AN14185

DCDC Usage on MCXNx4x/Nx3x

Rev. 1 — 20 January 2024 Application note

Document information

Information	Content
Keywords	MCXNx4x/Nx3x, AN14185, DCDC
Abstract	This application note is designed to provide a better understanding of the on-chip DCDC module.

1 MCXNx4x/Nx3x DCDC overview

The MCXNx4x/Nx3x series MCU embeds an on-chip DC-DC converter, which is able to supply the digital core domain of the MCU. This DC-DC converter is configurable and can interact with several on-chip peripherals.

This application note is designed to provide a better understanding of the on-chip DCDC module. It offers a comprehensive guide on how to control both basic and advanced parameters, as well as how to configure the DCDC module to work efficiently with other peripherals.

1.1 Power supply scheme

Figure 1 shows the MCXNx4x power supply scheme.

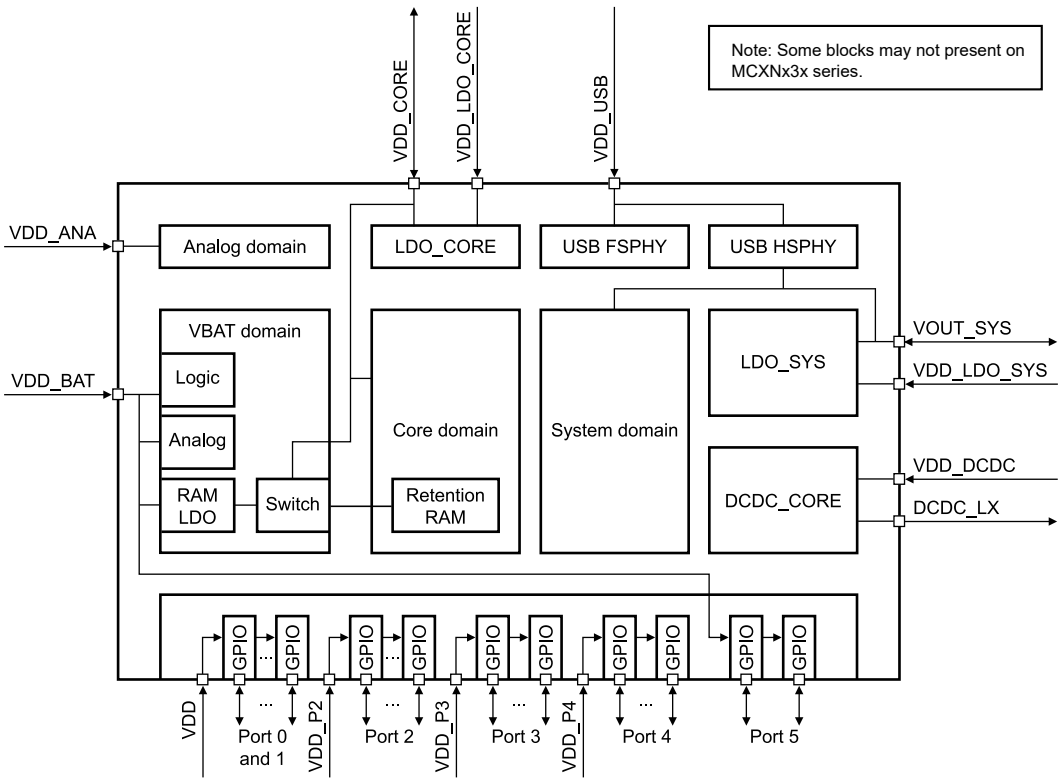

Figure 1. MCXNx4x power supply scheme

The MCXNx4x/Nx3x has several power domains, while the core DCDC module (DCDC_CORE) can be used to power the core domain to provide a highly efficient power source. Alternatively, the core domain can also be powered using the core LDO module (LDO_CORE), which can eliminate the need of an external inductor.

