nRF52805

Revision 1

Errata

v1.3

NORDIC® SEMICONDUCTOR

Contents

1 nRF52805 Revision 1 Errata

This Errata document contains anomalies and configurations for the nRF52805 chip, Revision 1 (CAAA-A00, CAAA-Ax0).

2 Revision history

See the following list for an overview of changes from previous versions of this document.

Version	Date	Change
nRF52805 Revision 1 v1.3	05.06.2023	• Added: No. 241. “Static 400 µA current after SAADC is disabled” • Added: No. 262. “CPU does not resume execution after CPU IDLE” • Updated: No. 246. “Intermittent extra current consumption when going to sleep”
nRF52805 Revision 1 v1.2	09.11.2020	• Added: No. 242. “NVMC operations during POFWARN cause the CPU to hang” • Added: No. 245. “CRC is wrong when data whitening is enabled and address field is included in CRC calculation” • Added: No. 246. “Intermittent extra current consumption when going to sleep”

Version	Date	Change
nRF52805 Revision 1 v1.1	18.06.2020	• Added: No. 15. “RAM[x].POWERSET/CLR read as zero” • Added: No. 20. “Register values are invalid” • Added: No. 31. “Calibration values are not correctly loaded from FICR at reset” • Added: No. 36. “Some registers are not reset when expected” • Added: No. 66. “Linearity specification not met with default settings” • Added: No. 68. “EVENTS_HFCLKSTARTED can be generated before HFCLK is stable” • Added: No. 77. “RC oscillator is not calibrated when first started” • Added: No. 78. “High current consumption when using timer STOP task only” • Added: No. 81. “PIN_CNF is not retained when in debug interface mode” • Added: No. 83. “STOPPED event occurs twice if the STOP task is triggered during a transaction” • Added: No. 88. “Increased current consumption when configured to pause in System ON idle” • Added: No. 136. “Bits in RESETREAS are set when they should not be” • Added: No. 155. “IN event may occur more than once on input edge” • Added: No. 156. “Some CLR tasks give unintentional behavior” • Added: No. 173. “Writes to LATCH register take several CPU cycles to take effect” • Added: No. 176. “Flash erase through CTRL-AP fails due to watchdog time-out” • Added: No. 179. “COMPARE event is generated twice from a single RTC compare match” • Added: No. 184. “Erase or write operations from the external debugger fail when CPU is not halted” • Added: No. 210. “Bits in GPIO LATCH register are incorrectly set to 1” • Added: No. 212. “Events are not generated when switching from scan mode to no-scan mode with burst enabled” • Added: No. 217. “RAM calibration values are not correctly loaded from FICR at reset” • Added: No. 219. “I2C timing spec is violated at 400 kHz”
nRF52805 Revision 1 v1.0	28.08.2019	Limited release

3 New and inherited anomalies

The following anomalies are present in Revision 1 of the nRF52805 chip.

ID	Module	Description	New in Revision 1
15	POWER	RAM[x].POWERSET/CLR read as zero	X
20	RTC	Register values are invalid	X
31	CLOCK	Calibration values are not correctly loaded from FICR at reset	X
36	CLOCK	Some registers are not reset when expected	X
66	TEMP	Linearity specification not met with default settings	X
68	CLOCK	EVENTS_HFCLKSTARTED can be generated before HFCLK is stable	X
77	CLOCK	RC oscillator is not calibrated when first started	X
78	TIMER	High current consumption when using timer STOP task only	X
81	GPIO	PIN_CNF is not retained when in debug interface mode	X
83	TWIS	STOPPED event occurs twice if the STOP task is triggered during a transaction	X
88	WDT	Increased current consumption when configured to pause in System ON idle	X
136	System	Bits in RESETREAS are set when they should not be	X
155	GPIOTE	IN event may occur more than once on input edge	X
156	GPIOTE	Some CLR tasks give unintentional behavior	X
173	GPIO	Writes to LATCH register take several CPU cycles to take effect	X
176	System	Flash erase through CTRL-AP fails due to watchdog time-out	X
179	RTC	COMPARE event is generated twice from a single RTC compare match	X
184	NVMC	Erase or write operations from the external debugger fail when CPU is not halted	X
210	GPIO	Bits in GPIO LATCH register are incorrectly set to 1	X
212	SAADC	Events are not generated when switching from scan mode to no-scan mode with burst enabled	X
217	RAM	RAM calibration values are not correctly loaded from FICR at reset	X
219	TWIM	I2C timing spec is violated at 400 kHz	X
241	SAADC	Static 400 µA current after SAADC is disabled	X
242	NVMC	NVMC operations during POFWARN cause the CPU to hang	X
245	RADIO	CRC is wrong when data whitening is enabled and address field is included in CRC calculation	X

