Use of Low-Power Thread and Zigbee® With the SimpleLink™ Wireless MCU Family

ABSTRACT

This application report discusses the power performance of the SimpleLink™ CC2652 device for Thread and Zigbee® applications. After a brief introduction on the classes of end equipment these technologies enable, Section 2 focuses on the power envelope of a typical battery-powered application and computes an estimated battery lifetime for products powered by the CC2652 device. Section 3 and Section 4 describe other advantages the CC2652 device platform brings to further reduce current consumption and energy waste during the active mode of operations. Section 5 summarizes the low-power benefits of SimpleLink CC2652 platform and the most common applications that can be designed using the device.

Contents

List of Figures

List of Tables

Trademarks

SimpleLink, Texas Instruments, LaunchPad, Code Composer Studio are trademarks of Texas Instruments.
Arm, Cortex are registered trademarks of Arm Limited (or its subsidiaries).
IKEA is a registered trademark of Inter IKEA Systems B.V.
PHILIPS is a registered trademark of Koninklijke Philips N.V.
OSRAM is a registered trademark of OSRAM GmbH.
ENERGY STAR is a registered trademark of U.S. government.
Zigbee is a registered trademark of Zigbee Alliance.
All other trademarks are the property of their respective owners.

1 Introduction

With new functionalities added to their traditional primary role, internet-of-things (IoT) applications are fundamentally transforming the way we use and operate devices. Smart devices are now integrated with a distributed computing networking system, so they process user interactions, generate, and analyze data while always staying connected.

A traditional occupancy sensor is simple compared to a more modern multipurpose sensor that integrates temperature and humidity, dynamically harvests daylight luminance, and sends lighting control commands based on inputs fed from a remote device. Another example of a device evolving past its traditional purpose is in the lighting domain—compare a legacy incandescent light bulb and a more modern connected LED light, with incorporated energy-sensing capability that communicates with other smart light bulbs and products within the same space.

Such transformation in device functionality requires more embedded computational power, storage capability, security, and connectivity functions. In terms of product roles, this paradigm shift inevitably affects the total energy consumed by the devices. IoT technologies and platforms must solve the challenge of providing new-found applications integrated with the same product, while also ensuring the same life expectancy as their legacy implementations.

Battery life is tremendously valued by consumers because it cuts down the bill of materials and the associated costs of battery replacement, while also enabling convenience of use and ease of maintenance. In parallel with the increased environmental concerns, regulatory authorities around the world continue to introduce mandatory power requirements for connected devices when their primary function is off. Therefore, the current consumption of devices that need to stay connected must be tightly controlled.

Low-power performance is a distinctive attribute of IoT products that is pervasive to every facet of the device operation: while in sleep mode, when they are connected and actively exchanging data on the network, or when they are processing data from incorporated sensors or through peripheral interfaces.

The SimpleLink CC2652 wireless MCU addresses product specification from a low-power standpoint for applications such as home and building security, flow metering, environmental monitoring, asset management, and incident detection. The CC2652 device supports a variety of low-power connectivity technologies in the 2.4-GHz spectrum (including Thread and Zigbee), which are offered through the SimpleLink CC26x2 Software Development Kit (SDK).

Thread and Zigbee technologies use mesh network types to provide interoperability and low-power operation benefits between connected devices. These technologies are very popular for smart home and building spaces, grid operation, factory automation, retail automation, and asset tracking.

The CC2652 device from Texas Instruments™ is designed with the lowest power performance in sleep mode, active mode, and during sensor and data processing The CC2652 device harmoniously aligns with the choices of Thread and Zigbee protocols, accentuating the ability of those technologies to provide more environment-aware solutions and extended battery life for products in the IoT.

Figure 1 shows the access flow of the Thread and Zigbee cloud.

