**Battery-less sensor nodes for 802.15.4/ZigBee wireless networks**

*Authors:*
*M. Meli, M. da Silva*

*Contact address:*
*Prof. Dr. Marcel Meli*
*Zürcher Hochschule für Angewandte Wissenschaften (ZHAW)*
*Institute of Embedded Systems (InES)*
*Technikumstrasse 9*
*CH-8400, Winterthur, Switzerland*
*Email : marcel.meli@zhaw.ch*

**Abstract:**

**Battery-free wireless sensor nodes are very convenient in many applications. They eliminate the costs associated with installing batteries and reduce maintenance headaches. For these reasons, they are gaining in popularity. Such devices usually require the use of an energy source that harvest power from the surroundings. Because of the necessity of harvesting energy and the related costs, there are limitations to the kind of application and wireless protocol that can successfully be implemented, especially when energy is produced intermittently. The interest in 802.15.4-based wireless systems such as ZigBee has also led to its consideration for battery-less nodes. In this paper we briefly discuss some of the challenges faced when using 802.15.4/ZigBee in applications where little energy is available. We then show that some of the recently introduced microcontrollers featuring FRAM memory allow interesting gain in energy, making it even easier to overcome the difficulties.**

## 1. Introduction

Several wireless systems are currently competing to become the dominant protocol in wireless automation. Installing, controlling and running thousands of nodes is no easy task. The power autonomy necessary for the nodes means that issues such as battery failure and their replacement are important.  It is therefore normal that the desire to get rid of batteries (or at least reduce the frequency of their replacement) has become an important issue in wireless sensor networks.
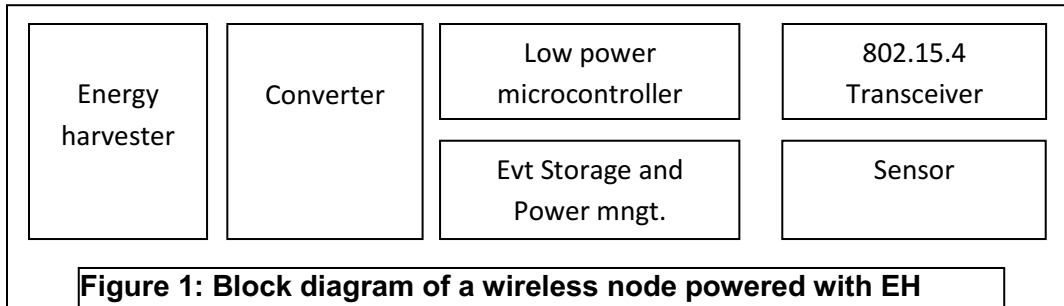
The obvious solution is to harvest energy from the surroundings, and use it to process and communicate. Harvesters and the associated technology come with a price that should be kept minimal for a market relevant product. In order to deal with the problem of limited energy, a wireless protocol can be designed to work on very small amounts of energy. This usually comes at the price of flexibility (or something else). One can also design a wireless protocol with much flexibility in mind. This approach introduces a lot of overhead, increasing the energy demands. ZigBee is an example of the second case, where overhead and sophistication has increased the energy needs.

Despite this, the fact that many manufacturers have adopted 802.15.4 (the lower layers of ZB) has led to a healthy competition. This competition means that manufacturers are interested in distinguishing their products, also by reducing the energy needs. Microcontrollers and transceivers used for 802.15.4 can only get better, making it increasingly easy to design 802.15.4 battery-free nodes. Some manufacturers have announced battery-less ZigBee compatible products, and the trend will probably continue.

In this work, we review some of the difficulties related to low energy and ZigBee, and we look at what new devices could bring.

## 2. Some problems facing 802.15.4/ ZigBee nodes powered by EH

A typical node will include the elements in Figure 1. When enough energy is available, some of it can be store for "bad times". This is generally the case with thermo harvesters. There are also cases where very little energy is produced. The system starts from a "power off" situation, uses the energy and returns in a "power off" state.

| Energy harvester | Converter | Low power microcontroller | 802.15.4 Transceiver |
|---|---|---|---|
| | | Evt Storage and Power mngt. | Sensor |

**Figure 1: Block diagram of a wireless node powered with EH**

Difficulties in adding battery-less nodes to 802.15.4-based networks have been discussed several times at the ZigBee European Developer's conferences [3,4,11,13] and probably in other forums. While we completely disagree with the "no go" or "impossible" claims that have been voiced by some for obvious competition reasons, it will also be naïve to think that there are no problems. We will therefore give a small overview here of energy-related issues. The reader is advised to check the references for more information.