ID	Module	Description	New in Revision 1
246	System	Intermittent extra current consumption when going to sleep	X
262	POWER	CPU does not resume execution after CPU IDLE	X

Table 1: New and inherited anomalies

3.1 [15] POWER: RAM[x].POWERSET/CLR read as zero

This anomaly applies to Revision 1, build codes CAAA-A00, CAAA-Ax0.

Symptoms

RAM[x].POWERSET and RAM[x].POWERCLR read as zero, even though the RAM is on.

Conditions

Always.

Consequences

Not possible to read the RAM state using RAM[x].POWERSET and RAM[x].POWERCLR registers. Write works as it should.

Workaround

Use RAM[x].POWER to read the state of the RAM.

3.2 [20] RTC: Register values are invalid

This anomaly applies to Revision 1, build codes CAAA-A00, CAAA-Ax0.

Symptoms

RTC registers will not contain the correct/expected value if read.

Conditions

The RTC has been idle.

Consequences

RTC configuration cannot be determined by reading RTC registers.

Workaround

Execute the below code before you use RTC.

```
NRF_CLOCK->EVENTS_LFCLKSTARTED  = 0;
NRF_CLOCK->TASKS_LFCLKSTART     = 1;
while (NRF_CLOCK->EVENTS_LFCLKSTARTED == 0) {}
NRF_RTC0->TASKS_STOP = 0;
```

3.3 [31] CLOCK: Calibration values are not correctly loaded from FICR at reset

This anomaly applies to Revision 1, build codes CAAA-A00, CAAA-Ax0.

Symptoms

RCOSC32KICALLENGTH is initialized with the wrong FICR value.

Conditions

Always.

Consequences

RCOSC32KICALLENGTH default value is wrong.

Workaround

Execute the following code after reset:

```
*(volatile uint32_t *)0x4000053C = ((*(volatile uint32_t *)0x10000244) & 0x0000E000) >>
13;
```

This code is already present in the latest system_nrf52.c file.

3.4 [36] CLOCK: Some registers are not reset when expected

This anomaly applies to Revision 1, build codes CAAA-A00, CAAA-Ax0.

Symptoms

After watchdog timeout reset, CPU lockup reset, soft reset, or pin reset, the following CLOCK peripheral registers are not reset:

- CLOCK->EVENTS_DONE
- CLOCK->EVENTS_CTTO
- CLOCK->CTIV

Conditions

After watchdog timeout reset, CPU Lockup reset, soft reset, and pin reset.

Consequences

Register reset values might be incorrect. It may cause undesired interrupts in case of enabling interrupts without clearing the DONE or CTTO events.

Workaround

Clear affected registers after reset. This workaround has already been added into system_nrf52.c file. This workaround has already been added into system_nrf52840.c file present in MDK 8.11.0 or later.

3.5 [66] TEMP: Linearity specification not met with default settings

This anomaly applies to Revision 1, build codes CAAA-A00, CAAA-Ax0.

Symptoms

TEMP module provides non-linear temperature readings over the specified temperature range.

Conditions

Always.

Consequences

TEMP module returns out of spec temperature readings.