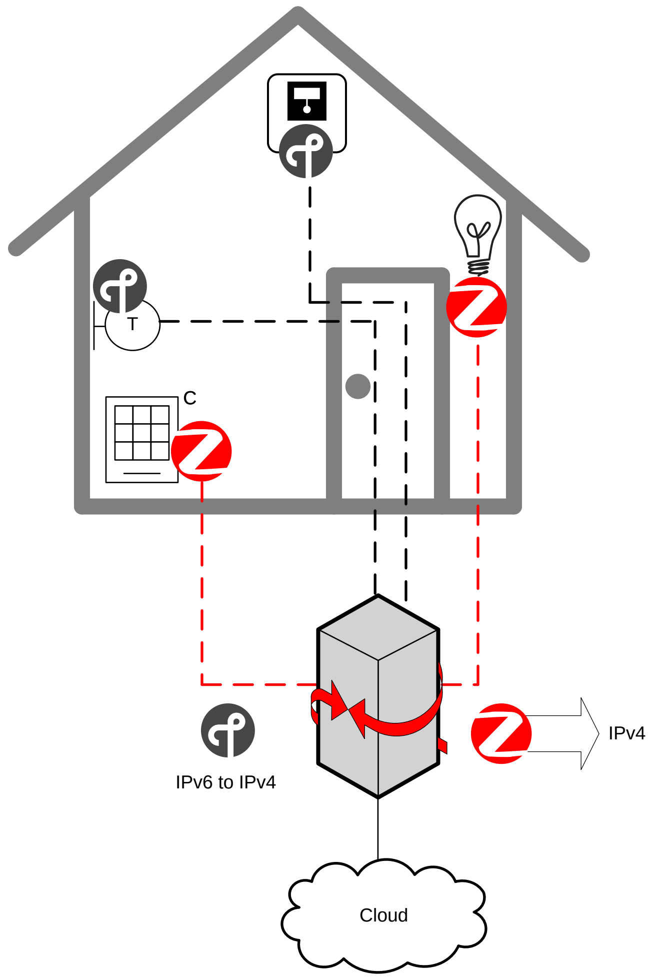

Figure 1. Thread and Zigbee® Cloud Access Flow

2 Power Envelope Minimization

Figure 2 shows the Thread and Zigbee networking protocol architecture.

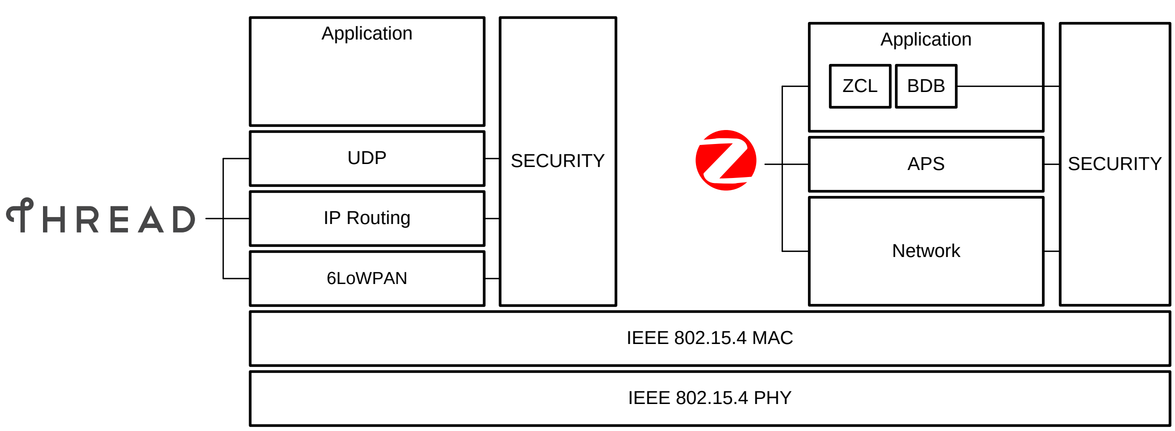

Figure 2. Thread and Zigbee® Protocol Layering

Thread and Zigbee technologies rely on the asynchronous mode of operation within the IEEE 802.15.4 specification implementing the medium access control (MAC) and the physical (PHY) stratums. Asynchronous mode is best suited for battery-operated devices that have their radio off most of the time and generate data in an ad-hoc manner (that is, when an event occurs or in response to a user interaction like a button press).

In 802.15.4 terminology, devices that are mostly sleeping are called reduced function devices (RFD), or informally *end devices*. In the mesh topology scheme, the end devices join and operate in the network through a node that has an always-on receiver and acts as their proxy (in 802.15.4 verbiage, this node is the end device (that is, the *parent*).

Zigbee and Thread have similar networking operations and use the 802.15.4 link the same way. End devices can re-parent to another device if the connection to the original parent is dropped because they only communicate with their one-hop-away parent irrespective of the location of the node for which the application packet they generate is intended.

