



## Power Optimization Guide for ADuCM302x Processors

Contributed by Mahesh Natarajan, Sachin Dwivedi, and Dileep Divakaran

Rev 1 – February 19, 2016

### Introduction

Choosing a low-power MCU can be a tough task, as it involves poring through datasheets to analyze electrical specifications. It is often difficult to relate these numbers to applicable system-level use cases. Evaluating various power modes while considering peripheral operations emulating real use case scenarios is an essential step in choosing the right MCU for a power-sensitive application. Some of the key aspects to evaluate when choosing an MCU for low-power applications include:

- availability of low-power modes and their impact to the ability to retain the contents of SRAM
- power consumption with the RTC running while the rest of the system is in a low-power mode
- wakeup times from low-power modes
- supply voltage range from an application standpoint
- power consumption in active mode
  - core activity – example algorithm processing
  - peripheral activity - DMA operations
  - simultaneous core and peripheral activity
- flexibility in choosing core/peripheral clock frequencies that meet system requirements while keeping the power consumption in check
- hardware DMA blocks that enable the CPU to be in low-power mode during peripheral activity

The ADuCM302x processor is an ultra-low-power integrated mixed-signal microcontroller system for processing, control, and connectivity. The MCU system is based on an ARM® Cortex®-M3 processor, offering up to 33 MIPS of peak performance at 26 MHz combined with a collection of digital peripherals, embedded SRAM and flash memories, and an analog subsystem which provides clocking, reset and power management capability in addition to an ADC subsystem.

The ADuCM302x processor is one of the very few low-power MCUs in the market that offers a cache controller. Programs that repeatedly access the same data or instructions can make effective use of cache memory, thereby reducing the overall power consumption.

The power consumption of an MCU largely depends on two factors, the operating voltage and the frequency at which the system operates. ADuCM302x processors incorporate several power modes that are extremely useful in building battery- or self-powered (energy harvesting) applications.

This EE-note discusses the power modes of the ADuCM302x processor in detail and provides example power measurements for several scenarios, with the intent of helping developers choose the power modes that best fit their low-power application requirements.

## ADuCM302x Processor Power Management

ADuCM302x processors incorporate a highly customizable power management and clocking system that offers a great deal of flexibility to application developers to strike the perfect balance between power and performance. The power management blocks consists of integrated regulators, a clock gating scheme, and switches to offer flexibility to apply to numerous application scenarios.

The power management system consists of:

- an integrated 1.2 V LDO and an optional buck regulator
- integrated power switches for low standby current in hibernate mode
- power gating to reduce leakage in sleep modes
- a power supply monitor with selectable voltage range

### ADuCM302x Processor Power Modes

The power management system provides the following low-power modes:

- Active mode with customized clock gating features
- Flexible Sleep mode with smart peripherals
- Hibernate mode with optional SRAM retention capability
- Shutdown mode with no SRAM retention

Each mode provides a low-power benefit with potential functionality trade-offs. [Table 1](#) summarizes the status of system blocks in each of the low-power modes.

Functional Block	ARM Cortex-M3 Core	BUCK	PERIPHERAL -DMA	HF-XTAL	HFOSC	LFXTAL	PLL	LFOSC	RTC0	RTC1	ADC	SRAM	FLASH
Active Mode	ON	User	User	User	User	User	User	ON	User	User	User	ON	ON
FLEXI Mode	OFF	User	User	User	User	User	User	ON	User	User	User	ON	ON
Hibernate Mode	OFF	OFF	OFF	OFF	OFF	User	OFF	ON	User	User	OFF	ON*	OFF
Shutdown Mode	OFF	OFF	OFF	OFF	OFF	User	OFF	OFF	User	OFF	OFF	OFF	OFF

\*: Retainable SRAM size is configurable

Table 1. Power Mode System Block States



The orange “User” blocks mean that the functional block can be configured to be on or off in the user application code.

#### Active Mode

In Active mode (also called Full-on mode), the ARM Cortex-M3 is active and executes instructions from flash and/or SRAM. All peripherals can be enabled or disabled at the user’s discretion, and active mode power can be enhanced by optimized clock management.

## Power Optimization Options in Active Mode

There are several power-saving options available in active mode.

### Buck Converter

The optional integrated buck converter is a great feature to save power in active mode. The buck converter powers the linear regulator, which powers the digital core domain. It will enter bypass mode once the battery voltage (VBAT) falls below ~2.3V. Once in bypass mode, the buck converter output will follow the input. [Figure 1](#) shows the external circuitry recommended for buck-enabled designs.

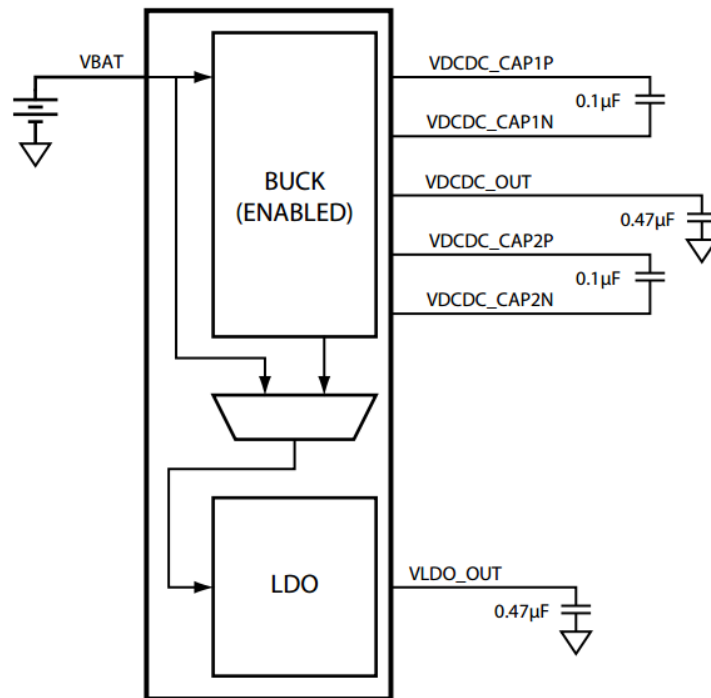


Figure 1. External Circuitry for Buck-Enabled Designs

As mentioned, [Figure 1](#) describes a buck-enabled design. For designs in which the optional buck converter is not used, the VDCDC\_CAP1P, VDCDC\_CAP1N, VDCDC\_OUT, VDCDC\_CAP2P, and VDCDC\_CAP2N pins must be left unconnected.



