TG-APP-Power 4.2 to 4.3 - 100

ETRX35x WIRELESS MESH NETWORKING MODULES

APPLICATION NOTE – POWER CONSUMPTION

Table of Contents

1 Introduction

In order to calculate the average power consumption (or battery life) of an ETRX3 series based product it is necessary to know the exact amount of power consumed for each individual activity.

When the device is a router or a coordinator this is straightforward as the unit will most of the time consume the stated RX current (26.5mA) as a router or coordinator is not supposed to go to sleep, but when it comes to end devices estimating the average power becomes more complicated.

With the release of EmberZNet4.3 Ember announced that the power consumption of sleepy end devices has been greatly reduced by taking advantage of improvements to radio operation. In addition to this a new CPU idle mode has been introduced to reduce the current consumption even further; the new idle state will not lead to any changes visible to the user in the Telegesis AT command set.

In order to analyze the advantages of ZNet4.3 the power consumption of a sleepy end device polling for data from its parent was examined.

As the basis of these tests identical test firmware compiled both on Znet4.2 and ZNet4.3 was used running on two ETRX357 modules.

It is expected that on the ETRX2 series power savings also similar to the ones described in this document can be achieved. With the ETRX2-PA and the ETRX357-LR the event timings are the same as reported here, but the current will be higher as stated in the corresponding datasheets.

2 Power Consumption When Polling for data

For all measurements described in this chapter a set of ETRX357 modules is used. One node is set to be a sleepy end device (SED) sleeping whenever possible, the other node is the network's coordinator. The main difference between an end device and a router/coordinator is that on an end device even when fully awake only the microcontroller is running and the radio only gets switched on when in use.

When not part of a network (and not idling) an ETRX357 will consume 7.5mA (with the ARM core being clocked at 24MHz for highest efficiency), because only its microcontroller is running. When scanning for PANs, trying to join a PAN or doing an energy scan, the radio is fully switched on and the unit will consume 26.5mA in RX mode and 31mA in TX mode when used with the default radio settings. When the output power is increased, the TX current will increase in the same way as the RX current increases when enabling boost mode. This however will have no effect on the transaction timings shown in this document.

The end device is attached to a Telegesis Development board and powered with 3.0V provided by the development board. The current consumption is measured in terms of a voltage drop across a 10Ω series resistor. Current consumption at higher voltages will be substantially the same as the EM357 chip's internal regulated voltage will remain constant.

For the measurements three ETRX357 modules from different batches were used for comparison and it was made sure that the captured waveforms are representative by performing multiple measurements and making sure the resulting waveforms were identical.

3 References

[1] "Measuring EM35x Power Consumption – 120-5068-0000A" by Ember Corporation

3.1 ZNet4.2 SED Polling → No Message Present

For this measurement the end device polls the parent (the COO) at regular intervals, whilst it is made sure that there is no data to be polled from the parent. It was found that the length of the interval between any two polls has no effect on the amount of energy consumed per poll, as long as it is made sure that the end device hasn't timed out of its parents neighbour table.

Ember are providing information [1] on how to disable the random CSMA backoff for the purpose of this measurement, but it was found that in ZNet4.2 the required API didn't exist. Because of this measurements were captured for both the shortest and the longest identified CSMA backoff times.