After the packet emitted by the sleepy device is successfully received by the parent, the sleepy device can immediately go back to sleep, regardless of whether the data has reached the destination. The packet is reliably routed through the distributed network and the mesh fabric while the end device is sleeping. Therefore, power consumption operations are optimized on these sleepy devices. End devices have no need to wake up regularly for housekeeping functions, to maintain a synchronous connection, or to synchronize the device clock with the rest of the network. Patterns of the wake-up cycle are driven only by the actual application traffic profile.

The downlink data flow is handled with short wake-up cycles. In the asynchronous mode of 802.15.4, parents buffer the data for their sleepy end-device children, and the data is extracted by the children when polling their parent. The 802.15.4 poll frames are very short. The parent sends back to its child a closed-loop acknowledgment packet (ACK), which indicates if there is data waiting to be downloaded by simply toggling a single bit in the MAC ACK packet header. This mechanism keeps the over-the-air message exchange short and minimizes the total power consumption. This asynchronous architecture allows flexibility for designers to adjust the frequency of the poll, thus trading off battery life and downlink latency, depending on the needs of the application.

Achieving the best power performance figures hinges on maintaining the total energy consumption at a low level during the active part of the wake-up cycle; therefore minimizing the power envelope (known as the *area under the curve*).

Figure 3 represents a typical sleep and wake-up power envelope of a Thread and Zigbee end device. Corresponding phases are analyzed to show the power implications and the contributions of the protocol configurations as well as the overall platform performance. Use the following sequence to understand the different phases of a wake-up cycle to transmit an 802.15.4 packet.

1. Exit sleep mode.
2. Start and calibrate the crystal oscillator.
3. Initialize the radio.
4. Switch to receive mode.
5. Clear the channel assessment (CCA)
6. Switch to transmit mode.
7. Transmit the IEEE 802.15.4 data frame (polling or application).
8. Switch to receive mode.
9. Receive the IEEE 802.15.4 acknowledgment frame.
10. Enter sleep mode.

Figure 3 shows that the time duration of the phases from 5 to 9 is controlled by the underlying timing of the 802.15.4 symbol rate and the data payload. Thread and Zigbee devices transmit data at 250 kilobits per second, and the size of certain packets (for example, the MAC ACK) or timing of operations (for example, CCA) is fixed by the standard. While the payload size is variable depending on the protocol overhead and application requirements, each byte in the payload adds 32 microseconds of transmit time to the data transmission phase (Step 7).

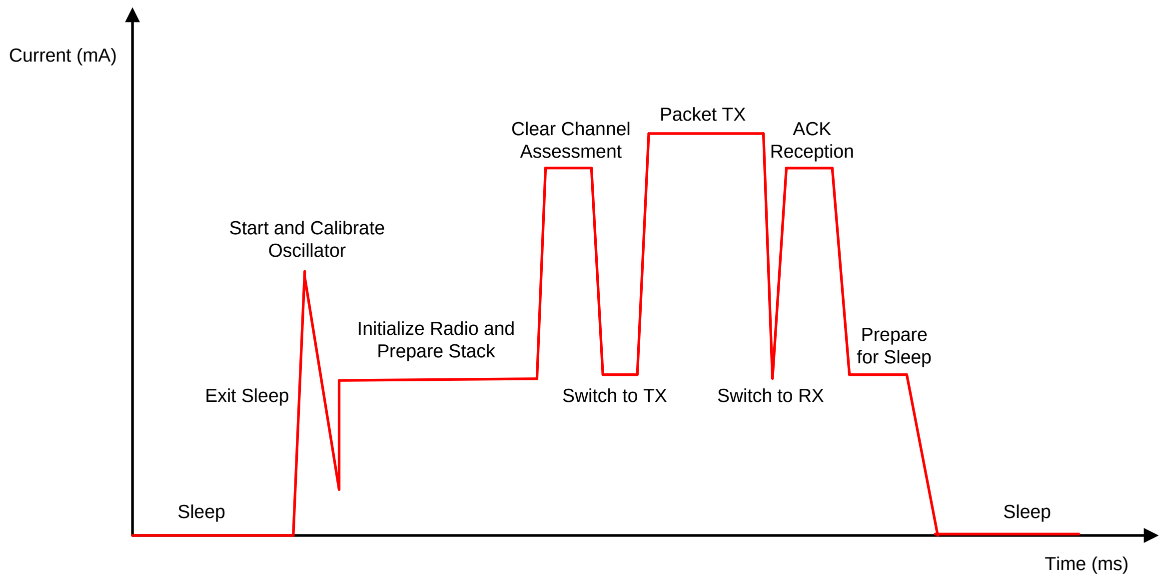

Figure 3. Typical Power Envelope in Thread and Zigbee®

A platform can potentially save the maximum amount of energy based on its ability to squeeze the power envelope, which can be achieved by:

- Reducing the non-protocol related activities (for example, 1 to 4 and 10) to keep the active time very short during those cycles
- Shaving the current peaks when the CPU is on and when the radio is transmitting or receiving

The SimpleLink CC2652 device integrates an Arm® Cortex®-M4F core, which can operate up to 48 MHz at 61 µA/MHz. The optimized SimpleLink MCU TI-RTOS-based power management infrastructure allows a quick operational start after wakeup. Therefore, from a radio standpoint, the device is ready to go within a few hundreds of microseconds while drawing only a few milliamperes before going back to sleep efficiently in few tens of microseconds.

The low-power performance of the SimpleLink CC2652 radio is implemented with a sophisticated and efficient mix of an Arm® Cortex®-M0 processor and a DSP core to accelerate the PHY procedures. This combination largely contributes to the total power envelope savings and scores excellently. With a receive consumption of only 6.5 mA and 6.3 mA of 0-dBm transmission current draw, the CC2652 device sends and receives data with peaks of current within the single digit milliampere consumption figure. Therefore, the designs of Zigbee- and Thread-based products can be entirely powered by coin cell batteries.

Between each active cycle, the device is sleeping. The sensitivity of the overall battery life to the sleep current of the wireless MCU is highly dependent on the use cases. The sensitivity increases when the active cycles are infrequent. This is the case with applications such as flow metering where data from the meter to the grid is reported every several tens of minutes or hours while the device is mostly sleeping.

As mentioned in Section 1, the SimpleLink CC2652 device is designed with a relentless attention for power consumption in all phases of the device. Sleep operations can be carried out below the 1-µA threshold with full 80KB RAM retention. The SimpleLink CC2652 device has excellent standby current performance across the full temperature scale. The device can operate below 1 µA until it reaches 40°C and can operate under the 3-µA threshold from 40°C to 85°C. This operating temperature range is especially important for alarms or security sensors that are mounted outdoors and may be subject to extreme temperature swings (for example, in the attic or in a garage).

Table 1 lists the average current consumption of typical procedures in Zigbee and Thread measured on a SimpleLink CC2652 evaluation module using the SimpleLink software development kit (SDK). The data was collected with devices operating at a transmission power of 5 dBm that were running Thread and Zigbee connectivity solutions from the SimpleLink CC26x2 SDK offering.

Table 1. Current Consumption of SimpleLink CC2652 for Zigbee and Thread Use Cases

Protocol	Polling Every Second	36-Byte Application Payload Sent Every Second	36-Byte Application Payload Sent Every 30 Seconds Combined With Polling Every Second	Estimated Battery Life (200 mAh)
Zigbee	21 µA	39 µA (APS payload)	9.1 µA	2.51 years
Thread	22 µA	37 µA (UDP payload)	9.6 µA	2.38 years

Protocol-related operations and size of the payload affect the overall energy consumption. Due to the intrinsic asynchronous nature of the 802.15.4 MAC and PHY layering, Thread- and Zigbee-based end devices that are battery operated communicate efficiently without exchanging keep-alive messages to maintain the connection with the network. If a parent disappears from the network, the sleepy end device can quickly recover re-parenting through the 802.15.4 MAC orphaning procedure, which completes in only two 802.15.4 messages.

Zigbee and Thread standard design choices that send application data consider the constraints of the underlying physical pipe (both in terms of throughput and maximum payload, which is set to 127 bytes for 802.15.4). Less energy is wasted when there are less active cycles.

Thread and Zigbee are purposely designed to minimize the overhead of the upper networking layers, encapsulating efficient application- and network-management packets.

