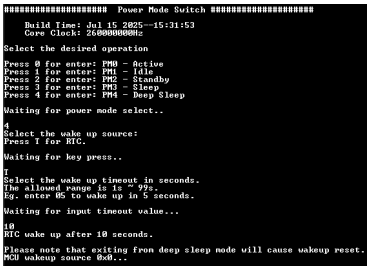
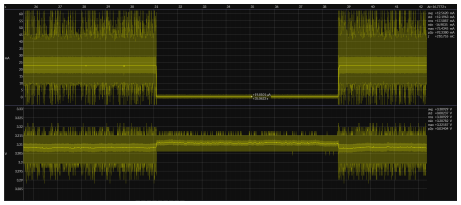
Power mode	Measured value (JP5)	Menu	Joule-scope view
PM3	170.9 µA		
PM4	19.8 µA		

Table 6. JP9 measures

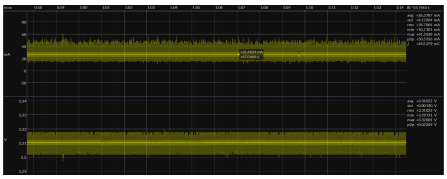
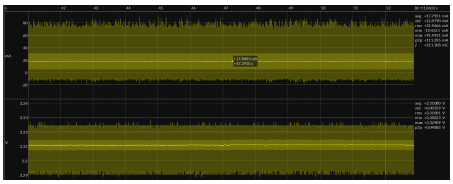
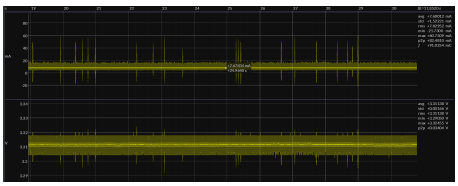
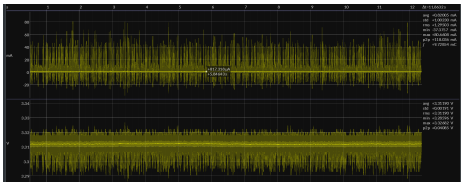
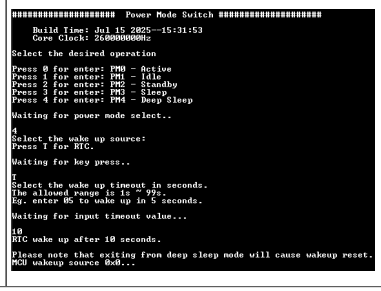
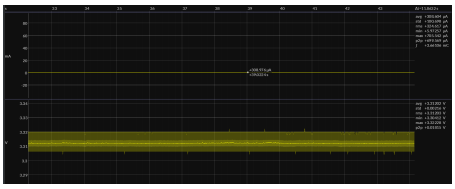
Power mode	Measured value (JP9)	Menu	Joule-scope view
PM0	26.3 mA		
PM1	17.79 mA		

Table 6. JP9 measures*...continued*

Power mode	Measured value (JP9)	Menu	Joule-scope view
PM2	7.68 mA		
PM3	0.82 mA		
PM4	306.4 µA		

10 Note about the source code in the document

The example code shown in this document has the following copyright and BSD-3-Clause license:

Copyright 2025 NXP Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials must be provided with the distribution.
3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR

BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

11 Revision history

Table 7 summarizes the revisions to this document.

Table 7. Revision history

Document ID	Release date	Description
AN14464 v1.0	03 September 2025	Initial public release

Legal information

Definitions

Draft — A draft status on a document indicates that the content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included in a draft version of a document and shall have no liability for the consequences of use of such information.

Disclaimers

Limited warranty and liability — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. NXP Semiconductors takes no responsibility for the content in this document if provided by an information source outside of NXP Semiconductors.

In no event shall NXP Semiconductors be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory.

Notwithstanding any damages that customer might incur for any reason whatsoever, NXP Semiconductors' aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms and conditions of commercial sale of NXP Semiconductors.