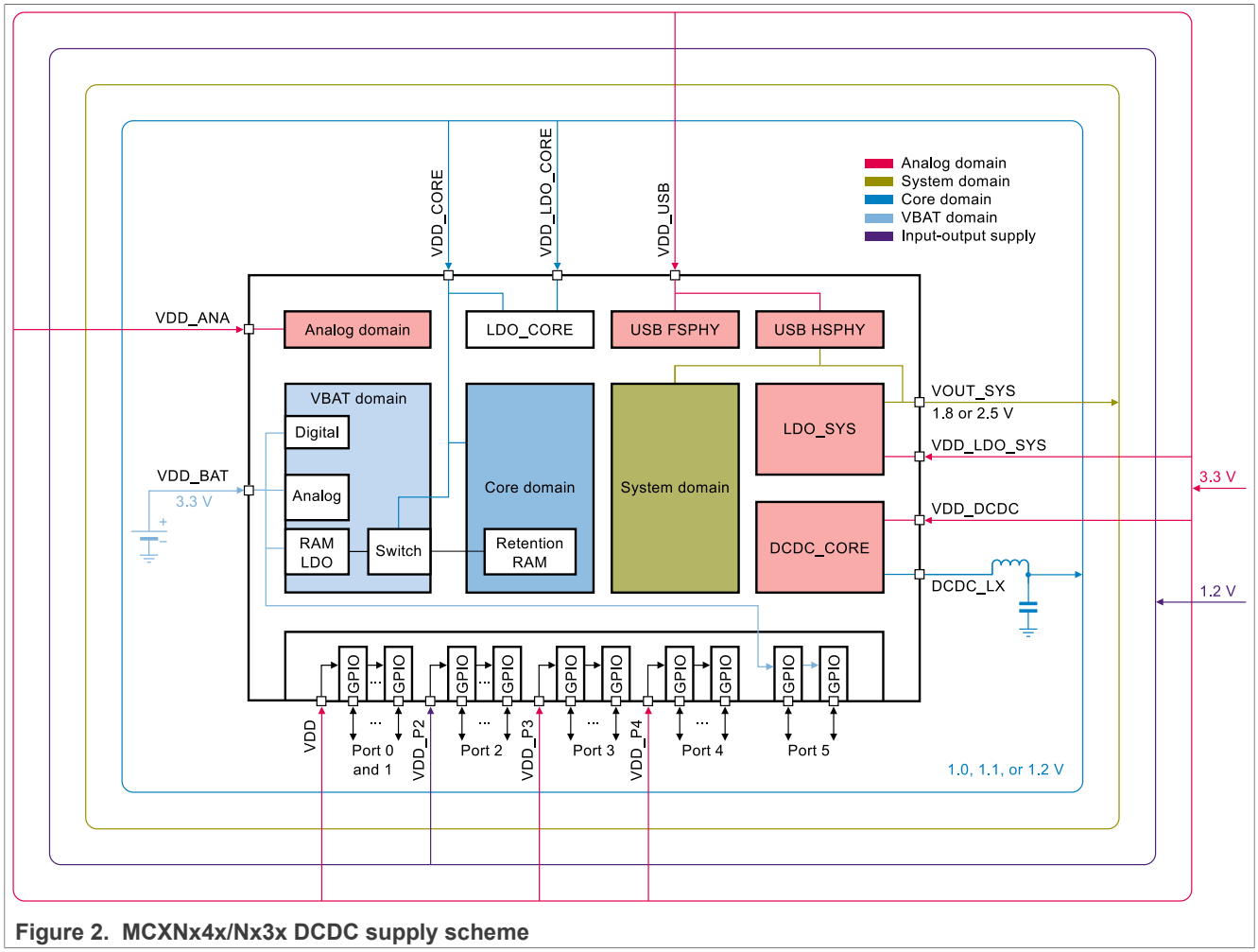

Figure 2. MCXNx4x/Nx3x DCDC supply scheme

In a typical configuration using the core DCDC module as the core domain supply, the VDD_DCDC input is connected to the system power, a power inductor is connected between DCDC_LX and VDD_CORE pins, and a filtering capacitor is placed on the output of the power inductor. In this configuration, the output from the LDO regulator is directly connected to its input, therefore, the regulator is bypassed, and the system is powered through VDD_CORE input from the DCDC.

1.2 On-chip DCDC module overview

The on-chip DCDC module is a synchronous rectification, burst mode DCDC converter that can provide up to 100 mA (approx.) load current. The DCDC controller drives a pair of complementary MOSFETs using a PWM signal based on the output voltage and current feedback. This feedback loop ensures a stable output across different loads.

The DCDC module has various operating parameters, including:

- Output voltage
- Switching frequency
- Power-saving modes (Drive strength)

These parameters can be adjusted as needed to achieve the highest efficiency under different load conditions. The DCDC module also provides a few of the auxiliary signal outputs indicating the internal operating status of the module. These signals can be used to synchronize the module with several analog / digital peripherals, such as ADC and CTIMER.

1.3 SPC module overview

The system power control (SPC) peripheral controls and monitors the operation of on-chip regulators and power modes. In general, the operating mode of the MCXNx4x/Nx3x can be divided into the following two modes:

- Active mode
- Low-power modes (Deep sleep, Power-down)

In active mode, the SoC operates normally, with all peripherals available. Various low-power modes are available for achieving different power consumption requirements by shutting down different power domains when not in use.

Note: *Sleep mode shares the active mode regulator configuration, while in the deep power-down mode, the core domain regulators are shut down.*

The regulators are controlled on a power mode basis. Two sets of configuration registers for active mode and all low-power modes are as follows:

- `SPC.ACTIVE_CFG` for active mode regulator settings
- `SPC.LP_CFG` for low-power modes regulator settings

The settings in `SPC.LP_CFG` only take effect when the SoC enters one of the low-power modes, while updating the settings in `SPC.ACTIVE_CFG` take effect immediately.

2 Control the DCDC parameters

This section describes how to control the operating parameters of the DCDC module.

2.1 Output voltage

This section provides an introduction of the output voltage parameter and how to adjust it through SPC.