- Thread uses 6LoWPAN adaptation, which is a standard-based technology that efficiently compresses in-mesh communication Internet Protocol (IP) headers. Thread performs best in terms of bytes reduction when the connectionless User Datagram Protocol (UDP) protocol is on top. Compression factors for UDP and IP headers can lower the packet overhead from 40 bytes to 2 bytes for the in-mesh local communication, which tremendously reduces packet processing time and air time. Link-layer packet forwarding is another important feature of the 6LoWPAN layer where nodes can quickly and efficiently route-forward the packets within the local mesh network without sending the packets to the network layer for further processing. This feature saves central processing unit (CPU) cycles and further improves power consumption.
- Zigbee devices use short 16-bit addresses to communicate with each other. The NWK, APS, Security, and ZCL layers are optimized by design for binary-based usage with packets that are lean and small with up to 79 bytes of application data in a single 802.15.4 packet. Typically, application packets (for example, an alarm notification or a light dimming command) are transmitted as a ZCL payload of a few bytes, enabling savings in terms of air and processing time. The savings reflect in parallel to the overall current consumption.

Table 2 lists the protocol differences between Thread and Zigbee stacks and summarizes the impacts on low-power operations.

Table 2. Thread and Zigbee® Stacks and Low-Power Implications

Layer	Zigbee®	Thread	Power Consumption Impact
Data Link layer	802.15.4 asynchronous	802.15.4 asynchronous	This layer allows for operations using batteries because sleepy devices only wake up to send and receive data.
Network layer	Zigbee mesh routing	IPv6/6LoWPAN	Both are optimized for operations on 802.15.4 with small packets.
Transport layer	APS	UDP	Although Zigbee is connection oriented and Thread is connectionless, they are both designed to create virtual end-to-end links with minimal overhead.
Application layer	ZCL binary	CoAP	Both are designed for embedded and power-constrained devices. ZCL is a binary representation of the cluster library; therefore it efficiently carries the application payload.

3 Power Consumption: The Lighting Use Case

Networks of connected devices can cooperate to produce efficient system energy savings. Let's take again the case of a series of connected light bulbs, with integrated occupancy and luminance sensors and communicating with each other and other class of devices, like environmental sensors and thermostats.

Using the data collected locally on the device and exchanged with the distributed network, current consumption induced by artificial lighting can be reduced by employing daylight harvesting techniques, or lights can be completely turned off when rooms, or hallways leading to them, are identified as vacant or rarely visited. Wireless sensor networks like Zigbee and Thread can efficiently enable sophisticated power management schemes, thus tearing down system costs and reducing broad carbon emissions.

State-run energy commissions or federal government programs (for example, ENERGY STAR®) were created to provide cost-saving, energy-efficient solutions for several categories of end equipment (for example, lighting). These programs set operation requirements for the end equipment. As products become smarter and more connected, the ENERGY STAR program includes energy thresholds for the connected products, which translate into savings at a system level.

The standby power-consumption requirements define the maximum amount of electric power consumed when the primary function of a lighting connected device is switched off. The ENERGY STAR specification indicates that wirelessly connected lamps shall draw no more than 500 mW in standby mode.

Connected lights may keep the receiver on in order to maintain the connection with the rest of the network, be ready to receive commands from actuators to quickly respond, and perform the corresponding operation. Therefore, leveling RF consumption becomes an important factor for ENERGY STAR concerns.

Over the last 5 years, numerous lighting manufacturers and popular consumer brands (for example, PHILIPS®, OSRAM®, and IKEA®) released Zigbee-based light bulbs, luminaires, and fixtures. Lighting has become one of the most popular entry points in the commercial space for the penetration of smart home services.

Thread and Zigbee lighting devices typically function as routers—they must execute a potential incoming actuating command within a certain latency (typically less than 100 ms), and they must also perform the role of routing packets within the network and buffering the messages for their children. As part of its role in the network, the receiver of the lighting device is always on (while the lighting primary function may be off).

The down-voltage conversion and clamping losses affect the overall power consumption of a connected light bulb by draining a few hundreds of mW. The power consumed by the wireless MCU while in receive mode also contributes significantly to power consumption.

The SimpleLink CC2652 wireless MCU has an outstanding current consumption performance of 6.5 mA in receive mode. Consider a 3.6-V operating voltage and a current consumption of 23.4 mW, which in turn represents only approximately 4.7% of the requirement for power budget in standby mode as set forth by ENERGY STAR. Notably, this performance reduces the contributions to the overall standby power from the RF microcontroller, providing an important benefit for ENERGY STAR considerations and reduces the average power drained when the primary function of the connected light is switched off.

