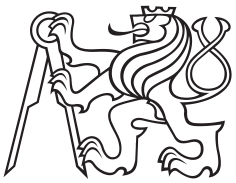
Master's Thesis



Czech
Technical
University
in Prague

F3

Faculty of Electrical Engineering
Department of Measurement

Wireless radiation monitoring system for particle accelerators

Martin Cejp

Supervisor: doc. Ing. Jan Fischer, CSc.
Field of study: Cybernetics and Robotics
August 2020

MASTER'S THESIS ASSIGNMENT

I. Personal and study details

Student's name: **Cejp Martin** Personal ID number: **417292**

Faculty / Institute: **Faculty of Electrical Engineering**

Department / Institute: **Department of Measurement**

Study program: **Cybernetics and Robotics**

Branch of study: **Cybernetics and Robotics**

II. Master's thesis details

Master's thesis title in English:

Wireless Radiation Monitoring System for Particle Accelerators

Master's thesis title in Czech:

Bezdrátový systém pro monitorování radiace v urychlovačích částic

Guidelines:

Design and implement a wireless radiation monitoring system for particle accelerators. The solution should include a data acquisition unit and data transmission. The system will be able to log the data into a central database using the LoRa standard for data transmission. Use the existing hardware platform for data acquisition ("BatMon"), and develop embedded software for the device with an emphasis on reliability and minimization of current consumption during battery-powered operation. Select a proper software solution for aggregation and presentation of the collected data or propose a custom solution. Validate the complete monitoring system experimentally. Evaluate the effect of ionizing radiation on the reliability and parameters of the monitoring system.

Bibliography / sources:

[1] Baumann, R. & Kruckmeyer, K.: "Radiation Handbook for Electronics". Texas Instruments, 2019. Available online: https://ti.com/radbook
[2] Spiezia, G. et al.: "A New RadMon Version for the LHC and its Injection Lines". In IEEE Transactions on Nuclear Science, vol. 61, no. 6, pp. 3424-3431, Dec. 2014.
[3] Brucoli, M. et al.: "Floating Gate Dosimeter Suitability for Accelerator-Like Environments", in IEEE Transactions on Nuclear Science 64 (2017): 2054-2060.

Name and workplace of master's thesis supervisor:

doc. Ing. Jan Fischer, CSc., Department of Measurement, FEE

Name and workplace of second master's thesis supervisor or consultant:

Date of master's thesis assignment: **15.01.2020** Deadline for master's thesis submission: **14.08.2020**

Assignment valid until:
by the end of winter semester 2021/2022

doc. Ing. Jan Fischer, CSc. Supervisor's signature	Head of department's signature	prof. Mgr. Petr Páta, Ph.D. Dean's signature

III. Assignment receipt

The student acknowledges that the master's thesis is an individual work. The student must produce his thesis without the assistance of others, with the exception of provided consultations. Within the master's thesis, the author must state the names of consultants and include a list of references.

. | Date of assignment receipt | Student's signature

Acknowledgements

This thesis builds upon years of work carried out at CERN, in particular by the EN-SMM-RME section led by Salvatore Danzeca.

I must thank my more experienced colleagues whose help has been invaluable in the beginning, especially Matteo Brucoli for sharing his expertise in preparing radiation tests and interpreting their results.

My supervisor Jan Fischer has provided excellent guidance and important feedback, and I greatly appreciate that he agreed to take on this responsibility.

Special gratitude then goes to my family and those close to me, for their continuous, unconditional support.

Declaration

I declare that the presented work was developed independently and that I have listed all sources of information used within it in accordance with the methodical instructions for observing the ethical principles in the preparation of university theses.

This thesis was created during a Technical Studentship at the European Organization for Nuclear Research (CERN).

In Prague, August 14, 2020

.....

Abstract

This thesis describes the implementation of a wireless, battery-powered radiation monitoring system for particle accelerators. The system is based around a sensor node using a floating-gate dosimeter and static random-access memories as radiation sensor. Data is transmitted using LoRaWAN, which enables long communication range without excessive infrastructure costs. The sensor node is modular, allowing reuse of the microcontroller-based platform for other purposes than radiation measurement.

Furthermore, a suitable database schema and user interface for data retrieval are presented. The work concludes by discussing results of performed tests, including in radiation-test facilities and in the Large Hadron Collider (LHC) tunnel.

Keywords: data logging, dosimetry, embedded systems, floating-gate dosimeter, FGDOS, InfluxDB, LoRa, LoRaWAN, low-power, radiation-tolerant, single-event effects

Supervisor: doc. Ing. Jan Fischer, CSc.

Abstrakt

Práce se zabývá realizací systému monitorování radiace v urychlovačích částic, založeného na senzorické jednotce s provozem na baterie a bezdrátovou komunikací. Pro měření radiace je použit dozimetr s plovoucím hradlem (FGDOS) a paměť typu SRAM. Přenosová technologie LoRaWAN umožňuje snížení nákladů oproti komunikaci po kabelech. Senzorická jednotka je modulární, takže vzniklou platformu, sestávající z mikrokontroléru, modemu a datové paměti, je možné použít pro další aplikace.

Součástí práce je také návrh databáze měření a uživatelského rozhraní pro prohlížení získaných dat. Systém byl podroben testu v laboratoři pro ozařování gamma částicemi a v tunelu urychlovače LHC.

Klíčová slova: bateriový provoz, bezdrátový přenos dat, dozimetrie, FGDOS, InfluxDB, LoRa, LoRaWAN, vestavěné systémy, záznam dat

Překlad názvu: Bezdrátový systém pro monitorování radiace v urychlovačích částic

Contents

Figures

Tables

Chapter 1

Background and motivation

In its mission to push the boundaries of science and our understanding of the universe around us, the European Organization for Nuclear Research (CERN) operates the largest particle physics laboratory in the world. Six accelerators and one decelerator form a network enabling progressive acceleration of particle beams before delivering them to one of the several installed experiments. In the Large Hadron Collider, protons have reached energies of up to 6.5 TeV, yielding a collision energy of 13 TeV[1] for a pair of beams. The accelerator complex is being continuously upgraded to reach ever-higher energies and enable new experiments, such as the High Luminosity LHC project that is currently underway.

Through several mechanisms, accelerators produce fields of electromagnetic and particle radiation that extend beyond the accelerator's interior. The main mechanisms are:

- imperfect collimation of the beam, causing particles to deviate from the desired trajectory and collide with the beam pipe (*beam losses*)
- collisions between particles of the beam and residual particles in the beam pipe, which is depressurized to between 10^{-7} and 10^{-9} Pa
- beam-beam collisions in experiments and beam-target collisions in *beam dumps*

Of particular concern is ionizing radiation, due to its adverse effects both on human health and on technical equipment. In addition, free neutrons emitted during operation of the machine can cause *activation* of matter in the machine itself and surrounding facilities. Activated material continues to produce radiation as a *secondary source*, even after the primary source has been shut down or removed.

To limit the scope, this work focuses solely on radiation protection of *equipment*, rather than *personnel*. Accelerators contain large amounts of electronics needed not only to ensure good performance, but also to protect the accelerator itself from the massive energies of beams within. Cost constraints dictate that most of these electronics be based on commercial off-the-shelf components, which do not have a guaranteed radiation performance.

[1] 13 TeV $= 2.083 \cdot 10^{-6}$ J per collision

1.1 Radiation field in the LHC and the associated risks

The LHC is surrounded by a *mixed radiation field*. This field is made up of various particle types, with an energy spectrum spanning more than 12 orders of magnitude. Radiation threatens the reliability of electronics in the LHC via several mechanisms, which can be categorized into two broad classes: **cumulative effects** and **single-event effects**. Cumulative effects cause gradual deterioration of the properties of semiconductors. Single-event effects manifest as sudden disturbances or errors in the operation of circuits.

It is single-event effects (SEEs) that are an immediate concern for reliability: according to [1], in 2012, radiation-induced errors caused 70 out of the 409 premature beam aborts, leading to over 300 hours of downtime. It can take several hours before a new beam is ramped up to full power, and therefore precious time for physics experiments is being lost. The effects of cumulative effects are not so dramatic, but they can also lead to sudden failure in complex systems and they add up to increased costs when equipment must be replaced.

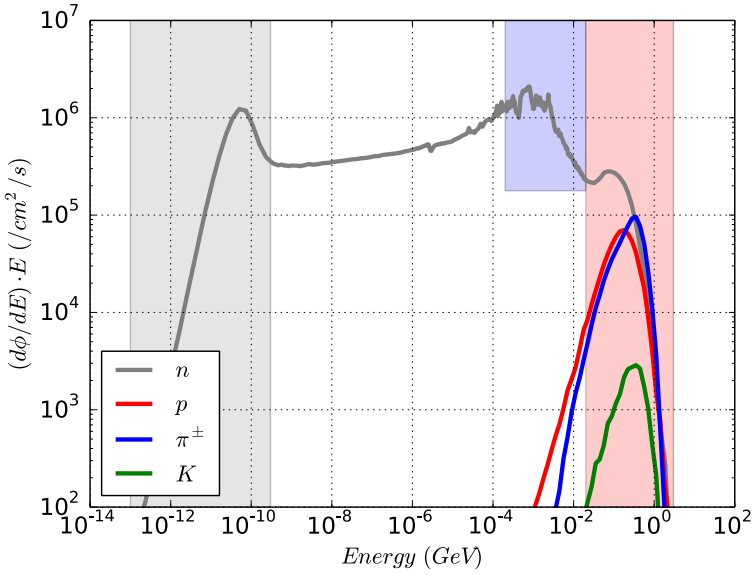


Figure 1.1: Simulated lethargy spectra for the SEE-relevant hadrons at an LHC-like test location at the CHARM test facility. The different shaded regions represent, in an approximate manner, the thermal neutron (gray), intermediate neutron (blue) and HEH fluxes (red). From [1].

In order to describe the radiation field quantitatively, introducing a bit of terminology will be necessary:

Flux denotes the number of particles passing through a unit of area per unit of time. Typically expressed in units of $\mathrm{cm}^{-2} \cdot \mathrm{s}^{-1}$ in relation to a specific class of particles, e.g. *hadron flux* or *electron flux*.

Fluence can be understood as flux integrated over a period of time – the number of particles that would have passed through an area of unit size: cm^{-2}.

Cross-section refers to the frequency of some *event* occurring in an electronic device proportionally to the flux, and can be therefore understood as the physical size of the sensitive part of the component, multiplied by a sensitivity factor (the proportion of particles that cause an event). The cross-section is reported with respect to a specific device, the class of particles considered, and the type of events counted. For example, if a device with a *neutron SEU cross-section* of 0.01 mm^2 is exposed to a flux of 10^3 neutrons $\cdot\, \mathrm{cm}^{-2} \cdot \mathrm{s}^{-1}$, on average 1 SEU will occur every 10 seconds.

Ionizing dose is the cumulative energy per unit of mass, deposited by ionizing radiation into object, expressed using the SI unit *gray*; $1\,\mathrm{Gy} = 1\ \mathrm{J} \cdot \mathrm{kg}^{-1}$. The time derivative of dose is *dose rate*.

According to [2], expected radiation levels in most of the LHC tunnel area vary between 10^7 and 10^{12} $\mathrm{cm}^{-2} \cdot \mathrm{year}^{-1}$ in high-energy hadron flux and 0.01 to $1000\,\mathrm{Gy} \cdot \mathrm{year}^{-1}$ in ionizing dose. As another example, figure 1.1 illustrates the energy spectrum of SEE-causing hadrons in the LHC.

1.1.1 Basic mitigation of radiation effects

In some cases, the undesirable radiation effects can be mitigated through one of the following techniques.

1. *Reducing the flux by increasing distance.* For isotropic sources, of the radiation the flux diminishes with the squared distance from the source. Equipment placed further away will thus experience lower fluxes. However, certain critical equipment (collimators, magnets, beam monitors) must be, by definition, placed very close to the beam, in which case this mitigation is not suitable.
2. *Reducing the flux by shielding the sensitive equipment.* The appropriate material and mass of shielding depends on the particle and energy spectrum of the radiation field. In particle accelerators, blocks of concrete are often used due to its low cost and ability to stop both gamma and neutron radiation. However, in critical areas the required thickness would be on the order of several meters, which again might be impractical for electronics that need to be a part of the accelerator itself.
3. *Reducing the effective cross-section.* This is a subject of active research. Techniques such as redundancy and radiation hardening can reduce the frequency of events induced by a given flux. Different semiconductor manufacturing technologies also vary in sensitivity to begin with.

Because mitigation can not always be fully applied, it is important to have an understanding of the radiation field in the area before equipment is installed, and to retain an ability to monitor it during operation. When new facilities and experiments are designed, the radiation field is first estimated by a computer simulation. However, the simulation can only be as good as the input data and model; it cannot account for all micro-scale effects, or changes in the environment due to daily use. Therefore, radiation measurement, or *dosimetry*, is required to keep an accurate insight into the concerned areas.

1.2 Radiation monitoring systems at CERN

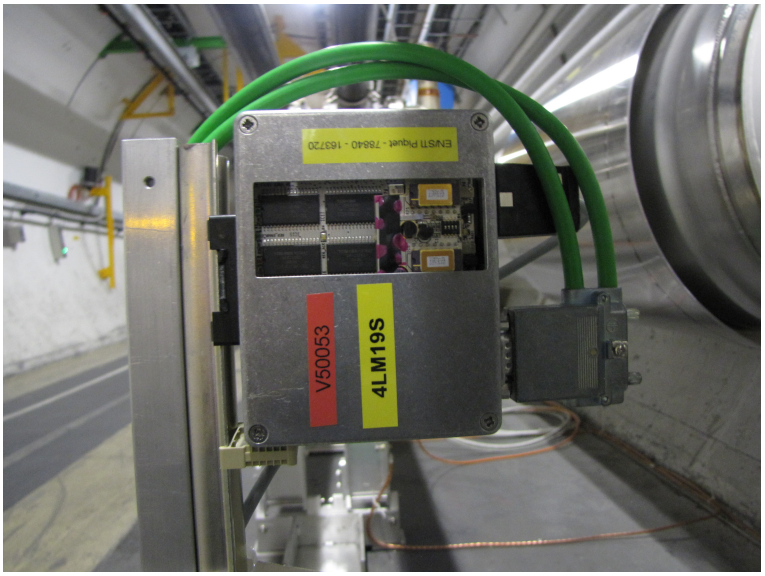


Figure 1.2: A RadMon installed in the LHC. The serial number *V50053*, and device ID *4LM19S* can be seen. Note the heavy fieldbus cables (green). Image from [3].

Radiation levels in the LHC are currently monitored by a measurement system called *RadMon* [2][4]. In the LHC alone, it comprises over 500 measurement devices, which are connected to a network using *WorldFIP*, an evolution of the FIP fieldbus. The devices monitor the following aspects of the radiation field: ionizing dose per day, flux of high-energy hadrons and thermal neutrons (*SEE-inducing flux*), and flux of 1-MeV equivalent neutrons (*DD-inducing flux*). The measured data is periodically transmitted via the fieldbus and logged in a central database.

The WorldFIP fieldbus is organized into *segments*. A segment can contain up to 32 devices sharing a common set of copper wires. Each segment is controlled by a FEC (Front-End Computer). A downside of WorldFIP is the amount and cost of required wiring; at CERN, in total over 450km of WorldFIP cables is installed [5].

The choice of WorldFIP has another drawback; in order to set up new measurement points, the necessary communications infrastructure must be put in place first. This process, including planning, procurement and installation, can take several months. In contrast, regular breaks in the machine operation (*technical stops*) last only a few days, leaving mere hours for any intervention.

These constraints led to the development of the first-generation of portable radiation monitor at CERN, now referred to as the "old BatMon". This is a battery-powered device, derived from the RadMon and equipped with sensors for measuring TID[2] and the rate of single-event effects. Because the device is not connected to any data network, the measurements must be read out by physically accessing the device and inserting a memory

[2]Total ionizing dose

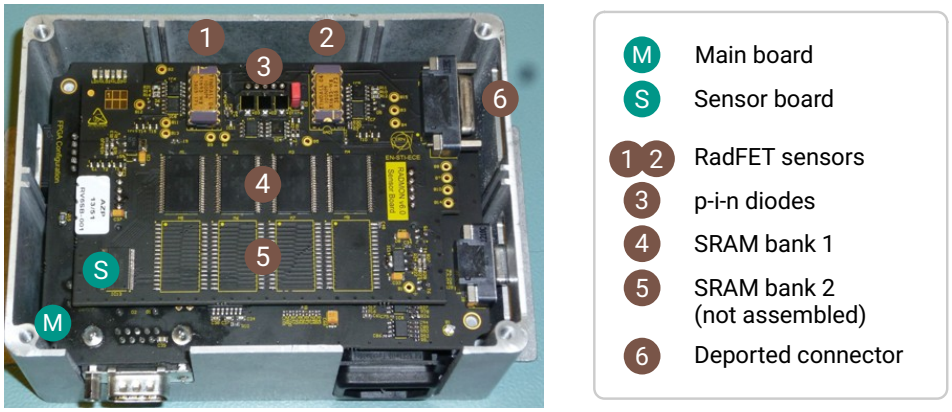


Figure 1.3: Components of the RadMon. Adapted from [2].

card. For a high number of devices, this procedure scales poorly, and the duration of a technical stop becomes once again the limiting factor.

The old BatMon is not particularly suitable for long-term deployment, either, because its high current consumption limits the battery lifetime to between 3 and 4 months.

1.3 Concept of the New BatMon

To address the limitations of the presented solutions, a new solution was sought, satisfying the following requirements:

- no cabling needed for data transmission: wiring deployment is costly both in terms of material and in terms of time; during technical stops, the available time is on the order of only several hours
- measurement: total ionizing dose and SEE-inducing flux, with resolution/sensitivity same or better vs RadMonV6
- radiation resistance: competitive with RadMonV6, which is built to a design TID of 250–300 Gy [2]
- choice between mains-powered and battery-powered operation: while AC power is generally available in the accelerator complex, the coverage is not 100 % and it imposes constraints on placement of the device. Therefore, the device should be able to operate on battery power, with a lifetime of at least 10-12 months.

Based on these constraints, the concept of a *New BatMon* was proposed: a wireless, distributed measurement system. Sensors would be deployed as part of compact, battery-powered *sensor nodes* that can be placed freely for short-term tests, or fixed in position for long-term deployments. Wired fieldbus shall be replaced by a network based on appropriate technology to transmit both measurement data and status information. All data should be

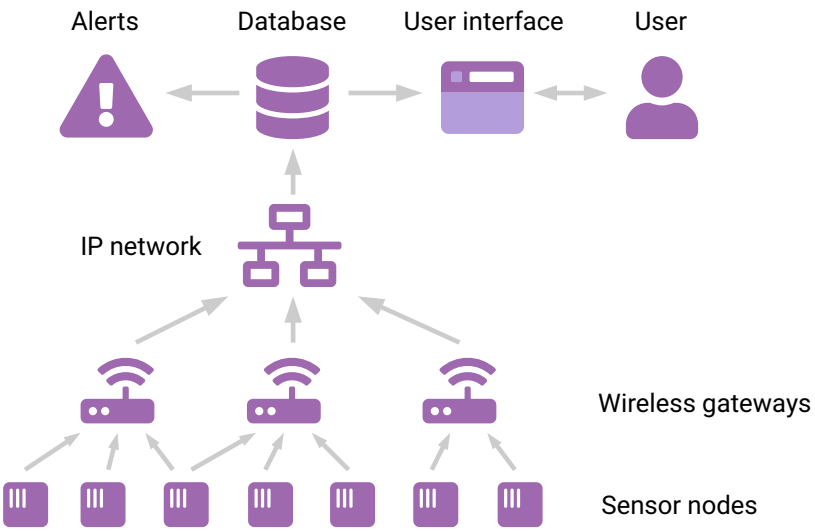


Figure 1.4: Proposed system architecture for the New BatMon

stored in a central database, and where necessary, made accessible to the interested users. (figure 1.4)

All electronic components should be commercial off-the-shelf. Radiation hardness is to be assured by choice of components with known radiation performance and thorough testing. The following chapters detail the considerations and trade-offs that led to selection of appropriate technologies and components for the system, followed by description and justification of choices made in the system implementation.

