AN108

Measuring the Power Consumption on CC2530ZNP Using CC2530 ZNP Mini Kit

By Suyash Jain & Morten Braathen

Keywords

- *ZigBee*
- *CC2530ZNP*
- *MSP430*
- *ZNP Mini Kit*
- *Application Acknowledgement*
- *Current Consumption*
- *Battery Lifetime*
- *Periodic Transmission*
- *Sensor Monitor Network Sample Application*

1. Introduction

To design a ZigBee product in a short time frame, use the ZigBee Network Processor, CC2530ZNP. The CC2530ZNP has the Z-Stack[TM] [1] loaded in the CC2530 [2] and requires a simple microcontroller to run the application code. It provides a simple, off-the-shelf ZigBee solution without requiring designers to learn the complexities of a full ZigBee stack. The CC2530ZNP also allows developers the flexibility to use their existing host processor to run the application code while the CC2530ZNP supports the networking component of the system.

A set of API functions described in the ZNP Interface specification document [3] forms the communication interface to the CC2530ZNP. With just a few commands required to operate the chip, the developer can stay focused on the application rather than the network protocol.

CC2530ZNP Mini Kit [4] is a miniaturized hardware platform demonstrating ZigBee embedded networking and low power consumption for a node with an MSP430F2274 and a CC2530.

This document will present power consumption measurements and battery lifetime calculations based on the CC2530ZNP Mini Kit running the Sensor Monitor Network Sample Application, where the CC2530 ZNP is configured as an End Device. The Sensor Monitor Network Sample Application implements a manufacturer specific ZigBee application profile with two end point types: Source and Sink. The Sink is mapped to the logical device type Coordinator and the Source is mapped to the logical device type: End Device or Router. The Sink receives temperature and voltage samples sent periodically from the Sources. In between the data transmissions, the End Device goes to sleep to save power.

All of the measurements are done on the CC2530ZNP Mini Kit, with only one Sink and one Source in the ZigBee network.

This document also discusses additional factors and configuration options in the application that influence the battery lifetime of the system.

Note that there are many factors that influence the overall power consumption and that the results presented in this document should only be regarded as indicative for what is possible to achieve in systems with similar hardware.

2. Abbreviations and Acronyms

ACK	Acknowledgement
ADC	Analog to Digital Converter
APS	Application Support Sub-Layer
CSMA-CA	Carrier Sense Multiple Access, Collision Avoidance
DCO	Digitally Controlled Oscillator
GPIO	General Purpose IO
IO	Input Output
MAC	Medium Access Control
OTA	Over the Air
RX	Receive
TX	Transmit
USB	Universal Serial Bus
VCP	Virtual Communication Port
VLO	Very Low Power Oscillator
ZASA	ZigBee Accelerator Sample Application
I	Current (Ampere, A)
V	Voltage (Volt, V)
R	Resistance (Ohm, Ω)
mA	Milliampere (1e-3 A)
uA	Microampere (1e-6 A)
nA	Nanoampere (1e-9 A)

Table of Contents

3. System Overview

3.1 CC2530 ZNP Mini Target board

The CC2530ZNP Mini Development Kit is designed so that developers can understand the ZigBee Network Processor concept without having to port a lot of software to get up and running. The ZNP allows adding ZigBee interface to any application using a simple serial interface.

(a) (b) (c)

Figure 1: ZNP Mini Kit (a) ZNP Mini Kit Target board (b) eZ430 USB Stick with target board (c) ZNP Mini Kit Battery board

The CC2530 ZNP Mini Kit platform provides debugging capabilities of the MSP430 and a Virtual Communication Port (VCP) over the USB interface. The key features of the platform are:

- MSP430 debugging capability
- Virtual serial communications port over USB
- Exposed MSP430 General Purpose IO for peripheral connectivity
- User Interface capabilities (LEDs, button and light sensor)

The CC2530 Target Board can either be connected to the USB emulator (for programming and UART communication) or it can be connected to the supplied battery boards.

For more specific information, see the CC2530 ZNP Mini Kit Wiki resource guide [5].

3.2 Software

The software used in the following experiments and analyses is available in source code format on the CC2530 ZNP Mini Kit product web page. For an in-depth description of the software functionality, please consult the sensor network monitor sample application document included with sensor monitor network sample application install [6].

.

The software can be modified and debugged using the IAR Embedded Workbench for MSP430 or Code Composer Studio.

4. Measurement Setup and Theory

4.1 Instrumentation

The general idea of the current consumption measurement is to visualize the current profile on an oscilloscope by measuring the voltage drop across a fixed resistor. The set up is illustrated in the Figure 2.

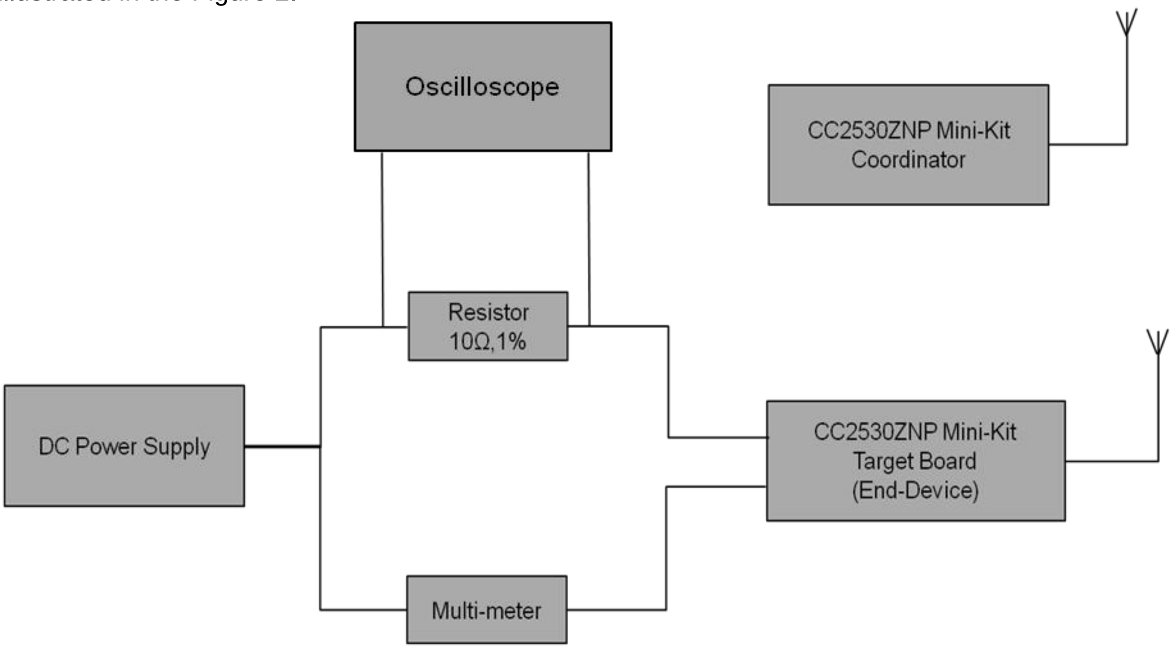

Figure 2: Measurement Setup

The oscilloscope will provide a graphical representation of the voltage drop across the resistor. Since there is a linear relationship between the voltage and current, the same graphical representation will illustrate the current consumed by the system.