Right to make changes — NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

Suitability for use — NXP Semiconductors products are not designed, authorized or warranted to be suitable for use in life support, life-critical or safety-critical systems or equipment, nor in applications where failure or malfunction of an NXP Semiconductors product can reasonably be expected to result in personal injury, death or severe property or environmental damage. NXP Semiconductors and its suppliers accept no liability for inclusion and/or use of NXP Semiconductors products in such equipment or applications and therefore such inclusion and/or use is at the customer's own risk.

Applications — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

Customers are responsible for the design and operation of their applications and products using NXP Semiconductors products, and NXP Semiconductors accepts no liability for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the NXP Semiconductors product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products.

NXP Semiconductors does not accept any liability related to any default, damage, costs or problem which is based on any weakness or default in the customer's applications or products, or the application or use by customer's third party customer(s). Customer is responsible for doing all necessary testing for the customer's applications and products using NXP Semiconductors products in order to avoid a default of the applications and the products or of the application or use by customer's third party customer(s). NXP does not accept any liability in this respect.

Terms and conditions of commercial sale — NXP Semiconductors products are sold subject to the general terms and conditions of commercial sale, as published at https://www.nxp.com/profile/terms, unless otherwise agreed in a valid written individual agreement. In case an individual agreement is concluded only the terms and conditions of the respective agreement shall apply. NXP Semiconductors hereby expressly objects to applying the customer's general terms and conditions with regard to the purchase of NXP Semiconductors products by customer.

Export control — This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from competent authorities.

Suitability for use in non-automotive qualified products — Unless this document expressly states that this specific NXP Semiconductors product is automotive qualified, the product is not suitable for automotive use. It is neither qualified nor tested in accordance with automotive testing or application requirements. NXP Semiconductors accepts no liability for inclusion and/or use of non-automotive qualified products in automotive equipment or applications.

In the event that customer uses the product for design-in and use in automotive applications to automotive specifications and standards, customer (a) shall use the product without NXP Semiconductors' warranty of the product for such automotive applications, use and specifications, and (b) whenever customer uses the product for automotive applications beyond NXP Semiconductors' specifications such use shall be solely at customer's own risk, and (c) customer fully indemnifies NXP Semiconductors for any liability, damages or failed product claims resulting from customer design and use of the product for automotive applications beyond NXP Semiconductors' standard warranty and NXP Semiconductors' product specifications.

HTML publications — An HTML version, if available, of this document is provided as a courtesy. Definitive information is contained in the applicable document in PDF format. If there is a discrepancy between the HTML document and the PDF document, the PDF document has priority.

Translations — A non-English (translated) version of a document, including the legal information in that document, is for reference only. The English version shall prevail in case of any discrepancy between the translated and English versions.

Security — Customer understands that all NXP products may be subject to unidentified vulnerabilities or may support established security standards or specifications with known limitations. Customer is responsible for the design and operation of its applications and products throughout their lifecycles to reduce the effect of these vulnerabilities on customer's applications and products. Customer's responsibility also extends to other open and/or proprietary technologies supported by NXP products for use in customer's applications. NXP accepts no liability for any vulnerability. Customer should regularly check security updates from NXP and follow up appropriately.

Customer shall select products with security features that best meet rules, regulations, and standards of the intended application and make the ultimate design decisions regarding its products and is solely responsible for compliance with all legal, regulatory, and security related requirements concerning its products, regardless of any information or support that may be provided by NXP.

NXP has a Product Security Incident Response Team (PSIRT) (reachable at PSIRT@nxp.com) that manages the investigation, reporting, and solution release to security vulnerabilities of NXP products.

NXP B.V. — NXP B.V. is not an operating company and it does not distribute or sell products.

Trademarks

Notice: All referenced brands, product names, service names, and trademarks are the property of their respective owners.

NXP — wordmark and logo are trademarks of NXP B.V.

Contents