embeddedworld2022
Exhibition&Conference
... it's a smarter world

Using New Memory technologies to Reduce the Energy Requirements of LPWAN Nodes

Marcel Meli
University of Applied Sciences, School of Engineering
Institute of Embedded Systems
Winterthur, Switzerland
marcel.meli@zhaw.ch

Stefan Kunz
University of Applied Sciences, School of Engineering
Institute of Embedded Systems
Winterthur, Switzerland

***Abstract*— The reduction of energy consumption is an important requirement in low-power nodes such as those used in LPWAN or WPAN. This is even more important when dealing with nodes powered by harvested energy. It is therefore important to investigate how energy can be saved. All nodes consume energy during processing and communication. The type of memory used in the nodes can enable new modes that help lower energy requirements. Technologies such as FRAM, MRAM, ReRAM allow interesting improvements compared to Flash, the dominant non-volatile memory technology in current embedded systems. The status of the application may be saved and devices with high leakages completely disconnected from the power supply, resulting in substantial energy savings in nodes that spend much time in low-power modes. In this work, some of the architectures allowing energy savings are discussed. Investigative measurements of a commercial 8-bit ReRAM microcontroller are presented.**

Keywords—ReRAM; RRAM; FRAM; MRAM; energy harvesting; LPWAN; WPAN; low-power

I. INTRODUCTION AND STATEMENT OF THE ISSUES

Many wireless nodes spend a good part of their time in low power modes. They typically wake-up following an event trigger, go into high gear to perform a given task, then go back to sleep. The trigger that wakes them up may come from a timer interrupt that schedules the activities. It may also come from sensors linked to the application or from other events. The node might then take some measurements, process the data, communicate, and then go back into an appropriate low-power mode in order to save energy. In general, designs tend to keep the node as long as possible in the lowest power mode in order to maximize energy savings and thus battery life. In cases where harvested energy is used, the need to save energy is more acute, because of the scarcity of the energy but also because the energy requirements can strongly influence the dimensioning of expensive elements such as the harvester, the energy storage and the power management. In many applications, the node spends more time in low-power mode than in active modes where the operating current is high (several milliamperes). For LPWAN systems, the peak current consumption often comes during the wireless communication. For instance, a LoRa radio transmitting with +14dBm at 3.0V requires tens of milliamperes. Considering the duty cycle limitation that is typically 1% (in Europe for 868 MHz systems), the time spent transmitting is a small fraction of the total time. The device also needs time and resources for other tasks related to the application. But that is unlikely to cover the remaining 99% of the time.

One mode in which devices consume less energy and yet are ready to quickly restart on interrupts is the "RTC on with memory backup". It is a mode where most parts of the MCU are inactive, the internal RTC is activated and driven by a low-power oscillator and vital contents of the application are kept in SRAM (for instance some variables). In that mode, the device may be woken up by the RTC or another event. The current consumption is usually in the range of the microampere to a few microamperes at room temperature. For example, the devices of the EFR32BG24 family of SiLabs typically require a current of 1.3µA in EM2 deep sleep mode with 16 kB RAM retention and the RTC running from LFRCO [1]. The values in other modes can be seen in Fig. 2. The current goes up to 4.2µA in the case of full radio RAM retention. It can also be seen from Fig. 1 that an increase in the device's temperature results in a dramatic rise in the low-power mode current. That behavior is not unique to this device. Fig. 3 shows low-power modes supply currents for the Nordic NRF52840, a popular device, used in many wireless

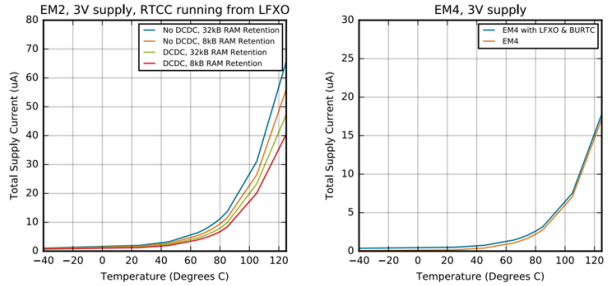

Fig. 1. Dependency of low-power modes current with temperature. Modes EM2 and EM4 typical supply current. Taken from [1].

This work is the result of an internal project of the ZHAW Institute of Embedded System with some aspects related to the AMANDA Project sponsored by the Horizon2020 program under the grant agreement ID: 825464.

nodes. It can also be seen on that figure that its low-power modes consume several microamperes.

Considering a 2µA current (RTC on and RAM retention) and a voltage of 3V, for 50% of the time, the resulting energy per day is nearly 260mJ. At a temperature of 40 degrees C, this will more than double, increasing exponentially with temperature. Nodes that are exposed to the sun will see their temperatures rise faster, depending on the country and the time of the year.

Parameter	Symbol	Test Condition	Min	Typ	Max	Unit
Current consumption in EM2 mode, VSCALE0	I_{EM2_VS}	256 kB RAM and full Radio RAM retention, RTC running from LFXO[1]	—	4.2	—	µA
		256 kB RAM and full Radio RAM retention, RTC running from LFRCO[1]	—	4.2	—	µA
		16 kB RAM and full Radio RAM retention, RTC running from LFXO[1]	—	1.8	—	µA
		16 kB RAM and full Radio RAM retention, RTC running from LFRCO[1]	—	1.9	—	µA
		16 kB RAM and full Radio RAM retention, RTC running from LFRCO in precision mode[1]	—	2.8	—	µA
Current consumption in EM3 mode, VSCALE0	I_{EM3_VS}	256 kB RAM and full Radio RAM retention, RTC running from ULFRCO[1]	—	3.9	—	µA
		16 kB RAM and full Radio RAM retention, RTC running from ULFRCO[1]	—	1.5	—	µA

Fig. 2. Power supply current of in low-power modes for devices of the EFR32BG24 Wireless SoC Family. [2]

Therefore, hundreds of millijoules may be consumed every day to keep the device in low-power modes with the contents of the RAM being retained. This is a non-negligible amount of energy for a small battery or an application with a small solar cell. For instance, for an item running on a CR2032 coin cell battery (250mAh), with an average application current consumption of 5µA, the expected battery lifetime can be estimated as 250mAh/5µA, that is about 5.7 years. Adding 2µA results in 7µA and thus a battery lifetime of about 4 years. There is a substantial reduction of the lifetime. Clearly, higher current values in low-power modes will further reduce that lifetime.