Chapter 2

Radiation measurement

In chapter 1, the context of this work was introduced – increasing electronics reliability in CERN accelerators by monitoring radiation levels. This chapter aims to give an overview of technical means for radiation measurement, and justify the choices made in this area. First, however, some background for this complex topic must be built up.

2.1 Classification of radiation effects in electronics

While the underlying physics is complex, involving different classes of particles and often non-linear behaviors, it is practical to group the effects based on the **mechanism** and whether the effect causes **permanent changes**. The same effects that are usually considered undesirable can be also used to measure various aspects of the radiation field; therefore, we avoid labeling them as *good* or *bad*. The basic classification of radiation effects was introduced in section 1.1 and the following sections will expand upon it.

This classification is also reflected in the quantities used to describe the findings; a measurement is only useful if its results are represented in an accurate, understandable way. For a given point in a radiation field, the most precise description would be a set of energy-flux spectra (*lethargy spectra*), one for every kind of particle that can be encountered; however, in practice, this representation is not easily obtained, nor is such detailed description usually necessary. More often, the field is described using quantities based on its different *effects*: ionizing dose rate (relevant for cumulative effects), thermal neutron flux and high-energy hadron flux (relevant for SEE) and 1 MeV equivalent neutron flux (relevant for displacement damage).

2.1.1 Cumulative effects

Cumulative effects can be defined as macroscopically observable variations in the characteristics of an electronic circuit or component due to the gradual degradation of some electronic parameters [6]. Generally, we consider them *irreversible*, although in specific cases circuits may "heal" by annealing after removing the radiation field. The specific changes

will depend on the type of circuit (analog, digital or mixed), manufacturing technology and type and energy of the impinging particles, among other factors. It is difficult to generalize these effects and quantitative figures often suffer from a high degree of uncertainty.

Ionizing dose effects

Ionizing radiation, such as gamma rays, induces generation, transport and trapping of holes. Conductors and semi-conductors have a high density free carriers, so any induced charge is quickly transported away or dissipated. Therefore, no permanent changes occur from a TID perspective.

Insulators, however, do not have this wealth of free carriers. In MOS and bipolar transistor technologies, SiO_2 is commonly used as the isolation material. As a result, energy deposited by radiation will readily create electron-hole pairs. If the mobility of the electrons and the holes were the same, the pairs would eventually recombine and *anneal out*. However, this is not the case; electron mobility in oxides is much higher than that of holes, so excess electrons are quickly removed, leaving behind the trapped holes creating a positive electric charge [7].

In transistors, the result of accumulating these micro-defects is increased leakage and shifting of the gate threshold. At first, the only observable effect might be increased current consumption. However, when enough damage accumulates, the transistor thresholds will shift so much that the circuit stops to work. When an electric field is present, these effects are further accelerated. This translates directly into a practical consequence: electronic devices powered during irradiation may degrade significantly faster; especially in modern technologies with thinner gate oxides [8]; see also figure 2.1.

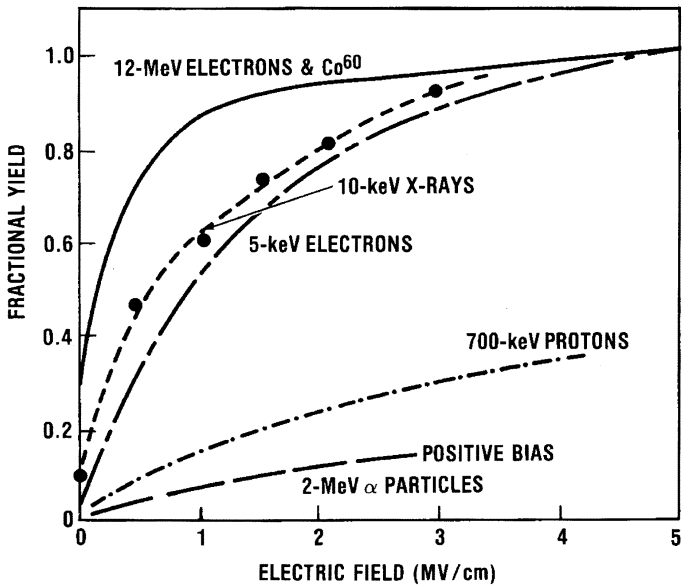


Figure 2.1: Experimentally measured fractional hole yield as a function of applied field, for a number of incident particles. From [8].

The acronym TID stands for *total ionizing dose*, a measure of energy absorbed per unit

of mass, expressed using the SI unit *gray*; $1\,\mathrm{Gy} = 1\ \mathrm{J} \cdot \mathrm{kg}^{-1}$. Because we are discussing the *absorbed* energy, the material of the irradiated object must be also specified. In this work, it is assumed that the material of interest is always silicon. In other words, we use Gy as a shorthand for the standard explicit notation Gy(Si).

A related unit is the *dose rate*, commonly expressed in $\mathrm{Gy} \cdot \mathrm{h}^{-1}$. In literature, the older unit of *rad* can be also encountered, particularly in relation to space applications. $100\,\mathrm{rad} = 1\,\mathrm{Gy}$.

Displacement damage (DD)

Displacement damage is an instance of a non-ionizing radiation effect.

Modern semiconductor technologies almost invariably involve atoms in a regular lattice, cut from a monocrystal with extremely low defect density. When an incident particle elastically collides with an atom in the lattice, it can knock the atom off, creating a *vacancy* and an *interstitial defect* (figure 2.2). As a practical consequence, in bipolar-junction transistors, this increases the recombination rate in the base area, increasing the base current required for a given collector current, therefore reducing the transistor's current gain [7].

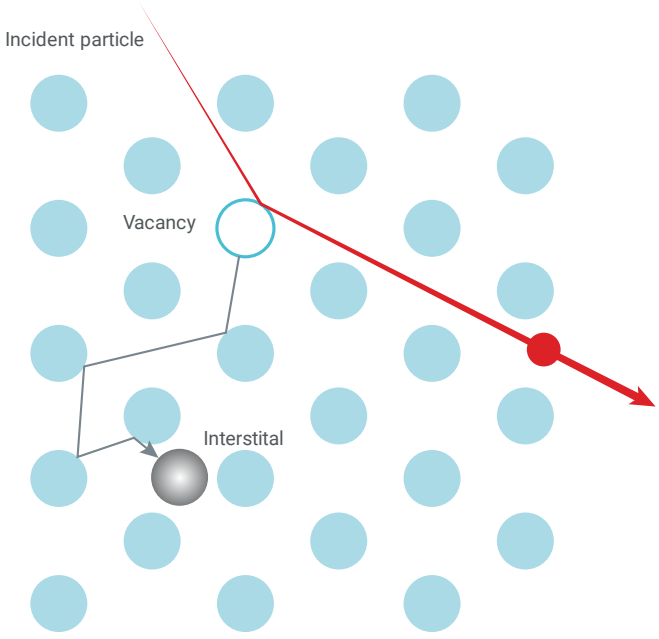


Figure 2.2: Vacancy (light gray) and Interstitial defect (dark gray) created in a silicon crystal lattice by an energetic particle. From [7].

On the other hand, MOS circuits are generally fairly robust against DD effects up to fairly high DD doses. At sufficiently high absorbed doses, mobility degradation and free-carrier reductions caused by DD ultimately lead to reductions in MOSFET device drive strength and switching speed. Because of the difficulty in quantifying displacement damage, what is

usually reported is the flux or fluence of DD-inducing particles, which are typically energetic electrons, protons and neutrons. Indirectly, gamma radiation can also cause DD effects via secondary (produced) electrons.

2.1.2 Single-event effects

Single-event effects are stochastic, and caused by a single passing particle. Non-destructive, or *soft* SEEs disrupt the circuit behavior and can corrupt data stored in a memory, but do not cause lasting damage. Destructive, *hard* SEEs, on the other hand, permanently damage or destroy the circuit.

Non-destructive single-event effects

Single-event transient (SET). An energetic ion passing through an electronic device will create ionized electron-hole pairs in its wake. The electric field will attract the electrons and the holes in opposite directions, causing a brief flow of charge – electric current. Eventually, all pairs will recombine or have their charge transported away and the device will return to a normal state. Figure 2.3 illustrates this process. Depending on which node in a circuit is struck, the effect may be observable; for example, as a voltage transient on the output of an operational amplifier, or as a spurious clock pulse in a synchronous circuit. SET is the archetype and originator of all other single-event effects.

Among other factors, the amplitude and duration of the event depends on the energy of the responsible particle, in ways that can be counter-intuitive; a higher energy of a particle does not always translate into higher energy transferred into the circuit. More detailed description of the physical background can be found in [7].

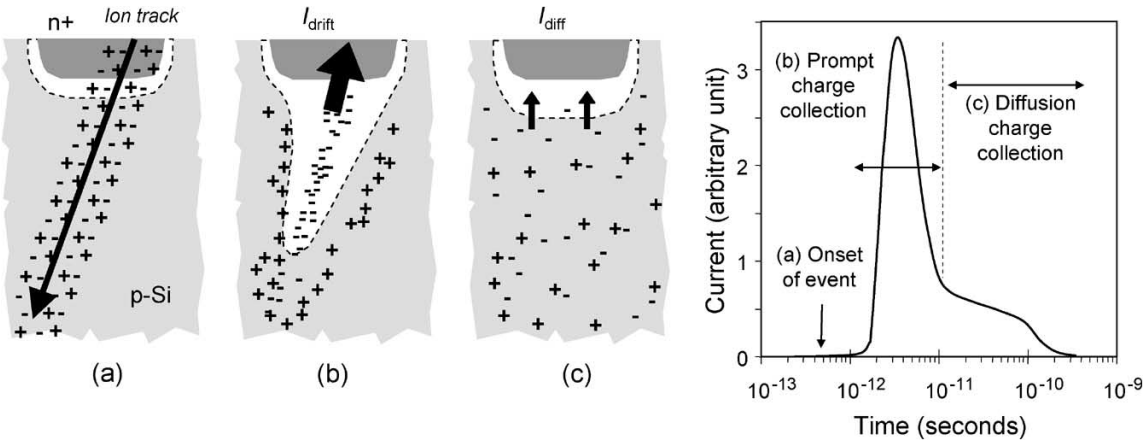


Figure 2.3: Charge generation and collection phases in a reverse-biased junction and the resultant current pulse caused by the passage of a high-energy ion. From [9].

Single-event upset (SEU). When a large enough single-event transient occurs in an analog or digital circuit with state (such as a sample-and-hold circuit or a digital memory),

it may corrupt the stored value. In a latches, flip-flops and memories, this can lead to a *bit flip* – changing the stored bit from 0 to 1 or from 1 to 0 (the mechanism of an SEU in static memory is further discussed in section 2.2).

The severity of the incident then depends entirely on the significance of the affected data. If a critical register or memory location is struck, the system may enter a state where it is no longer able to perform its function. This is called a Single-event functional interrupt (SEFI) and can be recovered only by resetting or power-cycling the affected system.

When manufactured in comparable technologies, and using no special mitigation, dynamic memories are more prone to upsets than static memories,[1] which in turn are more sensitive than general sequential logic. This is not only due to the much higher bit density of the former, but also due to different circuit topology (one *bit line* in a commercial DRAM connects to 64 or more bit cells).

Destructive single-event effects

Single-event latch-up (SEL). A latch-up is a well-known failure mode of CMOS integrated circuits. The totem-pole configuration of MOSFETs creates a PNPN structure, equivalent in behavior to a thyristor, or a pair of bipolar junction transistors connected in a positive feedback loop.

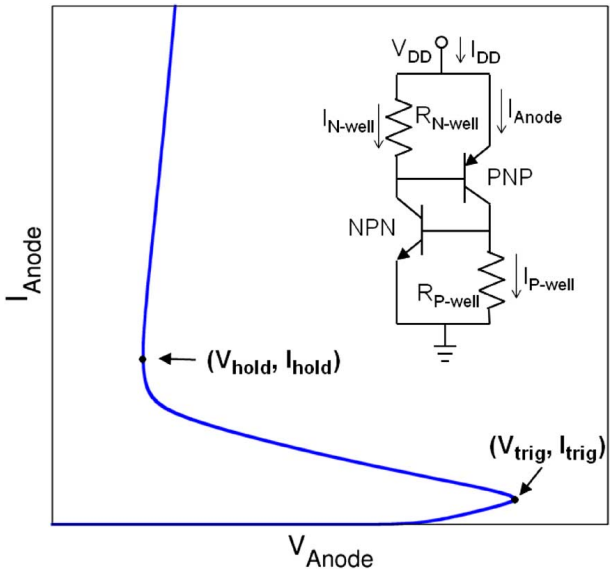


Figure 2.4: Typical I – V curve for positive current injection into a PNPN structure. Inset: Equivalent circuit for PNPN structure showing parasitic BJTs and well resistances. From [10].

During normal operation, this structure is inactive, but a sufficiently large voltage spike on an input pin can put one of the parasitic transistors into a forward-biased state. Due to the topology of the equivalent circuit, this creates a self-sustaining loop where each of the transistors keeps the other one saturated, as long as voltage is applied (figure 2.4). A

[1] Though SRAMs have a higher relative proportion of single-event *multiple-bit* upsets

short-circuit is created between the poles of the CMOS structure – typically the power rail and ground. If uncontrolled, the resulting high current can burn out the affected device, the power supply, or parts of the PCB.

A single-event latch-up occurs when a high-energetic particle injects current into the base of one of the parasitic transistors, thereby activating this structure.

The problem of latch-up is not being ignored – manufacturers of electronic parts routinely employ anti-latch-up techniques. Some of these techniques are process-based, for example adding guard rings to evacuate injected carriers [11]. Others work by suppressing the usual triggers of latch-ups, e.g. clamping input voltages via on-chip diodes; these techniques are not effective at preventing SEL.

A different approach to latch-up mitigation is using an external circuit to monitor the current consumption of the system. If the consumption rises above a defined level, the circuit immediately powers the system off to remove the forward bias of the parasitic BJTs. Afterwards, power is restored. In this way, the latch-up still results in an interruption of device operation (soft error), but permanent damage may be prevented.

2.2 Detection of single-event effects: static random-access memory

A common approach for measuring the SEE-inducing flux is to use Static Random-Access Memories (SRAMs). A conventional "6T" SRAM cell consists of 6 transistors, making up the storage for one bit, as shown in figure 2.5.

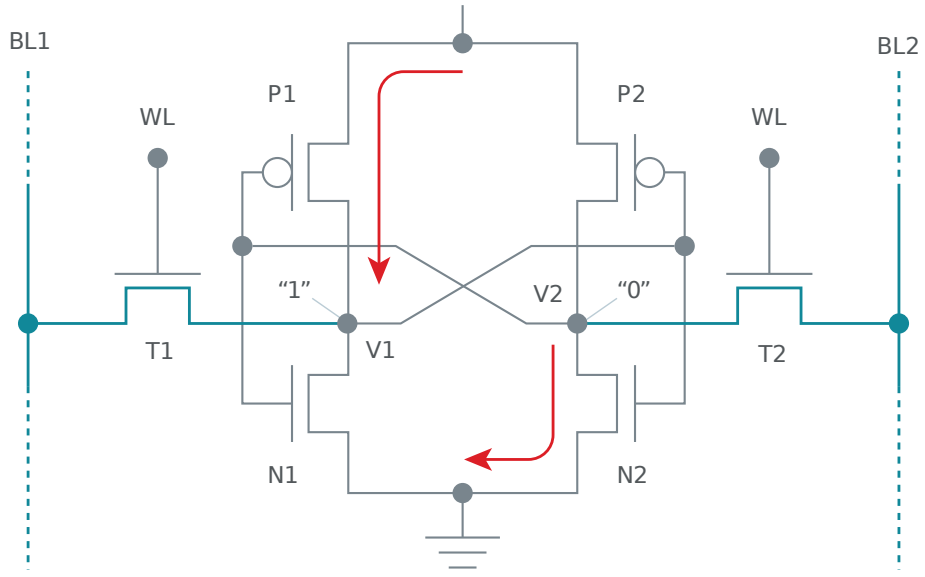


Figure 2.5: 6T SRAM bit cell in storage mode. The WL is off, so both pass transistors are off. The "1" data state is maintained on the left side by the P1 pull-up. The "0" data state on the right side is pulled down by N2. N1 and P2 are off. Adapted from [7].

The principle is the same as two inverters connected in a loop. These inverters are made up by transistors P1/N1 and P2/N2, respectively. In the example, the value "1" is stored in node V1, represented by positive voltage with respect to ground. This voltage keeps transistor P2 off and transistor N2 on, therefore holding the value of V2 in "0" (zero voltage). In turn, P1 is held on and N1 off, which reinforces the "1" on the left side. The circuit is stable.

To change the value stored in V1 to "0", WL and BL1 are brought low. The N-MOS transistor T1 is stronger than P1, and will overdrive it. This causes P2 to turn on and N2 to turn off, reinforcing the new state.

Because the state on each side is held only by one, relatively weak transistor, an energetic particle can inject enough charge to temporarily override the state, thus also changing the gate voltage on the opposite side, and putting the cell into a new stable state with inverse value. This is not simply a theoretical proposition, but a real, major concern for computer systems under radiation. The effect is stochastic, but given enough bit cells and enough time, remarkably consistent. Notably, due to the high uniformity of an SRAM die, the probability of the effect is about the same for every bit cell in a chip.

Due to these properties, SRAM is suitable as a sensor for certain types of particles. The mean rate of upset events in an SRAM is equal to the product of the particle flux and the SEU cross-section of the device:

$$\mathcal{E}\{f_{SEU}\} = \Phi \cdot \sigma \tag{2.1}$$

For commercial (not radiation-qualified) parts, the cross-section is not specified by the manufacturer, and is subject to variations in a manufacturing process. However, it can be determined by testing in a beam with well-defined flux, and the results can be extrapolated to the entire production lot.

Estimating the flux from number of upset events

With a calibrated cross-section σ and n bit flips counted during a time interval τ, the flux can be estimated as

$$\Phi = \frac{n}{\tau \cdot \sigma}. \tag{2.2}$$

Reality is usually more complicated; the radiation fields of interest are made up of different kinds of particles, in proportions that are unlikely to be the same as those experienced when calibrating the memory. For example, considering now two different classes of particles, thermal neutrons and high-energy hadrons, the expected number of events n during a time interval τ would be:

$$\mathcal{E}\{n\} = \tau \cdot \sigma_{ThN} \cdot \Phi_{ThN} + \tau \cdot \sigma_{HEH} \cdot \Phi_{HEH} \tag{2.3}$$

The equation now contains 2 unknowns, and cannot be unambiguously solved for flux, given just one measurement n. The situation can be remedied by taking another measurement with a different ratio of σ_{ThN} and σ_{HEH} – thereby *removing a degree of freedom* from the problem.[2] Consider:

$$\begin{aligned} n_1 &= \tau \cdot \sigma_{ThN1} \cdot \Phi_{ThN} + \tau \cdot \sigma_{HEH1} \cdot \Phi_{HEH} \\ n_2 &= \tau \cdot \sigma_{ThN2} \cdot \Phi_{ThN} + \tau \cdot \sigma_{HEH2} \cdot \Phi_{HEH}, \end{aligned} \tag{2.4}$$

or, in matrix form:

$$\begin{bmatrix} n_1 \\ n_2 \end{bmatrix} = \tau \cdot \begin{bmatrix} \sigma_{ThN1} & \sigma_{HEH1} \\ \sigma_{ThN2} & \sigma_{HEH2} \end{bmatrix} \cdot \begin{bmatrix} \Phi_{ThN} \\ \Phi_{HEH} \end{bmatrix}, \tag{2.5}$$

$$N = \tau \cdot \Sigma \cdot \phi. \tag{2.6}$$

If the ratios of $\sigma_{ThN1}/\sigma_{HEH1}$ and $\sigma_{ThN2}/\sigma_{HEH2}$ are different, Σ will necessarily be invertible, and the vector of estimated fluxes can be recovered by computing

$$\phi = \frac{1}{\tau} \cdot \Sigma^{-1} \cdot N. \tag{2.7}$$

In this example we demonstrated a case of 2 classes of particles in the field; however, the matrix approach scales to an arbitrary number. m distinct fluxes can be measured if Σ is a non-singular $m \times m$ matrix (note that some elements of Σ may be equal to zero). More generally, the number of rows of Σ may be *greater* than m, yielding an over-determined system. In that case, the Moore–Penrose pseudoinverse of the matrix Σ can be used to find the best-fit solution.