The buck converter is solely for processor usage. An external load cannot be connected to the buck converter output.

The buck converter can be enabled by setting the `CTL1.HPBUCKEN` bit, per the following code:

```
*pREG_PMG0_CTL1 |= (1<< BITP_PMG_CTL1_HPBUCKEN);
```

[Figure 2](#) compares the power consumption of the ADuCM302x processor computing prime numbers with the following conditions:

- VBAT = 3.0V
- HCLK = PCLK = 26MHz
- Cache disabled

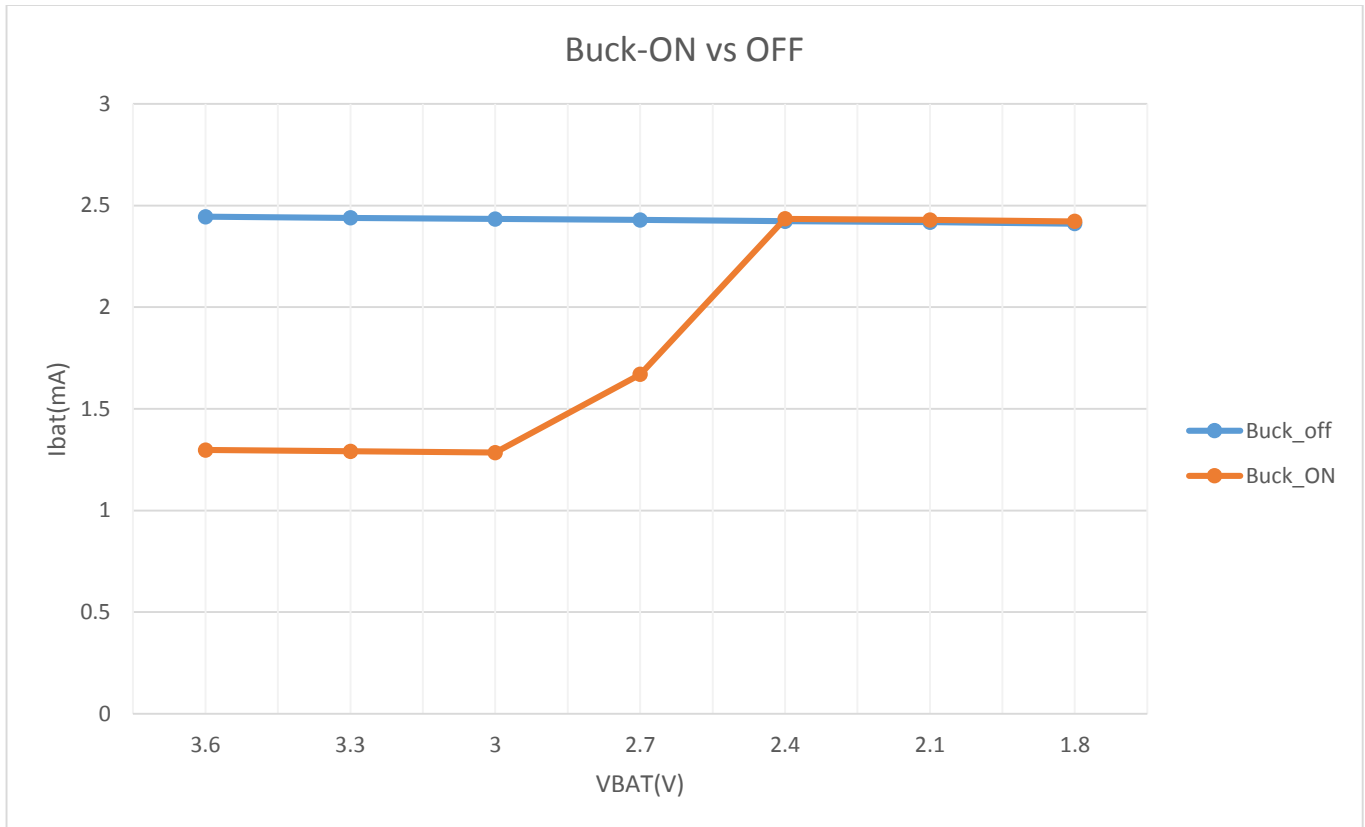


Figure 2. Impact of Buck Converter on Active Mode Power Consumption

As can be seen, the buck converter impacts current consumption positively at higher VBAT values. Specifically, there is roughly a 50% decrease in the active current when VBAT >= 3 V.

### Enabling Cache

Having a cache memory helps reduce the average time to access data from flash memory. For scenarios where the CPU is required to run an algorithm or access the same data repeatedly, having cacheable memory can help reduce the power consumption, as execution is instead from internal instruction SRAM. When the cache controller is enabled, 4 KB of instruction SRAM is reserved as cache memory.

By default, cache memory is disabled at start-up. To enable cache:

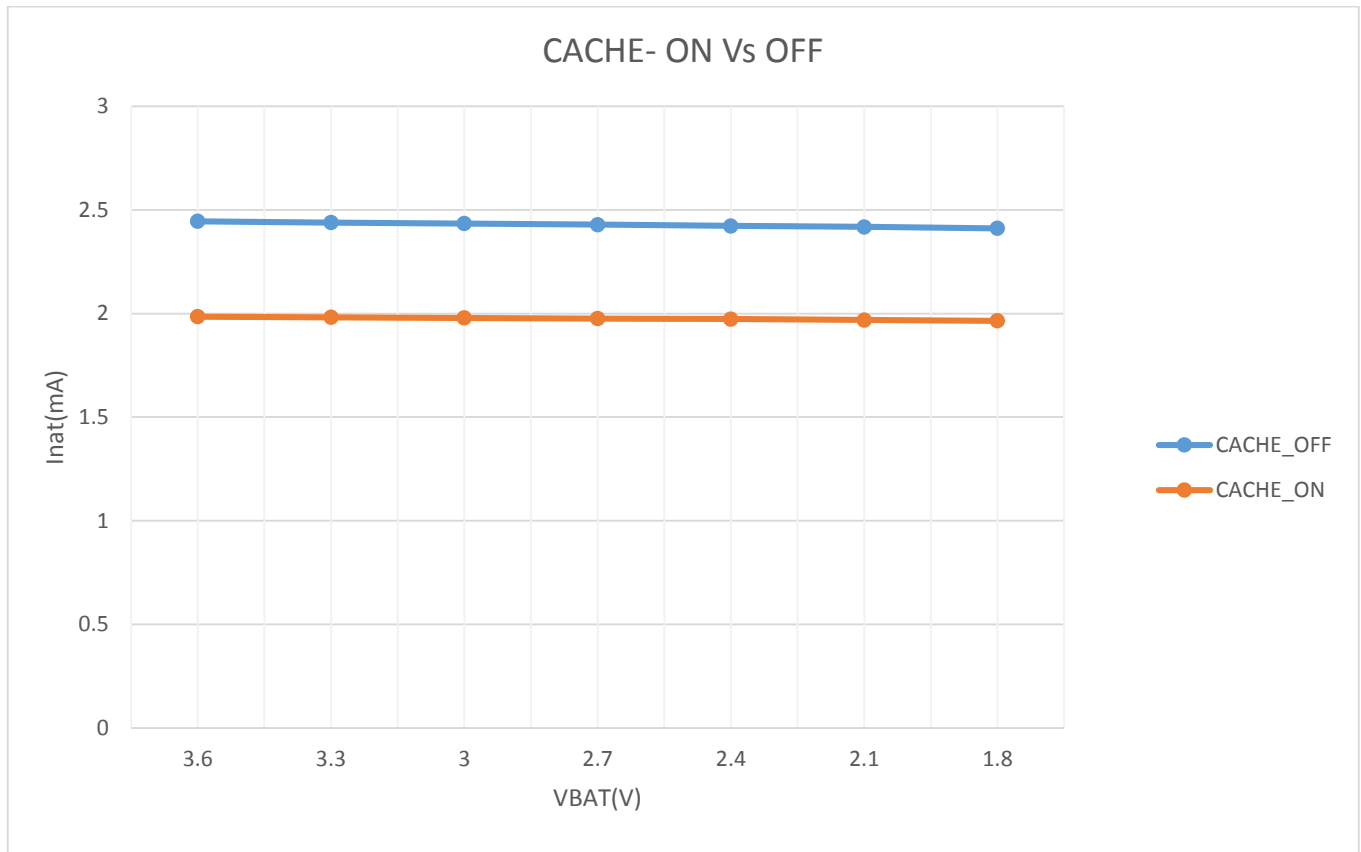
1. Read the `CACHESTAT` register bit 0 (cache enable status bit) to make sure that cache is disabled. Poll this bit until it is cleared.
2. Write the *user key* to the `CACHEKEY` register. For example:  

```
*pREG_FLCC0_CACHE_KEY = 0xF123F456;
```
3. Set the `ICEN` (instruction cache enable) bit in the `CACHESETUP` register:  

```
*pREG_FLCC0_CACHE_SETUP |= (1 << BITP_FLCC_CACHE_SETUP_ICEN);
```

[Figure 3](#) compares the power consumption of the ADuCM302x processor computing prime numbers with the following conditions:

- VBAT = 3.0V
- HCLK = PCLK = 26MHz
- Buck converter disabled



*Figure 3. Impact of Cache on Active Mode Power Consumption*

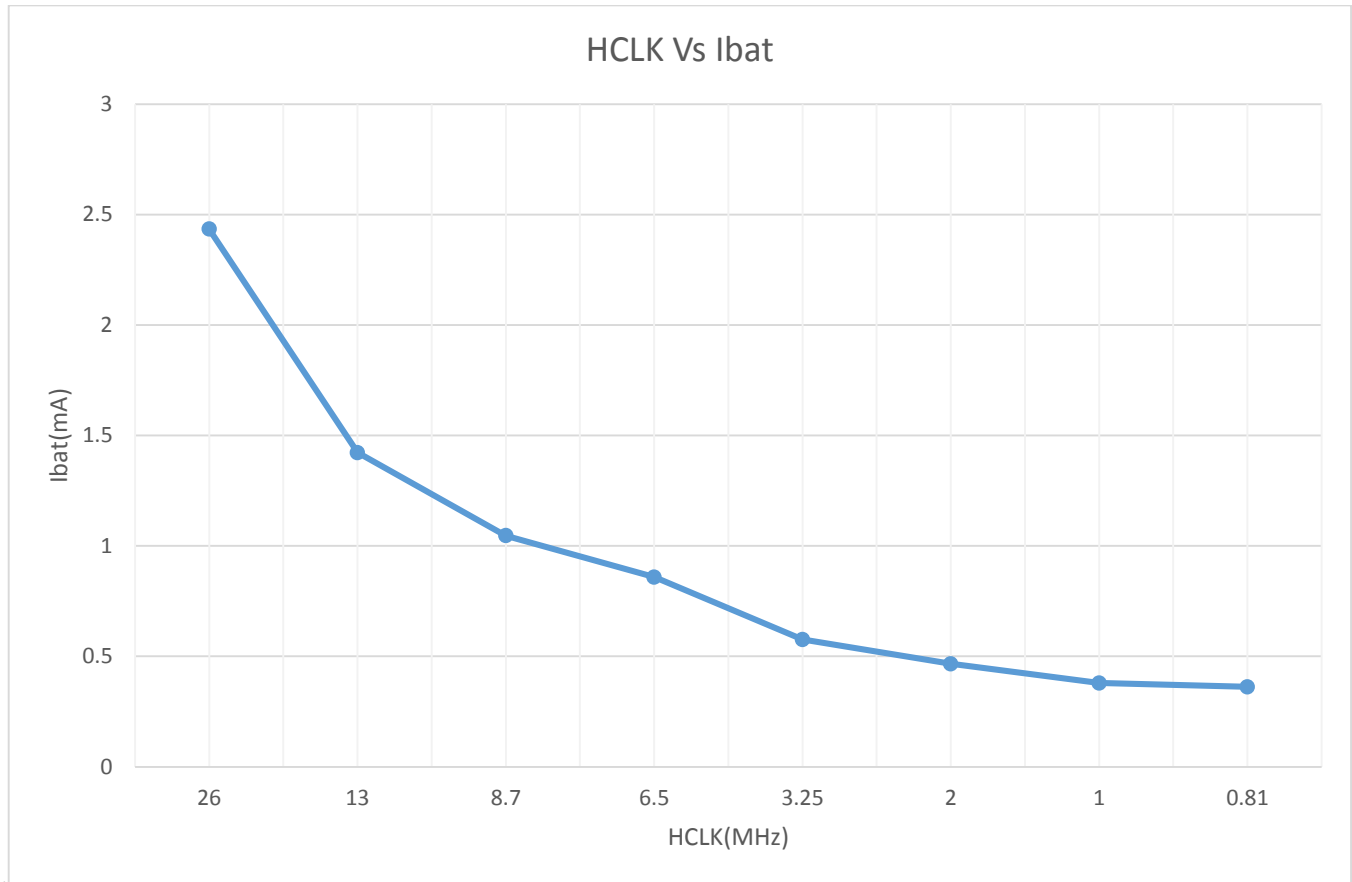
As can be seen, enabling cache reduces the average active current consumption by ~18%.