### 2.1. ZigBee header[1,2,3,13]

The header needed in a ZigBee frame is long. Several bytes are required to deal with the different parameters and layers. We will consider the example of a battery-less switch sending data to turn a lamp ON/OFF. A typical data packet for 802.15.4 application will look as following.

| octets | 2 | 1 | 0/2 | 0/2/8 | 0/2 | 0/2/8 | 0/5/6/10/14 | Variable | 2 |
|---|---|---|---|---|---|---|---|---|---|
| Fields MAC | Frame control | Sequence number | Dest. PAN | Dest. Addr. | Source PAN | Source Addr. | Security Header | Payload | FCS |
| Example | 2 | 1 | 2 | 2 | 0 | 2 | 0 | Variable | 2 |

It requires about 11 bytes + payload

A typical data packet on the ZigBee application layer will build on the example above, and looks as follows.

| Octets | 2 | 2 | 2 | 1 | 1 | 0/8 | 0/8 | 0/1 | var | var |
|---|---|---|---|---|---|---|---|---|---|---|
| Fields NWK | Frame control | Dest. Addr. | Source Addr. | Radius | Seq. Nber | Dest. Addr. | Source Addr. | Multicast cntr. | Src. Route subfr. | Frame payload |

| Octets | 1 | 0/1 | 0/2 | 0/2 | 0/2 | 0/1 | 1 | Variable | |
|---|---|---|---|---|---|---|---|---|---|
| Fields APS | Frame control | Dest. endpoint | Group Addr. | Cluster Id. | Profile Id. | Source endpoint. | APS counter | Payload | |

To be able to send switch commands for ZigBee, the following is needed:
- Clear channel assessment (switch on the receiver to verify channel activity)
- Transmit 30 + 2 + 6 = 38 bytes. (30 bytes of data + 2 bytes (FCS) on MAC layer, 6 Bytes for PHY overhead)

Receiving the ACK and resending the data will improve reliability.

The longer a frame is, the more energy is required. ZigBee frames are long. Lots of data is required just to turn a simple lamp ON or OFF. Despite these difficulties, it has been shown that an application such as a battery-less lamp switch can be implemented using ZigBee. Examples using a very small solar cell have also been demonstrated.

## 2.2. Joining a network [11]

A device needs to join a network before a meaningful exchange of application data can take place. Furthermore, applications need to be "informed" about their End Points (binding). These operations require the exchange of several packets of data, leading to an important use of energy.

Pairing using an "out of band channel" such as RFID has been suggested. Devices that should communicate together exchange the required parameters when they are brought together, or by using a pairing device such as a mobile phone accordingly equipped. This has the advantage of allowing a secure pairing to take place (Touch and Pair). It also makes the whole process more intuitive and simple for the end user. The figure below shows the block diagram of a node with dual ported memory for RFID pairing
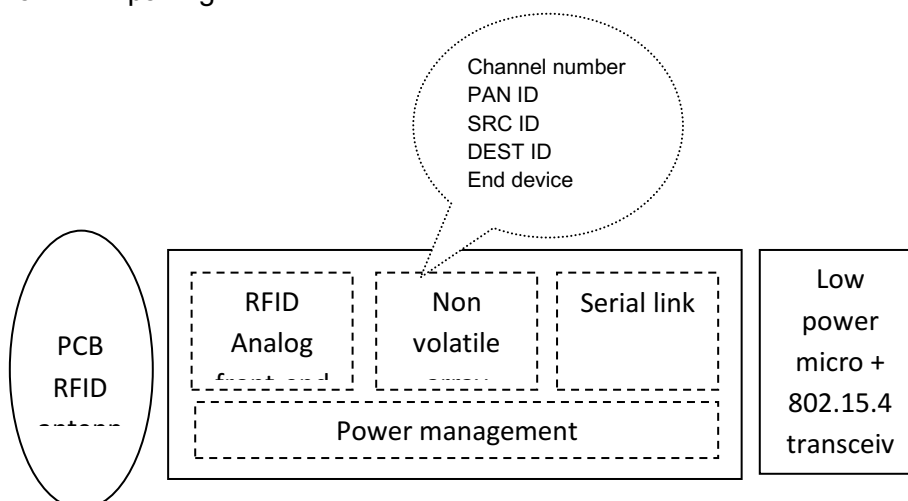


The data for pairing can be store in a special Dual Ported memory, accessible for the application microcontroller on one side and for the RFID reader on the other side. The microcontroller could also be programmed to emulate and RFID device. The important parameters are written before the data exchange using ZigBee starts.

When the battery-less node is energised, it simply reads those parameters and uses them for the communication.

What happens if the some parameters change after pairing? (The network master could for example change the channel where communication is taking place).