Other concerns for measurement accuracy

Resolution of the measurement can be increased by using multiple of the same SRAM in parallel, at the cost of increasing the amount of data to read out. Some other effects may impair the accuracy:

- If the measurement algorithm is "initialize–wait–readout" and two SEU-inducing particles hit the same cell during one measurement cycle, the second event will *mask* the first. The probability of this is usually sufficiently low to be negligible; it would manifest as a seemingly shrinking cross-section during the measurement cycle.
- *Scrubbing* of the memory – that is, counting flipped bits and re-initializing the memory contents – takes a non-trivial time, during which some new events may be missed. Highest accuracy is achieved if each cell is rewritten immediately after being read out and if the duration of the scrubbing operation is constant.

[2] In practice this is achieved by using multiple SRAMs from different product lines, or multiple chips with a different bias voltage applied as in [2].

- Occasionally, a single particle may upset multiple bits simultaneously. On average, this effect should reflect in the cross-section, but due to its lower occurrence, it tends to cause gross measurement errors on a short time-scale in devices suffering from this phenomenon.

Finally, an interesting observation is that the most accurate estimate of the SRAM cross-section might *not* be the most suitable for radiation measurement. Let us consider an extreme example, in which 10 % of all events are consistently missed as an artifact of the measurement algorithm; in this case, highest accuracy of measurement will be achieved using a cross-section obtained by a calibration procedure that suffers from the same systematic error. Thus, if the systematic error is not exactly known – as is usually the case in practice – it might be advantageous to calibrate with the exact same procedure as used for measurement, including all of its flaws.

2.3 Measurement of dose effects: radiation-sensitive field-effect transistor

A RadFET (RADiation-sensitive Field Effect Transistor) is a field-effect transistor manufactured with an emphasis on consistent and controlled radiation sensitivity. By means of the effects described earlier, accumulated dose alters the gate threshold voltage of transistors in general, which is exploited for TID measurement. The sensitivity is controlled by the gate oxide thickness; 100 nm, 400 nm and 1 µm have been used at CERN. For illustration, a 400 nm RadFET with grounded gate (also referred to as *0 V bias*) has an initial sensitivity of $40\,\mathrm{mV}\cdot\mathrm{Gy}^{-1}$. Figure 2.6 shows the non-linear dependence of sensitivity on bias voltage and oxide thickness.

RadFETs have been used in dosimetry applications for years; however, they have certain drawbacks that complicate their integration in a measurement system:

1. Analog circuitry is needed for read-out, which might be prone to EMI-induced noise and temperature drift. Furthermore, A/D converters embedded in low-cost microcontrollers and FPGAs generally trade off resolution for higher speed, making them a sub-optimal match (on the RadMon, this is partially compensated by adjustable voltage gain of the read-out amplifier)

2. High analog voltage (15 V) is needed for read-out, which necessitates the presence of another power rail on the board.

3. RadFETs suffer from a *fading* phenomenon, where the gate threshold rebounds after irradiation stops (15 % after 5 days for a 400nm device). Dis-entangling the effects of fading from effects of further irradiation might not be possible.

4. A choice of gate oxide thickness must be made to control the trade-off between sensitivity and TID.

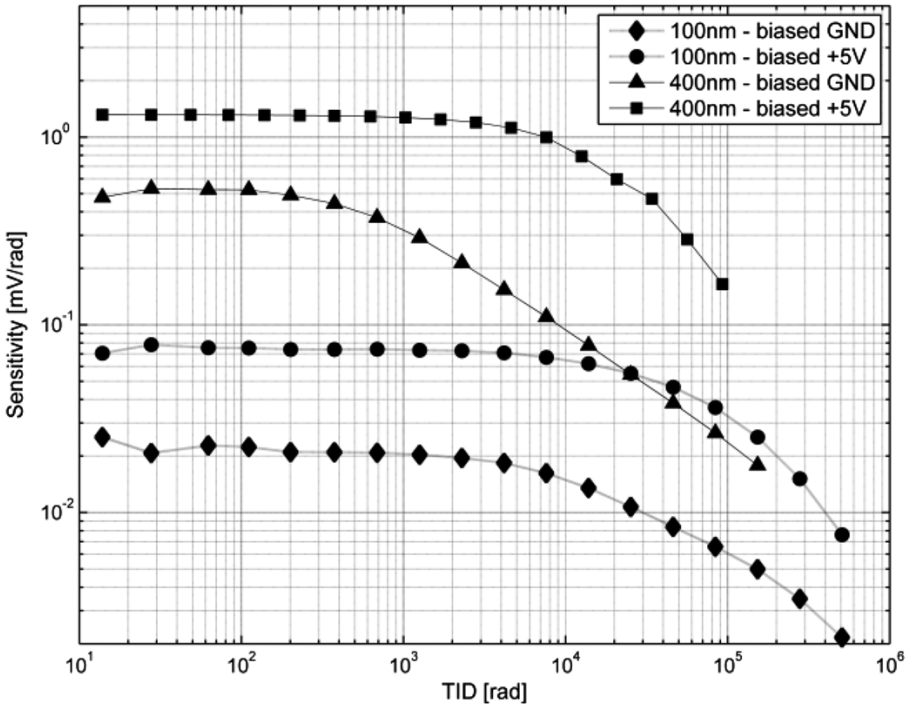


Figure 2.6: RadFET sensitivity as a function of the dose with different gate bias for irradiation with ^{60}Co at 40 °C. From [2].

2.4 Measurement of dose effects: floating-gate dosimeter

Recently, a new type of integrating dose sensor has become available, promising improvements where the RadFET suffers from limitations.

Floating-gate field-effect transistors were originally conceived for implementation of bit cells in non-volatile memories. The gate of such a transistor is isolated to eliminate leakage, and charge is deposited or removed using Fowler-Nordheim tunneling, a quantum process [12].[3] In a non-volatile memory, the presence (0) or absence (1) of charge in the gate encodes the stored value.

It has been observed that floating-gate transistors are susceptible to charge loss by irradiation; according to [13], the dominant mechanisms are: 1) the injection into the floating gate of the charge generated by the impinging radiation in the surrounding oxides, and 2) photoemission when the carriers in the floating gate get enough energy from the radiation to surmount the potential barrier. An opposite effect may happen through charge trapping in the tunnel oxide, however the oxide is usually sufficiently thin to make this effect negligible.

Inspired by these observations, a prototype floating-gate MOSFET dosimeter has been built and characterized by Álvarez et al. in [14]. The dosimeter is a silicon die manufactured in a conventional CMOS process, and packaged as an integrated circuit. It incorporates

[3] Hot Carrier Injection is another technique used in some cases, but not by the FGDOS.

a single floating-gate MOSFET along with a conventional MOSFET for temperature sensitivity compensation, a charging circuit, an evaluating circuit which converts the remaining charge to frequency, and an SPI configuration interface.

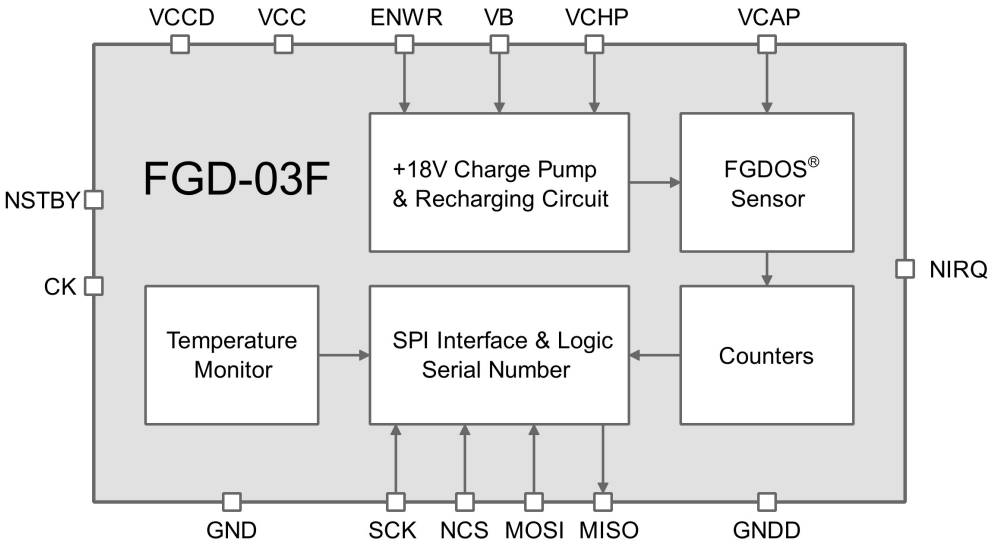


Figure 2.7: Block diagram of a modern FGDOS sensor. The commercially available integrated circuit packages two such modules with independent pin-outs. From [15].

Nowadays, the dosimeter is available as a commercial product, marketed as *FGDOS* (Floating Gate DOSimeter). Although the design has evolved and more support circuitry has been added, the principle is unchanged.

2.4.1 Operating principle of the floating-gate dosimeter

The operation of the dosimeter begins by charging up the gate capacitor. To charge the isolated gate, a voltage of 15–18 volts is needed to invoke the tunneling effect. The chip incorporates a charge pump which is able to generate this voltage from a common 5-volt rail. Then, as the capacitor is irradiated, it is discharged due to the effects described above [16].

At any time, the charge in the capacitor modulates (non-linearly) the drain-source current of the transistor [16]. When voltage is applied between drain and source, the current will vary correspondingly. The integrated circuit incorporates a current-to-frequency converter with a linear characteristic, which makes it possible to indirectly measure the gate charge as frequency of an AC signal. In older versions of a chip, this signal was the output of the chip; the latest revision includes an embedded frequency counter, so the read-out is fully digital.

The ratio between frequency change and the absorbed dose is expressed by *sensitivity*, defined as

$$S = \frac{\partial f}{\partial D} \tag{2.8}$$

and typically expressed in units of $\mathrm{kHz} \cdot \mathrm{Gy}^{-1}$.

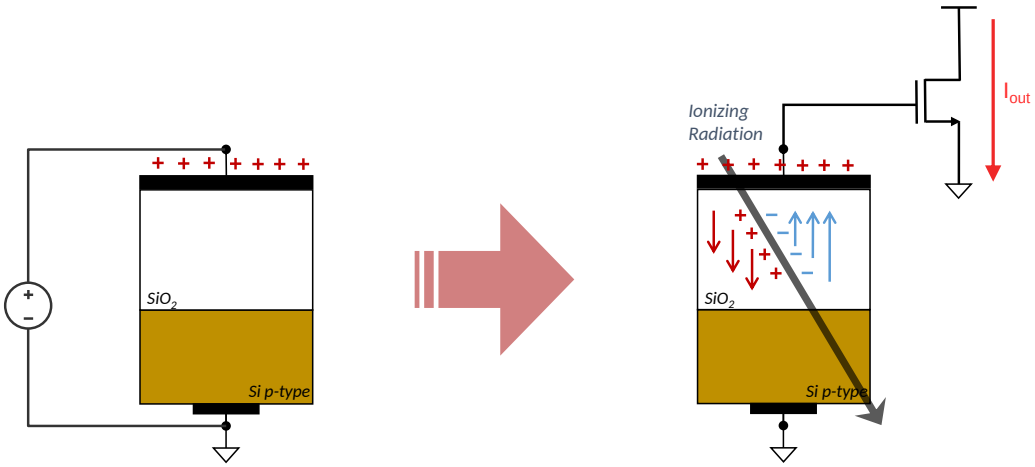


Figure 2.8: Operating principle of the floating-gate dosimeter. From [17].

Eventually, the gate capacitor would become completely discharged. At that point, the charging circuitry could be used to recharge the gate and resume measurement. However, as figure 2.9 illustrates, the response is highly non-linear over the full range of charge (the sensitivity is not constant). To compute the dose accumulated since the latest recharge, a calibration curve must be applied to the measured frequency. Furthermore, measurement resolution is diminished in the extremes due to the lower sensitivity. For these reasons, it may be desirable to constrain the operating range to a small *linear region*, where sensitivity is well-known and constant.

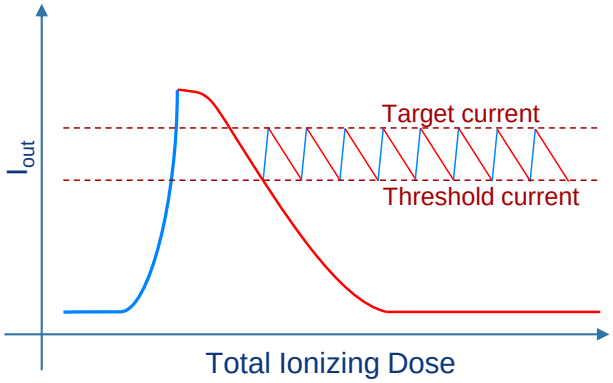


Figure 2.9: After being charged up (blue curve), the change of current due to unit amount of dose is highly non-linear over the operating range of the sensor (red curve). Linearity can be achieved by limiting the sensor to a narrower operating region where the sensitivity is sufficiently close to constant (in picture between *target* and *threshold*). Figure from [17].

2.4.2 Operation in autonomous mode

The sensor is capable of operating in a so-called autonomous mode, where the recharging is handled entirely by digital logic in the chip. The user first configures the sensor by setting

a *threshold frequency* and *target frequency*.[4] When the measured frequency drops below the threshold, the sensor automatically initiates a recharge. Once the target frequency is reached or exceeded, the recharge is stopped. The number of recharges performed is also counted by the digital logic and can be read out by the user along with the measured frequency at any time.

Autonomous mode can greatly simplify integration of the sensor into an application and allow the rest of the system to be put in low-power mode for extended periods of time. However, it might not be suitable for all use cases, for reasons explained below.

Complementary to autonomous mode is *manual* mode, where the user is fully in charge of starting/stopping the internal recharge mechanism, or even injecting recharge voltage through a dedicated pin on the chip.

2.4.3 Passive operation

The FGDOS also permits fully passive operation. In other words, after charging up the on-chip capacitor, the sensor can be completely powered down and the radiation-sensitive part will continue to measure absorbed dose. It then becomes the responsibility of the user, however, to power the sensor up often enough, read its value and perform a recharge if needed to stay within an operating range suitable for the application.

2.4.4 Sensitivity and linearity

The typical sensitivity of the sensor is specified in its the data sheet [15]; however, a non-negligible variance, presumably due to manufacturing tolerances, has been observed (further discussed in chapter 6). At the moment it is not clear if this variance can be compensated, or if the sensors require individual characterization to achieve maximum accuracy.

Over large doses (dozens and hundreds of grays), another effect begins to significantly impact measurement accuracy: ionizing radiation causes irreversible degradation of the sensor sensitivity.[5] There is no simple work-around for this effect; to correctly interpret measurements from a sensor that has absorbed a large dose, the entire previous lifetime of the sensor must be tracked, and the TID must be taken into account when evaluating the incremental dose. Any transient inaccuracies will cause long-term measurement errors because the newly calculated TID is used to compensate all subsequent measurements.

In [16], the temperature sensitivity of the FGDOS has also been investigated. Although the sensor contains circuitry to allow temperature compensation by the user, it was found to be inaccurate across the operating range of the LHC, and the authors propose another compensation method based on a look-up table.

[4] The official recommended range is 50 to 90 kHz, however due to overshoot of the recharge mechanism, accuracy improves after setting the target to 80 kHz.

[5] Characterized in [18] and also in chapter 6

2.4.5 Interpretation of readings

Although fundamentally the sensor measures dose by its effect on *charge* remaining in a gate capacitor, only the converted frequency is exposed outside.

If no recharge occurred, the dose absorbed between two frequency readings, f_1 and f_2, is calculated as an integral of the *reciprocal frequency sensitivity* (figure 2.10):

$$\Delta D = \int_{f_1}^{f_2} \frac{1}{S}\, df. \tag{2.9}$$

A problem with this expression is that it suggests the sensitivity is constant, while in reality, it depends on several factors, most importantly:

- instantaneous frequency (non-linear dependency)
- manufacturing variations of the radiation-sensitive volume on the die (linear dependency)
- TID absorbed during the sensor's lifetime (non-linear dependency)
- ambient temperature (not investigated, as compensation is available)

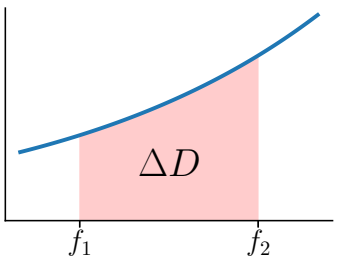


Figure 2.10: Integrating of the reciprocal sensitivity

Focusing only on the first 3 factors, the sensitivity of a particular sensor at a particular frequency, having absorbed a particular dose, is expressed as

$$S(S_0, f, D) = S_0 \cdot s_f(f) \cdot s_D(D), \tag{2.10}$$

with S_0 representing the initial sensitivity and s_f, s_D being scalar functions describing the relative sensitivity change with frequency and TID, respectively. To remove excess degrees of freedom, we fix $s_f(90\ \text{kHz}) = 1$ and $s_D(0\ \text{Gy}) = 1$. The integral then becomes

$$\Delta D = \int_{f_1}^{f_2} \frac{1}{S_0 \cdot s_f(f) \cdot s_D(D)}\, df. \tag{2.11}$$

Depending on the required accuracy and the operating conditions, some of the effects can be neglected. For example, if we constrain ourselves the a linear region of operation and ignore the long-term sensitivity degradation, the expression can be simplified to

$$\Delta D = \frac{f_2 - f_1}{S_0}. \tag{2.12}$$

Neglecting the manufacturing variation will cause a larger error (10-20 %), but in accelerator dosimetry applications even such uncertainties are not uncommon [2]. Then the nominal value of sensitivity, as specified in the sensor data sheet, may be used directly.[6]

[6] $60\,\text{kHz} \cdot \text{Gy}^{-1}$ in high-sensitivity mode, $10\,\text{kHz} \cdot \text{Gy}^{-1}$ in low-sensitivity mode.

Accounting for sensor recharge

The equations shown thus far share a common flaw; they do not account for periods of recharging the floating gate, which is a fundamental part of the sensor's operation. During recharge, the output frequency rises quickly, but the exact rate is not specified. Due to this, in order to achieve consistent results, the recharge must be performed in a feedback loop, regularly checking if the target frequency has been reached, which can take up to several seconds; whether this is achieved by the autonomous functionality described above, or driven externally, the mechanism is the same.

In any case, information about dose absorbed during the recharge is lost. Several approaches can be considered to cope with this issue, depending on the mode of operation used. In autonomous mode, the options are:

Ignore the problem. If the dose rate is low and stable, and the proportion of the time spent recharging is negligible relative to the time of discharge, it may be acceptable to simply discard the segments where frequency increased instead of decreasing. The data must be read out often enough that these segments are sufficiently short; in particular, if an entire recharge period passes between two samples, a systematic error will occur as illustrated in fig. 2.12, leading to grossly incorrect measurement.