2.1.1 Introduction

The core DCDC output voltage in the MCXNx4x/Nx3x series is selectable from the following four options:

- 0.7 V (Retention voltage, **only available in low-power modes**)
- 1.0 V (Mid voltage)
- 1.1 V (Normal voltage)
- 1.2 V (Overdrive voltage)

The output voltage of the DCDC module must be adjusted according to the operating frequency. In general, a higher operating frequency requires a higher supply voltage, and vice versa. In low-power modes, when the core domain is clock-gated and in the state retention mode, the operating voltage can be further reduced to the retention voltage for lower leakage while keeping core domain states and data.

Note: *For detail on frequency and voltage requirements, refer to the specific device data sheet.*

2.1.2 Adjusting output voltage through SPC

To adjust the DCDC output voltage, set the `SPC.ACTIVE_CFG[DCDC_VDD_LVL]` or `SPC[LP_CFG.DCDC_VDD_LVL]` to one of the following values:

- 2’b00 – 0.7 V static retention voltage, only available in LP_CFG register
- 2’b01 – 1.0 V
- 2’b10 – 1.1 V
- 2’b11 – 1.2 V

Note: *The `SPC.SC.BUSY` flag must be polled for voltage transition completion.*

Note: *The `CORELDO_VDD_LVL` field must be set to the same level as the `DCDC_VDD_LVL` field, even if the LDO is bypassed.*

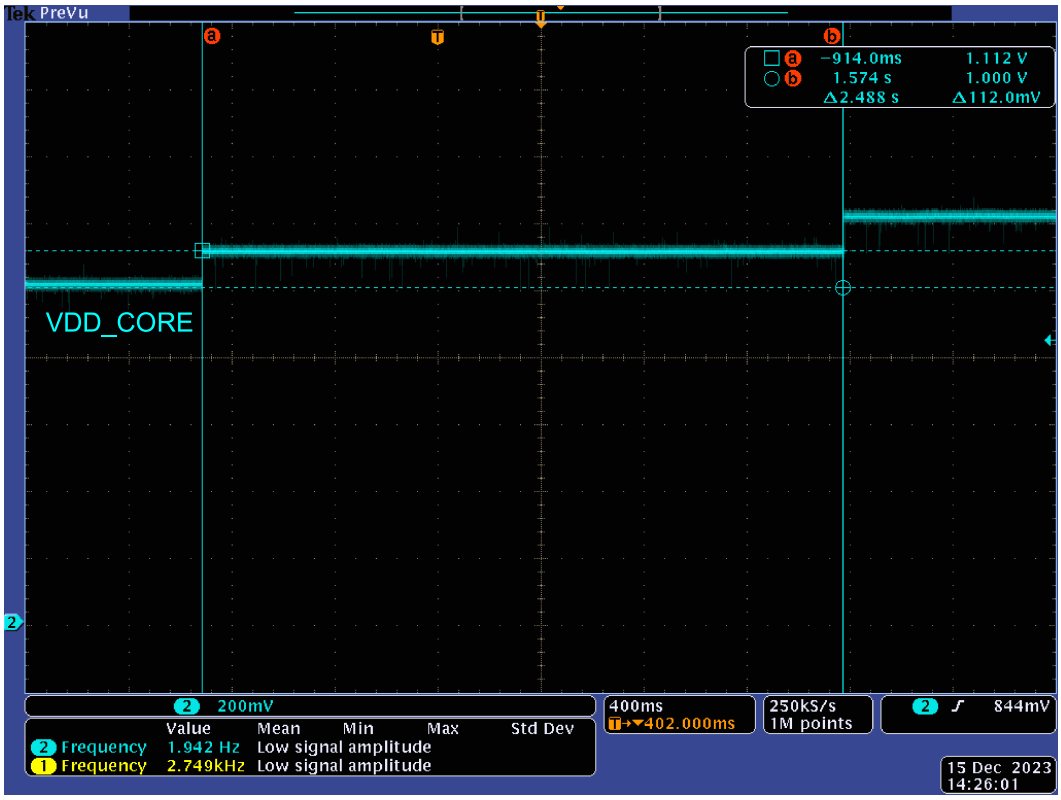

Figure 3. DCDC voltage scaling

A ramp up time is required before the new output voltage is stabilized.