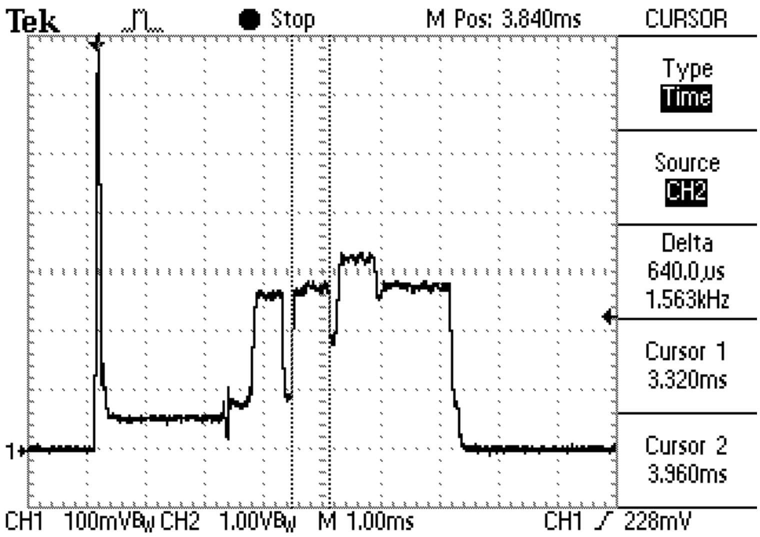

Figure 1: ZNet4.2 Polling with Short Backoff (x:1ms/div, y: 10mA/div)

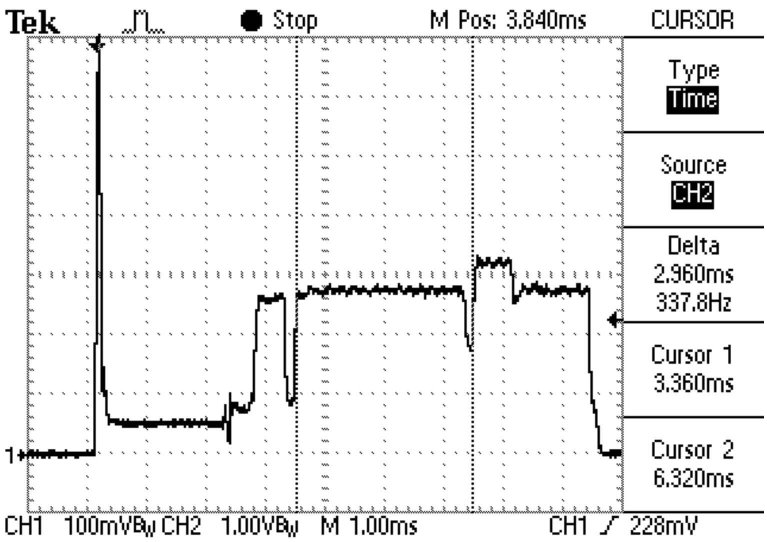

Figure 2: ZNet4.2 Polling with Long Backoff (x:1ms/div, y: 10mA/div)

In table 2 the timings shown in figures 1 and 2 are listed and the resulting power consumption is calculated. In ZNet4.2 a poll uses between 107µC and 169µC depending on the length of the random CSMA backoff.

3.2 ZNet4.3 SED Polling → No Message Present

The same measurement was then repeated with identical firmware, but with the ZNet4.3 stack compiled in instead of the ZNet4.2 stack. It was found that disabling the CSMA backoff as described by Ember resulted in a timing identical to the shortest one with CSMA backoff enabled, so to produce results comparable to the previous measurements, the shortest as well as the longest observed CSMA backoff times were considered.

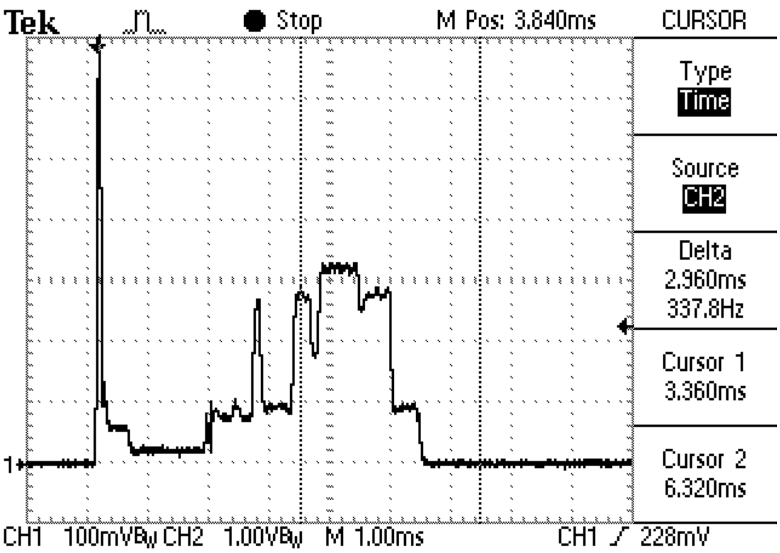

Figure 3: ZNet 4.3 Polling with no CSMA (short backoff) (x: 1ms/div, y: 10mA/div)