Factor in the recharges. The sensor internally tracks the number of recharges performed. Because f_{target} and $f_{threshold}$ are known (specified by the user), the following calculation can be done (see also figure 2.11):

$$\Delta f = f_1 - f_2 + (f_{target} - f_{threshold}) \cdot N_{recharge} \tag{2.13}$$

This does not compensate for the "lost" dose, but it prevents the gross errors caused by ignoring recharge. In practice, however, the configured f_{target} is not reached exactly; the autonomous recharge can only be stopped whenever the built-in frequency counter finishes one counting period. With the fastest possible setting, this happens once per 125 milliseconds. As a result, termination of the recharge is delayed and the frequency overshoots the configured target.

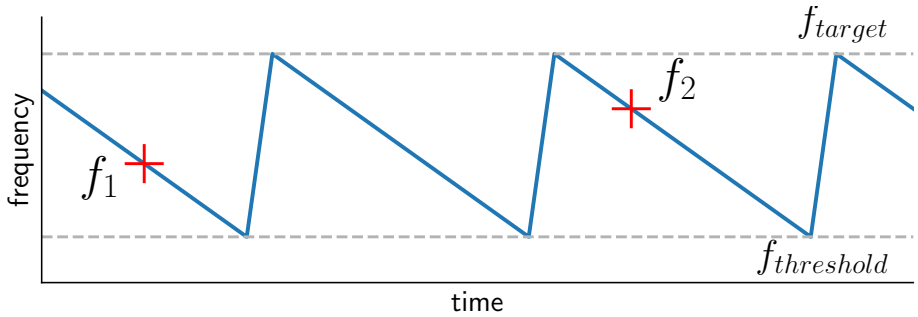


Figure 2.11: Frequency over time in autonomous mode with a constant dose rate

As can be seen, autonomous mode has fundamental flaws that limit the measurement accuracy, especially if the dose rate is considerable. In manual mode, more effort is required

to integrate the sensor and evaluate readings; however, new approaches to handle the recharge accurately become possible:

Extrapolate the dose rate. If the dose rate is very stable, the ΔD for segments of increasing frequency may be calculated from their duration and the dose rate. A method for estimating the dose rate, such as a low-pass filter, must be selected appropriately with regards to the operating environment. Figure 2.13 illustrates this approach.

Use two asynchronous sensors. If two independent sensors are available – as is the case with the used FGDOS – it can be ensured that they are never being recharged at the same time. The measurements from both can then be combined to remove any gaps. In addition, when both sensors are measuring, the values can be averaged to improve resolution by a factor of 2, and checked for mutual consistency to detect issues in the measurement system.

2.4.6 Summary

The FGDOS is an example of a modern radiation sensor that enables more accurate measurement in comparison to previously available options, while also adapting to a wide range of dose rates due to two configurable sensitivity levels and autonomous mode. To get reasonably accurate readings in low-dose-rate applications, autonomous mode can be used to simplify integration. However, when high accuracy is desired, it is necessary to understand its principle of operation and avoid, or at least be aware of, some of its fundamental limitations. The advantages of the FGDOS compared to RadFETs make it a more appropriate sensor for dose measurement on the BatMon.

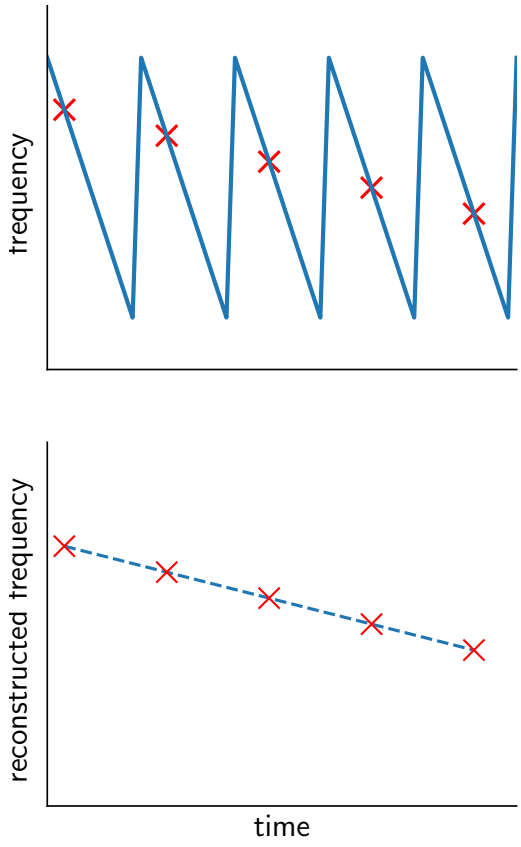


Figure 2.12: Systematic error due to undersampling

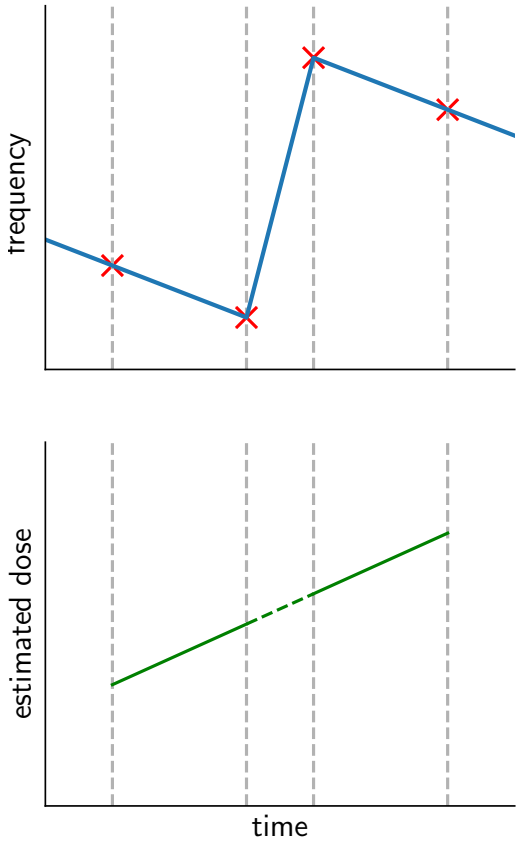


Figure 2.13: Extrapolating the dose rate

Chapter 3

Communications and data logging

Having acquired and pre-processed the measurement data, the next step is to cross the "air gap" between the devices and the network infrastructure. Simultaneously, the data is also stored on the device to cover for any connection issues.

The following data needs to be communicated:

1. Dose measurement (TID)
2. SEU counts or flux estimates
3. Diagnostic information

Considering the available resolution of the measurements, one data point can be encoded in about 40 bytes. Depending on the area of deployment, the frequency of reporting will generally vary between 1 per hour and 1 per day; for accelerated testing and diagnostics, a reporting rate as high as every 5 minutes may be needed.

3.1 Selection of appropriate wireless technology

The wireless technology market is incredibly diverse. The applications range from tiny devices powered by coin cells (wireless car keys) to massive arrays of satellite dishes receiving signals from deep space; correspondingly, different technologies are available to cover this spectrum of needs. To make an adequate choice in such wide landscape, a first necessary step is to identify criteria relevant to the application:

1. *Power efficiency*: to achieve long lifetime with battery power.
2. *Radiation hardness*: because the node incorporating the modem hardware must operate in a radiation environment. The modem does not need to be designed a as rad-hard specifically, qualification by testing is also possible.

3. Possibility and cost of *deploying own network infrastructure*: coverage is needed also in the tunnels deep underground; it should be easy and cheap to extend the network.

4. *Easy to incorporate*: preferably compact modem, lightweight antenna and not many additional components

5. *Availability* of modem hardware and networking equipment

6. *Data bandwidth and latency*: in this area, our requirements are very modest. Still, a higher throughput could improve data fidelity and allow closer monitoring of the system.

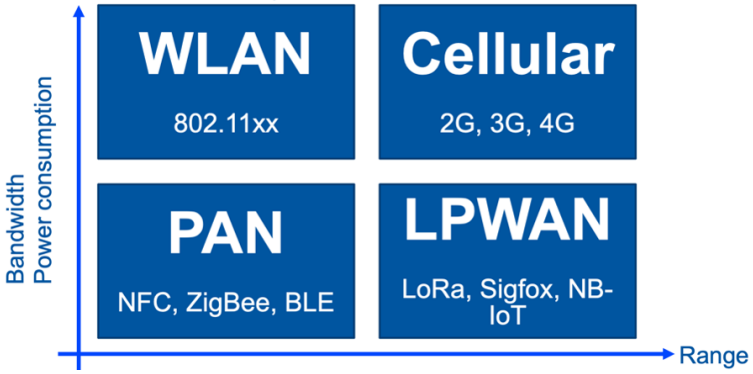


Figure 3.1: Classification of wireless technologies by bandwidth, power consumption and communication range. From [19].

Figure 3.1 depicts a coarse classification of wireless technologies by bandwidth, power consumption and communication range. For the application of radiation monitoring, short-range technologies are less interesting, because they would require any access points or gateways to be exposed to the radiation environment – this means that the range should be measured at least in 100s of meters. On the vertical axis, being a low-power application excludes current cellular technologies. This leaves a single quadrant to focus on: LPWAN.

3.1.1 Low-power wide-area networks

The low-power wide-area network (LPWAN) technology space have seen major development in the recent years, driven by increased demand for "Internet of Things" (IoT) applications, and enabled by advances in chip manufacturing and digital signal processing. As the name suggests, the common aspect of these technologies are low power consumption, low bandwidth, long range and low cost per device. Some of them also operate in license-free ISM[1] frequency bands.

In 2017, the CERN IT department evaluated several LPWAN technologies for adoption on site: LoRaWAN, NB-IoT, SigFox, Weightless-P, Weightless-N, Ingenu, SymphonyLink, QoVisio, Neul and Waviot. Ultimately, LoRaWAN has been selected for pilot deployment, and is progressively being rolled out. The parameters of LoRaWAN are compatible with

[1]Industrial, scientific and medical

the needs of the BatMon, and by adopting it, the system can take advantage of the ongoing deployment.

3.1.2 LoRa and the LoRaWAN ecosystem

LoRa specifically targets energy-constrained devices where data throughput requirements are low (on the order of dozens of bytes per hour). Due to its operation in license-free frequency bands, it is possible to deploy complete custom networks without any kind of administrative procedure (national and international rules still have to be followed; these limit, for example, the time-on-air and maximum transmission power). Note that *LoRa* refers to the technology powering the lowest levels of the stack – the modulation and physical layer.

LoRaWAN is a related protocol and standard which builds on top of LoRa to enable the co-existence of many nodes in a wide-area network, but also to standardize device provisioning and provide security features. The authoritative document is the LoRaWAN Specification, currently in version 1.1; however, we describe version 1.0.3 [20] because of more mature support.

A notable LoRaWAN implementation is *The Things Network* (TTN), a worldwide project to build an open, decentralized LoRaWAN network. This network is built up from gateways operated by a community of volunteers. The network can be used free of charge under a fair-use policy. The LoRaWAN network deployed at CERN is based on TTN software, but operates independently and is not available to the general public. Figure 3.2 illustrates the components of such network, which consists of gateways that provide the actual wireless coverage and a network server that manages the registered *end devices*. The data is then delivered to the appropriate application server, which can be independent of the network.

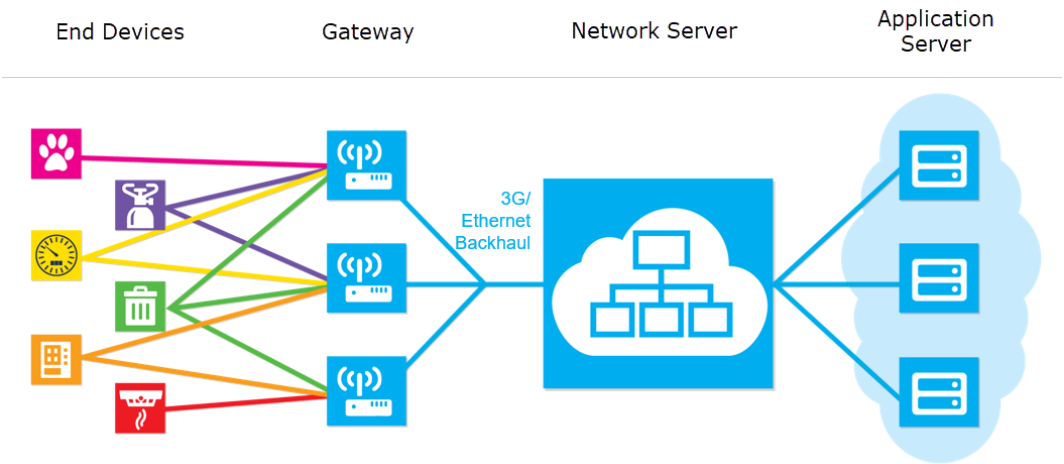


Figure 3.2: LoRaWAN architecture. From [21].

Finally, on the top level of the stack is the user application itself. The application can transmit payloads of limited size (up-link), as well as receive messages from the network (down-link).

3.2 LoRa physical layer

On the physical layer, LoRa employs a combination of established and novel techniques to achieve a balance between power consumption, bandwidth and operational range. Details of the encoding and modulation are proprietary, but related patents reveal much of the principles, and independent reverse-engineering efforts have been able to fill in the details. This is important, as it enables a more objective study of the technology, and may reduce the dependency on a limited number of vendors.

3.2.1 Chirp spread spectrum modulation

In a *chirp spread spectrum* (CSS) modulation, each bit of a message is transmitted as a series of chirps; that is, single tones with frequency linearly increasing or decreasing (fig. 3.3).

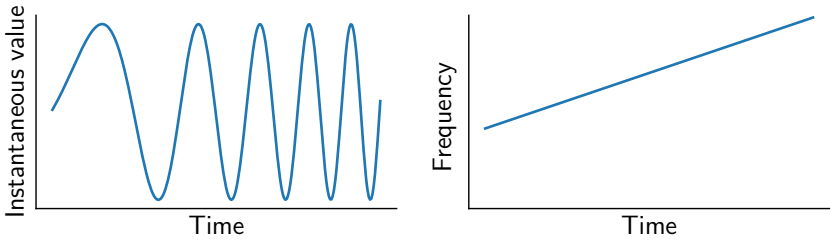


Figure 3.3: A chirp in time (left) and modulation (right) domains.

The result is that the signal is spread over a range of frequencies; *its bandwidth is increased*. This has several advantages for signal fidelity: transmissions are less susceptible to narrow-band noise, and destructive interference due to multi-path propagation (*multi-path fading*) is limited. Traditionally, additional power margin would have been needed to compensate for these issues.

The number of chirps used to encode every bit is determined by the *spreading factor* (SF), which effectively controls the trade-off between data rate and link reliability. The actual number of chirps is equal to 2^{SF}, with SF ranging from 7 to 12. Thus, lower SF equals a higher data rate. The node can select the value of SF to use every time a transmission is started and the receiving gateway will synchronize automatically. Strategies for choosing the SF vary based on the application requirements and are further discussed later.

To some extent, CSS also permits multiple simultaneous transmissions in a shared communication channel: two transmissions using a *different* SF will *not* disrupt each other, unless there is large disparity in received power [22]. The practical usefulness of this fact is however limited by the observation that the optimal SF for devices placed near each other tends to be the same (an extensive discussion of LoRa collisions can be found in [23]). This is in contrast with some other spread-spectrum techniques (e.g. direct-sequence spread spectrum, used by GPS and early Wi-Fi), which use orthogonal spreading sequences to implement *code division multiple access*, CDMA. The advantage of CDMA is that multiple transmitters can reliably use the same communication channel at the same time. LoRa does

not have this advantage, and because the nodes in a network are typically not synchronized, collisions of transmissions are bound to happen. It is then a matter of statistics to determine the expected collision rate, and additional mitigation if reliable transmission is needed.

3.2.2 Frequency channels

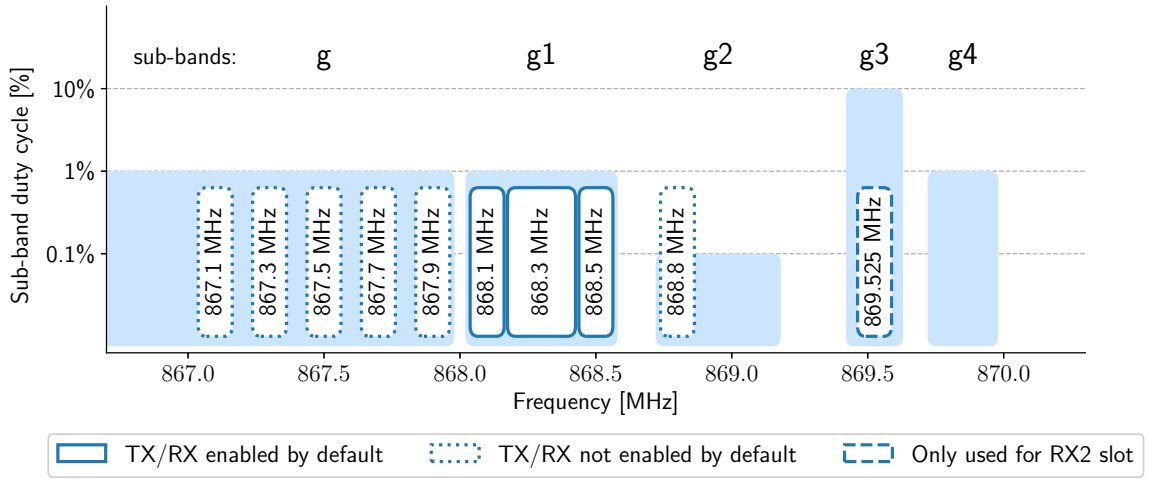


Figure 3.4: Channel map of the EU868 LoRaWAN band. At 868.3 MHz is the only channel permitting a bandwidth of 250 kHz; the others are specified to 125 kHz. The sub-band duty cycle limits are defined by European regulation (and thus are legally binding), while the channels and their properties are defined by LoRaWAN. Data from [24].

The spectrum used by LoRa is allocated to a number of channels. The *frequency plan* varies by region; figure 3.4 shows the EU channel map, including legal limits applicable in France and Switzerland (CERN is not exempted from these). The mentioned duty cycle limits apply to every device in the network, including the gateway. This is the main reason why by default, transmissions from end-devices are *unconfirmed*: no acknowledgment is requested from the gateway. In dense networks, this can also limit the rate at which new nodes are able to join.

Three of the channels are enabled by default and can be used by the node for join requests. Upon a successful join, the gateway will inform the node which other channels may be also used. To reduce the probability of collisions, the end device pseudo-randomly selects a new channel to use after every transmission. A gateway must therefore listen on all channels simultaneously.

3.3 LoRaWAN

LoRa as such is concerned only with delivering simple data frames to a gateway. Other responsibilities are left to layers higher in the stack, such as LoRaWAN. In LoRaWAN, an end device is identified by its *DevEUI*, a globally unique 64-bit identifier. After joining the network, the end device sends data to a specific *application*. All data is protected by two

layers of encryption as illustrated in figure 3.5; the gateway is able to access the header it needs, but it cannot eavesdrop on the communication between end device and application server.

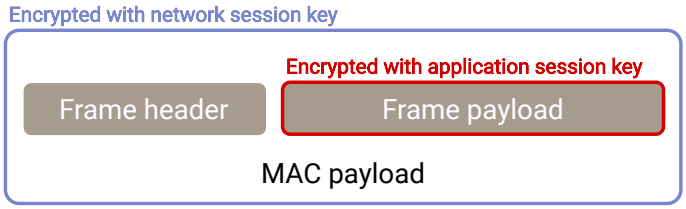


Figure 3.5: Communication between end device and network is protected from other users of the network. Furthermore, application data is encrypted end-to-end.

3.3.1 Joining the network

To connect to the network, 3 pieces of information have to be stored in the device during manufacture: *DevEUI*, *AppEUI* and *AppKey*.