Workaround

Execute the following code after reset:

```
NRF_TEMP->A0 = NRF_FICR->TEMP.A0;
NRF_TEMP->A1 = NRF_FICR->TEMP.A1;
NRF_TEMP->A2 = NRF_FICR->TEMP.A2;
NRF_TEMP->A3 = NRF_FICR->TEMP.A3;
NRF_TEMP->A4 = NRF_FICR->TEMP.A4;
NRF_TEMP->A5 = NRF_FICR->TEMP.A5;
NRF_TEMP->B0 = NRF_FICR->TEMP.B0;
NRF_TEMP->B1 = NRF_FICR->TEMP.B1;
NRF_TEMP->B2 = NRF_FICR->TEMP.B2;
NRF_TEMP->B3 = NRF_FICR->TEMP.B3;
NRF_TEMP->B4 = NRF_FICR->TEMP.B4;
NRF_TEMP->B5 = NRF_FICR->TEMP.B5;
NRF_TEMP->T0 = NRF_FICR->TEMP.T0;
NRF_TEMP->T1 = NRF_FICR->TEMP.T1;
NRF_TEMP->T2 = NRF_FICR->TEMP.T2;
NRF_TEMP->T3 = NRF_FICR->TEMP.T3;
NRF_TEMP->T4 = NRF_FICR->TEMP.T4;
```

This code is already present in the latest system_nrf52.c file and in the system_nrf52840.c file released in MDK 8.12.0.

3.6 [68] CLOCK: EVENTS_HFCLKSTARTED can be generated before HFCLK is stable

This anomaly applies to Revision 1, build codes CAAA-A00, CAAA-Ax0.

Symptoms

EVENTS_HFCLKSTARTED may come before HFXO is started.

Conditions

When using a 32 MHz crystal with start-up longer than 400 µs.

Consequences

Performance of radio and peripheral requiring HFXO will be degraded until the crystal is stable.

Workaround

32 MHz crystal oscillator startup time must be verified by the user. If the worst-case startup time is shorter than 400 µs, no workaround is required. If the startup time can be longer than 400 µs, the software must ensure, using a timer, that the crystal has had enough time to start up before using peripherals that require the HFXO. The Radio requires the HFXO to be stable before use. The ADC, TIMERs, and TEMP sensor for example can use the HFXO as a reference for improved accuracy.

3.7 [77] CLOCK: RC oscillator is not calibrated when first started

This anomaly applies to Revision 1, build codes CAAA-A00, CAAA-Ax0.

Symptoms

The LFCLK RC oscillator frequency can have a frequency error of up to -25 to +40% after reset. A +/- 2% error is stated in the Product Specification.

Conditions

Always.

Consequences

The LFCLK RC oscillator frequency is inaccurate.

Workaround

Calibrate the LFCLK RC oscillator before its first use after a reset.

3.8 [78] TIMER: High current consumption when using timer STOP task only

This anomaly applies to Revision 1, build codes CAAA-A00, CAAA-Ax0.

Symptoms

Increased current consumption when the timer has been running and the STOP task is used to stop it.

Conditions

The timer has been running (after triggering a START task) and then it is stopped using a STOP task only.

Consequences

Increased current consumption.

Workaround

Use the SHUTDOWN task after the STOP task or instead of the STOP task.

3.9 [81] GPIO: PIN_CNF is not retained when in debug interface mode

This anomaly applies to Revision 1, build codes CAAA-A00, CAAA-Ax0.

Symptoms

GPIO pin configuration is reset on wakeup from System OFF.

Conditions

The system is in debug interface mode.

Consequences

GPIO state unreliable until PIN_CNF is reconfigured.

3.10 [83] TWIS: STOPPED event occurs twice if the STOP task is triggered during a transaction

This anomaly applies to Revision 1, build codes CAAA-A00, CAAA-Ax0.

Symptoms

STOPPED event is set after clearing it.

Conditions

The STOP task is triggered during a transaction.

Consequences

STOPPED event occurs twice: When the STOP task is fired and when the master issues a stop condition on the bus. This could provoke an extra interrupt or a failure in the TWIS driver.

Workaround

The last STOPPED event must be accounted for in software.

3.11 [88] WDT: Increased current consumption when configured to pause in System ON idle

This anomaly applies to Revision 1, build codes CAAA-A00, CAAA-Ax0.

Symptoms

Using the mode where watchdog is paused in CPU Idle, the current consumption jumps from 3 µA to 400 µA.