In addition to the oscilloscope, a multi-meter is used measure, e.g., the current during sleep and other static states. The multi-meter is connected to the CC2530 ZNP Mini Kit Target Board to measure the current drawn by the MSP430 alone, the CC2530 alone or both.

4.2 Calculation of the average current consumption

The calculation of the current is based on the well known relation

$$V = I * R$$

Where V is the voltage, R is the resistance and I is the current. By measuring at the power supply side, the test system will observe the current consumed by the complete target board. Note that R should not be too large, since it will reduce the effective voltage over the target board itself, i.e.

$$V_{Target\ Board} = V_{Power\ Supply} - (I * R)$$

As long as I is just in the range of some tens of mA and R is relatively small, the overall effect of the voltage drop is negligible. Generally, we can say that

$$V_{Target\ Board} = V_{Power\ Supply} \quad if\ and\ only\ if \quad (I * R) \ll V_{Power\ Supply}$$

$$(I * R) \ll V_{Power\ Supply} \quad if\ and\ only\ if \quad V_{Power\ Supply} > 10 * (I * R)$$

Using R = 10Ω, it is determined that the above is true as long as I < 30 mA. In the subsequent sections, the user will see that this document is pushing the limit saying that the effect of R is negligible, since the peak current sometimes exceeds 30 mA. However, the current consumption of both the MSP430 and CC2530 are almost independent of the input voltage. In addition, if choosing a smaller value for R, the oscilloscope would need to capture accurately even lower voltage levels and the uncertainty in the measured values would increase. In order to keep the measurement system simple and easy to reproduce, the error introduced by the resistor is accepted.

Once the current is determined, the overall average current consumption can be found using the general formula -

$$I_{avg} = \sum_{i=0}^{n} \left(\frac{T_i}{P_i} * I_i\right) + \left(1 - \sum_{i=0}^{n} \left(\frac{T_i}{P_i}\right)\right) * I_{Sleep} \qquad ..(1)$$

$Where,$

$T_i = Time\ for\ which\ device\ consumes\ average\ current\ I_i$

$P_i = Total\ Time\ period\ for\ which\ average\ consumption\ is\ measured$

$I_{Sleep} = Current\ consumption\ while\ in\ sleep\ mode$

$I_{avg} = Average\ current\ consumption\ over\ period\ P_i$

Knowing $I_{i,}\ I_{Sleep,}$, T_i we can find I_{avg} based on the period of active sequences..

As final step, calculate the total life time of the system, knowing that,

$$\frac{Battery\ Capacity\ [mAh]}{Average\ Current\ [mA]} = Lifetime\ [h] \qquad ..(2)$$

The battery capacity will differ from one battery type to another. For normal AAA batteries available in stores, the capacity is typically 1200 mAh. Using AA batteries, capacities of up to 3000 mAh are possible.

In the following sections, assume that two AAA batteries are used.

4.3 Accuracy

To help determine the accuracy of the measurements, the system was set up to switch between three states consuming different amount of current. For the reference, an ampere meter was used to measure the static current draw in the states. The oscilloscope was used to measure the voltage across the resistor in each case, and then the readings were compared.

The CC2530 ZNP Mini Kit Target Board was programmed such that the CC2530 was set in Low Power Mode 3. The MSP430 toggled either the RED LED (State 2) or both LEDS (State 3) once every 5 seconds, using the VLO as clock source for an internal timer. The MSP enters LPM3 (State 1) after setting the LEDs.

With a calibrated ampere meter, we measured current in three states as:

State 1: 4.8uA
State 2: 3.38mA
State 3: 5.86mA

Using the Oscilloscope, the peak voltages in states 2 and 3, averaged over 8 samples, were measured to be:

State 2: 36.11mV
State 3: 63.61mV

Voltage measurements were done over a 10.26 Ω resistor; use this value to calculate the current. Table 1 summarizes the calculations

Description	***Ampere Meter***	***Oscilloscope***	***Diff Max***	***Diff Min***	***%Max***	***%Min***
State 1	4.8uA	-	-	-	-	-
State 2	3.32 mA	3.52 +/- 0.1 mA	0.3	0.1	9.2%	3.2%
State 3	5.86 mA	6.2+/- 0.1 mA	0.44	0.24	7.5%	4%

Table 1: Measurement System Characterization Results

As observed there is an offset in measurements between the ammeter and oscilloscope. In the following sections we will take this into account. However, when dimensioning a system it is always a good rule to add some margin to make sure that the values used will meet or exceed the final requirements. Thus many of the values are rounded to get approximate, yet representative numbers.

5. Current Consumption Measurements

5.1 Low Power Configuration of Sensor Monitor Network Application

An end-device operating as the data source in the sample application was configured as follows for low power consumption:

Behavior	Corresponding Constant	Value
No blinking of LEDs	APP_BLINK_LEDS	FALSE
Application layer acknowledgement is turned off	APP_DATA_CNF	FALSE
Send an application layer packet every 10 seconds	APP_REPORT_INTERVAL	10
Data polling is turned off	APP_POLL_INTERVAL	0

Table 2: Configuration Settings for the End-Device Low Power Configuration

When the nodes are up and running in the small network, a packet sniffer [7] can be used to visualize the packets going over the air. The figure below shows two data packets being sent. Each packet is acknowledged with a MAC ACK packet.

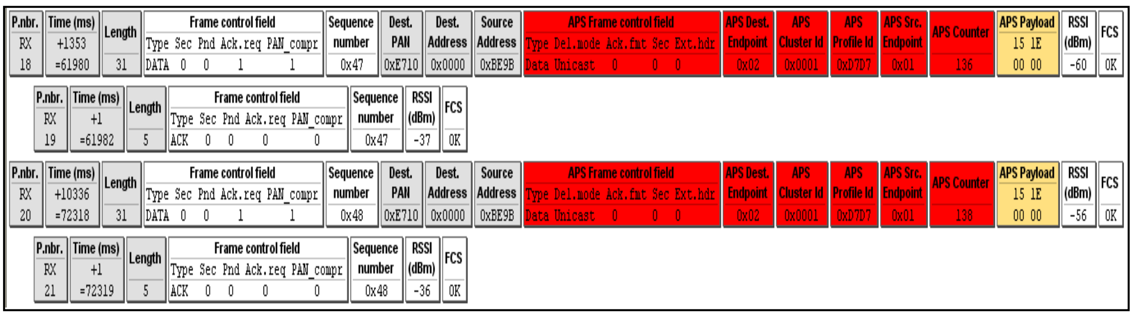

Figure 3: Transmission of data without Application Acknowledgement (2 Packets)

Using the oscilloscope, we can get a picture of the dynamic power consumption of the system during transmission of one of these packets.