DevEUI is the unique identifier introduced earlier, AppEUI is a 64-bit identifier of the application, and AppKey is a 128-bit encryption key known both to the device and the application, but not to the network provider. When the device intends to join the network, it takes the following steps [20]:

1. The device transmits a *join request*, in which it states its DevEUI and AppEUI.
2. A network server, if it recognizes the DevEUI and AppEUI, assigns a 32-bit device address (DevAddr), and provides a *session nonce* (used to derive keys) as well as some network parameters to the device.
3. The device uses the new keys to encrypt the following communication.

If a device stays within one network, the session keys can be stored for extended periods of time; there is no need to re-join the network after a power-cycle, for example. This can be particularly important in networks where many end devices turn on around the same time – as the gateway has to observe the strict duty cycle limits, it might be unable to send out join request replies in real time.

3.3.2 Implementation in the LMIC library

LMIC (originally *LoRaMAC in C*) is a library developed by IBM Research providing a LoRaWAN implementation for Semtech SX127x family of modems. The library is not tied to a specific processor or software platform; instead, it depends on a set of user-provided functions to handle concerns out of the scope of the library, such as timing or disabling/enabling interrupts. The library is high-level and handles internally all of the join procedure, frequency hopping and also implements region-dependent limits. The user

is exposed only to high-level operations like *Set DevEUI* or *Send packet*. On the one hand this makes for a low barrier of entry, on the other hand it may prevent an advanced user from making full use of the available options and capabilities.

Over time, the library has been extended and ported to other platforms by other developers. A version is available for the popular Arduino platform. At the time of writing, work is underway to implement LoRaWAN 1.1 in one of the independently continued projects.

3.3.3 Infrastructure

If one cannot (or does not wish to) rely on a third-party infrastructure, it is also possible to deploy a custom network from scratch. In the minimal case, a network can be as small as a single gateway running an embedded Network Server and an Application Server that simply forwards all received packets via a simple protocol like HTTP or MQTT.

When multiple gateways are used to form a network, proper configuration is required to let end-nodes seamlessly *roam* between areas covered by the different gateways.

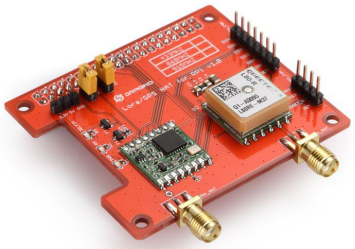


Figure 3.6: A low-cost LoRa + GPS module for the Raspberry Pi

Reception and transmission of LoRa frames requires appropriate hardware; one of the most cost-aggressive options is a "hat" for the Raspberry Pi single-board computer, such as the one in figure 3.6 which is available for around 35 €.

An example of a self-contained solution is the RAK7258 (figure 3.7). This indoor gateway is based on the open-source OpenWRT operating system and includes a Network Server implementation as well as an Application Server with LoRa packet forwarder. It can be connected to an IP network via Ethernet or Wi-Fi connectivity. Available for 150 €. A model incorporating LTE mobile connectivity is also available.

Figure 3.7: RAK7258 Micro Gateway

For the CERN LoRaWAN network, the Swiss-made LORIX One gateway has been used. This gateway is available in a weather-proof IP65 version, starting at 500 €. It can be used with a high-gain antenna or with *leaky feeder* cable.

In terms of software, production-quality free software solutions are readily available, such as ChirpStack, which is even used by major commercial hardware providers.

3.3.4 Practical concerns

LoRa and LoRaWAN together provide a set of means for building dependable sensor networks at a low cost. However, to achieve good and consistent results, some parameters must be chosen appropriately. This section discusses some of the choices that must be made.

Spreading factor and communications reliability

Spreading factor	Bandwidth [kHz]	Physical bit rate [bit/s]	Time on air [ms]
12	125	250	1811
11	125	440	988
10	125	980	453
9	125	1760	247
8	125	3125	134
7	125	5470	72
7	250	11000	36

Table 3.1: Bit rate as a function of used spreading factor and bandwidth, including time on air for an example 20-byte packet. Values from [25].

As discussed above and demonstrated in table 3.1, the spreading factor directly affects the data transmission rate. As LoRa is low-speed by design, the ultimate optimization goal is not to push as many bytes as possible per unit of time, but instead to maximize battery life. However, these two are closely linked: if the same amount of data can be transmitted in half the time, the energy expended by the radio will be also cut in half (assuming a constant transmit power). The total amount of data that can be sent while observing the regional duty cycle limitation is also increased. For these reasons, it would be optimal to always use the lowest spreading factor that successfully delivers the packet.

Adaptive Data Rate is a mechanism defined in LoRaWAN, whereby the end device can ask a gateway to *recommend* the spreading factor to use. The gateway can consider the measured signal-to-noise ratio and choose a SF according to the needed margin. This negotiation is suitable for fixed devices in a static environment. For moving devices, however, the reception quality can change rapidly, and the ADR mechanism would not be able to react quickly enough.

Therefore, a conservative choice must be made for mobile devices, taking into consideration network coverage in the area of deployment. For example, if most of the site is well-covered, but parts are shielded, confirmed uplinks can be used to detect when packets are lost, so that the end device can keep the data in memory and attempt re-transmission at a later time.

3.4 Offline logging of transmitted packets

As the reliability of wireless communication links cannot be always guaranteed, it is desirable to have an alternative data logging mechanism, which does not depend on external infrastructure, and which can be used as a sort of "flight data recorder" in case of device failure.

Therefore, the sensor node should also be equipped with a non-volatile data memory. The contents of this memory do not need to be accessible remotely, but in case of need can be read out by connecting a service cable. Emphasis is to be placed on robustness – localized data corruption should not render the whole subsystem inoperable, and it must also not negatively impact the normal operation of the device.

Chapter 4

Hardware of the sensor node

At the end of chapter 1 we presented a set of requirements for the system; many of those translate directly into requirements for the hardware design of the sensor node, which we re-iterate here:

- radiation resistance (250 Gy TID)
- resolution and sensitivity of measurement: satisfied by incorporating the set of sensors selected in chapter 2
- no cabling needed for data transmission: satisfied by using the LoRaWAN technology described in chapter 3
- choice between mains-powered and battery-powered operation

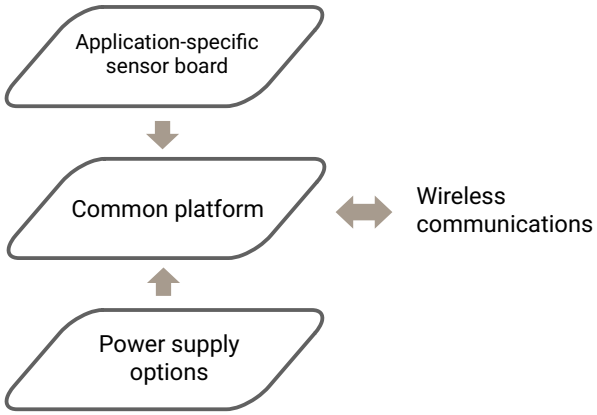


Figure 4.1: "Stacked boards" hardware concept for the sensor node

In addition, as development of more LoRaWAN-based applications is foreseen at CERN in the near future, it would be valuable to produce a radiation-tolerant base platform which could be later re-used for new projects. Ultimately, this goal is what drives the overall architecture of the hardware platform which is depicted in figure 4.1: a stack-up of

circuit boards with the *main board* hosting a microcontroller along with peripherals for communications and data logging, and a *sensor board* for radiation measurement, which can be replaced in future applications.

In line with this architecture, the following sections describe the design of the main board and the radiation sensor board. These boards were designed at CERN's Radiation-tolerant and Measurement Electronics department (EN-SMM-RME) between the years 2018 and 2019. Additional improvements were also proposed and implemented during work on the present thesis.

4.1 Main board

Figure 4.2 is a block schematic and figure 4.3 is a photo of the main board. Functionally, the board is rather simple. However, what might not be obvious at first glance is that important components have been selected based on how they had performed in radiation tests.

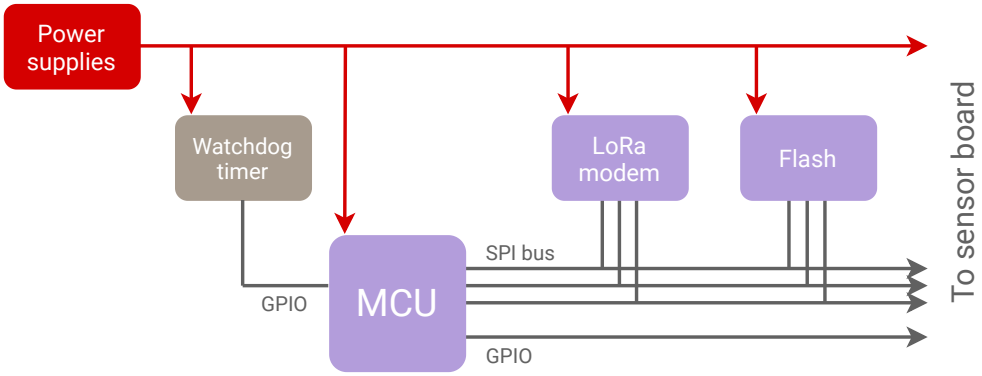


Figure 4.2: Block diagram of BatMon main board

The chosen format of the PCB enables re-use of mechanical parts and some accessories (described later) previously developed for the RadMon.

4.1.1 A note about radiation testing

To validate a device's reliability in a radiation field, *radiation tests* are carried out at specialized facilities. Such facilities are available at CERN and elsewhere in Europe and around the world. For instance:

- CC60 (CERN Cobalt-60 irradiation facility) – Equipped with a 10 TBq ^{60}Co gamma ray source.
- CHARM (CERN High energy AcceleRator Mixed field facility) – 24 GeV protons from the Proton Synchrotron (PS) hitting a selectable target to create a rich particle field.

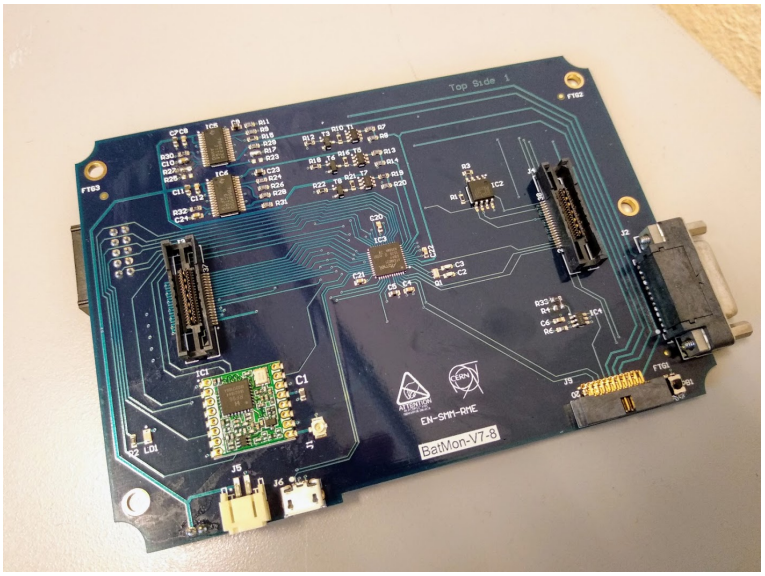


Figure 4.3: BatMon main board

- Proton Irradiation Facility of Paul Scherrer Institute (PSI PIF) – 230 MeV protons, optionally degraded to one of several discrete levels from 30 MeV to 200 MeV.

Due to the high number of devices that need to be tested (CERN has to compete for testing capacity with users in the space industry, for example), it would be impractical to test all components on a natural time-scale. Instead, *accelerated tests* are usually carried out, where the device is subjected to a much higher flux and dose rate for a shorter time. The time frame for such tests can be as short as 1 hour (when using beam time at PSI). For slower tests, several months may be needed to reach the required dose or fluence.

To make matters more complicated, the effects of a certain TID may also depend on the dose rate. In particular, linear bipolar devices like bipolar transistors and optical elements suffer from what is called Enhanced Low-Dose Rate Sensitivity (ELDRS). On the other hand, for CMOS technology devices, high-dose-rate irradiation is usually the worst case, so accelerated tests are feasible.

4.1.2 Microcontroller choice and suitability

The microcontroller unit (MCU) at the heart of the board is responsible for controlling all other components and executing the application. The chosen `ATSAMD21G18` is based on an ARM Cortex-M0+ processor core with 32 KiB SRAM and 256 KiB flash, along with various built-in peripherals. For this application, the following peripherals are particularly important:

- Serial Peripheral Interface (SPI) controller for interfacing with the LoRa module, external flash memory and sensor board

- UART[1] and USB for PC communication
- Internal watchdog (to increase chance of SEFI recovery)
- Real-time clock (to control intervals between measurements)
- Analog-to-digital converter (for self-monitoring of the board)

To verify its suitability, in February 2019 the MCU was tested at PSI Proton Irradiation Facility with 200 MeV protons up to a total dose of 252 Gy [26]. After absorbing this dose, the device still operated as expected. Different single-event effects were also tested and characterized:

Watchdog reset. The MCU was running with the internal watchdog enabled, reporting a counter that was incremented every second, via UART. On several occasions, the MCU restarted due to SEUs, presumably either because the program froze or because the SEU triggered a spurious watchdog reset (in the reported data, it is not possible to distinguish these conditions). No unrecoverable failures were observed.

Real-time clock (RTC) timing fault. The RTC was configured to periodically wake up the MCU from Standby mode. A tester was placed on the same PCB but out of the beam, to reset the MCU if it failed to wake up and provide the expected pulse signal. On several occasions, the MCU was reset by the tester. On one occasion, the MCU didn't recover after being reset and a power cycle was necessary.

External Interrupt Controller (EIC) fault. The EIC (which, despite the name, is a peripheral built into the MCU) was configured to trigger an interrupt when a rising edge was detected on a specified input pin of the MCU and respond by toggling an output pin. A tester generated this rising edge periodically and detected the response, to simulate the behavior of an external hand-shaking watchdog timer. If the MCU stopped responding, it would be reset by the tester, which happened on several occasion. On one occasion, the MCU didn't recover after being reset and a power cycle was necessary.

The test report does not mention the minimum number of consecutive missed replies that triggers a reset of the MCU.

Measure-output-save loop. In this most complex test, a 5-second measurement cycle was scheduled continuously. ADC, EXTI, RTC, SPI, UART and Watchdog peripherals were all used. Both the internal watchdog and an external one were used. The program locked up several times, but the MCU was always reset by one of the watchdogs (in the reported data, it is not possible to distinguish between them).

The reported results are summarized in table 4.1 and visually in figure 4.4. In addition, confidence intervals have been calculated based on the reported numbers of events and an assumed flux uncertainty of 10 %. Note that the covariance of the failure modes has not

[1] Universal asynchronous receiver/transmitter

been evaluated. In other words, some of the faults may be correlated, making their joint cross-section smaller than a sum of the individual cross-sections[2].

Test / type of fault	Events	Fluence $[\mathrm{p} \cdot \mathrm{cm}^{-2}]$	$\mathcal{E}\{\sigma\}$ $[\mathrm{cm}^2]$	95% confidence interval of σ
Watchdog reset	10	$3 \cdot 10^{11}$	$3.33 \cdot 10^{-11}$	$(1.57;\ 6.15) \cdot 10^{-11}$
Real-time clock (recoverable)	18	$5.65 \cdot 10^{11}$	$3.18 \cdot 10^{-11}$	$(1.85;\ 5.06) \cdot 10^{-11}$
Real-time clock (non-recoverable)	1	$5.65 \cdot 10^{11}$	$1.77 \cdot 10^{-12}$	$(0.04;\ 9.86) \cdot 10^{-12}$
External interrupt (recoverable)	14	$4.5 \cdot 10^{11}$	$3.11 \cdot 10^{-11}$	$(1.67;\ 5.24) \cdot 10^{-11}$
External interrupt (non-recoverable)	1	$4.5 \cdot 10^{11}$	$2.22 \cdot 10^{-12}$	$(0.04;\ 12.40) \cdot 10^{-12}$
Measure-output-save (resets)	22	$4.32 \cdot 10^{11}$	$5.09 \cdot 10^{-11}$	$(3.12;\ 7.76) \cdot 10^{-11}$

Table 4.1: Summary of SEE test results for ATSAMD21G18, including calculated cross-sections. Test at PSI, 200 MeV protons.

Considering a harsh environment with HEH flux of $5 \cdot 10^{11}\ \mathrm{cm}^{-2} \cdot \mathrm{year}^{-1}$, and one of the serious failure modes (non-recoverable RTC fail), we can estimate the mean time to failure (MTTF) as

$$MTTF = \frac{1}{\Phi \cdot \sigma}, \tag{4.1}$$

which evaluates to 58.9 weeks. For a leaner field of $5 \cdot 10^{10}\ \mathrm{cm}^{-2} \cdot \mathrm{year}^{-1}$, the longevity increases to 11.3 years, making this MCU a good candidate for the BatMon.

One disadvantage of this MCU is its low pin count, which leads to some necessary compromises in the sensor interface, as described further below.

4.1.3 Data memory choice and suitability

To store logged data and user configuration, a NOR-type SPI flash memory, part number `IS25LP128` is included. The memory has a capacity of 128 megabits (16 MiB), which should provide significant margin considering the low amount of data that needs to be stored per day. For example, at a sustained write rate of 64 bytes per hour, it would take nearly 30 years to fill the memory up. Although the lifespan of the device might be even longer, it is not expected that such long data retention would be needed.

[2]Sufficiently enough to make such sum entirely meaningless.

The memory was tested at PSI in 2018 using 200 MeV protons with a flux of $2.3 \cdot 10^8$ $\mathrm{p \cdot cm^{-2} \cdot s^{-1}}$ [27]. Two tests were carried out, one the device unpowered (OFF mode), and one with the device powered (POWER DOWN mode). 3 chips were used in each test.

In OFF mode, no data corruption or functional issue occurred up to 500 Gy. The devices failed between 500 Gy and 1000 Gy, and their current consumption increased by a factor of 3.3. In POWER DOWN mode, 2 of the 3 samples failed between 200 Gy and 300 Gy, after which no data could be read out anymore.

4.1.4 LoRa module choice and suitability

LoRa connectivity is realized by the low-cost RFM95W module. This module was tested as part of an Adafruit Feather M0 evaluation board at PSI in 2018 using 230 MeV protons with different fluxes between $4.3 \cdot 10^7$ and $2.56 \cdot 10^8$ $\mathrm{p \cdot cm^{-2} \cdot s^{-1}}$ [28]. The evaluation board contains the aforementioned module along with a Cortex-M0-based MCU. Only the module was irradiated.

The device was configured to continuously transmit LoRa packets (the transmission period is not mentioned in the test report), to which a tester would reply by sending its own packets. A watchdog was configured on the MCU so that if no packet would be received for 20 seconds – because the module failed to either sent its packet or to receive the reply –, the board would reset. On one occasion, a bit was flipped in the transmitted packet.

Over a total of 830 Gy, the device had functional interrupts (summarized in table 4.2), but was recovered every time by the watchdog. By the end of the test, the current consumption of the device had risen to about 150 % of the initial value.

Test / type of fault	Events	Fluence $[\mathrm{p \cdot cm^{-2}}]$	$\mathcal{E}\{\sigma\}$ $[\mathrm{cm^2}]$	95% confidence interval of σ
Watchdog reset	19	$4.86 \cdot 10^{11}$	$3.91 \cdot 10^{-11}$	$(2.30;\ 6.14) \cdot 10^{-11}$
Data corruption	1	$4.86 \cdot 10^{11}$	$2.06 \cdot 10^{-12}$	$(0.04;\ 11.47) \cdot 10^{-12}$

Table 4.2: Summary of SEE test results for RFM95W, including calculated cross-sections. Test at PSI, 230 MeV protons.