### *Dynamic Clock Scaling*

Dynamic clock/frequency scaling is a proven method to reduce power consumption. The ADuCM302x processor has a very flexible clock architecture that allows dynamic modification of the CPU and peripheral clock frequencies. A combination of clock dividers and a PLL provide great flexibility in deriving an optimum system clock frequency that guarantees system performance while keeping the power consumption lower as compared to a fixed clock scheme. Programmable clock dividers are available to generate the clocks in the system, and the divisors can be configured on-the-fly.

[Figure 4](#) plots the power consumption of the ADuCM302x processor computing prime numbers with the following conditions:

- VBAT = 3.0V
- HCLK = PCLK (the source of the root clock is HFOSC)
- Buck converter disabled
- Cache disabled



*Figure 4. Impact of Core Clock Frequency on Active Mode Power Consumption*

As can be expected, power dissipation decreases as core clock frequency decreases.

### *Clock Gating*

The system is heavily clock-gated, using automatic clock gating techniques. Most peripherals are automatically clock-gated when the peripheral is disabled, such that the clock is running only when the peripheral is enabled. The exceptions are I2C, GPIO, and the general-purpose timer (GPTMR). These blocks need to be manually clock-gated using the `CLKCON5` register. The peripheral clock can be gated completely by setting the `CLKCON5.PERCLKOFF` bit.



Any access to the clock-gated peripherals will override the clock gate settings in the `CLKCON5` register.

For application scenarios where the core is processing data and no peripheral activity is desired, the peripheral clock (PCLK) can be turned off to save power. [Figure 5](#) shows the power consumption of the ADuCM302x processor computing prime numbers with the following conditions:

- VBAT = 3.0V
- Buck converter disabled
- Cache disabled
- HCLK = PCLK = 26MHz

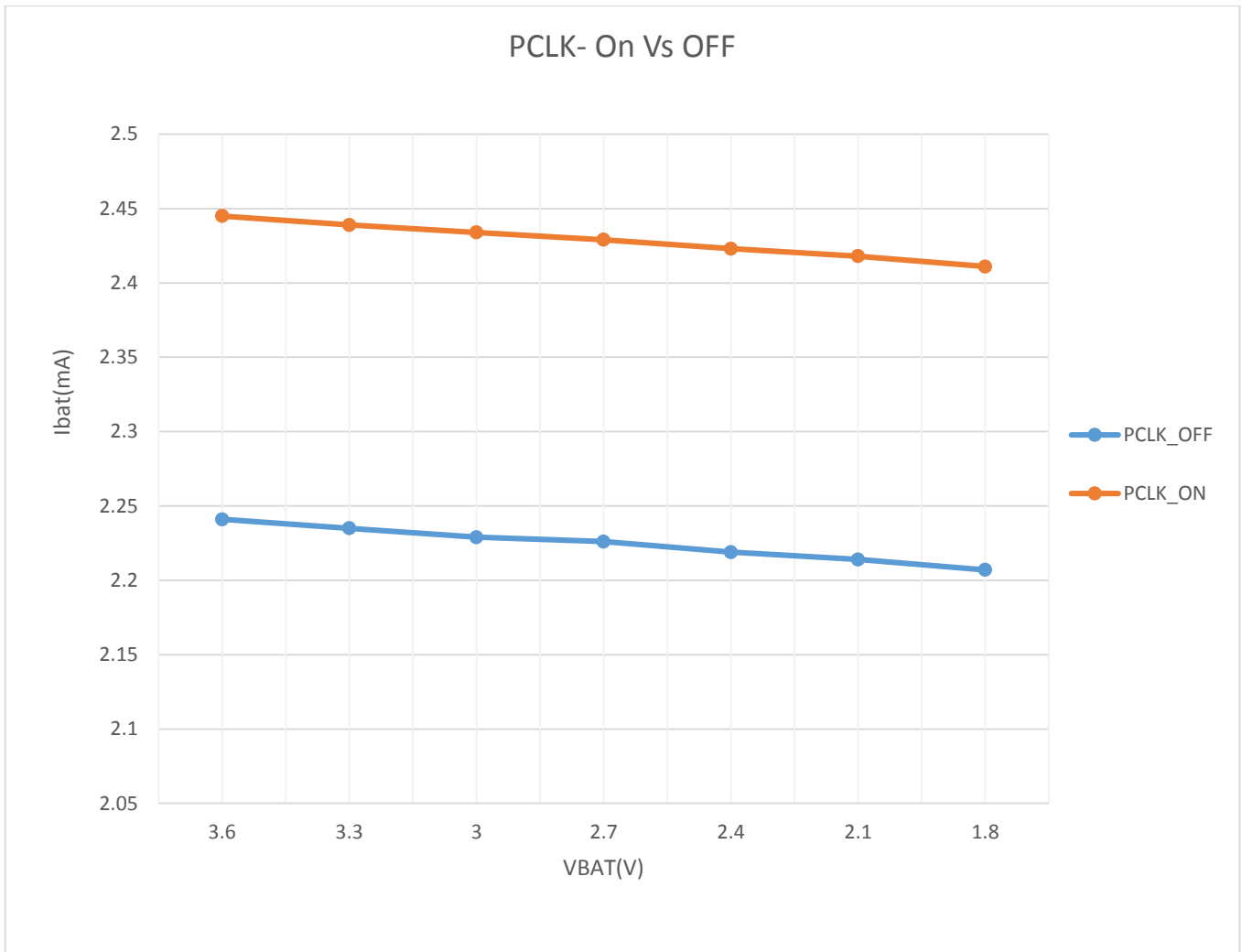


Figure 5. Impact of Peripheral Clock Gating on Active Mode Power Consumption

As shown, a ~0.2mA reduction in the active current is observed when the peripheral clock is gated.