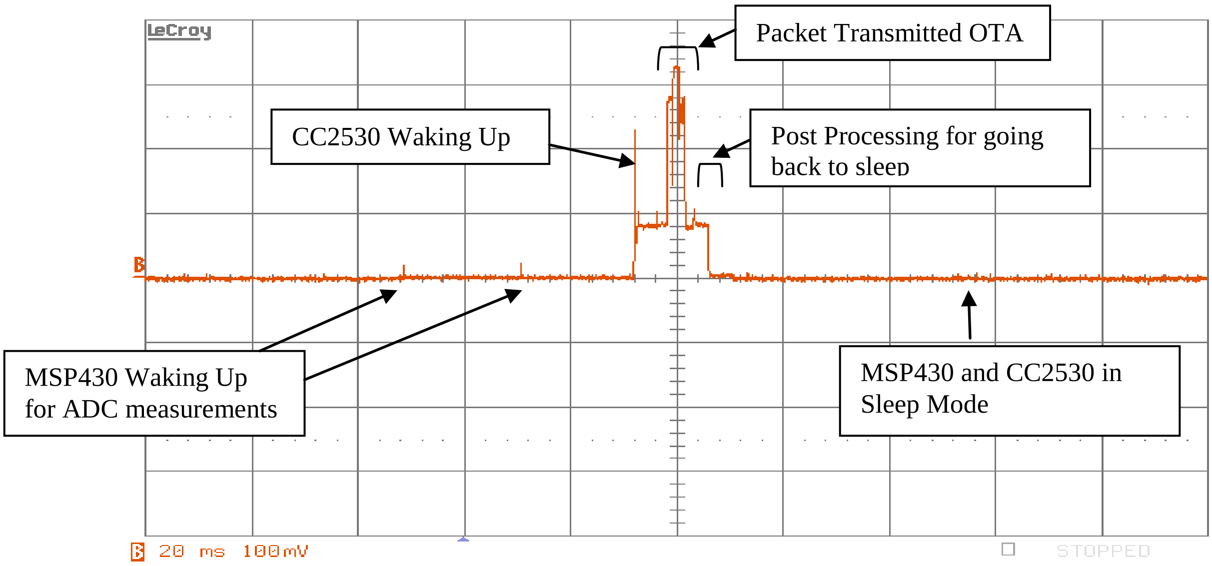

Figure 4: Transmission of data without Application Acknowledgement

Several events are shown in Figure 4.

The two small peaks before the large peak in the centre of the figure are the MSP430 waking up, starting the ADC for doing measurements. First, the temperature is measured. Next the on-board voltage level is measured. The time between the measurements is determined by the worst case start-up and settling time of the reference voltage for the ADC. This has been set to 20 ms in the sensor monitor demo application. The measurement itself only takes a couple of milliseconds. Note that once the ADC is turned on, the MSP430 core will be shut off, reducing power consumption to a minimum. The average current consumption during the 44 milliseconds used for measurements can be determined based on numbers from the MSP430F2274 data sheet.

The large peaks in the centre of the picture represent the CC2530 waking up, starting its local oscillators, crystals and timers. Next, the packet is prepared and sent over the air. At the end, the CC2530 prepares for going back down to deep sleep.

The major events, with approximate current and duration based on averaging values from several measurements, can be summarized in the table below. All of the numbers are typical. Some of the events, like CSMA-CA before beginning the data transfer might take a short time (down to 320 us) or a long time (several tens of ms) - depending on how much traffic is present on the selected channel. For details, see section 7.

Id	Description	Duration [ms]	Current [mA]	Power [ms*mA]
1	ADC Conversion (average for two samples, including waking up)	44	0.170	7.48
2	Synchronizing clocks and processing of data packet	6	8.5	51
3	CSMA-CA (RX), transmit packet and receive MAC ACK	3.5	28.9	101.15
4	Post processing and preparing for deep sleep	5	8.5	42.5
	Sum	58.5		202.13
	Average		3.46	

Table 3: Measured Current Consumption during TX without APS ACK (typical values)

To calculate the average current consumption for the system over time, we also need to find the current consumption when the system is in deep sleep mode. Using an ampere meter, the sleep current of the system was measured to be 4.8uA.

Now proceed to find the total average current consumption, based on (1), for the 10 seconds (= 10,000 ms) packet interval from the source to the sink.

Substituting in the formula values from Table 1 provides:

$$\left(\frac{58.5}{10000} * 3.46\right) + \left(1 - \frac{58.5}{10000}\right) * 0.0048 \text{ mA} = 0.0249\text{mA}$$

(2) can now be used to calculate the expected lifetime of the system:

$$\frac{1200 \text{ [mAh]}}{0.0249 \text{ mA}} = 48{,}024 \text{ hrs} = 2001 \text{ days}$$

If the end device is configured to transmit one packet every 10 seconds, with no application acknowledgement and no data polling, the board can operate for 2001 days (5.48 years) with two AAA batteries.

In the following sections, this document covers the various configuration options in Sensor Monitor Network Application influence the power consumption.

5.2 Application Acknowledgment

Depending on the reliability requirements of the system, the application acknowledgement (APS ACK) can be enabled or disabled. An APS ACK is the reply from the receiver of an application layer packet, saying that the packet was received by the recipient with no errors. Using APS ACK is the only way a ZigBee node can be sure that the packet it sent was received by the intended recipient. The MAC ACK, on the other hand, only indicates that the raw MAC packet was received correctly by a node one hop away in the network. In an alarm and security system or a sensor network monitoring critical parameters, it might be crucial that all transmitted packets are received by the recipient. However, in a non-critical system, losing a packet might not matter.

The Sensor Monitor Network Sample Application has the option to turn on and off the application acknowledgement with the constant APP_DATA_CNF in the file application_configuration.h. Setting it to TRUE will turn the APS ACK on.

Note that, in the Sensor Monitor Network Sample Application, setting APP_DATA_CNF to true will also enable the response poll rate. When an end-device sends a data packet, it can poll again with a shorter duration, specified by this parameter, if the application is expecting to receive an application level packet in response. This value is specified in milliseconds. In the sample application when APP_DATA_CNF is set to false the response poll rate is set to zero to save power. In the following, the user will see the effect this has on the total system power consumption.

Section 5 discusses the current consumption of the system when the application acknowledgement was turned off. Turning it on will change things quite a bit, since the active period will look quite different. In this case, the end device will have to request the acknowledgement packet from the receiver after having transmitted the packet. It must also wait to send the request until the recipient has actually received the packet. The time to wait will depend heavily on the system and number of hops. The default wait time is set to 100ms in the Sensor Monitor Network Application. After receiving the acknowledgement (as a response to the request), the end device will send a last data request in order to check whether additional packets are on their way to the node. If no packets are received, the device will enter deep sleep. The sequence described above can be visualized in a packet sniffer by capturing the packets going over the air. As many as 8 packets are involved.

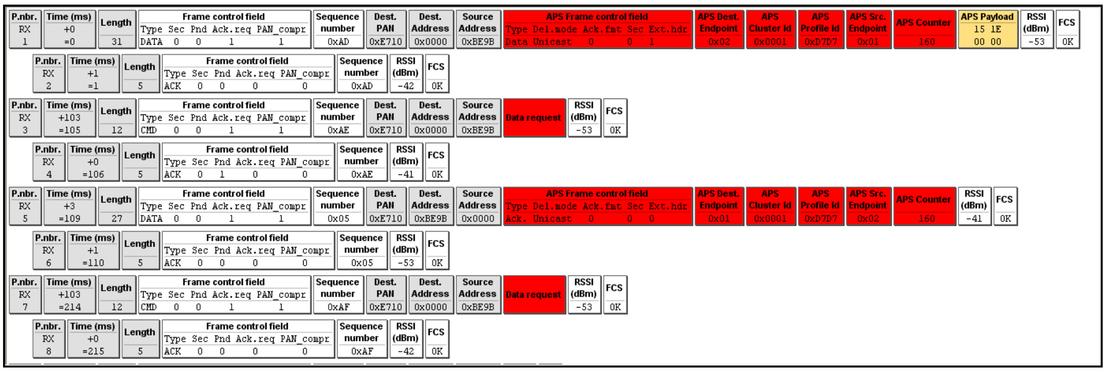

Figure 5: Transmission of data with Application Acknowledgement (1 Packet)

The current consumption profile of the sequence in figure 5 is shown in the Figure 6.