4.1.5 Watchdog timer choice and suitability

A good choice of external watchdog timer can increase the node's ability to recover from a single-event functional interrupt occurring in the MCU. The chosen TPL5110 has a settable period between 100 milliseconds and 2 hours. Once per this interval, the timer produces a rising edge on one of its pins, and expects a similar response from the MCU; if this response does not arrive before the next interval elapses, the MCU is forcibly reset. For the sensor node, a 60-second period was chosen as a compromise between recovery time in case of a fault and impact on power consumption due to the periodic wake-up.

The TPL5110 was tested in 2016 at CHARM (position 13, copper target, no shielding) [29]. The focus of the test was to verify the stability of timer period w.r.t. TID, to check if it

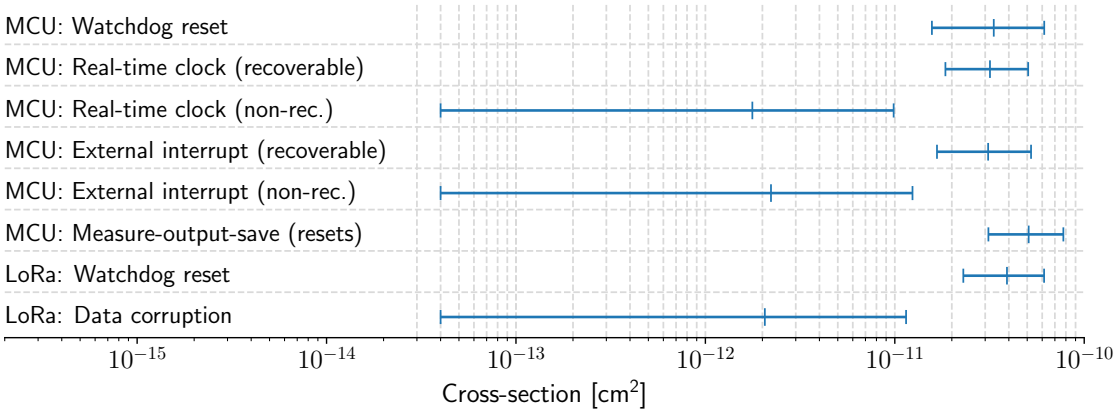


Figure 4.4: Comparison of the SEE-induced fault cross-sections for the MCU and LoRa modem, with 95 % confidence intervals. The generally large uncertainties are due to low numbers of events in the tests.

retains its ability to cut the power to the device when it receives an end-of-operation (DONE) signal, and finally, to judge the device's latch-up sensitivity. 3 chips were simultaneously exposed to a total dose of 340 Gy.

It was found that up to about 250 Gy, the timer period stayed within 0.5 % of the initial value. After 280 Gy, the timing error of one of the DUTs[3] rose sharply and at 320 Gy it failed completely (stopped switching the power output). For the other two devices, the timing error rose gradually beyond 250 Gy. No latch-up was observed during the entire run.

Statistically, the expected dose at which the period deviation exceeds 1 % evaluates to (294 ± 108) Gy (ssd). The high uncertainty is in large part due to the 35 % dosimetry uncertainty of CHARM.

4.1.6 Input/output interfaces

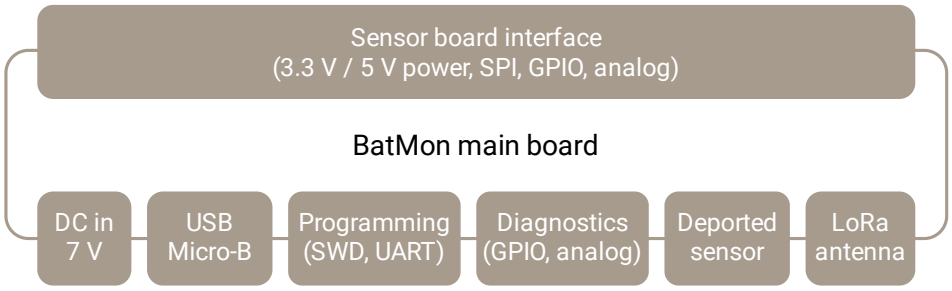


Figure 4.5: I/O interfaces of the main board

All of the board's numerous interfaces are shown in figure 4.5. Some of these contain redundant signals where needed to enable re-use of previously built accessories.

[3]Devices under test

The sensor board plugs directly into a pair of high-density board-to-board connectors (TE Connectivity 5767096-8). The connectors provide not only electrical connections, but also all mechanical support. The connectors contain two power rails, an SPI interface and a limited number of general digital and analog signals.

The main board also includes a USB Micro-B receptacle, connected directly to the MCU as a possible replacement for the UART interface. However, the USB interface has not been implemented in software because its principle of operation prevents capturing of diagnostic output from early boot stage of the MCU (before the USB stack is initialized), which limits its usefulness.

The *deported sensor* connector allows a sensor board to be placed up to 50 meters away from the rest of the system, so that measurements can be taken in high-radiation areas while exposing only the minimum subset of necessary components.

4.2 Radiation sensor board

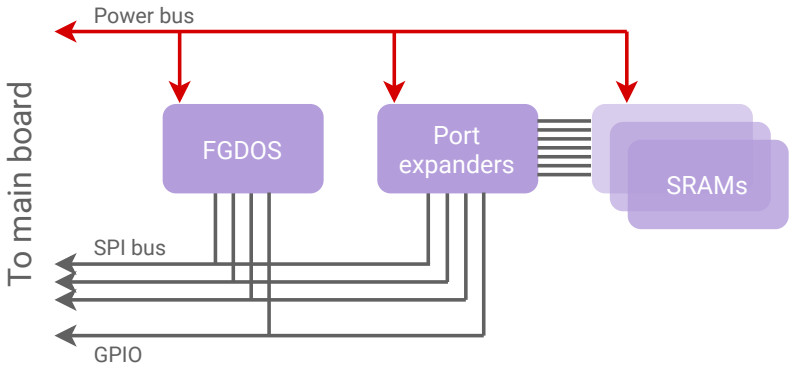


Figure 4.6: Block diagram of the radiation sensor board. Bus widths are only illustrative.

The radiation sensor board (figures 4.6, 4.7) carries the FGDOS (1 chip containing 2 separate sensors) and two sets of 4 SRAM chips each, per table 4.3. This version of the board does not contain a separate sensor for displacement damage.

Manufacturer, part number	Technology	Lot	σ_{ThN} [cm^2 · bit^{-1}]	σ_{HEH} [cm^2 · bit^{-1}]
Cypress CY62167GE	65 nm	1649	$< 1 \cdot 10^{-15}$	$\mathbf{4.461 \cdot 10^{-14}}$
Toshiba TC554001AF-70L	400 nm	0013	$\mathbf{2.40 \cdot 10^{-13}}$	$4.99 \cdot 10^{-14}$

Table 4.3: SRAMs used on the sensor board. Cross-sections apply only to the specified lot. σ_{ThN} of CY62167GE is provisional.

Because of the limited number of I/O pins available on the MCU, the memories do not

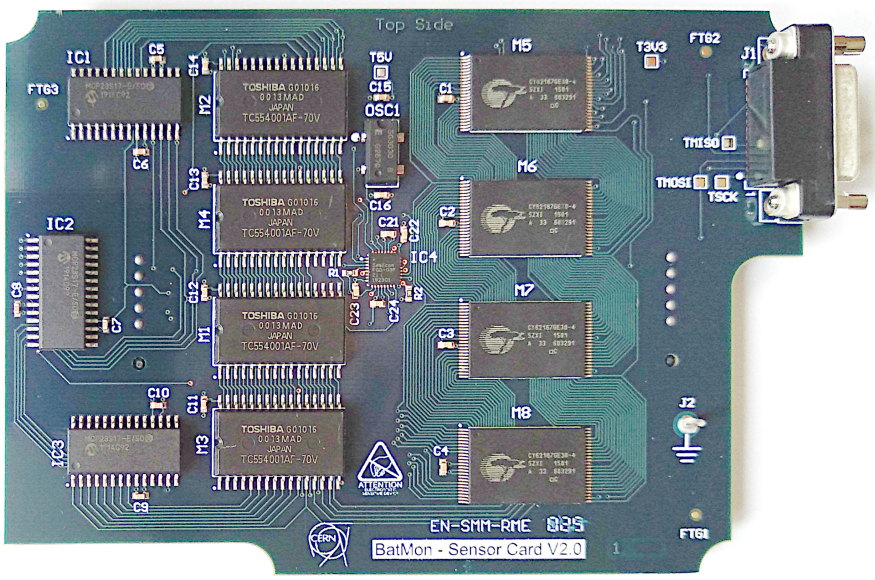


Figure 4.7: Radiation sensor board. Notice the same deported connector present on the main board. FGDOS is the small-footprint chip in the center of the board.

interface to it directly. Instead, three 16-pin port expanders (part no. `MCP23S17`) are used, connected to the common SPI bus. This marks a negative impact on scrubbing speed of the SRAMs – it takes over 3 minutes for all 8 chips.

4.3 Peripherals and accessories

4.3.1 Battery board

The BatMon is build from the ground up to operate without wires, but the main board does not carry any power source of its own. The battery board (figure 4.8) is a separate PCB, which would be typically mounted on the bottom of the enclosure, with the main board on top. The battery board takes 4 size C primary cells based on lithium chemistry, which have a nominal voltage of 3.6 V and capacity of 8.5 Ah each. They are connected in a 2s2p configuration (2 in series × 2 parallel) to provide approximately 7 volts. The board is also equipped with a power switch.

4.3.2 Break-out board

To aid in development and testing, but also in configuration for deployment, a “break-out” board has been produced (figure 4.9). This board connects to the BatMon main board via a flat cable, and makes its I/O interfaces more conveniently accessible:

- standard 4mm banana sockets for power input (7 V DC)

Figure 4.8: Assembled battery board including vertical supports for mounting of the main board

- a USB Type-C connector to access the serial interface (UART) of the BatMon via an on-board adapter chip
- a 10-pin, 1.27mm pitch header for programming

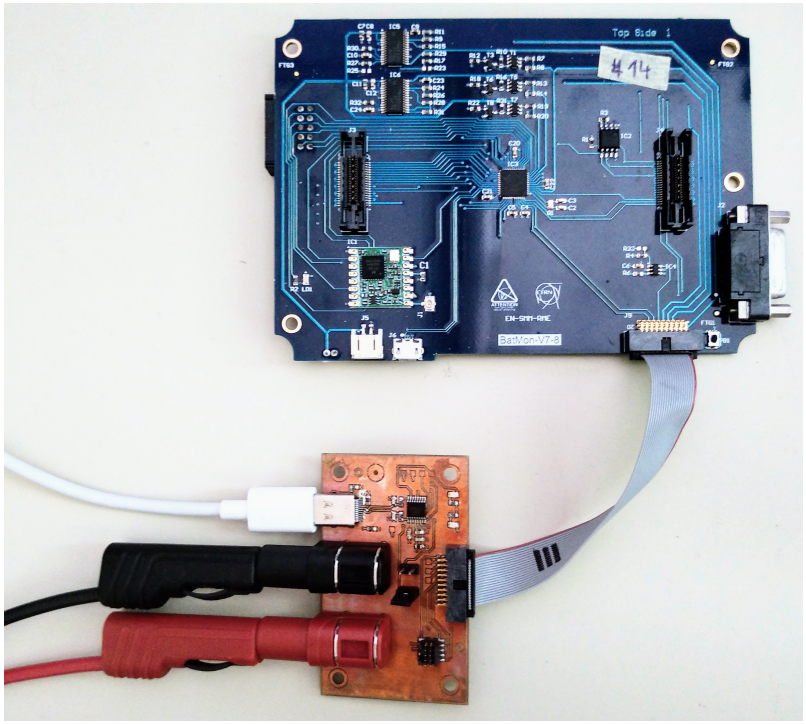


Figure 4.9: BatMon with a break-out board connected

Chapter 5

BatMon software platform

The architecture of the software package for the sensor node closely reflects the architecture of the hardware presented in chapter 4: a modular system with a base Internet-of-Things platform, upon which different sensor solutions can be implemented. This is illustrated in figure 5.1. The software package is written in the C language. Components are selected at compile time and linked to produce a monolithic firmware image.

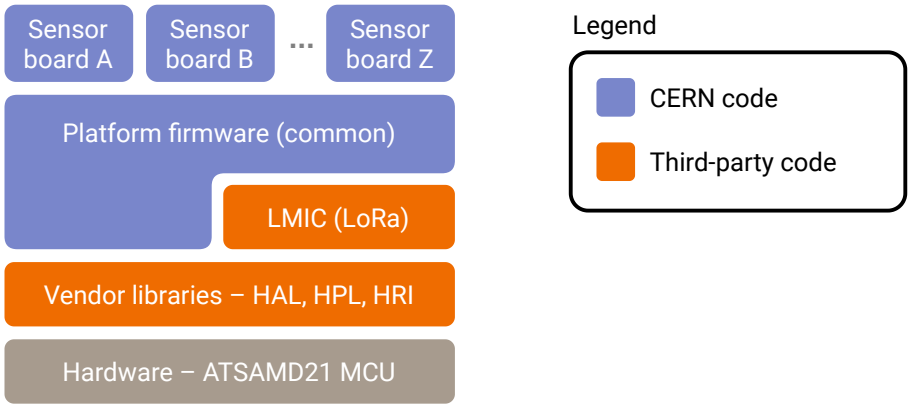


Figure 5.1: Architecture of BatMon sensor node software

To encourage code re-use, the shared *platform firmware* implements all functionality needed by the main board, as well as some auxiliary functions that are generally useful in development of sensor board drivers. To this end, the platform firmware exposes an application programming interface (API) that the driver can use to access these functions. Conversely, the drivers provide a unified interface for integration into the platform.

The modules are shown in more detail in figure 5.2 and some of them are further described below.

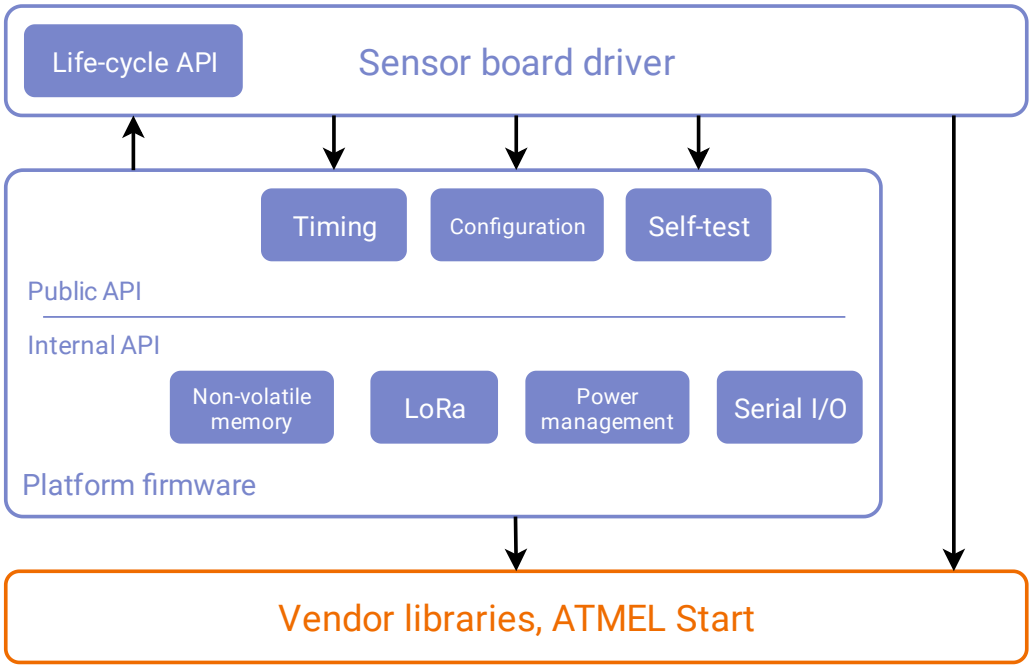


Figure 5.2: Modules and APIs comprising the platform firmware. Arrows indicate direction of function calls

5.1 Hardware and vendor libraries

The ATSAMD21 microcontroller is based on the ARM Cortex-M0+ processor core with up to 32 KiB SRAM, up to 256 KiB flash, and a variety of hardware peripherals. The Cortex-M0+, in turn, is an implementation of the 32-bit ARMv6-M architecture.

The vendor (Microchip, previously Atmel) provides a collection of software components comprising the proprietary *Advanced Software Framework*:

- HRI (Hardware Register Interface) – the lowest layer of abstraction. It provides register definitions and helper functions (get, set, toggle and others; both for entire registers and individual fields in them).
- HPL (Hardware Proxy Layer) – builds on top of HRI to implement primitive operations in the hardware peripherals. Provides a higher degree of portability than HRI, at the possible cost of reduced efficiency.
- HAL (Hardware Abstraction Layer) – builds on top of HPL to provide high-level abstractions. As an example, for communication peripherals, HAL implements buffering and asynchronous operations. Provides a higher degree of portability than HPL, but depending on the selected options, can incur high overhead and increase code size.
- Atmel START is a graphical configuration tool that enables the user to select options for peripherals and add high-level software building blocks into the project. It can significantly accelerate initial project setup and help in areas that are usually tricky

to configure, such as clocks. Atmel START relies on code generation, which means that only necessary code will be included, however it also means that it is difficult to integrate into applications requiring high re-configurability. It is directly comparable to STM32CubeMX, a similar tool for STM32 series microcontrollers.

5.2 Platform firmware

Besides the APIs introduced above and further detailed below, the platform firmware also implements the main loop of the program. After power-on, all hardware is initialized and diagnostic information is printed to the serial output. If the device has been configured, it also begins to connect to the LoRaWAN network. From there onwards, the program runs in an infinite loop (figure 5.3).

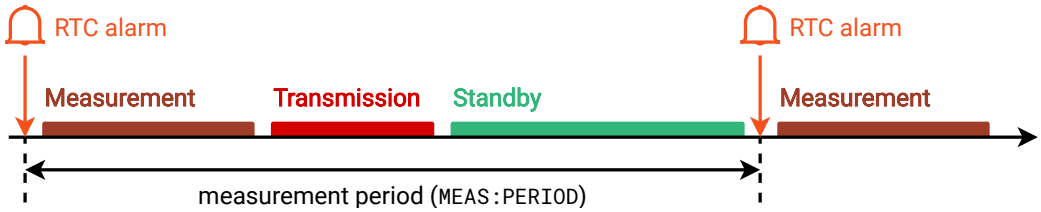


Figure 5.3: A measurement cycle

At the beginning of every iteration, measurement is performed. This operation is fully deferred to the sensor board driver and, depending on the used sensor(s), can take from a fraction of a second up to several minutes for SRAM scrubbing. Afterwards, if the device has succeeded in joining the network, a data packet is composed and sent. The rest of the *measurement cycle* is spend in standby mode, with most peripherals shut down to minimize power consumption.

5.2.1 Power management module

Unlike other functional blocks, power management cannot be entirely *compartmentalized* – that is, wrapped up in a compact module with a simple interface. Instead, it requires consideration throughout the design of the entire application; every part needs to be prepared for the possibility of entering a different power state, losing access to parts of hardware and so on.

Nevertheless, the firmware does contain a power management module, which performs transitions between the different power states as depicted in figure 5.4. When it does so, power management functions of the other modules are called to perform additional necessary operations (for example, when the BatMon enters standby mode, the SPI flash module will send an *Enter Power Down* command to the external memory).