Conditions

When we enable WDT with the CONFIG option to pause when CPU sleeps:

```
NRF_WDT->CONFIG = (WDT_CONFIG_SLEEP_Pause<<WDT_CONFIG_SLEEP_Pos);
```

Consequences

Reduced battery life.

Workaround

Do not enter System ON IDLE within 125 µs after reloading the watchdog.

3.12 [136] System: Bits in RESETREAS are set when they should not be

This anomaly applies to Revision 1, build codes CAAA-A00, CAAA-Ax0.

Symptoms

After pin reset, RESETREAS bits other than RESETPIN might also be set.

Conditions

A pin reset has triggered.

Consequences

If the firmware evaluates RESETREAS, it might take the wrong action.

Workaround

When RESETREAS shows a pin reset (RESETPIN), ignore other reset reason bits.

Important: RESETREAS bits must be cleared between resets.

Apply the following code after any reset:

```
if (NRF_POWER->RESETREAS & POWER_RESETREAS_RESETPIN_Msk){
    NRF_POWER->RESETREAS =  ~POWER_RESETREAS_RESETPIN_Msk;
}
```

This workaround is implemented in MDK version 8.13.0 and later.

3.13 [155] GPIOTE: IN event may occur more than once on input edge

This anomaly applies to Revision 1, build codes CAAA-A00, CAAA-Ax0.

Symptoms

IN event occurs more than once on an input edge.

Conditions

Input signal edges are closer together than 1.3 µs or >= 750 kHz for a periodic signal.

Consequences

Tasks connected through PPI or SHORTS to this event might be triggered twice.

Workaround

Apply the following code when any GPIOTE channel is configured to generate an IN event on edges that can occur within 1.3 µs of each other:

```
*(volatile uint32_t *)(NRF_GPIOTE_BASE + 0x600 + (4 * GPIOTE_CH_USED)) = 1;
```

Important: A clock is kept on by the workaround and must be reverted to avoid higher current consumption when GPIOTE is not in use, using the following code:

```
*(volatile uint32_t *)(NRF_GPIOTE_BASE + 0x600 + (4 * GPIOTE_CH_USED)) = 0;
```

3.14 [156] GPIOTE: Some CLR tasks give unintentional behavior

This anomaly applies to Revision 1, build codes CAAA-A00, CAAA-Ax0.

Symptoms

One of the following symptoms may occur:

- Current consumption is high when entering IDLE.
- Latency for detection changes on inputs connected to GPIOTE channels is becoming longer than expected.

Conditions

The following tasks are in use:

Address	GPIOTE task
0x060	TASKS_CLR[0]
0x068	TASKS_CLR[2]
0x070	TASKS_CLR[4]
0x078	TASKS_CLR[6]

Consequences

Experiencing high current consumption during System ON Idle, or too long time from external event to internal triggering of PPI event and/or IRQ from GPIOTE.

Workaround

Instead of using TASKS_CLR[*n*], set CONFIG[*n*].POLARITY to HiToLo and trigger TASKS_OUT[*n*], with *n* = 0, 2, 4, 6.

3.15 [173] GPIO: Writes to LATCH register take several CPU cycles to take effect

This anomaly applies to Revision 1, build codes CAAA-A00, CAAA-Ax0.

Symptoms

A bit in the LATCH register reads '1' even after clearing it by writing '1'.

Conditions

Reading the LATCH register right after writing to it.

Consequences

Old value of the LATCH register is read.

Workaround

Have at least 3 CPU cycles of delay between the write and the subsequent read to the LATCH register. This can be achieved by having 3 dummy reads to the LATCH register.

3.16 [176] System: Flash erase through CTRL-AP fails due to watchdog time-out

This anomaly applies to Revision 1, build codes CAAA-A00, CAAA-Ax0.

Symptoms

Full flash erase through CTRL-AP is not successful.

Conditions

WDT is enabled.

Consequences

Flash is not erased. If the device has a WDT time-out less than 1 ms and is readback-protected through UICR.APPROTECT, there is a risk of permanently preventing the erasing of the flash.

Workaround

Try again.