- The obvious solution is that the nodes can be paired again in the same way as before (by using a Touch and Pair strategy).
- Another solution is to use extra power to receive the ACK and store the result of the communication in non-volatile memory. Processing done on the microcontroller can then be used to change the channel using a list of preferred channels. This will be clearer when considering the energy gain resulting from the use of the FRAM.

### 2.3. ZigBee Green Power

This is an initiative of the ZigBee Alliance to allow 802.15.4 devices running on a small energy budget to send their data with less energy constraints. Frames are shorter, compared to normal ZB frames. The specifications resulting from this initiative will be presented in due time by those responsible. They will not be discussed in this paper.
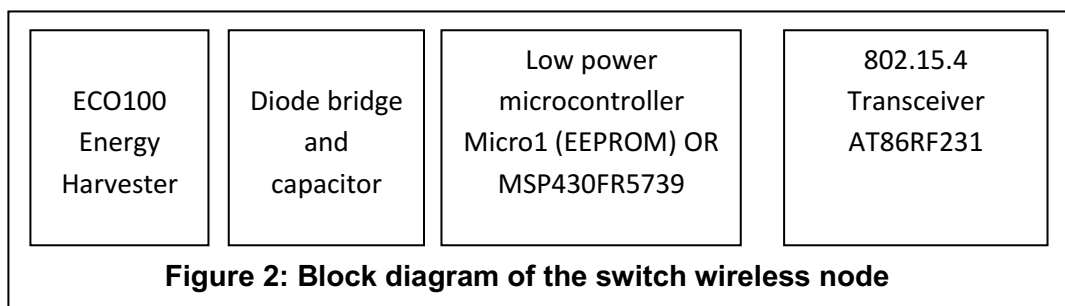
One should however note that using ZigBee frames requires more energy than using Zigbee Green Power frames. Therefore, what works for ZB frames will work even better with ZigBee Green Power.

### 3. Reducing energy needs with the help of a new microcontroller with FRAM

An illustration of the general trend towards the reduction of the energy requirements for wireless sensor nodes will be described in this part. This reduction is achieved by using technologies that allow the microcontroller, transceivers or sensors to consume less energy while doing the same work. Transceivers requiring half of the power of today's 802.15.4 radios have been announced. In this work, the new microcontroller of Texas Instruments featuring a non-volatile memory based on FRAM is used. This technology allows data to be written in non-volatile memory faster than when EEPROM or FLASH is used. Less energy is also needed. The memory positions can be updated billions of times (compared to tens of thousands of time with EEPROM, and even less with Flash).

### 3.1. Hardware

The main elements used to build the battery-less wireless node are shown below, and then briefly described.

| ECO100 Energy Harvester | Diode bridge and capacitor | Low power microcontroller Micro1 (EEPROM) OR MSP430FR5739 | 802.15.4 Transceiver AT86RF231 |
|---|---|---|---|

**Figure 2: Block diagram of the switch wireless node**

### FRAM Microcontroller [10]

The device used is the MSP430FR5739 from Texas Instruments. This microcontroller features an FRAM as non-volatile memory. The use of this technology opens the doors to very interesting concepts. The most interesting point in this work is the fact that parameters that need to be updated and kept while the wireless node is active can easily be stored in the FRAM, with little energy. This is very significant for devices powered intermittently. The state of the microcontroller can be saved, and then restored later when enough power is available. There are not many FRAM-based microcontrollers on the market. It some cases, an external serial FRAM memory can be attached to the microprocessor. The firm RAMTRON produces serial FRAM (http://www.ramtron.com/ )

### Microcontroller1

For comparisons with the FRAM technology, we used a second microcontroller that features EEPROM as non-volatile memory. The device is also good enough to allow ZigBee compatible frames to be sent, and update of the APS to be done. We chose not to name this device, in order to avoid misunderstandings. The main point in this work is simply to show the difference between using an EEPROM and using an FRAM. It is not the comparison of microcontrollers. We do not pretend here that the MSP430 is a better core.

### Transceiver [8]

As transceiver, the AT86RF231 from the firm ATMEL was used. It is a single chip radio designed for 802.15.4 applications in the 2.4 GHz frequency band. This radio works from 3.6 volts down to 1.8 volt. It communicates with the microcontroller over a serial link and a couple of IOs. In TX mode it typically needs up to 14 mA (+3dBm) and 12.5 mA while receiving.

### Energy harvester [6]

To allow a fair comparison with other wireless systems, an ECO100 has been used. This electro-dynamic energy harvester is manufactured by the firm EnOcean . It is used in many battery-free wireless switches or similar applications. Its output depends on the mechanical action and the way the harvester is loaded during that mechanical action.
We chose a very easy and low cost diode bridge-capacitor configuration. This means that the way the power-on-reset of the microcontroller works is important, since it will impact the amount of energy that is built up before the start of the system.