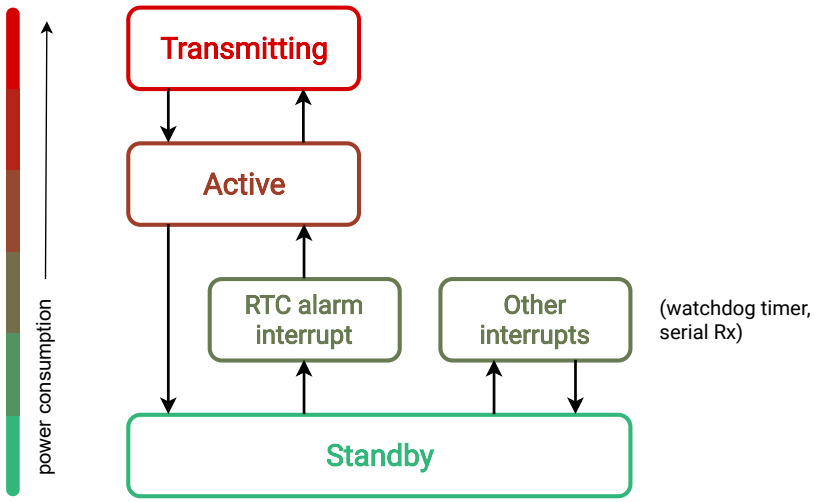


Figure 5.4: Power states and transitions between them

5.2.2 Communications module

Although the LMIC library (described in section 3.3.2) handles much of the LoRaWAN protocol, additional logic is needed for power state management, as well as for coping with network issues and for extraction of useful diagnostic information. Specifically, if connection cannot be established within a certain period of time, the device *backs off* from the connection attempt, goes into standby, an re-tries at a later time to avoid unnecessarily draining the battery (figure 5.5).

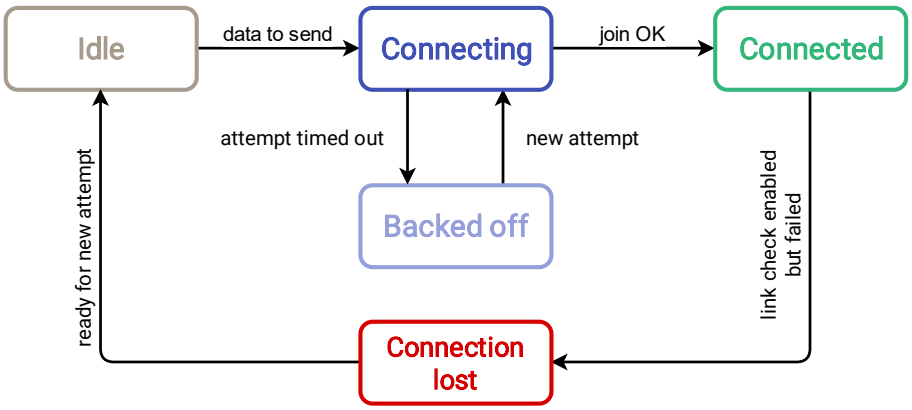


Figure 5.5: LoRa high-level connection states

In some parts, the library had to be extended to provide access to internal variables and to allow dynamically configuring some parameters (which would be otherwise fixed at compile time).

Packet payload format

Modularity of the platform is reflected also in the payload format: one part of it is defined by the BatMon platform, while the other is defined by the used sensor.

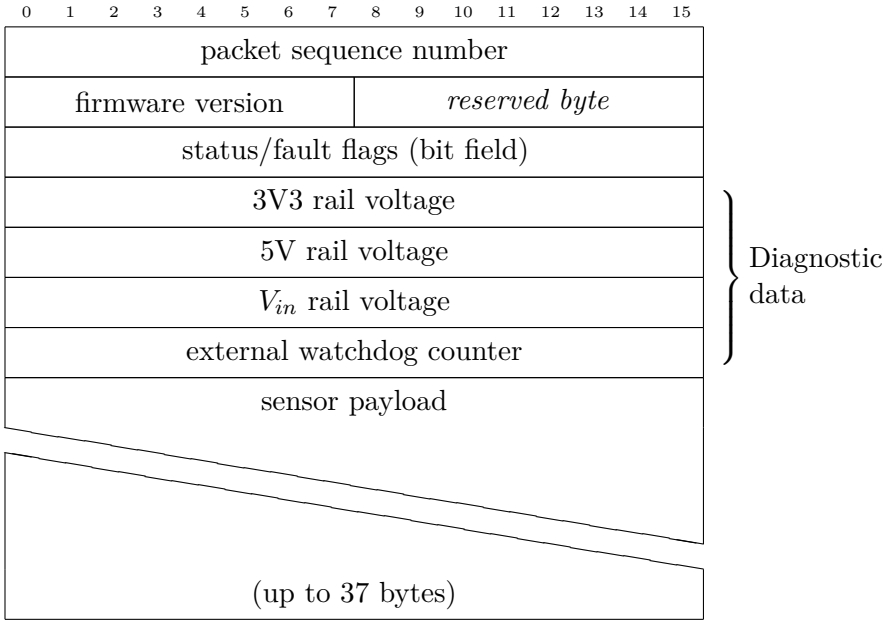


Figure 5.6: LoRa payload format (one row = 2 bytes)

The common part carries mostly status and diagnostic information; data valuable for the application will be embedded in the *sensor payload*.

It is worth noting that the sensor type ID is *not* included in the payload. Instead, this information is encoded in the LoRa port that the packet is sent to. Specifically, a unique port number is assigned to each combination of *protocol version* and *sensor type*.

The same format is also used for data logging to non-volatile memory (detailed in section 5.2.3). In fact, the packet is only composed once and then transmitted and logged simultaneously.

5.2.3 Non-volatile memory partitioning and formatting

The non-volatile memory subsystem serves two purposes: preserving node configuration and logging of measurement data.

The node actually contains two non-volatile memories: the internal memory of the MCU, which also contains its firmware, and an SPI flash memory on the main board. All of the external memory and a small part of the internal memory can be used by the application. The available memory regions are divided into partitions, as shown in figure 5.7.

I-Config and E-Config partitions are used to store configuration information (detailed below), while the Data partition is used to log measurements. The design and implementation of the formatting scheme is driven by specifics of the flash technology:

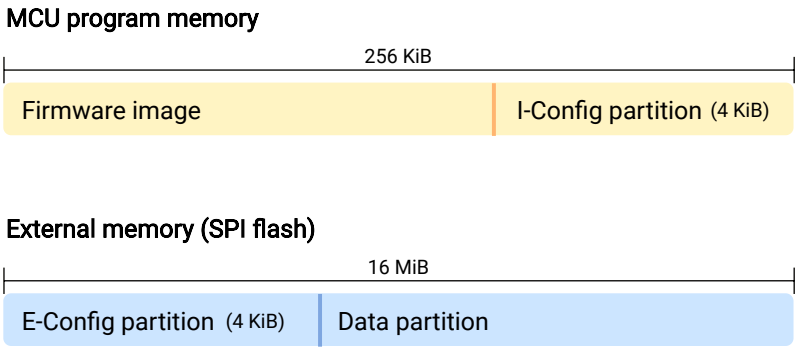


Figure 5.7: Partitioning of the non-volatile memories

- Memory must be *erased* before it can be *written*. Writing can only change bits from 1 to 0, an erase is needed to change 0 to 1.
- The granularity for erasing is larger than that for writing. For example, the `IS25LP128` can write individual bytes, but can only erase *sectors* of 4 KiB.
- Although byte-level addressing is used for writing, it is actually possible to alter individual *bits*, by setting the other bits in a byte write operation to 1. As it is not possible to change bits from 0 to 1 in a write operation, those bits will preserve their value.[1]

The partitions are first subdivided into sectors and finally into fixed-size *records*. From the point of view of the application, one record is the minimum writable unit. The size of the record has been chosen as 128 bytes. On the one hand, using records of fixed, power-of-two size simplifies addressing and many other operations, on the other hand, if the used length of every record is much smaller than its total size, a significant portion of the memory capacity will be wasted. However, the latter is not a significant concern considering the projected measurement frequency and the available memory capacity, so this trade-off is made with the aim of increasing robustness of the memory subsystem.

The Config partitions will typically not contain more than 2 records (one for platform configuration, one for sensor configuration and/or calibration), and will be rarely overwritten, therefore it is not necessary to particularly optimize the storage and erase pattern. The Data partition, on the other hand, is large enough and will see enough traffic to be a possible bottleneck in performance or reliability.

Organization of the data partition

On a high level, the Data partition can be understood as a vector of records with constant time *push back* and *pop front* operations, or equivalently a first-in-first-out queue. This

[1] This does not necessarily hold true for memories from other vendors. Some explicitly warn that overwritng the same location multiple times may cause unexpected data corruption, or even damage some bit cells by discharging them too deeply.

allows, for example, to use the memory as a "back buffer" of data that might need to be re-transmitted later, until the network confirms reception of a range of records.

On a lower level, the Data partition is implemented as a ring buffer with one pointer for writing, and another for reading. Upon device start-up, it is necessary to recover these pointers; a naïve solution might be to simply store them at a fixed address in the memory, however this brings several issues:

- The minimum erasable unit of the flash memory is one sector (4 KiB), of which only a few bytes would be used
- The number of erase/write cycles of each memory location is limited; for the used IC, it is 100 000 cycles. For example, with one cycle every 5 minutes, this budget is exhausted in 348 days.
- If the power to the device or the memory happens to be cut before the new write is completed, the pointers would be corrupted or lost.

A better solution might be to not erase the sector every time, but instead keep storing the updated pointers at subsequent addresses until the sector is full, and only then erase it. This solves the issue of erase/write cycles for many realistic applications, but there is still a small window of opportunity for losing the pointers when the sector becomes full and needs to be erased. There was also a concern of localized memory corruption or permanent damage due to ionizing radiation. For these reasons, a more robust solution was sought, one that would not rely on any single memory location.

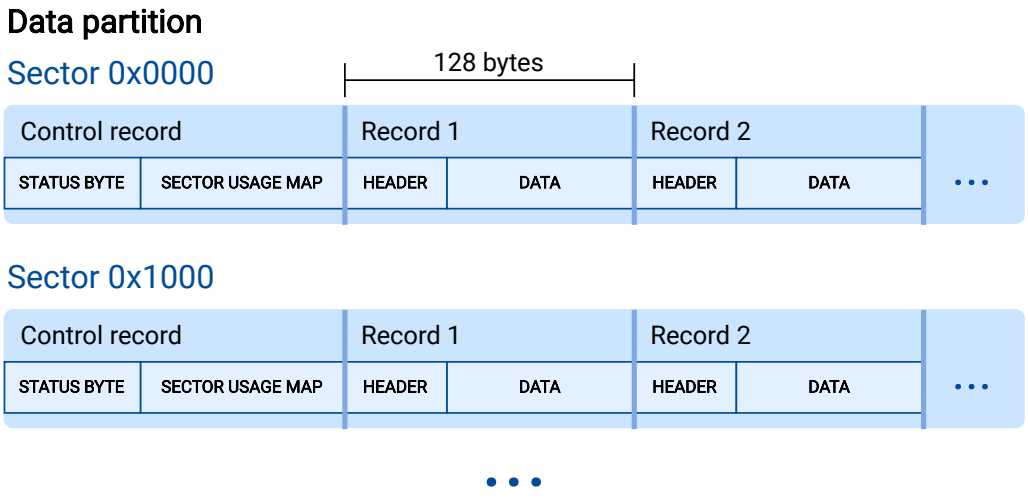


Figure 5.8: Sector formatting in the Data partition

Figure 5.8 depicts the chosen approach. Instead of storing the value of the pointers, sectors and records are flagged in a way that makes it possible to recover the pointers, even in presence of localized corruption.

The first byte in every sector is designated as a status byte. Because erasing the flash memory sets all bits to 1, the value of `0xFF` has been chosen to represent an empty sector.

The first time a record is written into the sector, a bit (U) in the status byte is set to 0 to indicate that the sector is partially used. When the sector is filled up, another bit (FU) is cleared to indicate this. Finally, once some records in the sector have been read and deleted, a third bit (D, for dirty) is set to 0. There is no "fully dirty" (FD) bit; instead, when the last record in a sector is removed, the entire sector is erased, making it available for new data.

When recovering the read and write pointers r and w, the following set of rules can be then used to quickly narrow the search down to specific sectors:

1. Scan the status byte of each sector in the partition. If all sectors have a status of `0xFF`, assume the memory is empty and set $w = r = 0$.
2. A partially used sector ($U = 0$, $FU = 1$) is a candidate to contain w.
3. A sector with $U = 1$, $FU = 1$ (empty) following a sector with $U = 0$, $FU = 0$ (fully used) is a candidate to have w point to its first record slot.
4. A dirty sector ($D = 0$) is a candidate to contain r.
5. If there is no sector with $U = 0$, assume that no records have been written and set $w = 0$.
6. If there is no sector with $D = 0$, assume that no records have been marked dirty and set r to point to the first position of a sector with $U = 0$ following a sector with $U = 1$. If there is no such sector, set $r = 0$.

Figures 5.9 and 5.10 illustrate some of the expected scenarios. To ensure that the logic works correctly, certain constraints must additionally be satisfied at all times:

- *At most one* sector can be partially used (having $U = 0$, $FU = 1$).
- *All* used sectors ($U = 0$), if any, must form one contiguous sequence, possibly wrapping from the end of the memory to the beginning.
- *At most one* sector may be marked dirty ($D = 0$).
- *At least one* sector must be either empty or only partially used ($FU = 1$). This implies that the capacity of the memory cannot be used in its entirety, otherwise the position of r and w would be ambiguous under this scheme. This could be worked around, for example, by using multiple bits for each of the U, FU and D variables, allowing to make a distinction among two fully used sectors.

When one of more of these constraints are broken, the algorithm may yield multiple candidate sectors. In that case, it is assumed that memory corruption has occurred, and a slower fall-back algorithm must be used to scan individual records, determine which are valid, and restore the constraints before more records can be written or read.

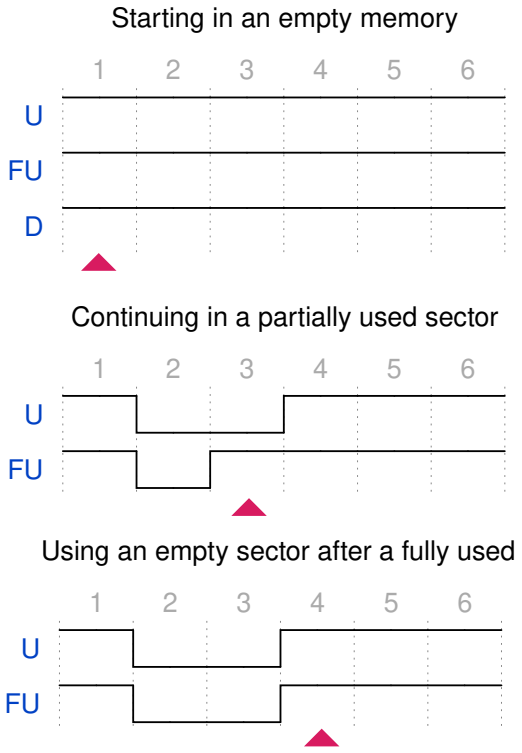


Figure 5.9: Narrowing down the search: write pointer

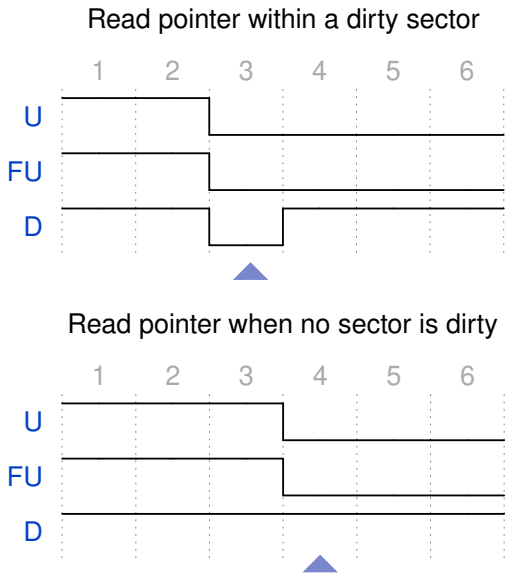


Figure 5.10: Narrowing down the search: read pointer

Once the search has been narrowed down, the corresponding *sector usage map* is read (see again figures 5.8 and 5.11). This map tracks the usage of record slots within the sector, in a manner very similar to how sector status bytes track the usage of sectors. Because records cannot be written partially, record slots can only be in one of 3 states: empty ($U = 1$, $D = 1$), used ($U = 0$, $D = 1$) and dirty ($U = 0$, $D = 0$).

The read and write pointers can be finally found by finding the first non-dirty ($D = 1$) and non-used ($U = 1$) records, respectively, within their candidate sectors (figure 5.12).

Record slot format

Figure 5.13 shows the structure of record slots. A fixed-length area is provided for the contents of the record, but the actual number of used bytes is noted. The application-level record type is encoded by a combination of two fields: the 1-bit C determines if the record contains configuration (1) or data (0). The *tag* field then encodes the sub-type (for data, the same assignment as for LoRa port numbers is used).

A *CRC-16* checksum is also included so that integrity of the record can be verified. The checksum is computed over the header and the valid part of the payload, however this introduces a self-referential problem, because the checksum field itself is part of the header. The issue is resolved by setting the CRC field to 0 for purposes of checksum calculation.

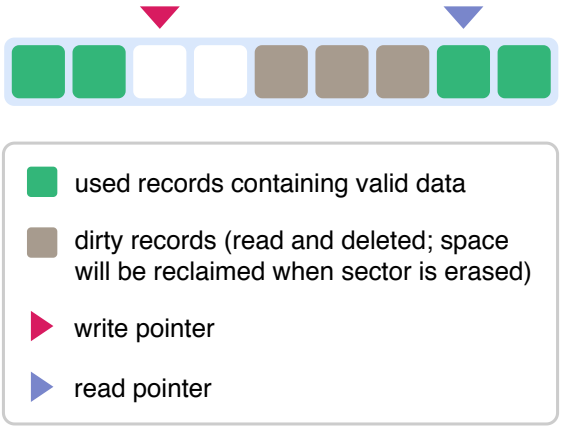


Figure 5.11: Various record states and corresponding read/write pointers

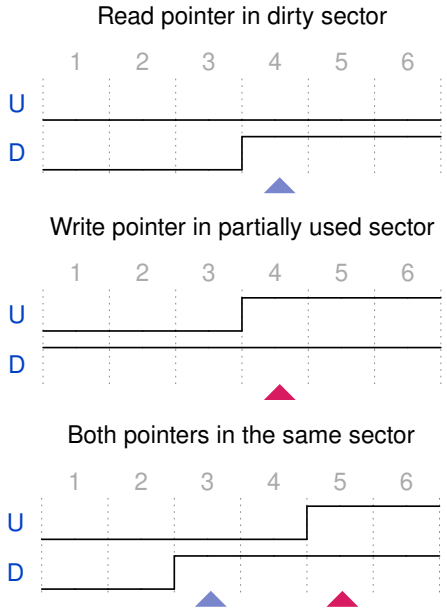


Figure 5.12: Recovering the exact pointers

5.2.4 Serial interface for control and configuration

Operation of the node is controlled by several variables. Their values may vary between deployments (e.g. the chosen measurement period) or may even need to be unique for each device (LoRa DevEUI and authentication keys).

The default values for these variables are specified in the firmware source code. This would mean, however, that for every deployed device, a unique firmware build was needed. Not only this adds extra work, it also makes version tracking more difficult.

For this reason, these variables can be changed independently of the firmware and saved in one of the Config partitions described in section 5.2.3. The device exposes a text-based command interface, where the command structure and naming conventions are inspired by the SCPI standard [30]. This interface can be used to query or set the variables (5.1) and to invoke a few general commands (5.2).