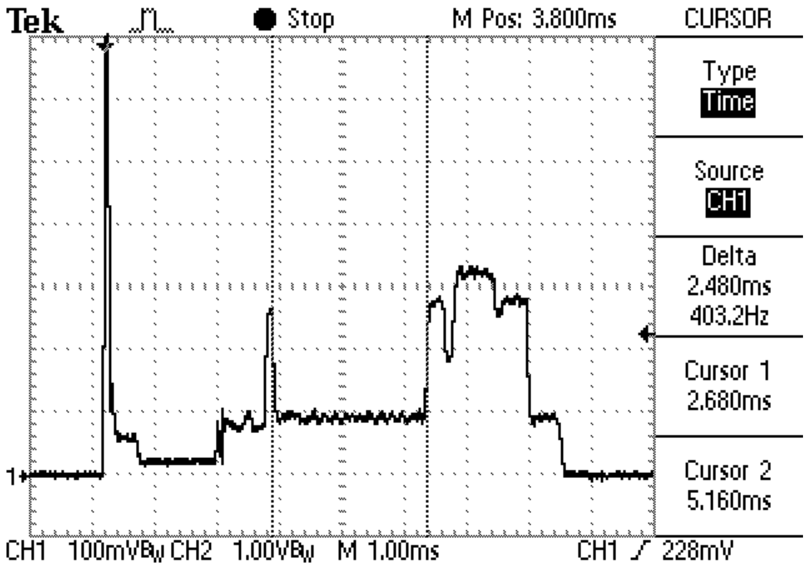

Figure 4: ZNet4.3 Polling with Long Backoff (x: 1ms/div, y: 10mA/div)

In table 2 the timings for the captures above are shown. The resulting power consumption is between 74µC for a poll with no (short) CSMA backoff and 88µC for a poll with the longest observed CSMA backoff. This represents a power saving of 30% to 52% depending on the CSMA backoff time. In a real application the backoff time is random so the average will lie between these two extremes.

Mode	Current	Time (ZNet4.2)	Time (ZNet4.3)
Sleep	>1µA	?	?
In-rush Current	60mA	120µs	120µs
HAL Start	5mA	2.1ms	400µs
Crystal Start-up and Cal	2.5mA	-	1.28ms
Radio Init 1	7.5mA	480µs	760µs
Radio Init 2	26.5mA	480µS	160µs
Transmit Preparation	7.5mA	200µs	480µs (2.440ms)
CSMA	26.5mA	640µs (2.96ms)	440µs
Send	31mA	640µs	640µs
Listen for reply	26.5mA	1.28ms	520µs
Shutdown	7.5mA	200µs	440µs
Sleep	>1µA	?	?
Energy Consumption		107µC (169µC)	74µC (88µC)

Table 1: Polling for Data without Idling

3.3 ZNet4.3 SED Polling → No Message Present, CPU Idling

Next, the measurement with the short (no) CSMA backoff was repeated with the new Idle state enabled.