In Active mode, the four techniques described in this section can be combined to achieve maximum power savings.

### Flexible Sleep (Flexi) Mode

Flexible Sleep (Flexi) mode is extremely useful in scenarios where the core has to wait for a peripheral data transfer to complete before it can start processing. In Flexi mode, the core is clock-gated while the remainder of the system is active. This mode can be used to substantially reduce active power when a very-low-speed activity is expected to complete (e.g., reading a certain number of bytes from a sensor) before the processor must be woken up to process the data.

Consider a scenario where the CPU configures a SPI DMA and needs to wait for the DMA to complete. [Figure 6](#) shows the power consumption of the ADuCM302x processor transferring data over SPI using DMA accesses with the following conditions:

- VBAT = 3.0V
- Buck converter disabled
- Cache disabled
- PCLK = 6.5MHz
- SPI\_DIV = 49

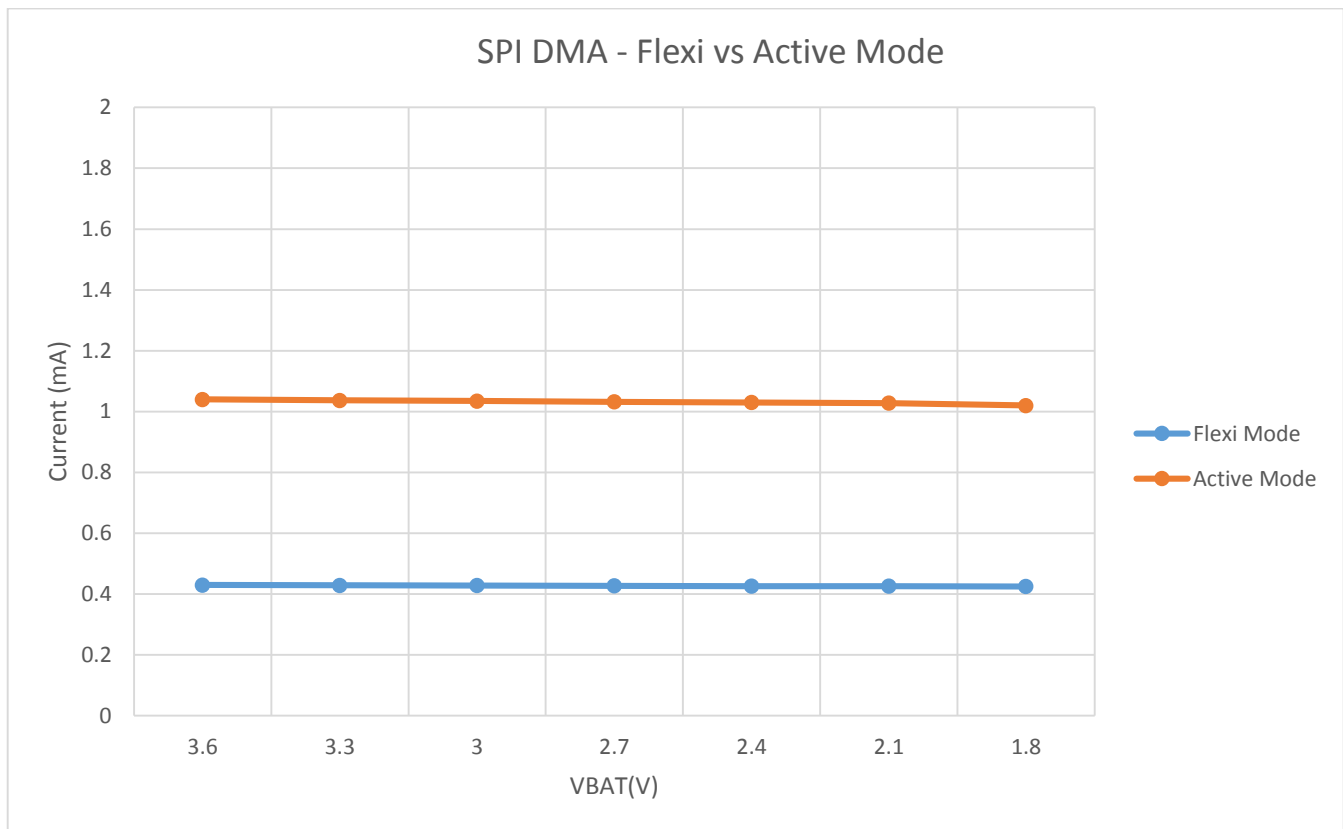


Figure 6. Impact of Flexi Mode on Power Consumption

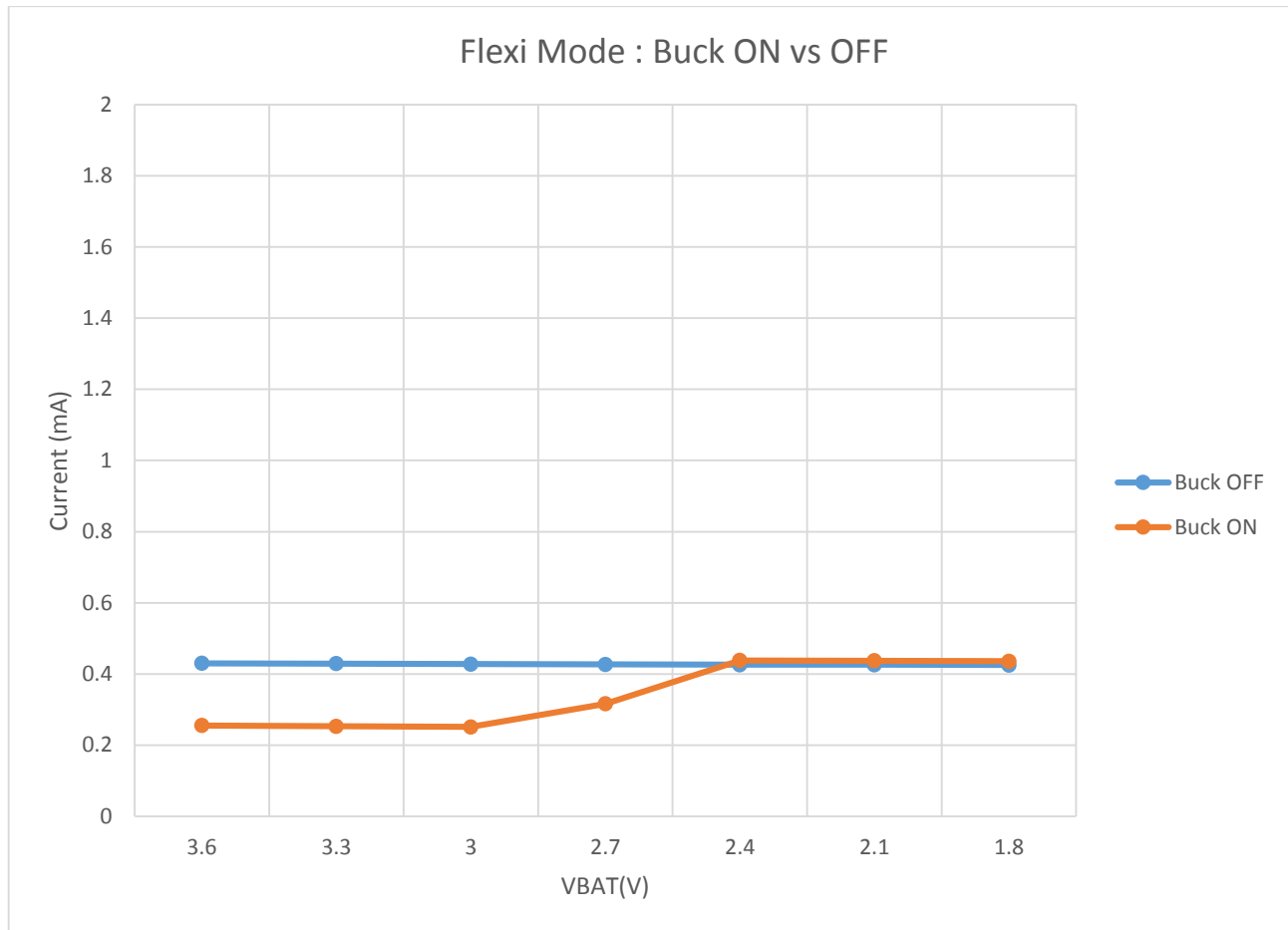
As shown, there is nearly a 66% savings in power when Flexi mode is used while the DMA is ongoing rather than keeping the core in Active mode.

There are a number of wake-up sources that can be used to exit Flexi mode (e.g., DMA interrupts, external interrupts, timer interrupts, etc.), and it typically takes only one CPU clock cycle to exit.



The buck converter can also be enabled in Flexi mode to save even more power. [Figure 7](#) shows the power consumption of the ADuCM302x processor across VBAT in Flexi mode with the buck converter on while transferring data over SPI using DMA accesses with the following conditions:

- VBAT = 3.0V
- Cache disabled
- PCLK = 6.5MHz
- SPI\_DIV = 49



*Figure 7 : Impact of Buck Converter on Flexi Mode Power Consumption*

As can be seen, a similar power improvement pattern to the buck converter’s impact to Active mode can be observed ([Figure 2](#)). Specifically, when VBAT  $\geq$  3 V, a 50% improvement in power is observed.

#### *Hibernate Mode*

In Hibernate mode, the ARM Cortex-M3 core and all the digital peripherals are off with configurable SRAM retention, port pin retention, a limited number of wake-up interrupts, and (optionally) an active RTC. All GPIO pin states are retained in hibernate mode. The ADuCM302x processor also incorporates a very unique feature called Sensorstrobe™ in the RTC block, which enables ultra-low-power sensor data measurement.

Before entering Hibernate mode, most of the peripherals which are enabled must be programmed to undergo a specific sequence to gracefully enter/exit the Hibernate mode, and several System MMRs and peripheral

registers are retained while in Hibernate mode. For more details, refer to the relevant peripheral chapters in the *ADuCM302x Mixed-Signal Control Processor with ARM® Cortex®-M3 and Low-Power Management Hardware Reference Manual*<sup>[1]</sup>.

### Configurable Retainable SRAM

The ADuCM302x processor supports SRAM block sizes of 8 KB (default), 16 KB, 24 KB, or 32KB to be retained while in Hibernate mode. The more SRAM that needs to be retained, the higher the power will be while in Hibernate mode, as shown in [Figure 8](#).