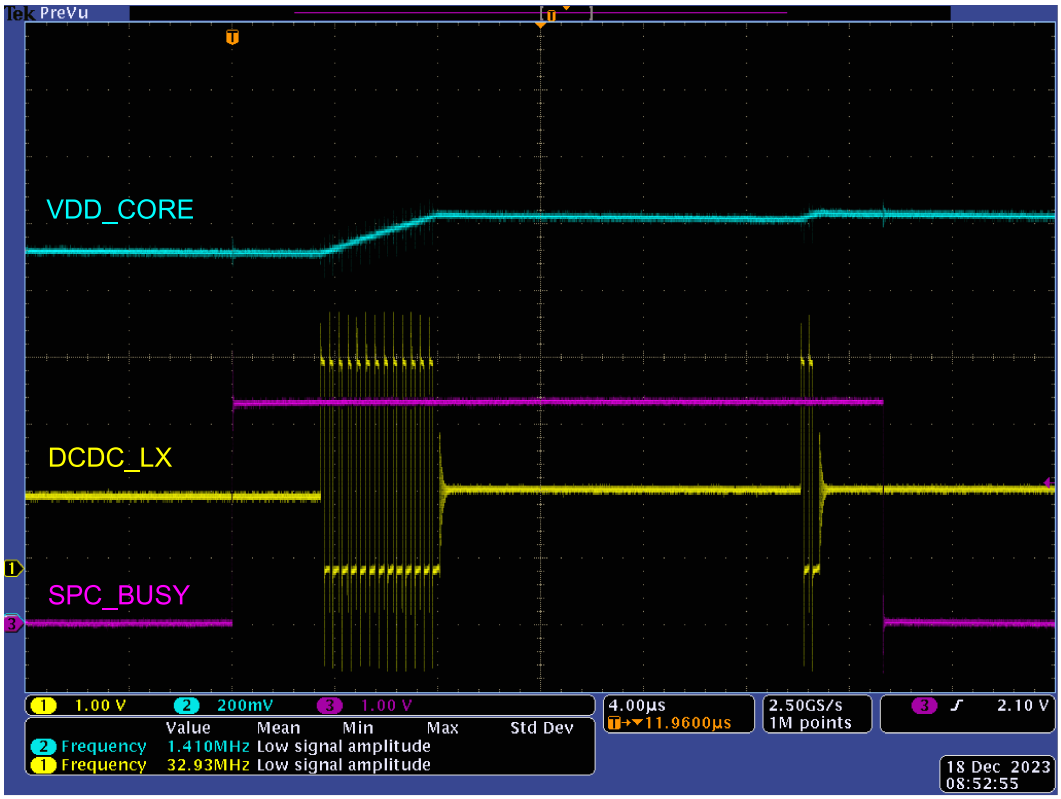

Figure 4. Output voltage ramp up diagram in Active mode

2.2 Drive strength

This section provides an introduction of the drive strength parameter and how to adjust it through SPC.

2.2.1 Introduction

The on-chip DCDC has various low-power features, and the drive strength is one of the power-saving features. For most of the cases, when the SoC is in Active mode, the DCDC uses Normal drive strength, however, the efficiency decreases when the load decreases or the SoC enters low-power modes. By configuring the DCDC into Low drive strength mode, light-load efficiency is improved by limiting the inductor current. The following drive strength modes are available in the MCXNx4x/Nx3x series:

- Normal drive strength
- Low drive strength
- Pulse Refresh mode (**Only available in low-power modes**)

2.2.2 Pulse Refresh mode

The Pulse Refresh mode can achieve even better efficiency by switching off the DCDC module for a specific, timer-controlled period in certain low-power modes. This approach can significantly reduce the quiescent current.

In Pulse Refresh mode, the SPC peripheral uses a low-speed clock source from the RTC module (LP_OSC clock) to clock an internal self-reloading counter. When the counter expires, the DCDC is automatically enabled to allow one single burst to happen. The burst replenishes the bulk capacitor of VDD_CORE to reach the

targeted voltage level. After the burst is completed, the DCDC is powered off with its output floating to avoid leakage current and wait for the next refresh request issued by SPC.

2.2.3 Adjusting drive strength through SPC

The drive strength of the DCDC module can be adjusted from the SPC.ACTIVE_CFG and SPC.LP_CFG registers, by alternating the DCDC_VDD_DS field from the above registers to one of the following values:

- 2'b00: Pulse Refresh mode (**Only available in low-power modes**)
- 2'b01: Low drive strength
- 2'b10: Normal drive strength

Note: *The DCDC requires the SPC internal bandgap to be enabled only when operating in Normal drive strength mode. Therefore, the SPC internal bandgap can be disabled **after** the DCDC is configured to Low drive strength mode to reduce quiescent current and must be switched back on **before** the drive strength is switched to normal again. The bandgap can be configured in the same registers SPC.ACTIVE_CFG and SPC.LP_CFG, by modifying the BGMODE field to one of the following values:*

- 2'b00: Bandgap disabled
- 2'b01: Bandgap enabled; buffer disabled
- 2'b10: Bandgap enabled; buffer enabled

The DCDC normal drive strength is available when bandgap is enabled with the buffer is either on or off (either 2'b01 or 2'b10 can be used).