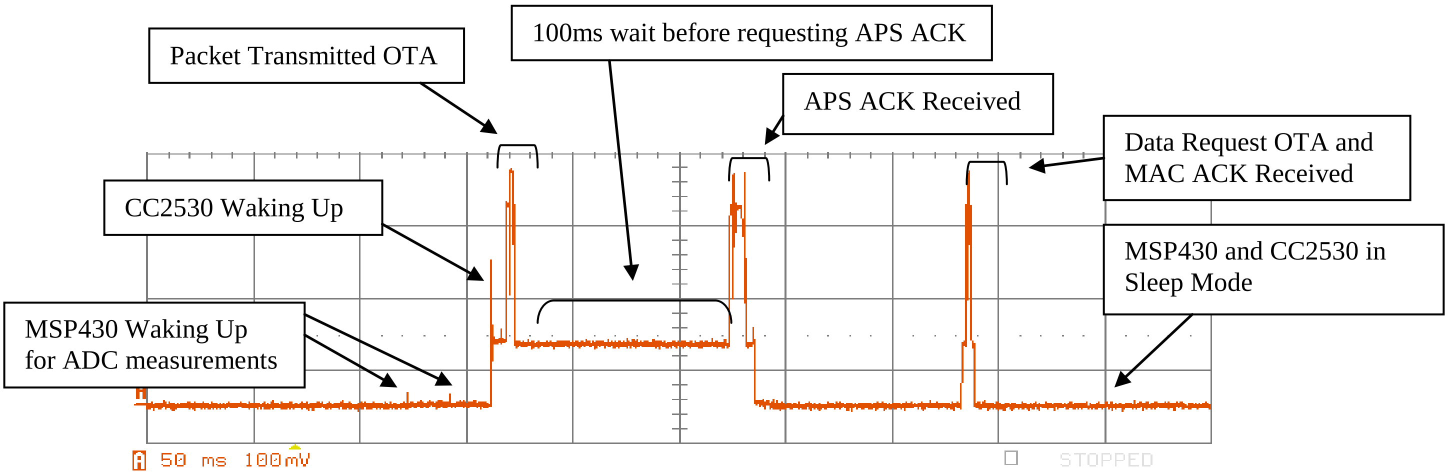

Figure 6: Transmission of data with Application Acknowledgement

Based on the graph above, the user can recalculate the average current consumption for the active period. Using a simplified approach, divide the sequence in different phases. Then measure the approximate current and duration for each phase. A more detailed description can be found in section 7.

Id	Description	Duration [ms]	Current [mA]	[ms*mA]
1	Acquire Sample (ADC on)	44	0.170	7.48
2	Synchronizing clocks and processing of data packet	7	8.5	59.5
3	CSMA-CA (RX), transmit packet and receive MAC ACK	3.5	28.9	101.15
4	Wait before requesting APS ACK	100	8.5	850
5	Request and Receive APS ACK	8	28.9	231.2
6	Post processing of packet	4.5	8.5	38.25
7	Enter LPM2	100	0.0052	0.52
8	Request data (Single Poll)	6.8	28.9	222.53
	Sum	273.8		1484.62
	Average		5.42	

Table 4: Measured Current Consumption during TX without APS ACK (typical values)

The sleep current in this case will still be 4.8 µA. Using (Eq. 1), find the average current consumption, for the 10 seconds (= 10,000 ms) packet interval from the source to the sink:

$$\left(\frac{273.8}{10000} * 5.42\right) + \left(1 - \frac{273.8}{10000}\right) * 0.0048\ \mathrm{mA} = 0.1530\mathrm{mA}$$

The lifetime of the device is now

$$\frac{1200\ [\mathrm{mAh}]}{0.1530\ \mathrm{mA}} = 7843\ \mathrm{hrs} = 326\ \mathrm{days}$$

The value above can be compared with the 2000 days when APS ACK is turned off. Note that in some scenarios, it might make sense to turn on APS ACK just for every n^{th} packet. Then the node will make sure that the packet goes all the way through the network, and the node can realign with the network if some of the routers or devices in the network are down.

Based on (1) and the measurements from section 5, it is relatively easy to calculate the effect of only using application acknowledgement for a few packets. As an example, calculate for a system where every 10^{th} packet requires an acknowledgement. Provides

$$\left(\frac{1}{10}\right) * \left(\frac{273.8}{10000} * 5.42\right) + \left(\frac{9}{10}\right) * \left(\frac{58.6}{10000} * 3.46\right) + \left(1 - \left(\frac{1}{10}\right) * \left(\frac{273.8}{10000}\right) - \left(\frac{9}{10}\right) * \left(\frac{58.6}{10000}\right)\right) * 0.0048\ \mathrm{mA} = 0.0371\ \mathrm{mA}$$

The expected lifetime is now

$$\frac{1200\ [\mathrm{mAh}]}{0.0378\ \mathrm{mA}} = 31704\ \mathrm{hrs} = 1321\ \mathrm{days}$$

5.3 Data polling

The only way an end device will be able to receive data from another node is to periodically send data requests to the associated device. If there are packets destined for the end device, the packet can be sent once the end device asks for them. In some systems, where the end device is never expected to receive any application data, polling can be disabled.

The CC2530 can be configured to automatically poll for data (sending data requests) without involving the host MCU. The MCU will only be notified when data is received. In the Sensor Monitor Network Application, turn polling on by setting APP_POLL_INTERVAL to a non-zero value. The value given will be the poll interval in milliseconds. Setting it to 0 will turn polling off.

The Figure 7 shows the data request packets from the packet sniffer.

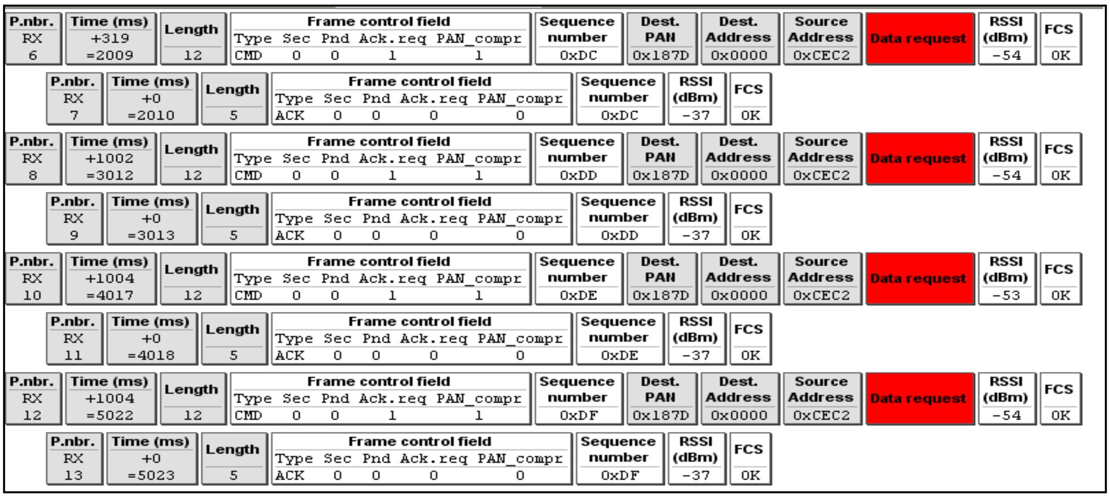

Figure 7: Periodic Data Requests (4 Data Requests)