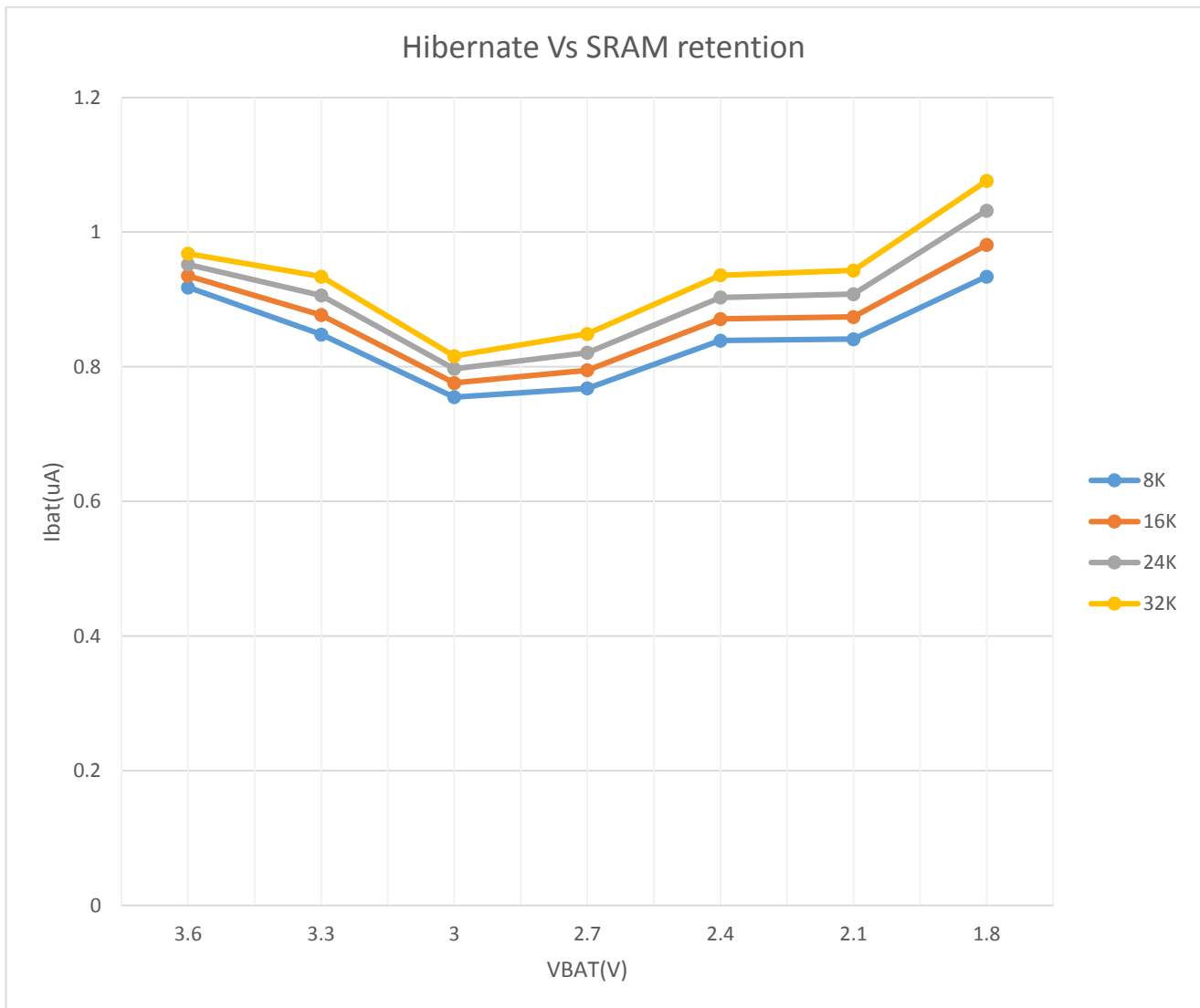


Figure 8. Hibernate Current with Various Sizes of Retained SRAM

SRAM retention size can be configured by setting the appropriate bits in the `SRAMRET` register. For example, to enable 32 KB of SRAM to be retained while in Hibernate mode:

```
*pREG_PMG_PWRKEY = 0x4859;
```

```
*pREG_PMG_SRAMRET |= ((1 << BITP_PMG_SRAMRET_SRAM_RET1_EN) |
                       (1 << BITP_PMG_SRAMRET_SRAM_RET2_EN));
```



If parity is enabled, initialization of non-retained SRAM regions may be required upon waking from Hibernate mode.

### Wakeup Sources

The following events are capable of waking the part from Hibernate mode:

- external interrupts 0-3
- RTC0/1 interrupt
- battery voltage range interrupt
- UART RX pin activity



Of the two real-time clocks, RTC1 is the recommended wake-up source from Hibernate mode, as RTC0 (and only RTC0) can be used for exiting [Shutdown Mode](#), should both modes be used by the application.

The wakeup time from hibernate mode from any of these events is ~10µS.

### RTC Clock Sources

The ADuCM302x processor offers two clock choices for the RTC1 block:

- low-power internal RC oscillator (LFOSC)
- external crystal oscillator (LFXTAL)

The choice of which to implement comes down to a trade-off between accuracy and power consumption. The LFXTAL will be more accurate (depending on the crystal manufacturer) compared to the LFOSC, but the LFOSC will dissipate less power, as depicted in [Figure 9](#).

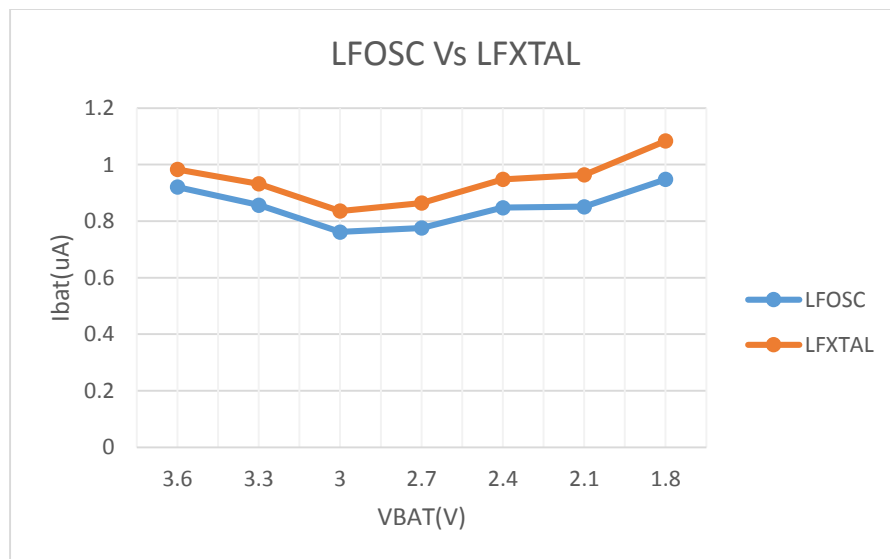


Figure 9. Hibernate Current with RTC1 as Wake-Up Source (LFOSC vs LFXTAL)

### Sensorstrobe

Sensorstrobe allows the ADuCM302x processor to be used as a programmable clock generator in all power modes, including Hibernate mode. In this way, the external sensors can have their timing domains mastered by the ADuCM302x processor, as the OPC can output a programmable divider from FLEX\_RTC, which can operate up to a resolution of 30.7  $\mu$ s. The sensors and microcontroller are synchronous, which removes the need for additional resampling of data to time-align it.