### Voltage conversion and energy management.

The transfer of the available energy can be optimised by using the appropriate electronic circuit before the load. Such a circuit will also be used to limit the voltage generated and possibly keep it at a optimal value to minimise energy loses. This was not done in this work

## 4. Tests and results

2 configurations of wireless battery-free switches were built using the elements above. The first switch used the microcontroller with EEPROM (Micro1), and the second switch was based on the FRAM microcontroller (MSP430FR5739). Both switches used the same transceiver, Atmel AT86RF231.
Software was written to allow ZigBee compatible frames to be sent, and the effects observed on a lamp controlled using a ZigBee compatible receiver (from the firm Adhoco in Winterthur, Switzerland). The messages were used to toggle a lamp.
A sniffer was also used to observe the frames sent.
The current profile of each of the switch was recorded.
- Using a stable power source
- Using the ECO100 energy harvester as energy source

The software written for the tests was kept simple, with only the needed functions. The important parameters are kept in a table. The values that should be changed are modified after the start of the microcontroller, and the table is loaded into the transceiver for sending.
In this way, switch status or sensor values can be computed and directly written in the proper positions in the table. Microcontroller low power modes are used, in order to reduce energy needs.

The results are shown in the figures below 3A, 3B, 4A, 4B, 5,6.

They basically confirm that:
- Sending ZB compatible frames to control a lamp is possible with both microcontrollers combined with the transceiver. (Figures 3A/3B 4A/4B)

- In cases where non-volatile memory is needed, EEPROM can be used. It will however require more energy than FRAM.

- It is even possible to send more than 1 frame (Figure 5, Figure 6)

It can be seen that using FRAM enables substantial energy savings. That energy can be used for other tasks. For example

- to receive an ACK signal and use if for channel change

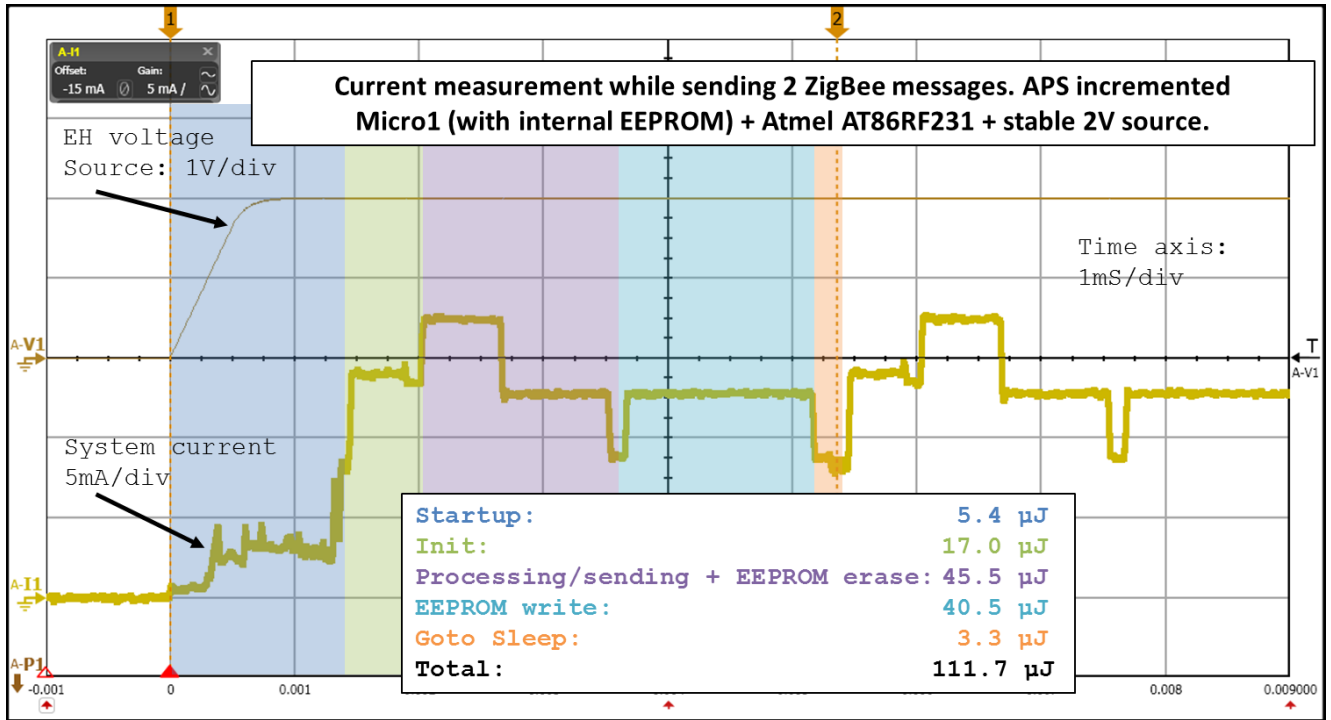- to send a longer frame

- to send the same frame a second time

**Current measurement while sending 2 ZigBee messages. APS incremented Micro1 (with internal EEPROM) + Atmel AT86RF231 + stable 2V source.**