3.17 [179] RTC: COMPARE event is generated twice from a single RTC compare match

This anomaly applies to Revision 1, build codes CAAA-A00, CAAA-Ax0.

Symptoms

Tasks connected to RTC COMPARE event through PPI are triggered twice per compare match.

Conditions

RTC registers are being accessed by CPU while RTC is running.

Consequences

Tasks connected to RTC COMPARE event through PPI are triggered more often than expected.

Workaround

Do not access the RTC registers, including the COMPARE event register, from CPU while waiting for the RTC COMPARE event. Note that CPU interrupt from this event can still be enabled.

3.18 [184] NVMC: Erase or write operations from the external debugger fail when CPU is not halted

This anomaly applies to Revision 1, build codes CAAA-A00, CAAA-Ax0.

Symptoms

The erase or write operation fails or takes longer time than specified.

Conditions

NVMC erase or write operation initiated using an external debugger. CPU is not halted.

Consequences

The NVMC erase or write operation fails or takes longer time than specified.

Workaround

Halt the CPU by writing to DHCSR (Debug Halting Control and Status Register) before starting NVMC erase or write operation from the external debugger. See the ARM infocenter to get the details of the DHCSR register.

Programming tools provided by Nordic Semiconductor comply with this.

3.19 [210] GPIO: Bits in GPIO LATCH register are incorrectly set to 1

This anomaly applies to Revision 1, build codes CAAA-A00, CAAA-Ax0.

Symptoms

The GPIO.LATCH[n] register is unexpectedly set to 1 (Latched).

Conditions

Set GPIO.PIN_CNF[n].SENSE at low level (3) at the same time as PIN_CNF[n].INPUT is set to Connect (0).

Consequences

The GPIO.LATCH[n] register is set to 1 (Latched). This could have side effects, depending on how the chip is configured to use this LATCH register.

Workaround

Always configure PIN_CNF[n].INPUT before PIN_CNF[n].SENSE.

3.20 [212] SAADC: Events are not generated when switching from scan mode to no-scan mode with burst enabled

This anomaly applies to Revision 1, build codes CAAA-A00, CAAA-Ax0.

Symptoms

SAADC stops working.

Conditions

Any of the following:

- Switching from multiple channels to single channel when BURST is disabled and acquisition time < 10 µs.
- Switching from multiple channels to single channel when BURST is enabled.

Consequences

SAADC does not generate the expected events.

Workaround

Execute the following code before changing the channel configuration:

```
volatile uint32_t temp1;
volatile uint32_t temp2;
volatile uint32_t temp3;

temp1 = *(volatile uint32_t *)0x40007640ul;
temp2 = *(volatile uint32_t *)0x40007644ul;
temp3 = *(volatile uint32_t *)0x40007648ul;

*(volatile uint32_t *)0x40007FFCul = 0ul;
*(volatile uint32_t *)0x40007FFCul;
*(volatile uint32_t *)0x40007FFCul = 1ul;

*(volatile uint32_t *)0x40007640ul = temp1;
*(volatile uint32_t *)0x40007644ul = temp2;
*(volatile uint32_t *)0x40007648ul = temp3;
```

After the workaround is executed, the SAADC configuration is reset. Before use all registers must be configured again.

3.21 [217] RAM: RAM calibration values are not correctly loaded from FICR at reset

This anomaly applies to Revision 1, build codes CAAA-A00, CAAA-Ax0.

Symptoms

RAM calibration value is initialized to the wrong value.

Conditions

Always.

Consequences

Statistical margin to data retention failure when retaining RAM in System OFF is lower than intended for high temperature operating conditions.

Note: This failure has not been reported or reproduced under test at the time of publication.

Workaround

Apply the following code after any reset:

```
*(volatile uint32_t *)0x40000EE4ul |= 0x0000000Ful;
```

This workaround is implemented in nRF MDK v8.25.0 and later.

3.22 [219] TWIM: I2C timing spec is violated at 400 kHz

This anomaly applies to Revision 1, build codes CAAA-A00, CAAA-Ax0.

Symptoms

The low period of the SCL clock is too short to meet the I2C specification at 400 kHz. The actual low period of the SCL clock is 1.25 µs while the I2C specification requires the SCL clock to have a minimum low period of 1.3 µs.