The active sequence of the data request is represented with the following current profile on the oscilloscope:

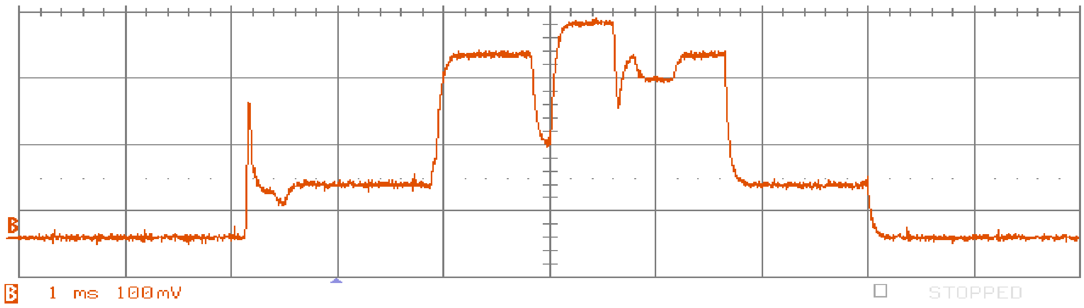

Figure 8: Periodic Data Request

Looking at multiple sequences of the data request, the approximate current and duration is measured for the major events. Once again, the numbers are based on averaging results from several measurements. See further details in section 7.

Id	Description	Duration [ms]	Current [mA]	[ms*mA]
1	Wake Up	1.9	8.5	16.15
2	CSMA-CA (RX), transmit packet and receive MAC ACK	3.5	28.9	101.15
3	Prepare for Sleep	1.5	8.5	12.75
	Sum	6.9		130.05
	Average		18.84	

Table 5: Measured Current Consumption during Data Poll (typical values)

In this case, the sleep current was measured to be 5.2 µA. The increase in the sleep current is caused by a timer on the CC2530 that has to be up and running to handle the timing of the periodic data requests.

Given a polling period, calculate the average current consumption using (1). For an example, set the poll period to 1 second (1000 ms). Thus:

$$\left(\frac{6.9}{1000} * 18.84\right) + \left(1 - \frac{6.9}{1000}\right) * 0.0052 \text{ mA} = 0.1351\text{mA}$$

The lifetime of the device is now

$$\frac{1200\ [mAh]}{0.1351\ mA} = 8878\ hrs = 369\ days$$

The calculations above do not take into account the scenario where the polling device actually receives a packet. This document has briefly discussed the subject for the end device requesting an application acknowledgement. That scenario is a bit different, though, since the recipient of the APS ACK will always end the transaction with a MAC ACK. For the general case, the recipient would have to parse the packet and determine whether it should prepare an APS ACK and it send back to the original transmitter. Only if no APS ACK is required, the device can go to sleep after sending the MAC ACK.

Thus, as for the transmitter, the user must investigate the three different cases

1. Data request with no data
2. Data request with data, no APS ACK required
3. Data request with data, reply with APS ACK

The idea of the Sensor Network Monitor Application is that the end devices just transmit the data and never really receive anything other than the MAC ACK and the APS ACK if enabled. As a consequence, cases 2 and 3 above will not be investigated.

5.4 Combining Transmission and Polling

In the previous sections, this document has discussed the average current consumption for three different scenarios:

1. Transmission of data without data acknowledgement
2. Transmission of data with data acknowledgement
3. Automatic data requests

The sleep current will be different depending on whether polling is enabled or not.

To get the full information of the current consumption, combine the various configurations and tune the system to get the required lifetime. Using the numbers from the previous paragraphs, provides a graphical representation of the system lifetime as a function of the transmit period and poll period.

Figure 9, shows the difference between two systems, where one is always using data acknowledgement, whereas the other does not. By combining ACK ON and OFF for every nth packet, any lifetime between the two graphs can be achieved

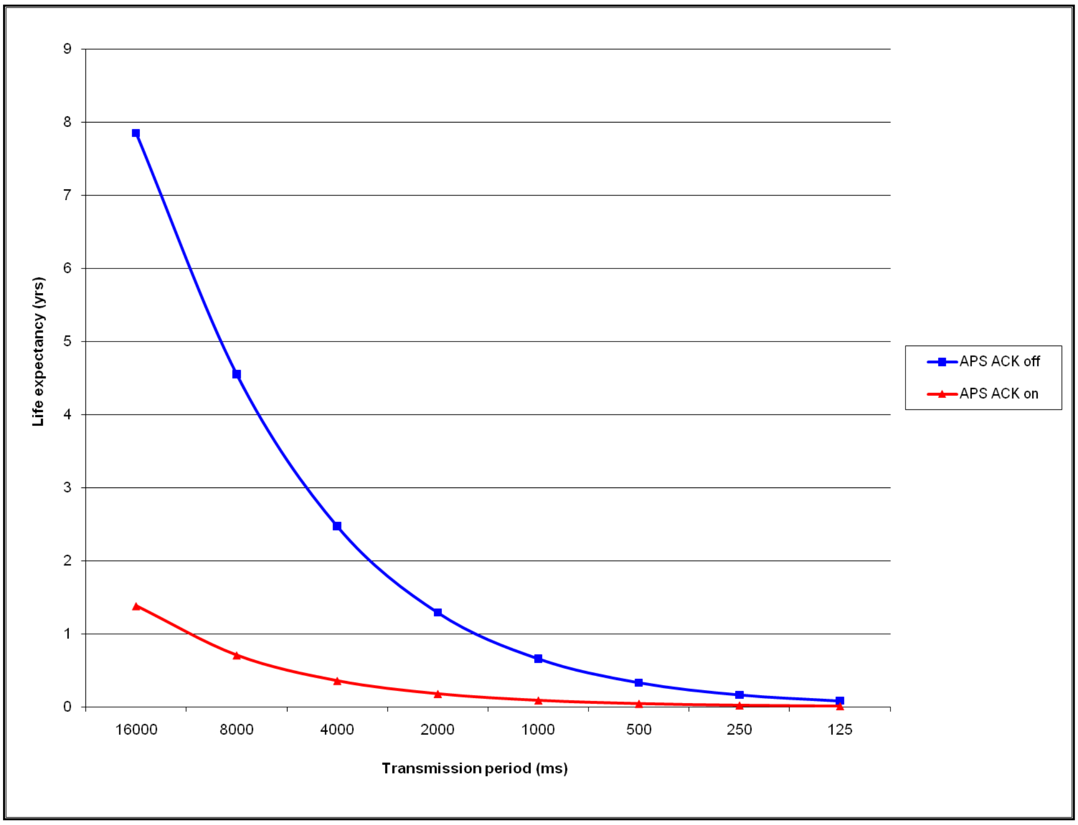

Figure 9: Years of Operation vs. Transmission Period

Figure 10 shows how the two parameters: poll period and transmission period, together give the total lifetime of the system. For this example, this document used the numbers for transmission where application acknowledgement was turned off.