EH voltage Source: 1V/div

Time axis: 1mS/div

System current 5mA/div

| Startup: | 5.4 µJ |
|---|---|
| Init: | 17.0 µJ |
| Processing/sending + EEPROM erase: | 45.5 µJ |
| EEPROM write: | 40.5 µJ |
| Goto Sleep: | 3.3 µJ |
| Total: | 111.7 µJ |

**Figure 3A : A ZB compatible message with APS modification in non-volatile memory is sent using a stable 2V supply**



**Current measurement while sending 1 ZigBee message. APS incremented MSP430FR5739 (FRAM micro) + Atmel AT86RF231 + stable 2V source.**

Stable voltage Source: 1V/div

System current 5mA/div

Time axis: 0.5mS/div

| Startup: | 5.7 µJ |
|---|---|
| Init: | 6.0 µJ |
| Processing/APS update | |
| Sending: | 25.9 µJ |
| Goto Sleep: | 1.7 µJ |
| Total: | 39.3 µJ |
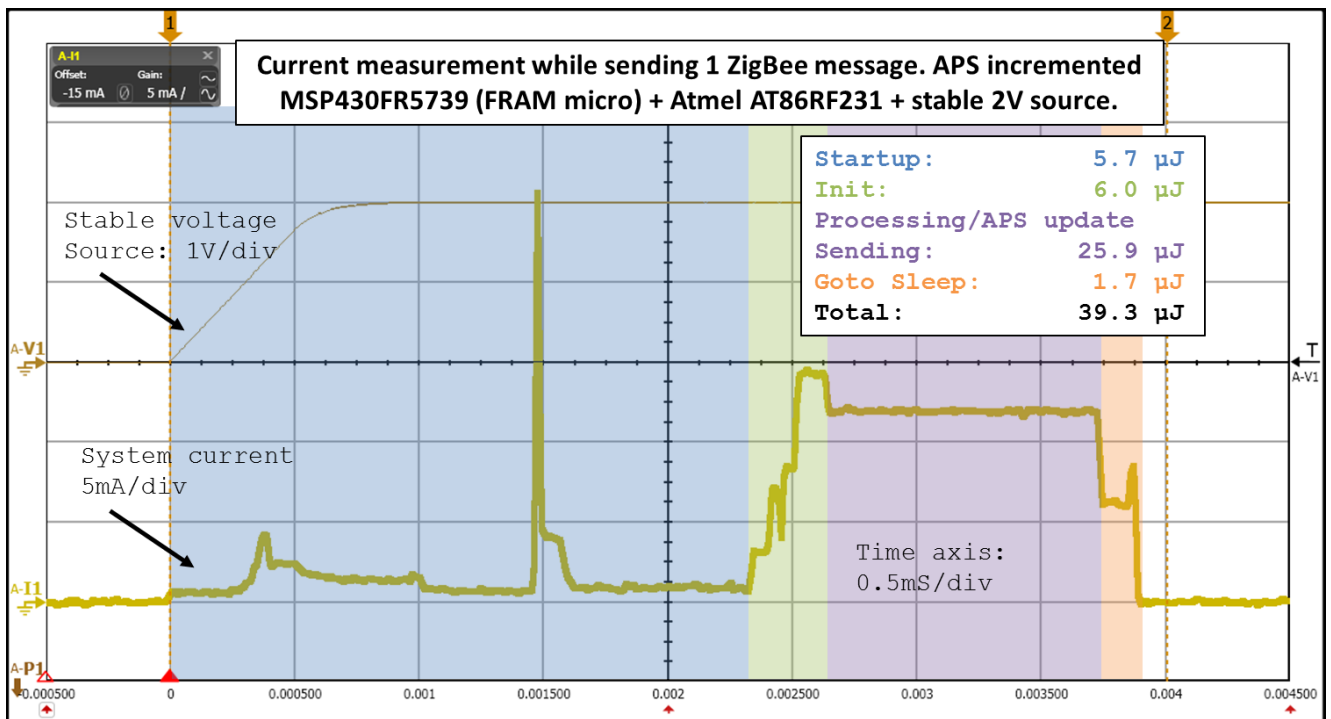
**Figure 3B : Less energy is required using FRAM**
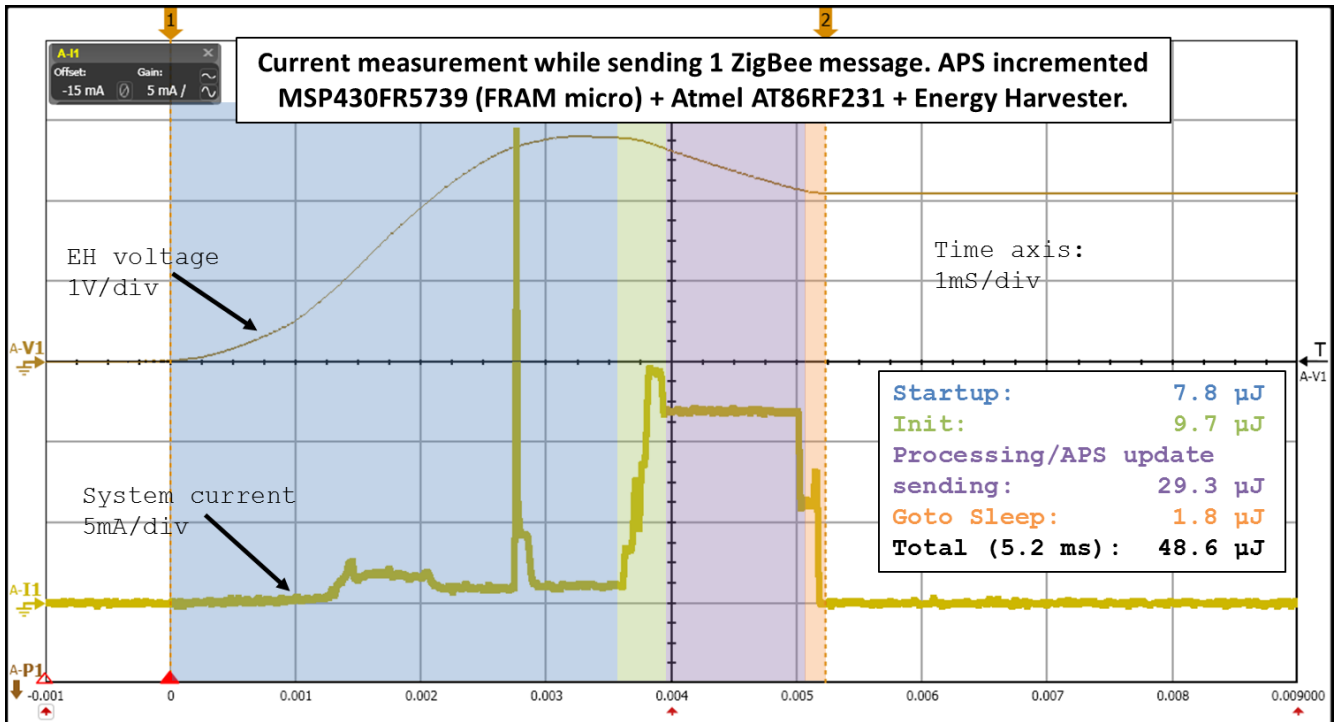
**Figure 4A : A ZB compatible message with APS modification in non-volatile memory is possible on less than 50 µJ using the energy harvester.**
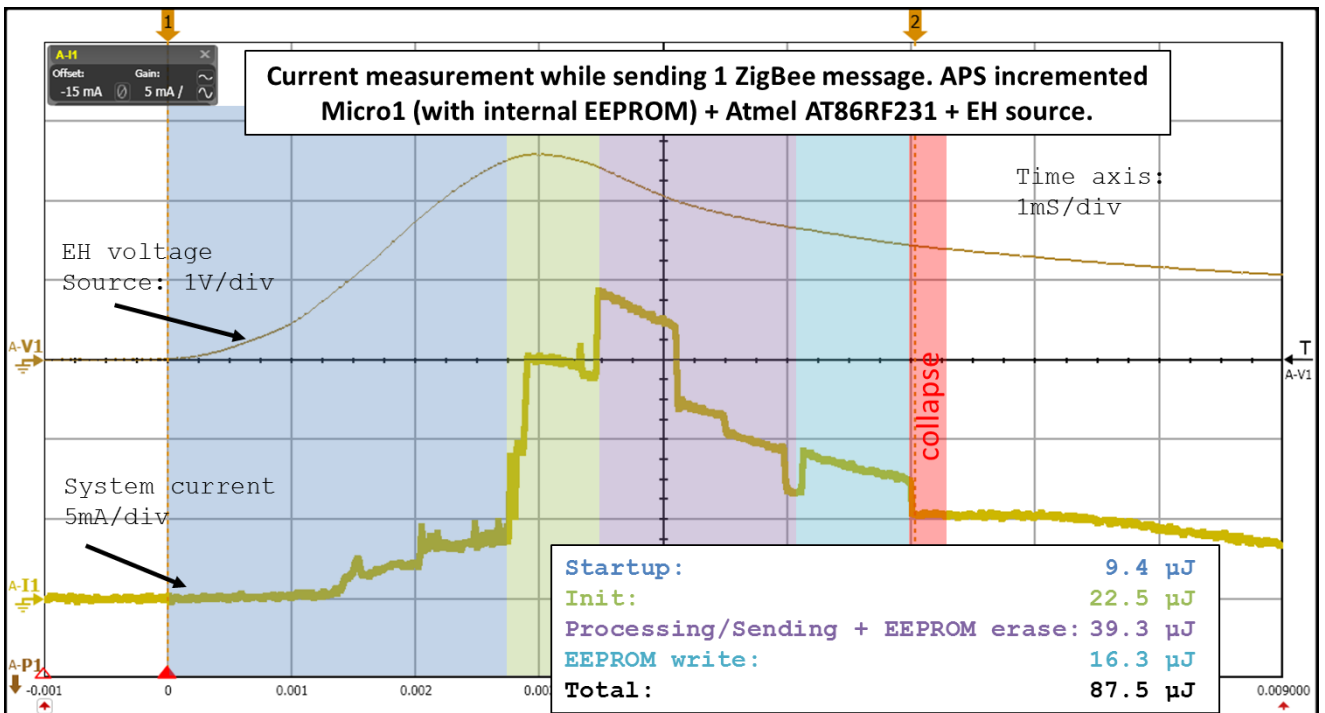