In the case of a system powered with harvested energy, there will also be consequences affecting the harvester, power management and storage used.

- The energy storage will be depleted faster when the system is not harvesting and mostly relying on saved energy. For instance, at night if solar harvesters are used in an outdoor setting.
- The cut-off point at which harvesting stops may be shifted (because of the load current), meaning that the system will stop harvesting earlier. That results in less energy being harvested. For instance, in the case of solar harvesting, the light intensity required for the harvesting process to be useful will be higher. The storage will be activated earlier and will need to deliver energy longer.
- In some cases, the node may run out of energy more often and consequently switch off. This will result in more frequent loss of status (variables values are lost) and possibly more energy losses when reinitializing the system once the power outage is over.
- It is also important to remember that frequent charge/discharge cycles can seriously shorten the lifetime of certain types of storage elements, and thus contribute to early dysfunction of the node.

Can some of the energy lost in low-power modes be saved? If yes, how and with which limitations?

One way of alleviating the issues mentioned above is to use low-power non-volatile technologies such as FRAM, ReRAM or MRAM. They may be combined with (or used in place of) Flash memory technologies often found in microcontrollers.

5.2.1.1 Sleep

Symbol	Description	Min.	Typ.	Max.	Units
$I_{ON\_RAMOFF\_EVENT}$	System ON, no RAM retention, wake on any event		0.97		µA
$I_{ON\_RAMON\_EVENT}$	System ON, full 256 kB RAM retention, wake on any event		2.35		µA
$I_{ON\_RAMON\_POF}$	System ON, full 256 kB RAM retention, wake on any event, power-fail comparator enabled		2.35		µA
$I_{ON\_RAMON\_GPIOTE}$	System ON, full 256 kB RAM retention, wake on GPIOTE input (event mode)		17.37		µA
$I_{ON\_RAMON\_GPIOTEPORT}$	System ON, full 256 kB RAM retention, wake on GPIOTE PORT event		2.36		µA
$I_{ON\_RAMOFF\_RTC}$	System ON, no RAM retention, wake on RTC (running from LFRC clock)		1.50		µA
$I_{ON\_RAMON\_RTC}$	System ON, full 256 kB RAM retention, wake on RTC (running from LFRC clock)		3.16		µA
$I_{OFF\_RAMOFF\_RESET}$	System OFF, no RAM retention, wake on reset		0.40		µA
$I_{OFF\_RAMOFF\_LPCOMP}$	System OFF, no RAM retention, wake on LPCOMP		0.86		µA
$I_{OFF\_RAMON\_RESET}$	System OFF, full 256 kB RAM retention, wake on reset		1.86		µA
$I_{ON\_RAMOFF\_EVENT\_5V}$	System ON, no RAM retention, wake on any event, 5 V supply on VDDH, REG0 output = 3.3 V		1.29		µA
$I_{OFF\_RAMOFF\_RESET\_5V}$	System OFF, no RAM retention, wake on reset, 5 V supply on VDDH, REG0 output = 3.3 V		0.95		µA

Fig. 3 Power supply current in low-power modes at room temperature for the NRF52840 device. Taken from [3].

There are several reasons behind the use of Flash technology in embedded systems. It is programmable, which facilitate the update of programs thousands of times; it is well integrated in the manufacturing processes of MCUs, leading to acceptable costs. At this point of time, the overwhelming majority of MCUs use Flash, meaning that they are available in many flavors and lot of software has been written for them, whereas there are hardly any MCUs with FRAM/MRAM/ReRAM.

Flash technology has some disadvantages for as low-power applications:

- There is an important asymmetry in the time and energy required for the content update (write/program), with respect to the same for read operations. In other words, reading is much faster than writing/programming and requires less energy.
- The endurance is typically limited to tens of thousands of cycles, meaning that too frequent changes of the contents might lead to reliability issues.

Some of the typical properties of Flash memories are listed below, especially the NOR version that is often used as program memory in microcontrollers:

- Time to write a memory cell in the range 1ms – 0.1ms
- Read and write operating require high voltage
- Retention time is about 10 years
- Endurance 10^4 to 10^5 cycles

There are several activities dealing with the use of the non-volatile memory technologies to save energy. They are sometimes described as “Normally-off” Computing [9] or

"Intermittent Computing" [10] techniques. Our contributions have mainly been with regard to Energy Harvesting using TEGs or solar cells, combined with FRAM [26,27,29]. Here we discuss the way the different technologies can be used, the parameters that should be taken into account during optimization and the way this might be carried out. We also introduce an analysis of some of the low-power properties of one of the few ReRAM microcontrollers on the market.

II. Short description of some non-volatile memory technologies

Below are short descriptions of the non-volatile memory technologies that are of interest in this work. Commercially available examples of standalone devices or those embedded with MCUs are given.

A. FRAM

Ferroelectric capacitor can be used to store binary states. When the voltage is applied, the capacitor generates an electric field to polarize the ferroelectric dielectric. As a result, the dipoles in the crystal lattice of the ferroelectric align with the electric field. There are two stable states which are maintained even when the external electric field is removed [5]. Some of the properties are [4] [5]:

- Access time to a memory cell is 40 to 100ns
- Over 10^{14} read and write cycles possible
- Programming possible with voltage between 1.3V - 3.3V
- Data retention over 10 years

It is important to note that read operations are destructive and are therefore followed by (in-built) write operations.

Several FRAM serial memory devices of Fujitsu have endurances of 10^{13} read/write cycles [18]. The manufacturer offers devices with parallel, SPI, QSPI, communication interfaces, allowing the use of external FRAM in several architectures.