Note: *The SPC.SC.BUSY flag must be polled for drive strength transition completion.*

Note: *There is a limitation when switching the drive strength from normal to low when the output voltage is kept the same or higher than the current configuration. The output voltage must be raised to a higher level before switching the drive strength, then switching the voltage and drive strength to desired values at the same time.*

Note: *When switching the DCDC drive strength from low to normal, the voltage level must be maintained as the same level before switching the drive strength to low. Otherwise, an unexpected low-voltage detect (LVD) event may occur.*

2.2.4 Use Pulse Refresh in SoC low-power modes

As the Pulse Refresh mode is only available in certain low-power modes (except deep power-down mode), therefore, only the SPC.LP_CFG register has the corresponding settings. To use Pulse Refresh mode, the SoC must be put in one of the following low-power modes:

- Deep Sleep mode
- Power-down mode

Prior to entering these low-power modes, the low-power clock source feeding the SPC module must be configured and in running state from the VBAT module.

The clock source feeding to the SPC is from the RTC LP_OSC clock, and can be selected from either:

- OSC_32K (External 32.768 kHz crystal oscillator)
- FRO_16K (Internal 16.384 kHz FRO oscillator)

After the clock source is configured and running, the VDD_BAT domain clock must be enabled from either VBAT.OSCCLKE or VBAT.FROCLKE register, the RTC.CTRL register must be updated with the corresponding clock source selected as the LP_OSC clock.

The SPC refresh counter reload value controls the number of low-power clock cycles between two DCDC refresh bursts. A larger interval has a longer shutdown time, resulting in a lower average quiescent current, however, this causes more voltage drop before the next refresh burst comes in to replenish the core supply.

If the core voltage drops below the limiting value during the DCDC OFF cycle (below the retention voltage for low-power modes), the SoC may not function correctly. Similarly, if the interval is set to a value too small, the DCDC may not have enough time to complete a full burst cycle before powered down by SPC again. Therefore, it is crucial to program an appropriate value (N) into the `SPC.DCDC_BURST_CFG[PULSE_REFRESH_CNT]` field before entering the pulse-refresh mode. The burst interval can be calculated using the following formula:

$$t_{cycle} = \frac{1}{f_{ref}} \times (N+2)$$

Note: *The DCDC module regulates the burst period not to exceed the programmed voltage level, therefore, no overvoltage occurs even if the refresh interval is low.*

Figure 5 shows an example that results into ~8 ms of the DCDC OFF period.