4 Power Consumption: The Sensor Processing Use Case

Thread and Zigbee technologies have been adopted in different classes of battery-operated devices for smart home and smart building applications (for example: occupancy, door and window sensors, smoke detectors, and environmental sensors such as temperature and humidity).

As mentioned in Section 1, products are becoming more sophisticated and integrating different sets of functionality (for example, additional peripherals and sensor interfaces). Additional contributions to the total system power consumption come from the leakage on these peripherals and their interaction with the main microcontroller.

The design of the SimpleLink CC2652 MCU strives to minimize the overall current consumption in sensor processing operations. Figure 4 shows the dedicated embedded processing core (the Sensor Controller CPU) of the device, which can be programmed to perform sensor processing and serial-like interfacing (for example: ADC sampling and analysis, threshold comparison, and SPI interfacing) with no need to wake up the main Arm Cortex-M4F core. The Sensor Controller CPU is a 16-bit RISC machine that can use up to 4KB of SRAM for program and data at both 2 MHz and 24 MHz. Therefore, the device allows for fast wake up and processing of peripheral sensors, while also reducing the overall power consumption because it can operate efficiently at 11 µA/MHz.

For example, sampling an ADC input with 12-bit conversion at a one second wake-up cycle can be carried out at a factor of less than 1 µA. The Sensor Controller can also handle more complex tasks (for example, measuring the flow of a water or gas meter at below 2 µA, a 16-Hz measurement). At similar current levels it is possible to read out (through the SPI) a digital sensor at 100 Hz and offer optimized filtering algorithms for applications (for example, fall detect or advanced motion analysis).