The MSP430 MCU family of Texas Instruments includes several devices with integrated FRAM. For example, the MSP430FR5043. The FRAM cells have a minimal endurance of 10^{15} for read/write cycles and a fast write operation of 125ns. The minimal data retention is between 10 years (85 degrees C) and 100 years (25 degrees C) [19].

B. ReRAM (or RRAM)

The ability of a ReRAM to store binary information is based on the properties of a memristor. A memristor (a portmanteau of memory and resistor) is a non-linear passive component which can be LRS (low resistance state) or HRS (high resistance state) depending on an external applied voltage [8]. There are several variants of a ReRAM. Some of the properties are [4], [7]:

- Access time to a memory cell is lower than 10ns
- Over 10^{12} write cycles possible
- Programming possible with voltage between 1.3 - 3.3V
- Data retention over 10 years

One of the main advantages of ReRAM is its low-cost and (apparent) simplicity. However, despite early expectations, it has proven difficult to commercially produce ReRAM and several firms gave up their work on that technology [20]. The last years have seen a renewed commercial interest in that technology. ReRAM cells are more limited in the number of times they can be overwritten, compared to FRAM and MRAM. Depending on the application, this aspect can prove (very) limiting. For its MB85AS12MT serial ReRAM, Fujitsu give a write endurance of 500K cycles [17]. Another product of the same manufacturer, the MB85AS8MT, has a write endurance of 1 million cycles. For the Panasonic MCU with integrated ReRAM (on the market since 2013), the write endurance of the program memory is 1000 cycles. For the data memory, that endurance is 100K cycles [14].

C. MRAM

The non-volatile memory element exploits the properties of the ferromagnetic material, in which the magnetic alignment is slightly influenced by electric fields. An MRAM memory cell is made up of two magnetic layers with an insulator between them. One layer is magnetically soft (free layer) and the other magnetically hard (fixed layer) (and therefore unchanging). The binary information is stored in the magnetically soft layer. The magnetic field difference between the two layers is used to distinguish between low and high storage content. Some of the properties are: [8]

- Read and write time to a memory cell is below 10ns
- Over 10^{13} read and write cycles possible
- Programming possible with less than 2V
- Data retention over 10 years

Everspin Technologies [21] offer serial MRAM memories with parallel, SPI and QSPI interfaces. The MR25H10 for example has a retention time of 20 years and the manufacturer claims unlimited write endurance [22]. Greenwaves and Ambiq offer MCUs with embedded MRAM [23, 24, 25], focusing on high performance.

III. Saving energy

Thanks to some of their characteristics such as fast read access, low energy to read and write cells, high endurance, FRAM/MRAM/ReRAM can be used to store information before switching a device off. It means that the status of the application (that is in variables) can be saved and then kept at low energy costs (to save and restore). Several architectures are possible, depending on the application and the devices available. However, one needs to be careful about the write endurance limit of the memories.

A. Memory embedded in the microcontroller

One method is to use a microcontroller that already embeds the needed memory technology. This is the case for applications that run on the FRAM versions of the MSP430. For applications requiring more computing performance, devices such as the Apollo4 or the GAP9 offers integrated MRAM [24, 25]. At the low-end, the 8-bit MCU of Panasonic integrates ReRAM.

B. *Serial FRAM/ReRAM/MRAM and Flash Microcontroller*

It is also possible to add an external serial memory to existing Flash MCUs. This allows the node to be augmented with the advantages of using such memory devices, without having to change the MCU. The appropriate size of the serial memory can be chosen to keep the costs down. A disadvantage is the extra energy required because of the need for appropriate external signal drivers (compared to internal signals if the memory is on the MCU die).

IV. USING THE TECHNOLOGIES

When using those NV-memory technologies, the characteristics of the devices and the requirements of the application should be kept in mind and the whole system designed accordingly. For example, one needs to consider the required lifetime of the node, the type of energy supply (energy harvesting or battery) and obviously the cost. One way of saving energy is to switch off the embedded system (or a large part of it) when it is not needed. Critical parameters are kept in the NV-memory. Fig. 4 and Fig. 5 illustrate the block diagrams of such a system, in the case energy harvesting is used and in the case batteries are used.

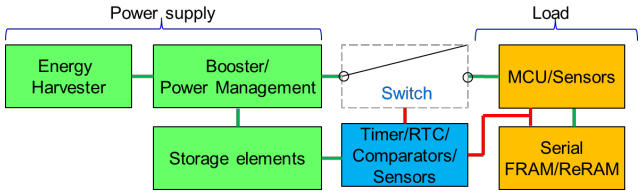

Fig. 4. Block diagram when powering te device with energy harvesting

The load is separated from the energy harvesting/power management with an electronic switch that can be controlled by

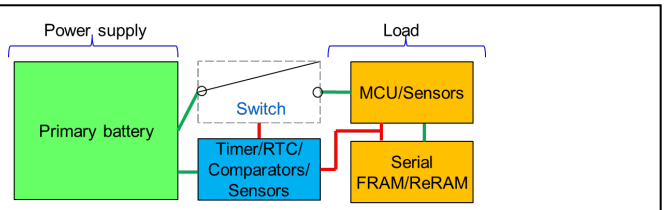

Fig. 5. Block diagram when powering te device with energy harvesting

a timer/RTC and/or by voltage comparators on the power management side. Low power sensors could also be used. Care should be taken to avoid powering the MCU through the connections to the timer/RTC/sensor. The timer or RTC may be a standalone device. Some of those commercially available require less than 100nA (at room temperature). The accuracy of the timer depends on the constraints of the application. The timer may also be integrated in the power management or booster device. The advantage of such an approach is that it reduces the current consumption when devices are not active. This in turn makes the energy harvesting process more efficient.