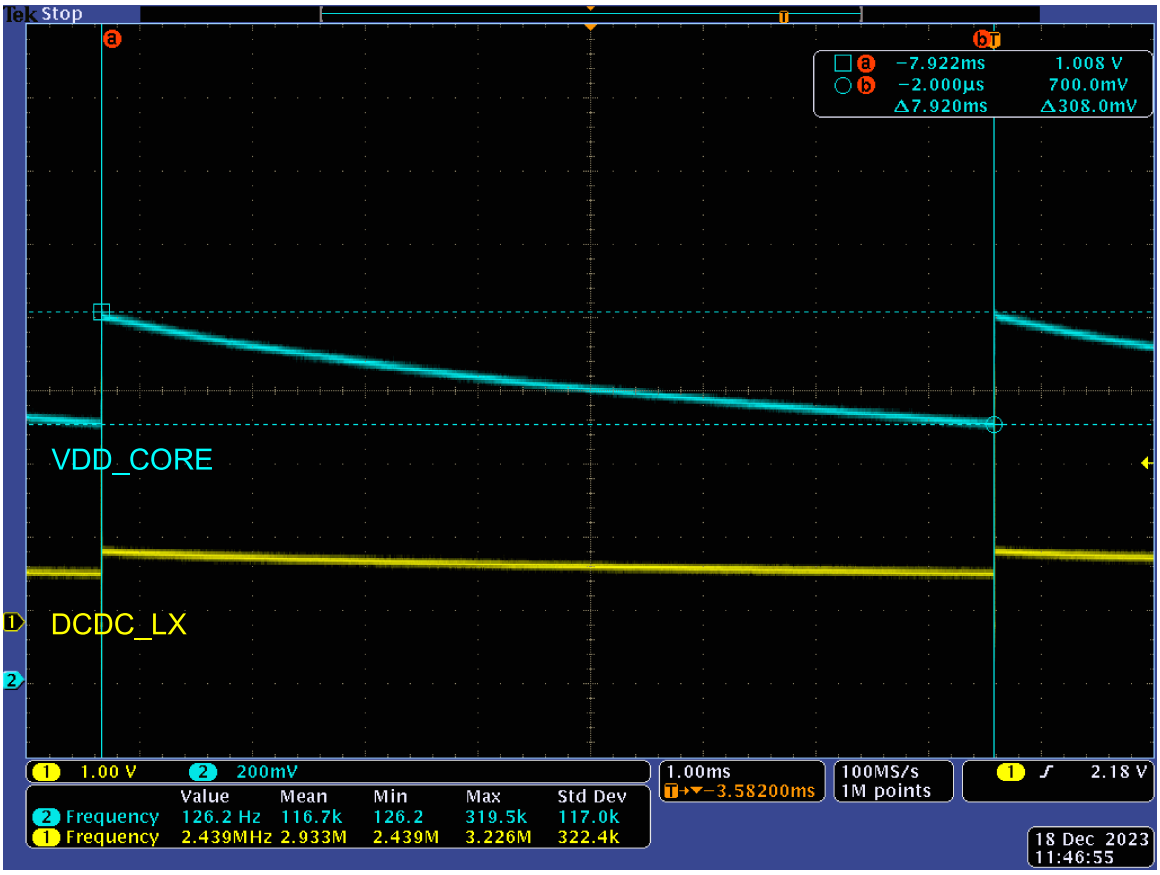

Figure 5. DCDC Pulse Refresh in Deep Sleep mode

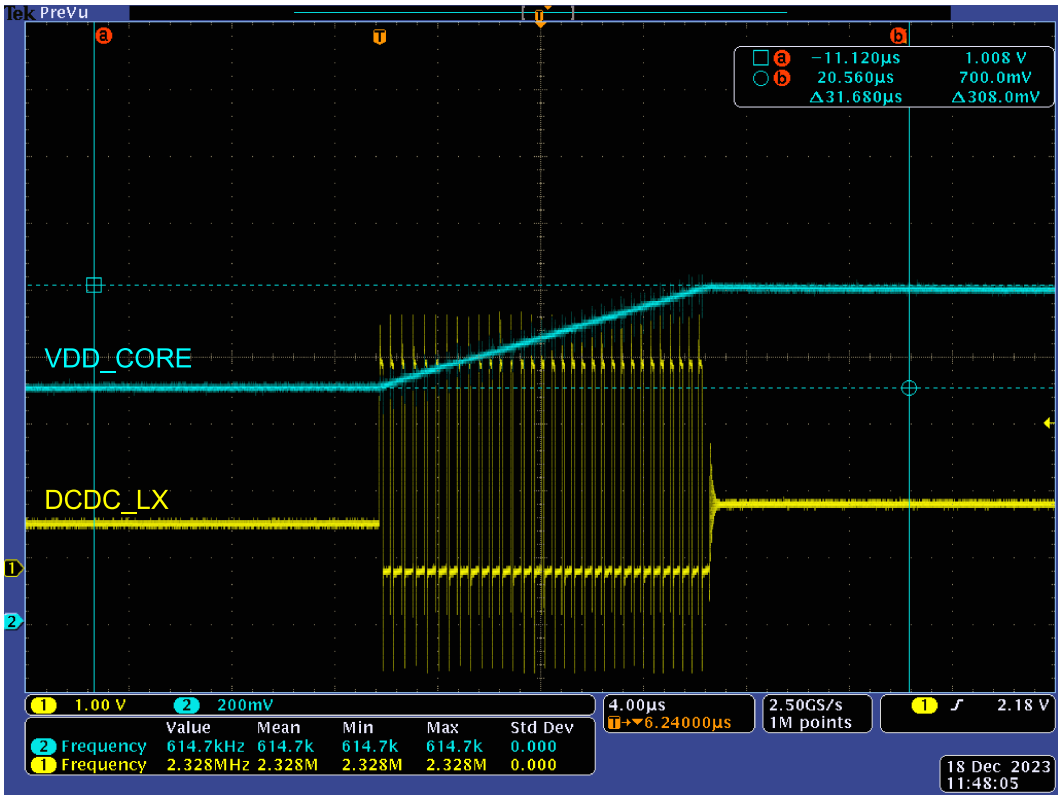

Figure 6. Refresh burst controlled by SPC counter

In a single refresh mode burst shown in Figure 6, the DCDC is enabled by the SPC counter event.

2.3 Switching frequency (frequency stabilization)

This section provides an introduction of the switching frequency parameter and how to adjust it through SPC.

2.3.1 Introduction

The on-chip DCDC module works in burst mode, which means that the length of a burst period and the interval between two bursts can vary depending on the load condition of the core domain. Within a single burst, the switching frequency can be adjusted to a fixed value by the user, if desired. The frequency is controlled either by the DCDC itself or using "frequency stabilization" mode. However, using the latter may limit the transient response.

Frequency control can be helpful if a specific switching frequency or its harmonic frequencies have to be avoided, or in layout space limits, which requires nondefault inductor and capacitor values.

Higher switching frequencies can work with smaller inductor and output capacitor, while there is a trade-off between switching loss and transient response. In general, a higher switching frequency increases the switching loss of the integrated MOSFET, however, improves the transient response across the load. Therefore, a proper inductor and switching frequency should be selected based on the application and PCB layout to achieve high efficiency and performance.

2.3.2 Controlling burst frequency from SPC

The burst frequency is normally automatically controlled by the DCDC module. To enable manual control of the switching frequency, program an appropriate value to `SPC.DCDC_CFG[FREQ_CNTRL]`, then enable the

frequency stabilization mode from SPC.DCDC_CFG[FREQ_CTRL_ON]. According to the SoC specification, typical switching frequencies are selectable from 3 MHz to 8 MHz.