### Shutdown Mode

Shutdown mode is the deepest sleep mode, in which all of the digital and analog circuits are powered down. The state of the digital core and the SRAM memory content are not retained; however, the state of the pads are preserved, as is the wakeup interrupt configuration.

The configuration of the pads is preserved and locked after waking up from Shutdown mode. The user needs to unlock the state of the pads by writing the value 0x58FA to the PGM\_TST\_CLR\_LATCH\_GPIOS register, preferably inside the ISR routine:

```
*pREG_PMG0_TST_CLR_LATCH_GPIOS = 0x58FA;
```

Additionally, the user must configure the appropriate wake-up source, which can be any of:

- external interrupts 0-2
- external reset
- battery falling below 1.6 V
- RTC0 timer

The RTC0 block can (optionally) be enabled in this mode, which allows for the processor to be periodically woken up by the RTC0 interrupt.



The clock source for RTC0 must be the LFXTAL, as the LFOSC is disabled in Shutdown mode.

Because the RTC0 block needs to be powered to serve as a wake-up source, it will add to the power dissipation while in Shutdown mode, as shown in Figure 10.

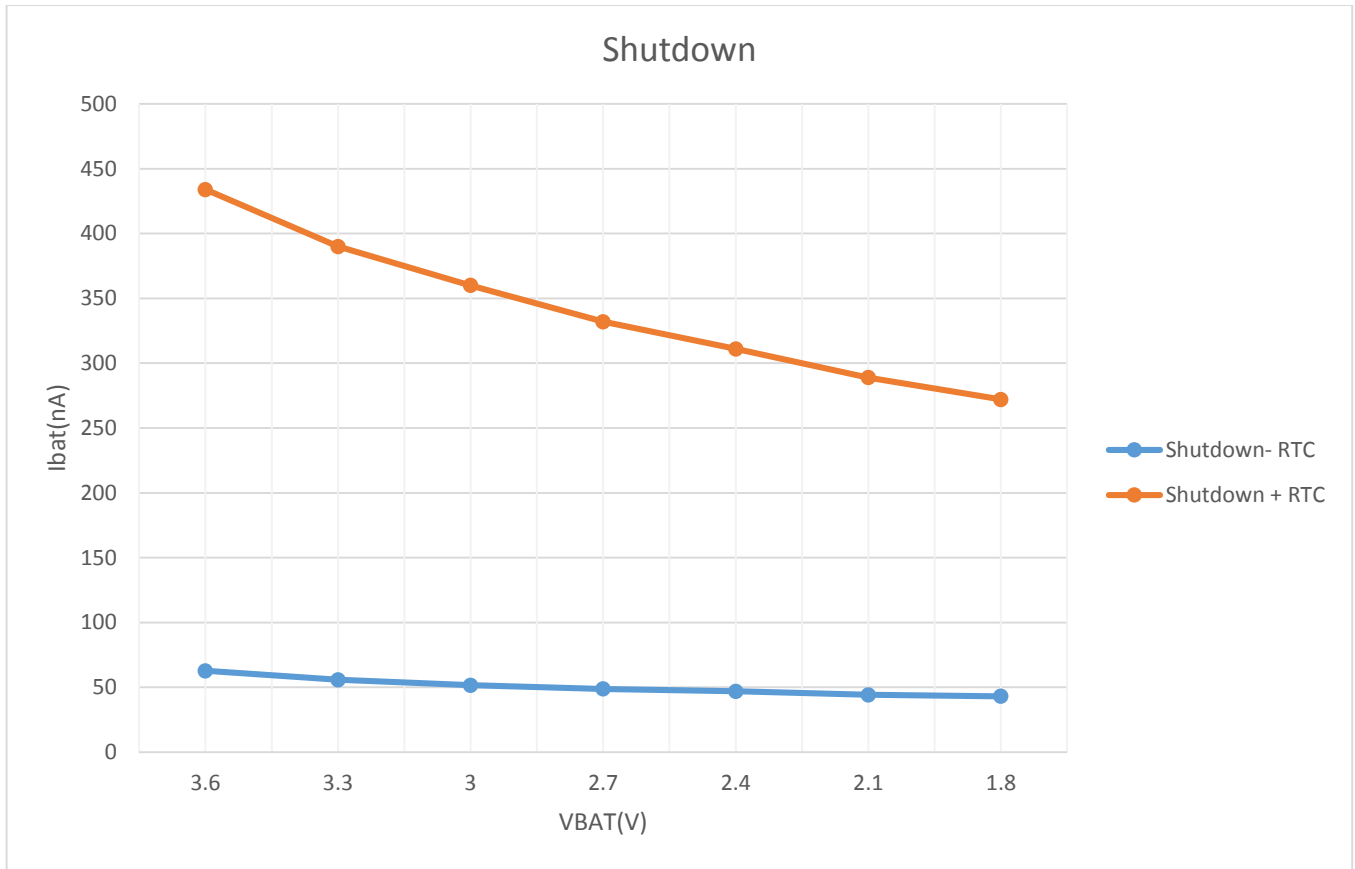


Figure 10. Shutdown Mode Current (External Sources vs RTC0)

When the part wakes up from Shutdown mode, the power-on reset (POR) sequence is followed, and code execution starts from the beginning.

## References

- [1] *ADuCM302x Mixed-Signal Control Processor with ARM® Cortex®-M3 and Low-Power Management Hardware Reference Manual*. Revision 0.3, February 2016. Analog Devices, Inc.
- [2] *ADuCM3027/ADuCM3029 Ultra Low-Power ARM Cortex-M3 MCU with Integrated Power Management Datasheet*. Rev PrF, February 2016. Analog Devices, Inc.

## Document History

Revision	Description
Rev 1 –Feb 19, 2016 by Mahesh Natarajan, Sachin Dwivedi, and Dileep Divakaran	Initial Release.