It is important to know how much energy is required to save and restore the variables that need to be kept. It is also important to know how much energy is required to restart the MCU, the sensors, the transceivers, including the associated clocks. The duration of these activities might also play a role. If the frequency of the events triggering power cycling is such that the energy to save/restore the status and to restart the load is more than the saved energy, then it is not worth it.

Frequent power cycles can have a negative impact on the lifetime of the product if the endurance limits of the NV memory are reached. For instance, updating counters at 5 minutes intervals will lead to over 100K cycles per year and over a million cycles after 10 years. Clearly, this is close to the guaranteed endurance of some ReRAM products that are on the market today. A similar problem might occur for FRAM elements if one is not careful about the number of read and write operations, because reads are followed by write operations.

Due to the store/restore/reboot energy overhead, switching off the embedded system does not always make sense for energy savings. One should compare various parameters to decide when to switch the system off and when not. This can be quite complex, especially if the system is powered with harvested energy and one tries to optimize that process while keeping costs down.

For instance, in the case of solar cells, harvesting at low light conditions helps work longer and gives more freedom in optimizing costs and harvesting time for the node. An example of such gains is presented in [26] where harvesting under 30 lux in order to power a LoRaWAN node is achieved. In recent and not yet published works, operation at even lower illuminations has been achieved [28]. We have built and tested several designs, powered with TEGs or with solar cells. Some examples of the architectures that have been used are:

- MSP340 microcontroller with on-chip FRAM and external LoRa transceiver.
- STM Cortex microcontroller with on-chip Flash memory, on-chip LoRa transceiver, off-chip serial FRAM.
- Apollo2 microcontroller with on-chip Flash serially linked to off-chip LoRa transceiver and serial FRAM.

In those designs, the power management IC already includes elements that help control the load and make decisions based on the energy input, the storage level, … etc. The EM8500/EM8502 devices that are used also include an interface that enables the MCU to read and set power management parameters. Storage levels, input energy and many other parameters can be read or programmed. To facilitate the designs, a configurator program

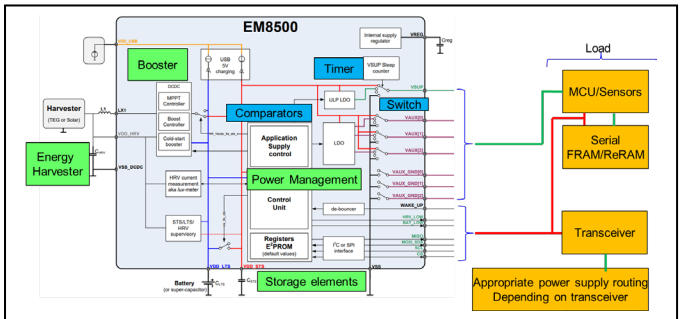

Fig. 6. Principle illustration of system using the EM8500 PM

was written to help visualize the settings of different parameters of the power management. Fig. 6 illustrates the system in a general way, showing the elements that are already included in the power management. These are: booster to allow work with single solar cells; power switch to control the load; low-power timer to control the switch; comparators to set different voltage levels for powering the load.

Fig. 7 and Fig. 8 illustrate some of the parameters that can be set

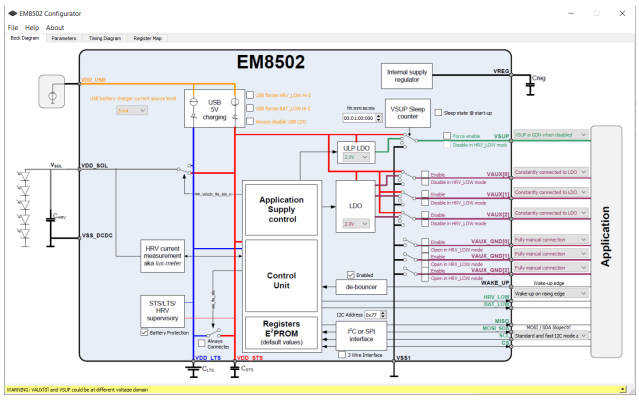

Fig. 7. Basic view of the configurator that allows setting of load switch and timer value.

to control the power management. Many of those values can be changed on the fly, conferring great flexibility to the system to optimize the energy harvesting operation.

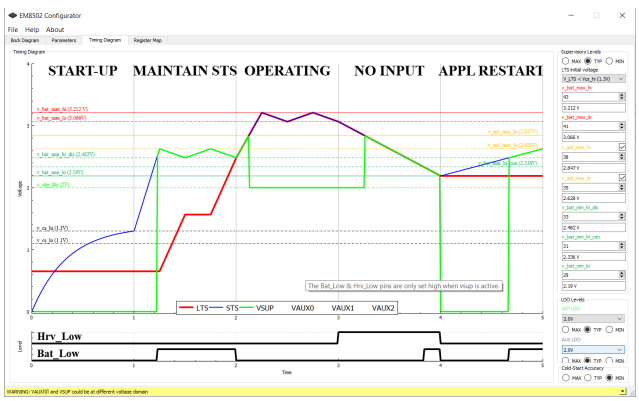

Fig. 8. Setting of comparator values to determine limits when the system may start or stop

Fig. 9 shows a chart that can be used to help make decisions regarding the power cycling of the embedded system, based on the application considerations.

A list of important parameters that are updated regularly is kept. This includes the input power (that can be read from the power management registers or estimated by measuring the storage levels at different times). The tracking of endurance by using appropriate counters is also important.

As long as the amount of energy coming from the harvester is sufficient and the storage element has enough energy, there is no need to switch the system off. Parameters can be kept in RAM. The endurance of NV elements must not be reduced by unnecessarily using them.

If the input energy is below a certain value or if the storage levels get dangerously low, it might be necessary to improve the harvesting process by reducing the load during low-power modes. For that, the load needs to be switched off, if possible, provided that the endurance limit has not yet been reached or is not dangerously close.