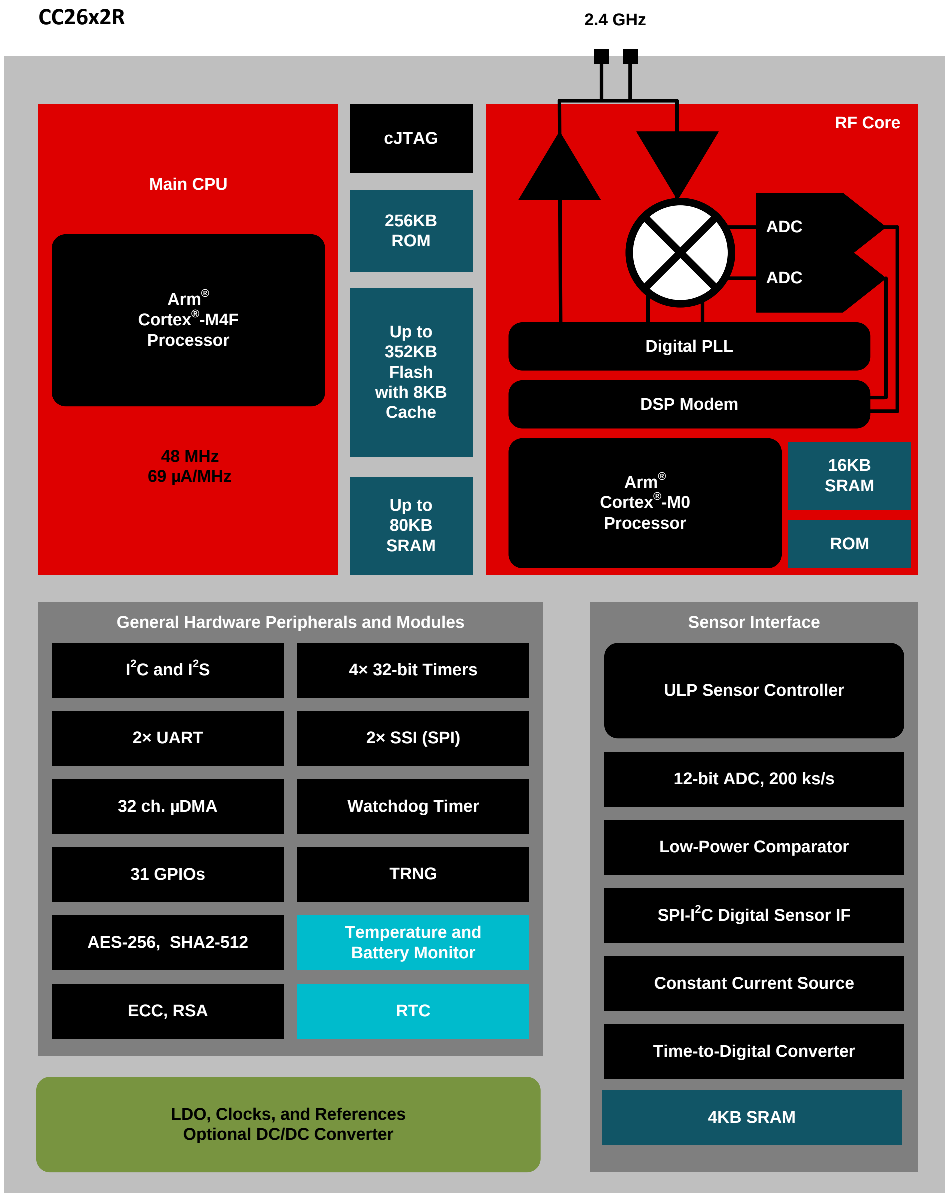

Figure 4. CC2652 Block Diagram With Sensor Controller CPU

The Sensor Controller CPU can be programmed with the SimpleLink Sensor Controller Studio, which is a graphical-user interface (GUI) tool that allows designers to configure the Sensor Controller CPU tasks and generate driver files that can be compiled for the native Arm Cortex-M4F core. The driver files can be loaded onto the Sensor Controller CPU SRAM for direct execution.

5 Conclusions

Thread and Zigbee technologies are used in a wide variety of products on the market today from battery-operated to mains-powered end device networks for lighting, safety and security, environmental sensing, energy management, HVAC systems, and more.

When in active mode the end devices participate in the networks by exchanging data over-the-air, by processing inputs coming from external sensors and peripherals, or when the device is dormant and waiting for an external event to be woken up. Combined with the intrinsic features and architecture of the protocols for IoT products, the SimpleLink CC2652 MCU platform augments the low-power performance capability of the Thread and Zigbee standards and ultimately provides cost savings, ease of maintenance, and greener solutions to the consumers. The TI SimpleLink device platform features the SimpleLink CC2652 silicon and provides a compelling low-power solution for Zigbee and Thread-based applications. This solution strives to achieve the best power performance during all the tasks the device executes across all modes of operation throughout its life cycle.

Thread and Zigbee connectivity stacks are built on top of the SimpleLink MCU with a set of tools and a software development environment that integrates a real-time operating systems (TI-RTOS), which includes power management framework and drivers. Therefore, the CC2652 MCU provides a complete solution for application developers. In terms of power consumption, the device solution is already optimized from the start of the application design.

Table 3 summarizes the feature set of the CC2652 device offering and the performance advantages brought to Thread- and Zigbee-based applications.

Table 3. Use Cases and Power Performance Advantages of the CC2652 SimpleLink™ MCU

Use Case	CC2652 SimpleLink™ MCU Feature	Performance Advantage
Zigbee and Thread temperature sensor	RF current consumption	Multiple years of battery life with typical traffic profile
	CPU current consumption	
	Fast wake up	
	TI-RTOS power management framework wake up and sleep management	
Zigbee and Thread lights	Receive current consumption	Less than 5% ENERGY STAR budget
	TI-RTOS power management framework standby management	
Analogue sensor with sensor processor operations	Sensor Controller CPU	Sampling an ADC input with 12-bit conversion at a one second wake-up cycle can be carried out at less than 1 µA

SimpleLink MCU SDKs come with examples for the connectivity stacks that are provisioned to support sleep operations through the TI-RTOS power management framework and are optimized accordingly.

With the accompanying set of tools such as Sensor Controller Studio and ENERGY TRACE (natively supported with the TI LaunchPad™ Development Kit and the TI Code Composer Studio™ IDE), flexibility is given to developers to analyze, configure, and tweak the application for the lowest power-consumption figures.

Reducing energy costs and carbon footprints are fundamental concerns for IoT applications, and designers are looking for technologies and platforms that strive to achieve the best power-consumption figures. When it comes to low power, the performance, completeness, and flexibility of the TI SimpleLink platform is unparalleled in the industry.