Note: *The SPC.SC.BUSY flag must be polled for switching frequency changes.*

3 Synchronizing DCDC with other peripherals

This section explains how to synchronize DCDC with other peripherals.

3.1 DCDC BURST_ACTIVE status output

This section describes the DCDC BURST_ACTIVE output signal and how to measure it using CTIMER.

3.1.1 Introduction

The DCDC module outputs various signals to the SPC. One of the output signals is a BURST_ACTIVE signal indicating whether there is an on-going burst carried out by the DCDC module. This signal is routed through INPUTMUX to CTIMER capture inputs, therefore, the pulse width and duty cycle of the burst can be measured. User code can also choose to synchronize with DCDC burst start or stop events using CTIMER capture interrupts.

3.1.2 Measuring DCDC bursts using CTIMER

The MCXNx4x/Nx3x series MCU has several Generic Counter/Timer modules (CTIMERs), each CTIMER instance has four capture inputs. As the DCDC burst signal operates at a relatively high frequency, the CTIMER instance must count at a high frequency (150 MHz Typ) to get the best accuracy.

To measure the cycle time and duty cycle from the internal BURST_ACTIVE signal, both rising edge and falling edge on this signal are required to trigger CTIMER capture events. Therefore, in this case, two capture inputs must be set through INPUTMUX.

- DCDC_BURST_ACTIVE to CTIMER capture 0
- DCDC_BURST_ACTIVE to CTIMER capture 1

Program the following INPUTMUX registers.

- INPUTMUX.CTIMERnCAP0[INP] = 7’b10110
- INPUTMUX.CTIMERnCAP1[INP] = 7’b10110

Configure the CTIMER input clock to its maximum (150 MHz) input clock. Afterward, set up two input capture channels for rising edge and falling edge. As interrupt handling takes a considerable amount of CPU time, disable capture interrupts for both channels.

Note: *Program CTIMER.CTCR[SELCC] to 4’b0001 to clear the counter on a new falling edge of the BURST_ACTIVE signal (capture channel 0), then enable the counter-clear feature by setting CTIMER.CTCR[ENCC] to 1.*

Note: *The capture event interrupt can be used to synchronize the software action with a start or stop event of the DCDC burst action, while the counter could be disabled in ISR to avoid repetitive interrupts.*

Figure 7 illustrates the measurement process.

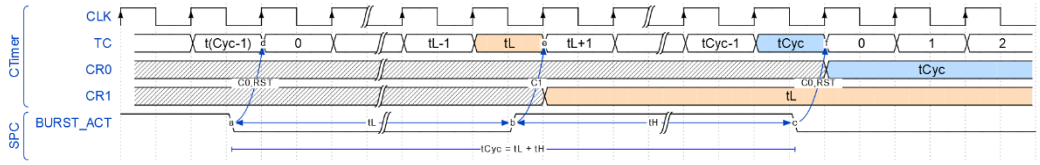

Figure 7. Burst time measurement diagram

After the counter started, it resets upon the falling edge of the BURST_ACTIVE signal. This indicates that a new cycle has been started by the DCDC module, and the counter value prior to the reset is captured into the CR0 register. When the burst is started, the BURST_ACTIVE rising edge is captured into the CR1 register, and the CR1 value indicates the low time of the current burst cycle. Then, the CR0 is updated when the next burst cycle starts, and the reading of the CR0 register is the total cycle time of the current DCDC period.

The above sequence happens periodically and automatically, and the CR0/CR1 values are always the cycle time and low time of one consecutive burst period, no matter which is captured earlier.

The low (idle) time of the burst cycle (t_L) can be calculated using the following formula:

$$t_L = \frac{1}{f_{timer}} \times N_{CR1}$$

Similarly, the full cycle of the burst cycle (t_{cyc}) can be calculated using the following formula:

$$t_{cyc} = \frac{1}{f_{timer}} \times N_{CR0}$$

This way, the final estimated DCDC load percentage (duty cycle) can be calculated using:

$$P_{act} = \frac{t_{cyc} - t_L}{t_{cyc}} \times 100\%$$

3.2 ADC "Quiet" period sampling

This section describes the quiet period and ADC quiet period sampling.

3.2.1 Introduction

When using DCDC as the core power supply, it is possible to introduce additional switching noise to the system. These switching noise could be coupled to the analog power supply rails or voltage reference sources, therefore have an impact on the SoC's analog performance.

The SPC has a mechanism to allow manually creating a "quiet" period prior to sensitive analog sampling, by synchronizing the start of this period with an internal ADC sampling trigger. The ADC samples in the noise-free "quiet period" and reduces the impact of the switching noise.

3.2.2 Request a Burst from SPC

To maximize the length of the "quiet" period of the DCDC, the software can request an additional burst from the DCDC. The DCDC synchronization output is triggered once this additional burst is completed and a full "quiet" period begins, yielding the longest ADC sampling time as possible.

To request a burst manually, the `SPC.DCDC_BURST_CFG[BURST_ACK]` flag must be cleared by writing a 1 to this bit, then the software can request a burst by setting the `SPC.DCDC_BURST_CFG[BURST_REQ]` bit. The `BURST_ACK` flag is set when the burst has completed.