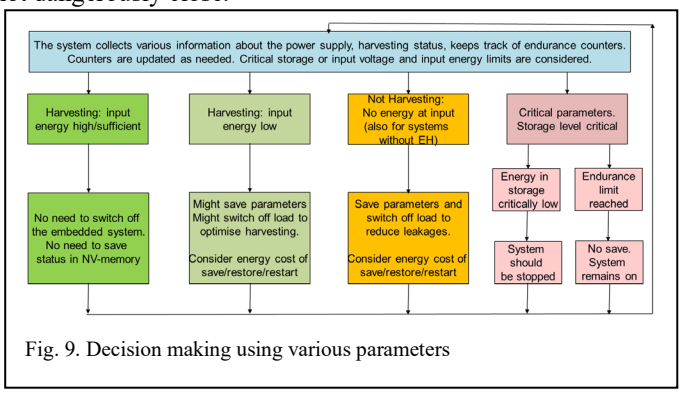

Fig. 9. Decision making using various parameters

When the system is not harvesting (or is a system with a primary battery), one may set a voltage/energy level below which the save/restore procedure should be started to save energy. The endurance of the memory element needs to be taken into account.

If the energy level is critically low, the system might need to be switched off to avoid damaging the storage elements (it depends on the technology of those storage elements).

If the endurance limit has been reached, parameters should no longer be stored in the NV memory, since their correctness is not guaranteed.

V. Working with an MCU that integrates ReRAM

There are not many commercial ReRAM MCUs. However, the last years have seen firms taking important steps to make embedded ReRAM a reality.

Weebit nano announced in April 2022 the completion of successful functional tests on their device that includes a RISC-V and ReRAM [11]. The firm claim a low-power and low-cost memory technology compatible with actual production methods. They also claim that "*Weebit ReRAM typically has 10x-100x better endurance than flash, handling between 100,000 and a million write cycles versus the typical 10,000 program/erase cycles that flash can manage*" [12].

Another firm that is looking into bringing ReRAM in the embedded world is Intrinsic Semiconductor Technology. They are "*raising money to fund a foundry for its ReRAM chip and to build a memory compiler so microcontroller makers can replace SRAM with cheaper, simpler ReRAM*" [13].

There is a product that has been on the market for about 10 years. In first steps towards using it in a wireless IoT node, some of its performances were analyzed and the results are partly presented here. Panasonic Inc launched an embedded ReRAM MCU MN101LR05D in 2013. The microcontroller has an AM13L CPU core with LOAD-STORE architecture with a maximum system clock of 10Mhz. The embedded memory is

divided into 64KB ROM and 4KB RAM. The term ROM is misleading because it refers to the ReRAM unit. In the ROM, a distinction is made between the program area (62KB - 41 KB) and the data area (2 KB - 16 KB). According to the manufacturer, the program area can be overwritten 1000 times and the data area 100000 times, byte by byte. The ReRAM is based on the type with a conductive filament (Ir/Ta2O5/TaOx/TaN stacked film) in a 1T1R arrangement. While the MCU can be operated from 1.1V to 3.6V depending on the operation mode, the programming voltage for the ReRAM is 1.8V to 3.6V. [14]. The specifications given by the manufacturer can be found in Table I.

TABLE I. KEY FEATURES OF PANASONICS RERAM EMBEDDED MCU [14].

Density	64KB
Power Supply	1.1～3.6V
Frequency	10MHz (1.8~3.6V) 1MHz (1.3~3.6V) 40kHz (1.1~3.6V)
Current	2.1mA@10MHz 0.22mA@1MHz 5.6µA@32.768kHz
Endurance	1K (Program Area) 100K (Data Area)
Retention	@ 85°C, >10years

The MN101LR05D can be set to different operating modes. The slow and normal modes differ primarily in the source of the system clock (Table II). The normal mode is operated with a maximum frequency of 10MHz and the slow mode with a frequency of 40kHz. The two modes also differ in the minimum operating voltages. For the slow mode with 40kHz, at least 1.1V must be applied, and for the normal mode with 10MHz, at least 1.8V must be applied. If the normal mode is operated with a maximum frequency of 1MHz, the supply must be at least 1.3V.

TABLE II. OVERVIEW OF THE OPERATING MODES OF THE PANASONIC MN101LR05D

Operation Mode	Clock an CPU Status				Memory	Max	Min.
	HCLK	SCLK	SYSCLK	CPU	ReRAM	SYSCLK	Power Supply
Normal	Active	Active	HCLK	Active	ON	10MHz/1MHz	1.8V/1.3V
HALT0	Active	Active	HCLK	Stop	ON	10MHz/1MHz	1.8V/1.3V
HALT2	Stop	Active	Stop	Stop	OFF	-	1.3V
STOP0	Stop	Stop	Stop	Stop	OFF	-	1.3V
SLOW	Stop	Active	SCLK	Active	ON	40kHz	1.1V
HALT1	Stop	Active	SCLK	Stop	ON	40kHz	1.1V
HALT3	Stop	Active	Stop	Stop	ON	-	1.1V
STOP1	Stop	Stop	Stop	Stop	OFF	-	1.1V

HCLK: High-speed oscillation clock, SCLK: Low-speed oscillation clock, SYSCLK: System Clock

A. *Measurements*

Several measurements were carried out using the AM13L-STK2 starter kit. Before each measurement, it was ensured that the input capacitance was completely discharged by short-circuiting the VDD to GND.

The microcontroller is powered according to Fig. 10. VDD30 is the power supply pin and VDD18 and VDD11 are internal power output pins. The figure shows the arrangement for measuring voltage and current.

After the start-up, the MCU is set into different operating and sleep mode (Table III). The power consumption is measured.