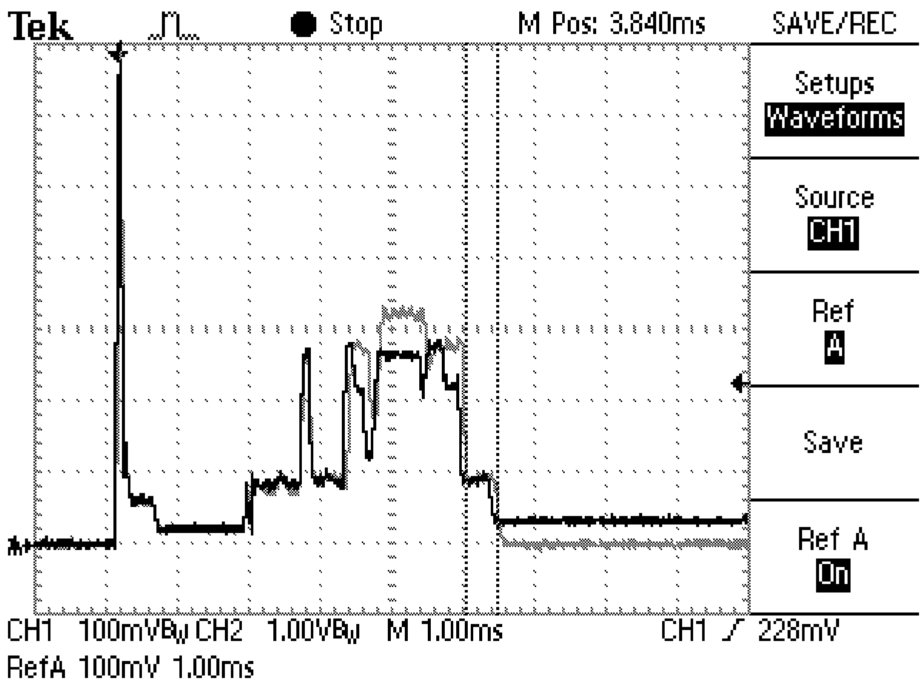

Figure 5: ZNet4.3 Polling with and without CPU Idle (x: 1ms/div, y: 10mA/div)

As a reference figure 5 shows the waveform of the previous measurement without idle in grey and the new measurement with CPU idle in black. It can be seen that the timings haven't changed because of the enabled CPU idle feature, but the current consumption of some of the activities has dropped.

Please note that the fact that the current doesn't go back to 1µA after completion of the poll is down to the fact that the test firmware was not adopted to fully support the new Idle mode. With an adopted firmware it is expected that the current will drop to the expected <1µA as soon as the poll completes.

Mode	Current	Time (ZNet4.3)
Sleep	<1µA	?
In-rush Current	60mA	120µs
HAL Start	5mA	400µs
Crystal Start-up and Cal	2.5mA	1.28ms
Radio Init 1	7.5mA	760µs
Radio Init 2	26.5mA	160µs
Transmit Preparation	7.5mA	480µs
CSMA	24mA	440µs
Send	26mA	640µs
Listen for reply	24mA	520µs
Shutdown	7.5mA	440µs
Sleep	<1µA	?
Energy Consumption		68µC

Table 2: ZNet4.3 Poll for Data with Idling

Looking at the new figures it can be seen that the CPU idling reduces the power consumption of a poll from 74µC to 68µC. This is slightly less than the 10% savings outlined in [1], but it is within the expected range.

The timings listed in the tables above are averaged over a number of samples, as there can be slight variations of the consumption between samples. All samples were taken with no additional traffic on the air, additional traffic could potentially increase the power consumption (collision prevention, and actual collisions taking place).

4 Average current

When a sleepy end device polls once per second, the charge in coulombs translates directly to average current, eg 68µC per event becomes 68µA average current consumption. If the polling interval is 2 seconds the average current falls to 34µA, and so on down pro rata.

5 Conclusions

The new Ember ZNet4.3 offers massive power savings compared to previous versions of the stack. Because of this it is highly recommended to migrate to firmware based on ZNet4.3 if battery life is a concern. The standard Telegesis AT Commandset firmware R304 is based on ZNet4.2 and the coming R305 is based on ZNet4.3.

6 Trademarks

All trademarks, registered trademarks and products names are the sole property of their respective owners.

7 Disclaimer

Product and Company names and logos referenced may either be trademarks or registered trademarks of their respective companies. We reserve the right to make modifications and/or improvements without prior notification. All information is correct at time of issue. Telegesis (UK) Ltd does not convey any license under its patent rights or assume any responsibility for the use of the described product

8 Contact Information

Website: www.telegesis.com

E-mail sales@telegesis.com

Telegesis (UK) Limited

Abbey Barn Business Centre

Abbey Barn Lane

High Wycombe

Bucks HP10 9QQ

UK

Tel: 01494 510199

Fax: 05603 436999