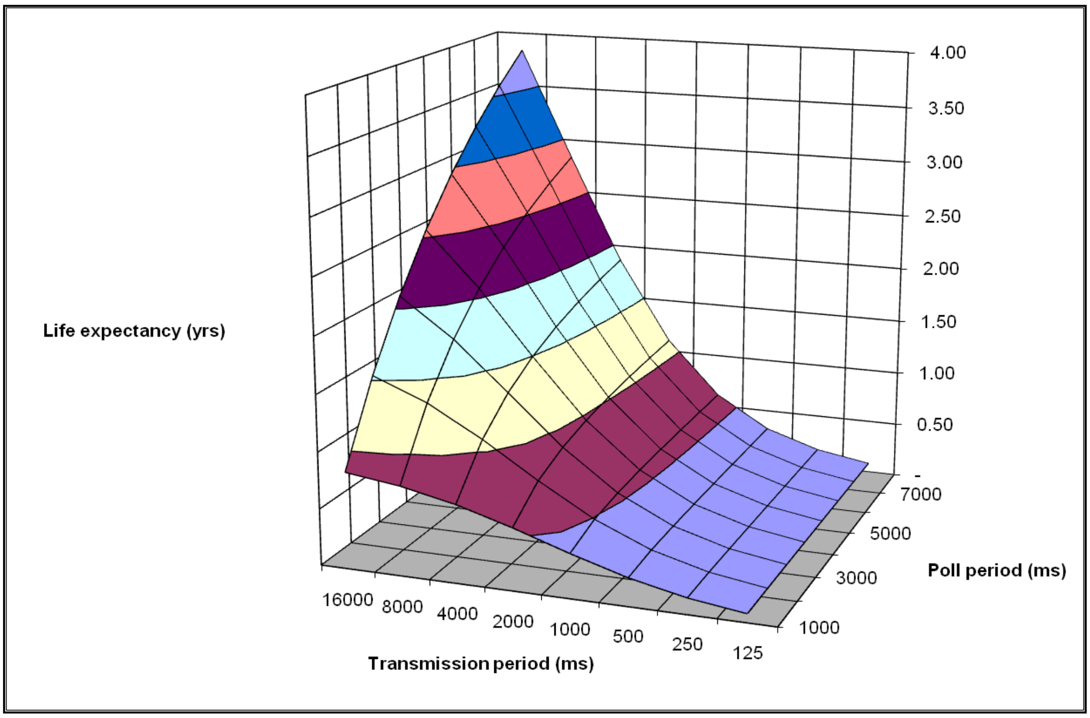

Figure 10: Years of Operation vs. Transmission and Poll Period

5.5 Retransmission of packets

Retransmission of packets that are not being acknowledged, either on the MAC or the APS layer, will have a direct impact on the battery lifetime. There are many reasons for losing packets: noise or interference on the RF channel, poor RF design reducing sensitivity or stability of the radio, operating close to sensitivity limit of radio (out of range), improper operation of the application, etc.

It is possible to do calculations for battery life similar to the ones in section 5.2 by considering the probability of packet loss and the number of retries. To some extent, it can be compared to increasing the overall packet rate. However, the time the node needs to stay in RX waiting for the ACK to arrive must be taken into account.

As a consequence, the following parameters must be determined for a full analysis:

1. Probability of packet loss
2. Timeout waiting for missing MAC ACK
3. Timeout waiting for missing APS ACK
4. Number of retries
5. Retry period

This will not be discussed further in this application note.

6. Other contributors to the total current consumption

This document has only covered how to change the current consumption by modifying the functional configuration of the end device. This section discusses other factors that should be taken into account for improving the overall battery life of the system.

6.1 Battery type

As briefly mentioned at the end of section 4, the battery type will have a direct impact on the lifetime of the system. Typical low cost household batteries are AA and AAA Alkaline batteries. The average capacity of one AA battery is 2500 mAh and 1200 mAh for one AAA battery. The capacity determines how many hours the battery will be able to supply a specific current. If your application drains 1 mA on average, the system can stay alive for 1200 hours using AAA batteries. The lifetime is doubled when using AA batteries.

Note that there are many other criteria than the capacity that should be carefully considered before selecting a battery type. Some of them are:

- Voltage
- Peak current
- Rechargeable
- Price
- Availability
- Size
- Material
- Self discharge current
- Environmentally friendly

6.2 Taking advantage of the MSP430

The MSP430F2274 used on the CC2530 ZNP Mini Kit Target Board is a good fit for the CC2530. The MCU has a short wake-up time and both the operating current and sleep current are low. Using the many low power features of the MSP430, it is possible to develop systems that consume just a fraction of the current other MCU's would consume.

It is still important that the MSP430 be used correctly in order to ensure as low an average current consumption as possible.

Since the CC2530 ZNP Mini Kit Target Board does not have any 32 kHz crystals connected to the MSP430, there are no traditional clock sources that can be used for an ultra low power sleep timer. For many systems, the only solution would be to use the internal high speed RC oscillator of the MCU as clock source. The current consumption in such "sleep" modes can be several hundred uA – wasting a lot of precious energy when the system is idle.

The MSP430F2274 has a low power, low frequency oscillator called the VLO. It is operational in the low power mode 3 and draws only a couple of hundred nA. By calibrating the VLO with the DCO during system initialization, the accuracy can be improved significantly. The VLO can then be used as the clock source for a sleep timer if the accuracy requirements are within the limits possible to achieve using the VLO.

Thus, even without a crystal, the system can perform periodic tasks and have a sleep current of just a few hundred nA. Using an ampere meter, we measured the sleep current on the MSP430F2274 in LPM3 with the VLO enabled to be 486 nA on the CC2530 ZNP Mini Kit.

Another parameter to have in mind is the operating frequency of the MCU core when it is active. The MSP430 makes it easy to set up the DCO to any frequency up to 16 MHz. However, the current consumption increases as the frequency increases. The system should be carefully designed such that the tradeoff between speed and current consumption is taken into account. The slower the clock, the longer time the transactions take and the longer the MSP430 has to stay awake. However, the higher the clock rate, the more energy you burn when the system is active.

For the application running on the eZ430-RF2530 ZNP Mini Kit, having a high clock rate is advantageous, since there are a lot of data being transferred over the SPI bus between the CC2530 and the MSP430. For both the devices, it is beneficial that these transactions (or really remote procedure calls) are effectuated as quickly as possible, such that each device can go to sleep mode immediately when its part of the transaction is finished. Thus, the Zigbee Sensor Monitor Network application has been set up to operate at 8 MHz in active mode.

Note that the selected DCO frequency sets a limit to the lowest operating voltage of the system. To safely operate at 8 MHz, the on board voltage has to be above 2.2V. The software should in addition take advantage of the fast transition times of the MSP430 from sleep to active mode. Whenever the software is waiting for an event, the software should implement delay routines that set the MCU in deep sleep and have interrupts to wake up the system - either by using timed periodic polling or by having a hardware trigger on the actual event. The use of spinning loops should be limited to extremely short delays (several microseconds or less).