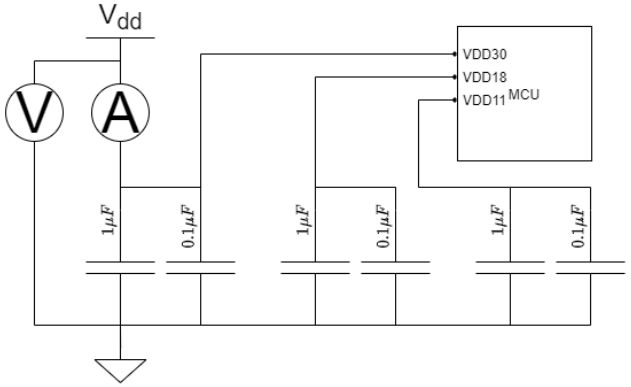

Fig. 10. Measurement setup

The measurement results as shown in Table III are mainly within the range of the manufacturer's specifications.

TABLE III. MEASURED VALUES STARTUP, OPERATING HALT AND STOP MODES

Setup	**Startup**	**Startup**		**Workloop / Halt0/ Halt2 / Stop0**
Clock speed	Voltage supply	Duration	Energy consumption	meanpower
10 MHz	1.8 V	184.00 ms	11.70 uJ	3632.0 uW / 540.17 uW / 24.36 uW / 0.33 uW
1 MHz	1.8 V	192.00 ms	11.56 uJ	408.21 uW / 142.88 uW / 18.35 uW / 2.39 uW
1 MHz	1.3 V	186.00 ms	6.74 uJ	280.61 uW / 93.27 uW / 8.00 uW / 0.51 uW
Setup	Startup	Startup		Workloop / Halt1 / Halt3 / Stop1
Clock speed	Voltage supply	Duration	Energy consumption	meanpower
40 KHz	1.8 V	179.00 ms	10.55 uJ	24.43 uW / 11.42 uW / 9.63 uW / 0.48 uW
40 KHz	1.3 V	167.00 ms	5.60 uJ	16.84 uW / 7.93 uW / 6.99 uW / 0.15 uW
40 KHz	1.1 V	170.00 ms	4.13 uJ	13.82 uW / 6.39 uW / 5.64 uW / 0.10 uW

The ReRAM can be read out in the normal or slow mode at any supply voltage and does not show any significantly higher power consumption. Data can be read after passing the start-up phase. The supply voltage for writing to the ReRAM must be at least 1.8V.

Fig. 11 to Fig. 14 show the start-up measurement with different settings. In each case, the ReRAM is written 10 times. The clock frequency of the CPU and the write frequency of the memory are varied. 2 or 20 bytes are written per write process. Writing to the ReRAM is done via a pre-compiled subroutine on the memory. The MCU provides a subroutine for writing 1 byte and another one for writing 2 to 64 bytes. The same subroutine

is always used in the following measurements. Therefore, the smallest written unit in our measurements is 2 bytes.

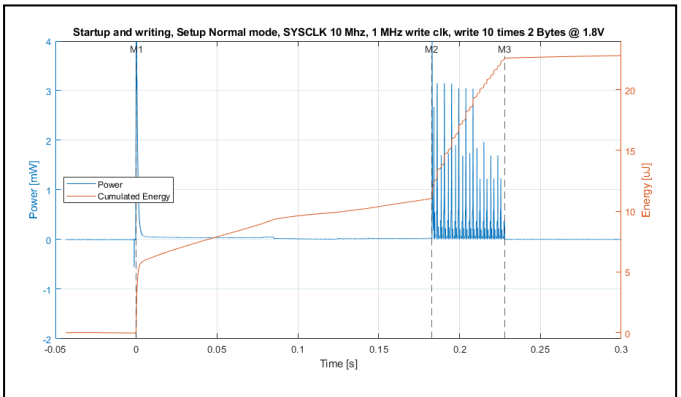

Fig. 11. Startup MCU in Normal Mode (SYSCLK: 10MHz) and then write 10 times 2 Bytes to ReRAM with min write speed (1 MHz write clk)

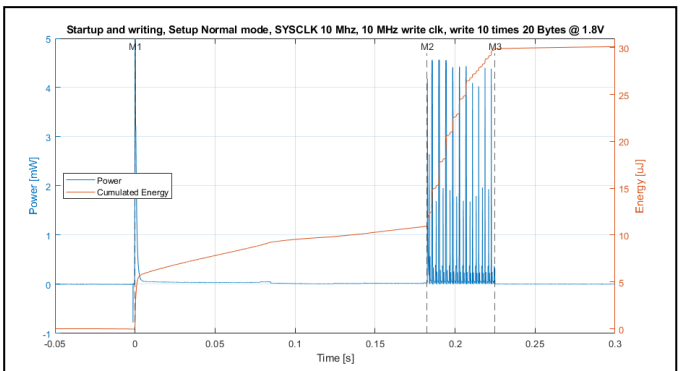

Fig. 12. Startup MCU in Normal Mode (SYSCLK: 10MHz) and then write 10 times 20 Bytes to ReRAM with max write speed (10 MHz write clk)

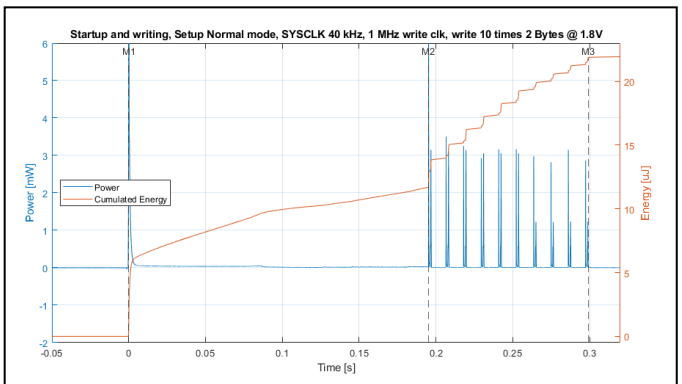

Fig. 13. Startup MCU in Slow Mode (SYSCLK: 40Khz) and then write 10 times 2 Bytes to ReRAM with min write speed (1 MHz write clk)

It is worth noting that the writing process does not always require the same amount of energy. It depends on whether the content (the resistance state) is changed or not when the cell is overwritten. In our measurements (shown in Fig. 11 to Fig. 14) the content of the memory cell was changed in the first 6 write accesses. In the remaining four write operations, the value stored in the memory cell was the one already in it (so no change). This