**Figure 4B : More energy is required if a microcontroller with EEPROM is used. Here the voltage gets dangerously low during the update of the EEPROM.**
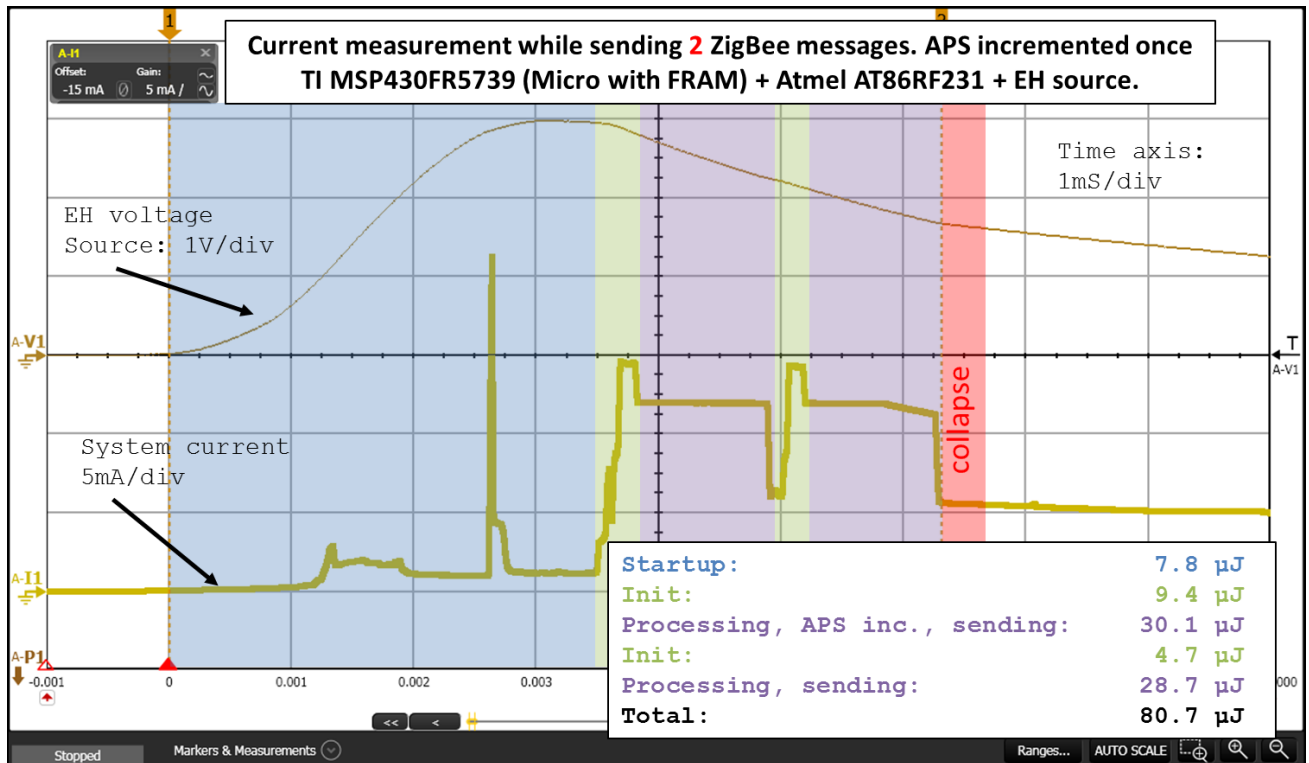
**Figure 5 : There is enough energy to update the APS and send 2 messages (complete frames).**