In addition to the clock, it is also important to make sure that all peripherals and GPIOs on the MSP430 are configured correctly to reduce the current consumption. Most peripherals are automatically turned off when the MSP430 enters a low power mode, but some things must be turned off by the user. For instance, the internal reference voltage for the ADC must be turned off by writing to the appropriate register when the ADC has been disabled.

The GPIOs can cause extensive current leakage if not set correctly. A typical example would be the case where one GPIO is configured as an input and the external line is floating, causing a leakage of potentially several hundred µA on that IO pad. If two signals are driving against each other (both are set as outputs), the current draw can be several mA.

It is recommended to go over the schematics of the system carefully to make sure that all of the pins on the MSP430 are configured for the lowest possible current consumption.

7. Deciphering the current consumption profile

The sections above only used average values and a simplified approach to calculate the average current consumption during an active period of the device. This section covers the details of various sections on the oscilloscope captures and explains what is happening on the device during a data transmission when application acknowledgment is enabled

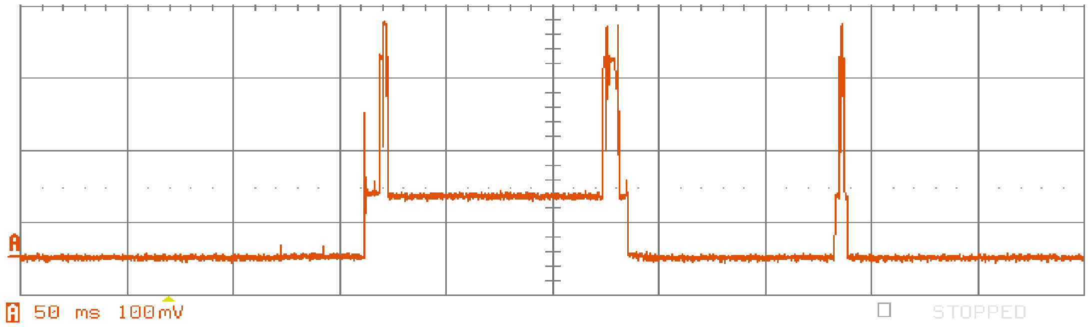

Figure 11: Transmission of data with Application Acknowledgement

Figure 11 gives the overview of the events during the transmission, but it is difficult to see and understand what is happening. By zooming in on the three peaks, it is easier to see and understand what is happening.

The three subsequent sections will give a detailed description of each of the three steps.

The values should only be regarded as indicative for the given hardware and software.

Please note that the values shown for the currents and durations are based on approximate measurements using the oscilloscope and to some extent a multi meter. The average value for several measurements has been used.

The values should only be regarded as indicative for the given hardware and software.

Step 1 – Transmit the Packet

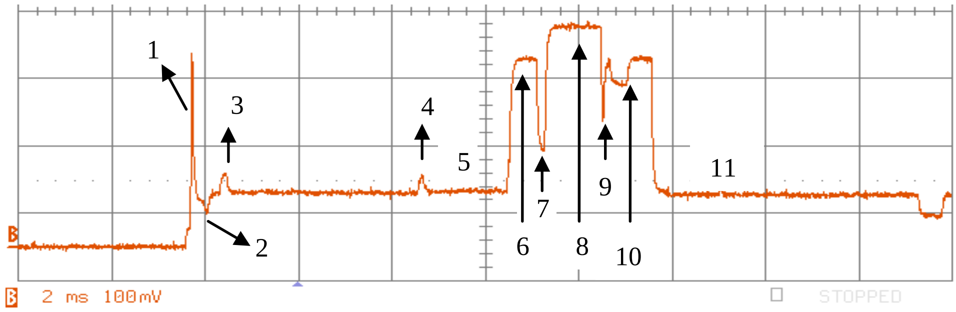

Figure 12: Transmit the Packet

Event	Description	[ms]	[mA]
1.	The MSP430 has finished the ADC measurements and wakes up the CC2530 to send the packet. The initial peak is caused by the startup of the internal 16 MHz RC oscillator and the 32 kHz crystal.	0.4	28
2.	The MSP430 sends the data and appropriate command over to the CC2530 so that it can start the transmission. The MSP430 will immediately go to sleep, waiting for the CC2530 to signal that it has processed the command	0.35	8.5
3	CC2530 starts the 32 MHz crystal and sets it up as the core clock. The command from the MSP430 is processed and the CC2530 starts synchronizing an internal timer in order to be able to transmit the packet.	4.2	8.5
4	CC2530 wakes up the MSP430, such that it can read out the return value for the command that was invoked. The MSP has finished the transmission step and goes back to sleep.	0.2	8.5
5	CC2530 sets up the radio and churns the packet through the ZigBee stack, preparing it for transmission.	1.7	8.5
6	CC2530 starts the Carrier Sense Multiple Access Collision Avoidance (CSMA-CA) algorithm. The radio is set in RX and CC2530 will assess whether the channel is clear for transmission. The duration of this step will vary if the channel is noisy. This is due to the random interval the device will defer from sending when the medium is sensed to be busy or noisy. The average duration over 20 observations is approximately 1 millisecond. The duration in the plot is 0.5 milliseconds.	1	26.5
7	Switch from RX to TX.	0.2	18
8	The packet is sent over the air.	1	31.5
9	Switch from TX to RX.	0.2	18
10	CC2530 receives the MAC ACK from the associated device in the network, indicating that the packet was received with no errors. Note that this does not mean that the application data was received by the intended recipient, only that the packet was transmitted and acknowledged on the MAC layer by the network node that was associated with the transmitter.	1	26.5
11	CC2530 enters an IDLE state, waiting to request the APS ACK from the recipient. The duration of this step will depend on the number of nodes and how many hops there are between the sender and transmitter. Sensor Monitor Network Sample Application sets this duration to 100ms.	100	8.5

Step 2 – Request and Receive APS ACK

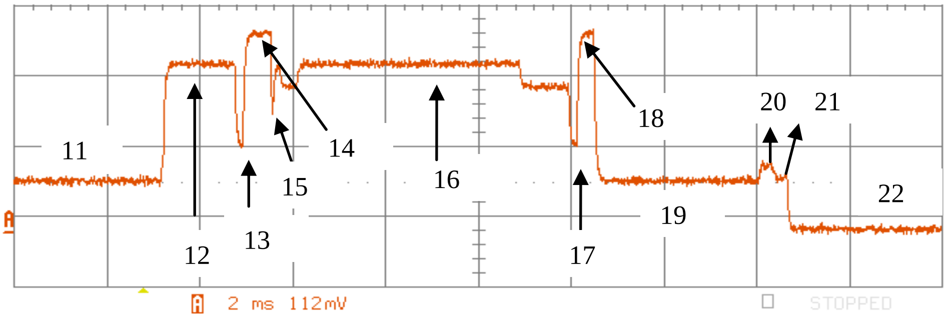

Figure 13: Request and Receive APS ACK

Event	Description	[ms]	[mA]
12	CC2530 wakes up and begins a new transmission sequence. The node has to request the APS ACK from the recipient. As in step 6, it runs the CSMA-CA algorithm.	1	26.5
13	Switch from RX to TX.	0.2	0.2
14	The data request packet is sent over the air.	1	31.5
15	Switch from TX to RX.	0.2	18
16	The node receives first the MAC ACK for the data request packet, then it stays in RX until it receives the APS ACK.	5	26.5
17	Switch from RX to TX.	0.2	18
18	The node sends a MAC ACK to confirm that it has received the APS ACK.	0.4	31.5
19	CC2530 processes the incoming APS ACK and prepares a response to the host (MSP430)	3.7	8.5
20	CC2530 wakes up the MSP430 to indicate that the packet was sent without failure to the recipient	0.3	11
21	CC2530 prepares for sleep and turns off the 32 MHz crystal	0.5	9