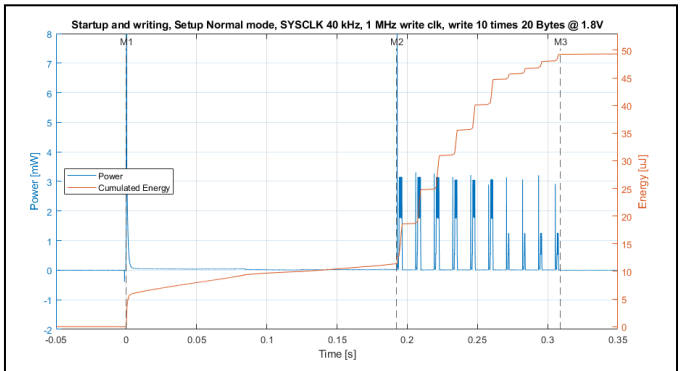

Fig. 14. Startup MCU in Slow Mode (SYSCLK: 40Khz) and then write 10 times 20 Bytes to ReRAM with min write speed (1 MHz write clk)

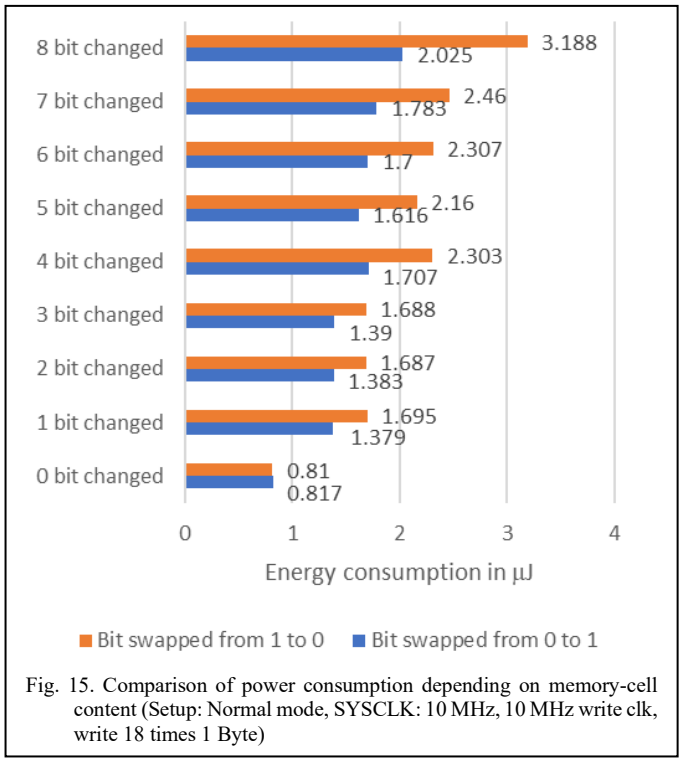

Fig. 15. Comparison of power consumption depending on memory-cell content (Setup: Normal mode, SYSCLK: 10 MHz, 10 MHz write clk, write 18 times 1 Byte)

effect is more pronounced as the number of bits that changes increases. A comparison is shown in Figure 12. (MCU in normal mode with system clock of 10 MHz at a maximum write clock of 10 MHz). One byte is written at the time, with different number of bit changes. The fewer the number of bit changes, the less the required energy for the write process. This can be used to define some variables in a way that the required energy is minimized.

An interesting feature of the MCU is the fact that a change in the supply voltage can trigger an interrupt. The MCU could be used in systems where the power supply is not constant, and the voltage fluctuates. This is illustrated in the fictitious scenario of Fig. 16. The MCU is initially not powered, and supply capacitors are discharged. A supply voltage of 1.1V is applied. The MCU starts in slow mode (with a system clock of 40kHz), is initialized and switches regularly between sleep mode and full load. The ReRAM can be read at an operating voltage of 1.1V,

but not written. After a short time. The supply voltage changes to 3V. In an interrupt routine, two times 20 bytes are written to the ReRAM. The MCU then returns to its previous task. After a short time, the supply voltage drops again. An adaptation to the supply voltage has the potential of reducing the complexity of the power management of the energy harvesting system while allowing the device to run longer, as the amount of energy available is reduced, albeit at lower CPU frequencies.

The measurement in Fig. 16 also shows that, as expected, the power consumption of the device strongly depends on the of the applied supply voltage. The average power of about 10µW at 1.1V is more than four times lower than at a supply voltage of 3V.

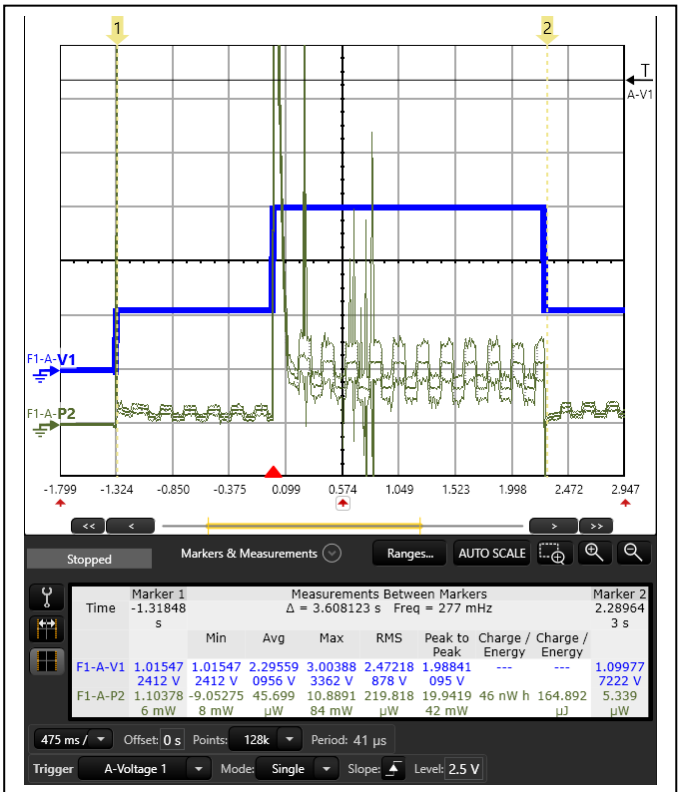

Fig. 16. Measurement of the supply voltage (1V/div) and the power (50uW/div) of a fictitious scenario

CONCLUSIONS AND FUTURE WORK

New memory technologies such as ReRAM, FRAM, MRAM offer possibilities to reduce the energy consumption of IoT nodes. They have various features, and it is important to take those into account in order to optimize energy consumption. In the case of system powered with harvested energy, the power management constraints are also important. MCU that embed those memory elements are appearing on the market and will bring interesting advantages and challenges.