Note: *The `SPC.DCDC_BURST_CFG[BURST_REQ]` flag is write-only.*

3.2.3 Synchronize ADC sampling with DCDC

The SPC uses another signal named `DCDC_BURST_TRIG_PULSE` to provide a synchronized trigger to the ADC, which must be enabled before requesting a burst manually. This output can be enabled by programming a 1 to the `SPC.DCDC_BURST_CFG[EXT_BURST_EN]` field.

Note: *This signal only activates when the current burst is requested by the software.*

To configure an ADC sampling using `DCDC_BURST_TRIG_PULSE` as the trigger, connect the signal to one of the four hardware inputs of the ADC through INPUTMUX by programming:

- `INPUTMUX.ADCm_TRIGn[TRIGIN]`: 8'b0000_1010

Configure an ADC command by setting the desired channel and sampling mode, then select one of the hardware triggers to trigger the corresponding ADC command by enabling the `ADC.TCTRLn[HTEN]` bit.

Request a burst using the SPC register operation described from the above chapter, and the ADC sampling will be automatically started when the DCDC enters a "quiet" period.

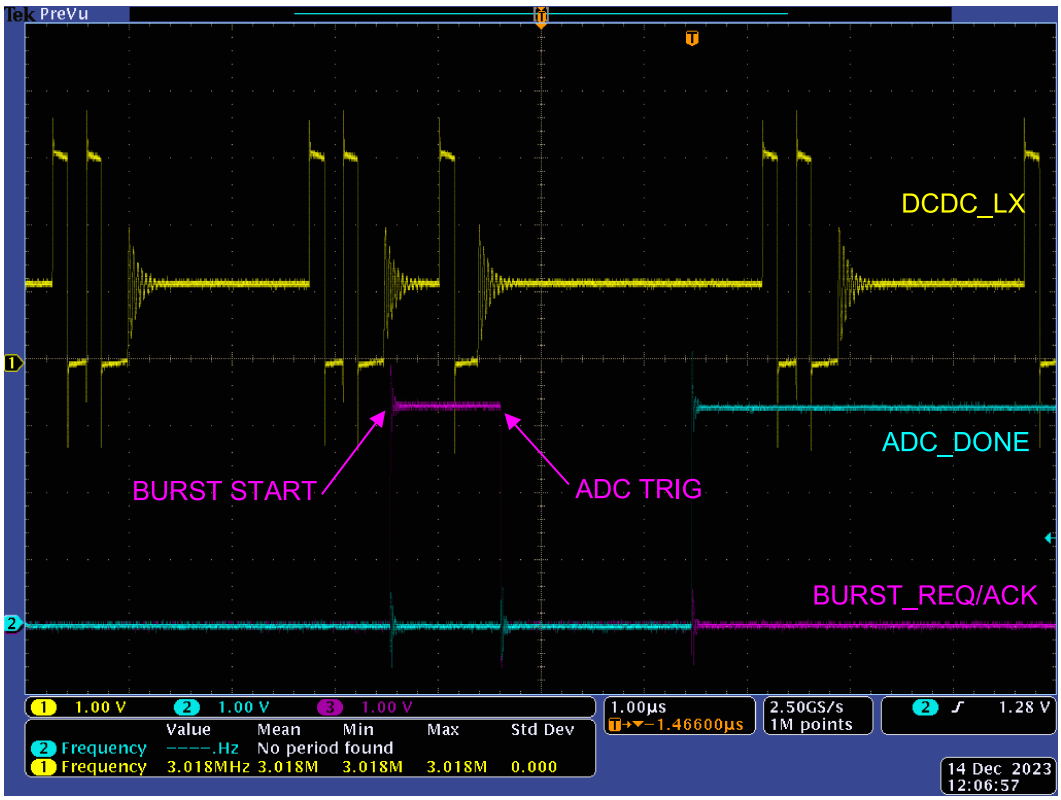

Figure 8. DCDC synchronized ADC sampling

In Figure 8, the additional manual burst can be observed when the software triggers the BURST_REQ. After the manual burst is completed, the BURST_ACK acknowledges the burst, while ADC sampling starts automatically. The ADC sampling completes within the "quiet" period after the burst (indicated by the `ADC_DONE` signal set from the ISR), before the next automatic burst happens.

Note: *It is recommended that the ADC sampling time should be configured in a way that it does not exceed the maximum estimated "quiet" period.*

4 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 2024 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 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.

5 Revision history

Table 1 summarizes revisions to this document.

Table 1. Revision history

Document ID	Release date	Description
AN14185 v.1	20 January 2024	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.

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.

MCX — is a trademark of NXP B.V.

Microsoft, Azure, and ThreadX — are trademarks of the Microsoft group of companies.

Contents

For more information, please visit: https://www.nxp.com

Date of release: 20 January 2024
Document identifier: AN14185