Step 3 – Request Additional Data

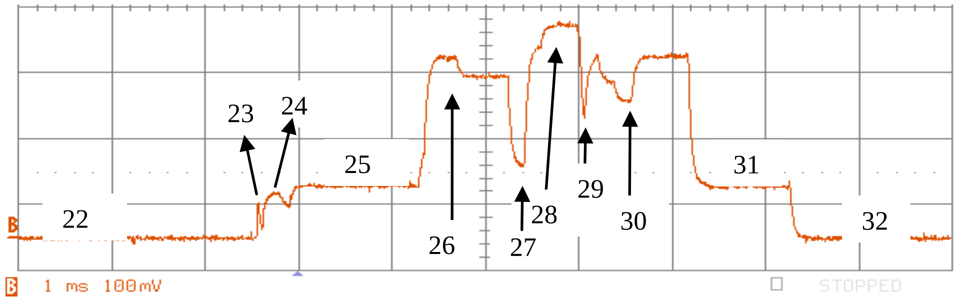

Figure 14 Request Additional Data

Event	Description	[ms]	[mA]
22	CC2530 is sleeping, but has only entered LPM2, as it needs to wake up one last time. Sensor Network Monitor Sample Application has configured CC2530 such that this duration is 100 ms.	100	0.0052
23	CC2530 wakes up from LPM2. Similar to step 1. The node will transmit a new data request just to make sure that there are no more pending data destined for the node before it enters deep sleep. Note that the MSP430 is not involved in this transaction.	0.05	5
24	CC2530 starts the 32 MHz clock.	0.5	7.3
25	Internal processing and preparing for transmit.	1.45	8.5
26	CSMA-CA	1	26.5
27	Switch from RX to TX.	0.2	18
28	The data request packet is sent over the air.	0.8	31.5
29	Switch from TX to RX.	0.2	18
30	Receive MAC ACK	1	26.5
31	Post processing and prepare for deep sleep.	1.6	8.5
32	Deep sleep.	10000	0.0048

8. Conclusion

This document has covered the basics of performing current consumption measurements and estimating the expected lifetime of a battery operated system. It then provided detail on the CC2530 ZNP Mini Kit platform and performed a series of measurements on the CC2530 ZNP Mini Kit Target Board to determine the lifetime of the system depending on various configuration options in the software.

In a two chip solution, it is clear that the two devices must operate "in harmony" with each other to get the best out of the system in terms of low power. Both devices must be used and configured so that they meet the functional requirements and at the same time do not waste power by staying awake when they could have been in a sleep state. By using the ultra low power MSP430 microcontroller and the CC2530 ZigBee Network Processor, it is possible to achieve several years of operation using common off-the-shelf batteries.

Two factors that influence the power consumption strongly are the transmission period and the poll period. Effectively increasing the transmission period can be done by making sure that data is only sent when it must be, and then send as much data as possible. It is e.g. better to transmit one packet with two samples rather than two packets with one sample each.

Another factor that will influence the power consumption is the condition of the RF channel. With much traffic on the channel, the duration of the CSMA-CA phase will be longer and will have a direct impact on the expected lifetime of the system. In addition, if a packet is lost due to interference, the packet may have to be retransmitted, which in turn will translate into shorter transmit periods and, thus a shorter battery lifetime.

As a consequence, it is difficult to calculate the exact battery lifetime, since the model of the system rapidly becomes complex. However, by having a basic understanding of the major contributors to the battery lifetime and then combining the various elements, it is possible to get good estimates that can be used as basis for the final system design.

9. References

[1]. Z-Stack Download [1]

[2]. CC2530 Second Generation System-on-Chip Solution for 2.4 GHz IEEE 802.15.4 / RF4CE / ZigBee (swrs081)

[3]. ZNP Interface Specification (swra312)

[4]. ZNP Mini Kit [4]

[5]. ZNP Mini Kit Wiki Resource Guide

[6]. Sensor Monitor Network Sample Application (swrc229)

[7]. Packet Sniffer [7]

10. Document History

Revision	Date	Description/Changes
1.0	10/3/2011	Initial release.

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

TI assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using TI components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any TI patent right, copyright, mask work right, or other TI intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information published by TI regarding third-party products or services does not constitute a license from TI to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of TI information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Reproduction of this information with alteration is an unfair and deceptive business practice. TI is not responsible or liable for such altered documentation. Information of third parties may be subject to additional restrictions.

Resale of TI products or services with statements different from or beyond the parameters stated by TI for that product or service voids all express and any implied warranties for the associated TI product or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

TI products are not authorized for use in safety-critical applications (such as life support) where a failure of the TI product would reasonably be expected to cause severe personal injury or death, unless officers of the parties have executed an agreement specifically governing such use. Buyers represent that they have all necessary expertise in the safety and regulatory ramifications of their applications, and acknowledge and agree that they are solely responsible for all legal, regulatory and safety-related requirements concerning their products and any use of TI products in such safety-critical applications, notwithstanding any applications-related information or support that may be provided by TI. Further, Buyers must fully indemnify TI and its representatives against any damages arising out of the use of TI products in such safety-critical applications.

TI products are neither designed nor intended for use in military/aerospace applications or environments unless the TI products are specifically designated by TI as military-grade or "enhanced plastic." Only products designated by TI as military-grade meet military specifications. Buyers acknowledge and agree that any such use of TI products which TI has not designated as military-grade is solely at the Buyer's risk, and that they are solely responsible for compliance with all legal and regulatory requirements in connection with such use.

TI products are neither designed nor intended for use in automotive applications or environments unless the specific TI products are designated by TI as compliant with ISO/TS 16949 requirements. Buyers acknowledge and agree that, if they use any non-designated products in automotive applications, TI will not be responsible for any failure to meet such requirements.

Following are URLs where you can obtain information on other Texas Instruments products and application solutions:

Products		**Applications**	
Audio	www.ti.com/audio	Communications and Telecom	www.ti.com/communications
Amplifiers	amplifier.ti.com	Computers and Peripherals	www.ti.com/computers
Data Converters	dataconverter.ti.com	Consumer Electronics	www.ti.com/consumer-apps
DLP® Products	www.dlp.com	Energy and Lighting	www.ti.com/energy
DSP	dsp.ti.com	Industrial	www.ti.com/industrial
Clocks and Timers	www.ti.com/clocks	Medical	www.ti.com/medical
Interface	interface.ti.com	Security	www.ti.com/security
Logic	logic.ti.com	Space, Avionics and Defense	www.ti.com/space-avionics-defense
Power Mgmt	power.ti.com	Transportation and Automotive	www.ti.com/automotive
Microcontrollers	microcontroller.ti.com	Video and Imaging	www.ti.com/video
RFID	www.ti-rfid.com		
OMAP Mobile Processors	www.ti.com/omap		
Wireless Connctivity	www.ti.com/wirelessconnectivity		
		TI E2E Community Home Page	e2e.ti.com

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265