The ReRAM device and the techniques presented here will be further investigated and used to build and evaluate LPWAN or WPAN IoT nodes. New architectures such as those embedding MRAM will also be investigated.

REFERENCES

[1] Datasheet(page 63)
https://www.silabs.com/documents/public/data-sheets/efr32fg22-datasheet.pdf

[2] Datasheet (page 33)
https://www.silabs.com/documents/public/data-sheets/efr32bg24-datasheet.pdf

[3] https://infocenter.nordicsemi.com/pdf/nRF52840_PS_v1.7.pdf (page 59)

[4] J. S. Lee, S. Lee and T. W. Noh, “Resistive switching phenomena: A review of statistical physics approaches,” Applied Physics Reviews, 2015.

[5] L. Stiny, Aktive elektronische Bauelemente, Springer Fachmedien Wiesbaden, 2019.

[6] K. Ando, S. Fujita, M. Hayashikoshi and Y. Fujimori, “Non-volatile Memories,” in Normally-Off Computing, Springer Japan, 2017, pp. 27 - 55.

[7] F. Zahoor, T. Z. A. Zulkifli and F. A. Khanday, “Resistive Random Access Memory ({RRAM}): an Overview of Materials, Switching Mechanism, Performance, Multilevel Cell (mlc) Storage, Modeling, and Applications,” Nanoscale Research Letters, vol. 15, 2020.

[8] S. Pal, S. Bose and I. Aminul, “Design of memristor based low power and highly reliable {ReRAM} cell,” Microsystem Technologies, 2019.

[9] T. Nakada, T. Shimizu and H. Nakamura, "Normally-off computing for IoT systems," 2015 International SoC Design Conference (ISOCC), 2015, pp. 147-148, doi: 10.1109/ISOCC.2015.7401761.
https://ieeexplore.ieee.org/document/7401761

[10] Sumanth Umesh, Sparsh Mittal,
A survey of techniques for intermittent computing,
Journal of Systems Architecture, Volume 112, 2021
https://www.sciencedirect.com/science/article/pii/S1383762120301430

[11] On-line press release
https://www.weebit-nano.com/news/press-releases/weebit-nano-demo-chips-integrating-its-embedded-reram-module-successfully-complete-functional-testing-phase/

[12] On-line information from firm about their product
https://www.weebit-nano.com/technology/reram-advantages/

[13] https://blocksandfiles.com/2022/03/09/intrinsic-semiconductor-technology-guru-talks-about-reramifications/

[14] Datasheet of MN101LR05D
https://industrial.panasonic.com/content/data/SC/ds/ds4/MN101L05__E.pdf

[15] Datasheet of EM8500
https://www.emmicroelectronic.com/sites/default/files/products/datasheets/8500-ds.pdf

[16] Datasheet of EM6502
https://www.emmicroelectronic.com/sites/default/files/products/datasheets/8502-ds.pdf

[17] Datasheet of device
https://www.fujitsu.com/jp/group/fsm/en/documents/products/reram/lineup/MB85AS12MT-DS501-00071-0v1-E.pdf

[18] Features of product
https://www.fujitsu.com/jp/group/fsm/en/products/fram/features/

[19] Datasheet of device
https://www.ti.com/lit/ds/symlink/msp430fr5043.pdf

[20] On-line article
https://semiengineering.com/what-happened-to-reram/

[21] Firm website
https://www.everspin.com/

[22] Datasheet of MR25H10
https://www.everspin.com/supportdocs/MR25H10CDC?npath=259

[23] https://www.eenewsanalog.com/en/greenwaves-stresses-gap9s-hearing-credentials/

[24] Gap9 product brief
https://greenwaves-technologies.com/wp-content/uploads/2022/04/Product-Brief-GAP9-Sensors-General-V1_13_new.pdf

[25] Brief description of apollo4 MCU
https://ambiq.com/apollo4/

[26] M. Meli et al., "Low Light Energy Autonomous LoRaWAN Node," 2020 IEEE 5th International Symposium on Smart and Wireless Systems within the Conferences on Intelligent Data Acquisition and Advanced Computing Systems (IDAACS-SWS), 2020, pp. 1-6, doi: 10.1109/IDAACS-SWS50031.2020.9297089
https://ieeexplore.ieee.org/document/9297089

[27] M. Meli, C. Furter, K. Karakostas, C. Kouzinopoulos and D. Tzovaras, "Low Power LoRaWAN Node Based on FRAM Microcontroller," 2020 IEEE 5th International Symposium on Smart and Wireless Systems within the Conferences on Intelligent Data Acquisition and Advanced Computing Systems (IDAACS-SWS), 2020, pp. 1-6, doi: 10.1109/IDAACS-SWS50031.2020.9297075.
https://ieeexplore.ieee.org/document/9297075

[28] Not yet published work. LPWAN node is powered with small solar cells down to 20 lux

[29] M. Meli. L. Hegetschweiler
Harvesting energy from trees in order to power LPWAN IoT nodes
Wireless Congress 2018, Munich, 21-22 March 2018
https://doi.org/10.21256/zhaw-18842