*Packet index: 1  Length: 32*
*Raw data (hex): 41 88 01 32 00 01 00 6F 79 48 00 00 00 6F 79 0A 01 00 0A 06 00 04 01 10*
***07** 11 53 02 46 C9*
*RSSI [dBm]: -52  Correlation value: 108  CRC OK: 1*

*Packet index: 2  Length: 32*
*Raw data (hex): 41 88 01 32 00 01 00 6F 79 48 00 00 00 6F 79 0A 01 00 0A 06 00 04 01 10*
***07** 11 53 02 46 C9*
*RSSI [dBm]: -53  Correlation value: 108  CRC OK: 1*

*Packet index: 3  Length: 32*
*Raw data (hex): 41 88 01 32 00 01 00 6F 79 48 00 00 00 6F 79 0A 01 00 0A 06 00 04 01 10*
***08** 11 53 02 46 C9*
*RSSI [dBm]: -55  Correlation value: 108  CRC OK: 1*

*Packet index: 4  Length: 32*
*Raw data (hex): 41 88 01 32 00 01 00 6F 79 48 00 00 00 6F 79 0A 01 00 0A 06 00 04 01 10*
***08** 11 53 02 46 C9*
*RSSI [dBm]: -56  Correlation value: 108  CRC OK: 1*

**Figure 6 : Data captured using a sniffer**

**Conclusions and future work**

Battery-free devices can be used in ZigBee or other 802.15.4 –based wireless networks. There are microcontrollers and radio that can send compatible frames on a low energy budget and a low cost. There is a clear tendency towards devices that need even less power (radio and microcontrollers). Because of the large amount of supplier of microcontrollers and 802.15.4 transceivers, we expect the next generations to run on even less energy.  In this work we have shown that devices with FRAM that are now available on the market lower the energy needs when data should be written in non-volatile memory. This represents a great improvement compared to EEPROM or FLASH memory.

In a future work, we will explore other areas in battery-free applications that can profit from the use of FRAM. We will seek to optimise the software of the FRAM microcontroller and introduce a better energy transfer electronic.

**Acknowledgements**

**References.**

[1]    802.15.4  specifications

[2]    ZigBee Alliance  www.zigbee.org  (specifications)

[3]    A battery-less switch for use in 802.15.4 / ZigBee applications
       M. Meli, A. Bluem, M. Gysel. Second European ZigBee Developer's Conference.
       Munich June 2009

[4]    Bidirectional 868/915 MHz wireless module powered with energy harvester.
       M. Meli, S. Blättler, V. Kunz, M. Gysel
       10. Wireless Technologies Kongress, September 2008, Bochum

[5]     AT86RF231 datasheet. www.atmel.com

[6]    EnOcean: www.enocean.com

[7]    lightningswitch: www.lightningswitch.com

[8]    ECO100 datasheet. www.enocean.com

[9]    Battery-less wireless Switch with EM9201. www.ines.zhaw.ch

[10]   Datasheet and development tools for FRAM MSP430. www.ti.com

[11]    A Touch and Pair system for battery-free 802.15.4/ZigBee
       Home Automation networks.  Condrau, Zimmermann, Gysel, Wuerms, Meli
       3rd European ZigBee Developers' Conference, Munich 2010

[12]    Harmony™ XB5R. Battery-less push-buton from Schneider Electric
       http://static.schneider-
electric.us/docs/Machine%20Control/Push%20Buttons%20and%20Operator%20Interface/Wi
reless%20Batteryless%2022%20mm/9001CT1105.pdf


[13]   ZigBee compatible sensor powered by one tiny solar cell
       M. Würms, M. Gysel, M. Meli; 4th European ZigBee Conference
       Munich April 2010

[14]   A world technological first for Schneider Electric with the introduction of a ZigBee-
       based wireless, batteryless switch
       From Schneider electric news, web site