Conditions

Using TWIM at 400 kHz.

Consequences

TWI communication might not work at 400 kHz with I2C compatible devices.

Workaround

If communication does not work at 400 kHz with an I2C compatible device that requires the SCL clock to have a minimum low period of 1.3 µs, use 390 kHz instead of 400kHz by writing 0x06200000 to the FREQUENCY register. With this setting, the SCL low period is greater than 1.3 µs.

3.23 [241] SAADC: Static 400 µA current after SAADC is disabled

This anomaly applies to Revision 1, build codes CAAA-Ax0.

Symptoms

Static current consumption between 400 µA and 450 µA occurs.

Conditions

SAADC is disabled after sampling with BURST when multiple channels have been enabled.

Consequences

Current consumption is higher than expected.

Workaround

Execute the following code after disabling SAADC:

```
volatile uint32_t temp1;
volatile uint32_t temp2;
volatile uint32_t temp3;

temp1 = *(volatile uint32_t *)0x40007640ul;
temp2 = *(volatile uint32_t *)0x40007644ul;
temp3 = *(volatile uint32_t *)0x40007648ul;

*(volatile uint32_t *)0x40007FFCul = 0ul;
*(volatile uint32_t *)0x40007FFCul;
*(volatile uint32_t *)0x40007FFCul = 1ul;

*(volatile uint32_t *)0x40007640ul = temp1;
*(volatile uint32_t *)0x40007644ul = temp2;
*(volatile uint32_t *)0x40007648ul = temp3;
```

After the workaround is executed, the SAADC configuration is reset and all registers must be configured again.

3.24 [242] NVMC: NVMC operations during POFWARN cause the CPU to hang

This anomaly applies to Revision 1, build codes CAAA-A00, CAAA-Ax0.

Symptoms

The CPU hangs.

Conditions

NVMC write or erase operation when POFWARN is asserted, and with low probability when POFWARN is asserted while an NVMC write or erase operation is ongoing.

Consequences

Code execution is halted.

Workaround

Disable POFWARN by writing POFCON before a write or erase operation. Do not attempt to write or erase if EVENTS_POFWARN is already asserted.

3.25 [245] RADIO: CRC is wrong when data whitening is enabled and address field is included in CRC calculation

This anomaly applies to Revision 1, build codes CAAA-A00, CAAA-Ax0.

Symptoms

CRC failures are reported.

Conditions

In RX, if data whitening is enabled and the CRC checker is configured to take the address field into CRC calculations.

Consequences

CRC failures are reported though received packet contents are good.

3.26 [246] System: Intermittent extra current consumption when going to sleep

This anomaly applies to Revision 1, build codes CAAA-A00, CAAA-Ax0.

Symptoms

Extra current consumption in the range of 350 µA when in System On Idle.

Conditions

A high-speed peripheral (CPU, CRYPTOCELL, USB, or CTRL-AP) accesses a RAM block which is being accessed by a low-speed peripheral through the DMA bus with a specific timing, and the high-speed peripheral has higher priority than the low-speed peripheral.

Consequences

Extra current consumption in System On Idle.

Workaround

Apply the following code after any reset:

```
*(volatile uint32_t *)0x4007AC84ul = 0x00000002ul;
```

Workaround consequences: Up to 40 µA current increase when the 16 MHz clock is used.

3.27 [262] POWER: CPU does not resume execution after CPU IDLE

This anomaly applies to Revision 1, build codes CAAA-A00.

Symptoms

CPU does not resume execution.

Conditions

The following conditions are present:

- CPU enters IDLE state between 32 to 48 16 MHz clock cycles before SPIM END event.
- SPIM is the only peripheral preventing System ON IDLE state.
- CPU is configured to resume execution after SPIM END event.
- PPI channels are not configured on the SPIM END event.
- SPIM END_START shortcut is not enabled.
- Constant Latency mode is not enabled.

Consequences

CPU does not execute.

Workaround

Connect the SPIM END event to an available PPI channel. The PPI channel does not have to be connected to a task end point.