The interface is line-oriented, with the following syntax:

- variables can be queried by specifying the name of the variable, followed by a question mark; e.g. `LORA:DEVEUI?`
- variables can be set by specifying the name of the variable, followed by a space and the new value; e.g. `LORA:DEVEUI 1122334455667788`
- commands may be sent using their respective syntax as per table 5.2

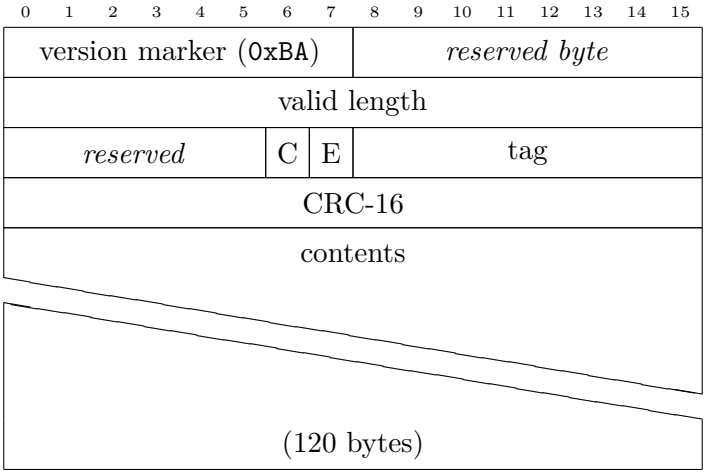


Figure 5.13: Record slot in non-volatile memory (one row = 2 bytes; 128 bytes in total)

Variable	Access	/ type	Default	Description
`LORA:APPEUI`	R/W	uint64	N/A	LoRa application ID
`LORA:APPKEY`	R/W	byte[16]	N/A	LoRa authentication key
`LORA:DEVEUI`	R/W	uint64	N/A	LoRa device ID
`LORA:DR`	R/W	enum	automatic	LoRa data rate
`LORA:LINKCHECK`	R/W	boolean	false	Enable adaptive data rate (ADR)?
`LORA:JOINDR`	R/W	enum	SF7/BW125	Initial data rate
`LORA:PORT`	R/W	uint8	N/A	LoRa port number for data packets
`MEAS:PERIOD`	R/W	uint32	300 seconds	Measurement period
`SYSTEM:FWCRC`	R	uint32	N/A	Firmware image checksum

Table 5.1: Variables exposed by the firmware

Implementation details

The board is equipped with two interfaces for serial communication: duplex UART (Rx/Tx) and USB. The considerable additional complexity of USB can be mostly abstracted by the vendor libraries, but both interfaces share a common caveat: the data is sent by the host computer *asynchronously* with respect to the program running on the MCU. To ensure that no data is lost, reception of the data in the MCU is handled in a high-priority interrupt service routine. (An alternative would be to use Direct Memory Access – DMA – to buffer incoming data automatically. Interrupts were used because they are simpler to configure and their additional processing load does not negatively affect the application, due to low amount of data exchanged)

When substantial work is to be performed in an interrupt routine, careful synchronization with the main thread is needed to ensure that shared data structures are not overwritten or corrupted in the middle of another operation. Similarly, hardware resources such as the SPI bus have to be protected from conflicting access. Not only can the amount of extra

Command	Description
`CONFIG?`	Output the current values of all configuration variables.
`CONFIG:SAVE`	Save current configuration to non-volatile memory.
`CONFIG:SCAN?`	Output the checksums of any saved configuration records and the checksum of the current configuration (in RAM).
`ERASE`	Erase all writable partitions. (figure 5.7)
`RESET`	Restart the node. Any unsaved configuration changes are discarded.

Table 5.2: General commands

code be substantial, it is also extremely difficult to verify its correctness, as multi-threading is subject to various race conditions and synchronization errors. To minimize these risks, the routine is tasked with only storing the input, and command execution is deferred to the main thread.

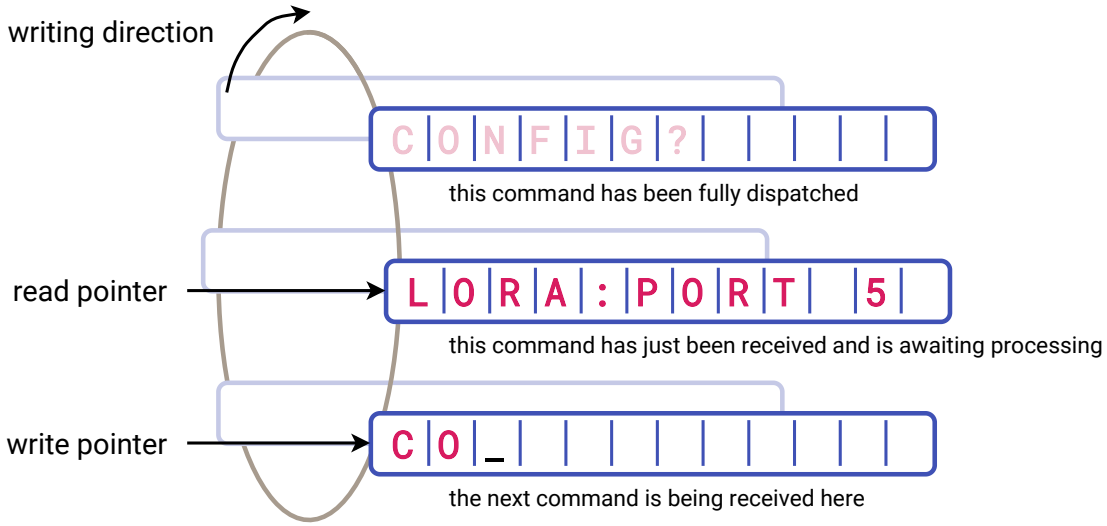


Figure 5.14: Buffering of the serial input

The data is stored in a ring queue of fixed-size buffers (figure 5.14), where every entry represents one command to be processed. When the end-of-line marker is received, the current buffer is marked as complete and handed off to the main thread for parsing and processing of the command. Further received data is stored in the next buffer. After the command has been processed, the buffer is marked available, and can be used again for reception. This way, only a minimum amount of information (namely, the read/write pointers) needs to be synchronized carefully.

Two parameters need to be selected: the capacity of each buffer must be sufficient to accommodate the longest permitted command,[2] and the queue depth (number of buffers) will determine how many commands can be "bursted" at once without losing any data. In practice, 4 buffers of 64 bytes have yielded satisfactory results.

[2] In practice, the longest command is `LORA:APPKEY <value>`, with 44 characters plus termination.

5.2.5 Timekeeping services

The platform provides two clocks: firstly, a monotonic clock with millisecond resolution, which starts as 0 at boot time and counts up whenever the MCU is running in active mode; secondly, a real-time clock with 1-second resolution, which continues to run also in standby mode.

Millisecond-resolution clock

The millisecond-resolution clock can be used for many timing-related tasks. Its caveat is that it is paused when the MCU is in standby mode – from the clock's point of view, no time passes when the node is asleep.

The resolution of the clock is 32 bits, which means that it will overflow every 2^{32} milliseconds, or 49.71 days of accumulated run time. The severity of this figure will depend on the deployment configuration – a node that wakes up for 5 minutes every day would need almost *40 years* to accumulate such run time. However, if we accept the possibility of nodes supplied with wall power, and taking measurements several times a day, clock overflow becomes entirely realistic.

Therefore, the value of the clock is interpreted using modular arithmetic; the absolute value of the clock loses any meaning, and the only permissible operation is subtraction of two timestamps, $t_2 - t_1$, where t_2 was taken after t_1. The difference of the two timestamps is taken modulo 2^{32} to obtain the number of milliseconds elapsed from t_1 to t_2.

Real-time clock

A real-time clock (RTC) is typically used to keep track of calendar time. However, the BatMon does not have a source of absolute time information (such as a GPS receiver), nor is it able to track time when the device is turned off. Therefore, when the device is powered on, the real-time clock starts on an arbitrarily chosen date, and is only used to track time *incrementally*, to schedule the periodic measurement cycles.

The RTC in the MCU provides an alarm functionality, which triggers an interrupt when the specified date and/or time is reached. At the end of a measurement cycle, before low-power mode is entered, the alarm is set so that the device wakes up for the next cycle.

5.2.6 Self-test functions

While high confidence in device reliability can only be established by extensive testing, some issues may be easy to discover by the device itself. A self-test is implemented for quick validation before deployment. It is not exhaustive, but it is very cheap. However, the test is *invasive*, in that it modifies the configuration of the sensor, writes to the flash memory etc. Therefore, it is not suitable to be executed periodically in the field, as it might create measurement errors and pollute the data log.

The results of the test are output through the serial interface, in a format that is human-readable ("engineer-readable", to be more precise), but also sufficiently structured to allow a hypothetical automated tool to parse and categorize detected errors in mass production.

5.3 Radiation sensor board

5.3.1 Sensor data processing

The sensor board driver is activated every time the measurement period elapses. Due to the drawbacks of the autonomous mode of the FGDOS, outlined in section 2.4.5, it is operated in manual mode, and thus its remaining charge must be sampled and, if needed, a recharge performed.

At the beginning of a measurement cycle, the SRAMs are overwritten with a predefined pattern. Due to the symmetrical nature of 6T SRAM, the probability of bits flipping from 0 to 1 is expected to be the same as for 1 to 0. In our implementation, a balanced pattern (equal number of 0s and 1s) is used, so this expectation can be verified in practice.

After a certain amount of time, the memory is read back and the data is compared with the pattern and the flipped bits are counted. Afterwards, the memory is overwritten again and the cycle can restart.

5.3.2 Payload format

As mentioned in section 5.2.2, the LoRa packet contains a dedicated space for sensor data. Its structure in case of the radiation sensor board is indicated in figure 5.15.

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
status/fault flags (bit field)
$\frac{\text{sensor 1 TID}}{100\,\mu\text{Gy}}$ (32 bits)
$\frac{\text{sensor 2 TID}}{100\,\mu\text{Gy}}$ (32 bits)
SRAM 1 bit flip count
SRAM 2 bit flip count
SRAM 3 bit flip count
SRAM 4 bit flip count
SRAM 5 bit flip count
SRAM 6 bit flip count
SRAM 7 bit flip count
SRAM 8 bit flip count

Figure 5.15: Sensor payload format (one row = 2 bytes; 26 bytes total)

The TID of the two FGDOS sensors is transmitted as a 32-bit value with 1 LSB[3] corresponding to 100 µGy. This gives a range of 429.5 kGy. For the SEU measurement, the counts for the 8 chips are sent directly. This is likely to change in the future to transmit the evaluated fluxes and uncertainty, but at the time of writing, the required resolution and range have not been specified.

5.4 Central database and user interface for data retrieval

Historically, data from the RadMon system would be logged into the *CERN Accelerators Logging Service* (CALS). This is a massive database focused on recording time-series data from CERN's particle accelerators and a wide range of accelerator support systems. This includes beam quality, cryogenics, power distribution, radiation protection, and others, but does not include physics data from experiments. CALS is universal and extremely scalable, accumulating 2.5 TiB of data per day from over 2.3 million signals [31]; however, that is also the source of some of its drawbacks, such as lack of in-depth documentation and clunky API for data retrieval.

While there is high probability that BatMon measurements will be eventually incorporated into CALS, one of the goals of the project is to re-think how data is presented and explore other technical possibilities. In recent years, two free software projects relevant to this task have risen to popularity: *InfluxDB* and *Grafana*. InfluxDB is a *time series database*, making it similar to CALS. It is not optimized for such large throughput, however it is lightweight, widely deployed, and boasts high-quality documentation. InfluxDB can store time series of integer, floating-point, boolean and text data. It provides a SQL-like language for querying and statistical processing, named *InfluxQL*.

Where a traditional RDBMS[4] has *tables* and *rows*, InfluxDB has *measurements* and *points*. A point represents a single data record that has four components: a measurement, tag set, field set, and a timestamp. A point is uniquely identified by the measurement it belongs to, its tags and the timestamp [32]. A suitable database schema for the demonstration prototype has been implemented in InfluxDB. The schema consists of 3 measurements: one for the estimated parameters of the radiation field (`batmon_meas`), one for node diagnostic data (`batmon_diag`), and one to track deployment location (`batmon_deploy`). All of them use a tag, `DeviceID`, to pair each data point to the correct node. `batmon_meas` and `batmon_diag` are filled with data from the LoRa packet, while `batmon_deploy` must be updated manually whenever a node is commissioned and installed. Figure 5.16 shows this structure.

Data logged into InfluxDB can be conveniently visualized in *Grafana*, a web-based application that lets the user build live dashboards from graphs, tables, and other *widgets*. These dashboards can be published for other users; thus an entire class of custom applications can be replaced. Figure 5.18 shows the implemented data visualization for the BatMon system, implemented as a set of Grafana dashboards. Data (TID, flux, device status) is queried directly from InfluxDB based on the user-selected display interval. A server-side

[3] Least significant bit

[4] Relational database management system, such as MariaDB or Microsoft SQL Server

Measurement
`batmon_meas`
Tags
`DeviceID`
Fields
`timestamp`
`TID`
`FluxThN`
`FluxHEH`

Measurement
`batmon_diag`
Tags
`DeviceID`
Fields
`timestamp`
`V_bat`
`V_ref`
`V_aux`
`ErrorCode`
`SensorErrorCode`

Measurement
`batmon_deploy`
Tags
`DeviceID`
Fields
`timestamp`
`Location`

Figure 5.16: The three *measurements* (in InfluxDB terminology) providing the data back-end for the prototype application

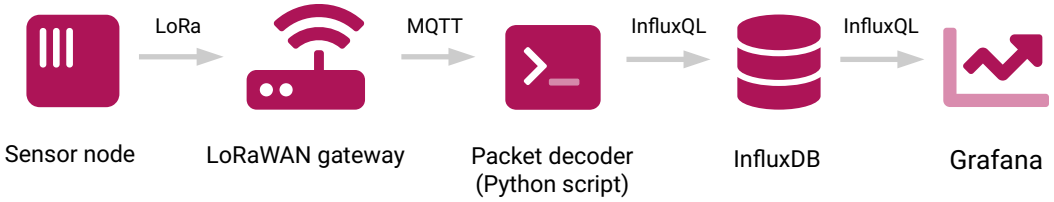


Figure 5.17: Data flow in the demonstration software stack

component of the solution listens for packets forwarded from the LoRaWAN Network Server, extracts device status and measurement data, and inserts them into the InfluxDB instance. The overall flow is depicted in figure 5.17.

The proposed solution has been implemented as a demonstration prototype, not a production-scale application; it is yet to be determined whether this proposed light-weight approach will be able to meet the needs of all users.

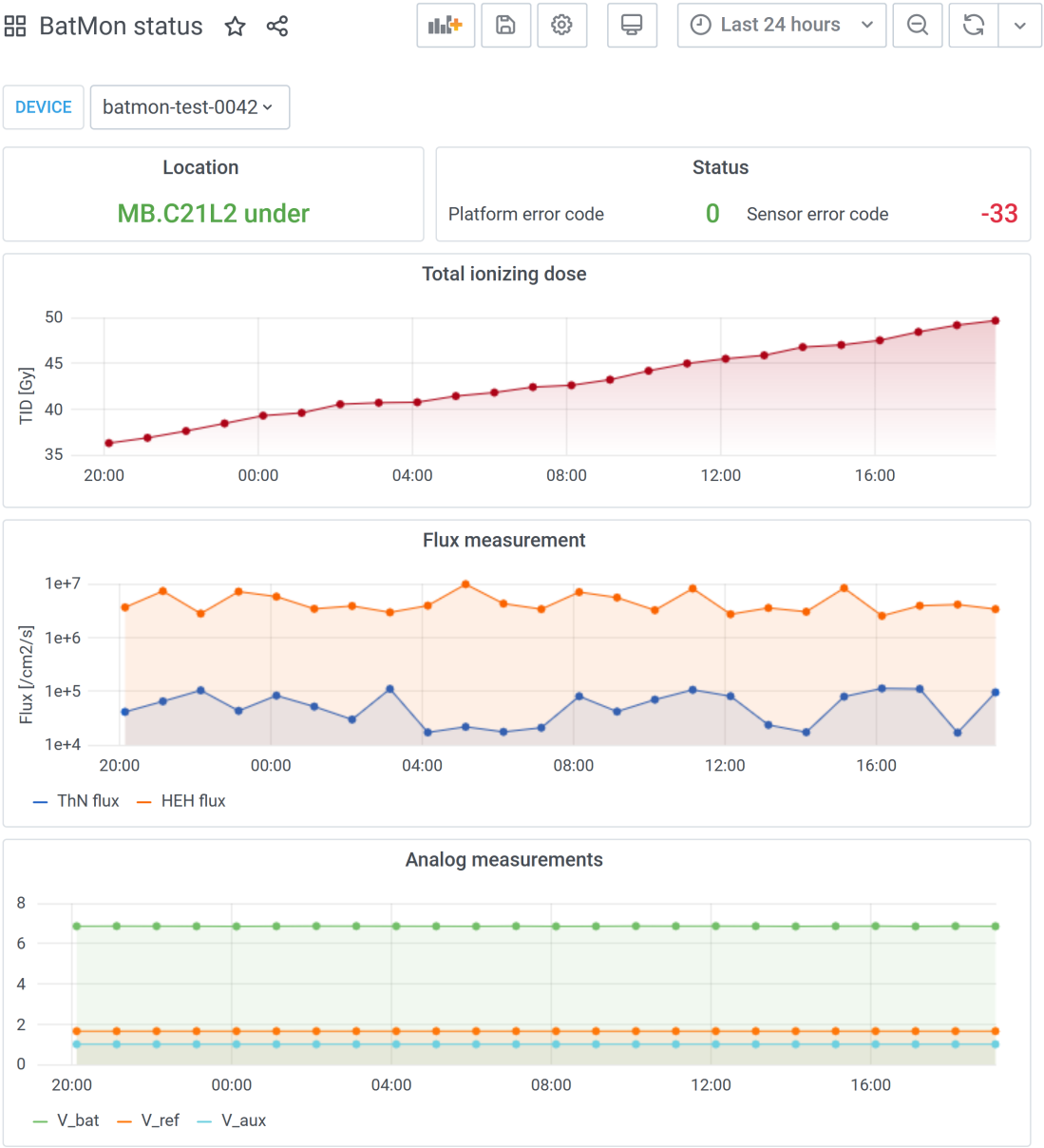


Figure 5.18: The interactive dashboard created in *Grafana*. On the top left, the user can select the device for which data will be displayed. On the top right, the time interval can be changed. Rest of the screen real estate is dedicated to intuitive visualization of the measurement and status data.

Chapter 6

Tests and validation

To gain confidence in the reliability of the system, functionality of the sensor node must be thoroughly tested. This is achieved through a combination of targeted artificial tests (e.g. accelerated exposure to radiation) and tests carried out in more realistic conditions, exercising many different aspects of the device at the same time. The most extensive of the test campaigns carried out was at the CC60 facility where multiple devices were irradiated over several weeks.

6.1 Overview of the CC60 facility

At the CC60 (CERN Cobalt-60) irradiation facility a ^{60}Co source is used to provide gamma rays for radiation testing of electronic devices (figures 6.1, 6.4, 6.3). The dose rate can be set between 120 mGy/h up to 20 Gy/h by adjusting the distance between the DUT and the source by means of a moving table (from 20 cm up to 5 meters).

Figure 6.2 depicts the nominal *calibration curve* – the relationship between position in the room and the dose rate at the time the source was installed (2014). However, as the half-life of ^{60}Co is only 5.27 years, the source activity – and thus also dose rate for any chosen position – decays fairly quickly. The up-to-date dose rate can be calibrated much more precisely using a portable ionization chamber.

Calibration. The ionization chamber is first mounted at the desired DUT position, and a cable is ran to the control room, where a data acquisition system is located. The chamber is exposed to the source for 10 seconds and the measured dose rate is recorded. Because the pneumatic mechanism for exposing the source is not perfectly predictable, the dose rate will vary slightly every time the source is exposed. For this reason, the expose-return cycle is repeated several times and the result is processed statistically; the standard deviation is typically about 2 %. Finally, after enough samples have been collected, the ionization chamber is replaced with the DUT without moving the fixture and the test itself can begin.

Other equipment. General-purpose cabling between the irradiation room and the control room makes it possible to apply power to the DUTs and perform measurements as they are

being irradiated (gamma radiation does not scatter, so this cable conduit does not create additional concerns for radiation protection).

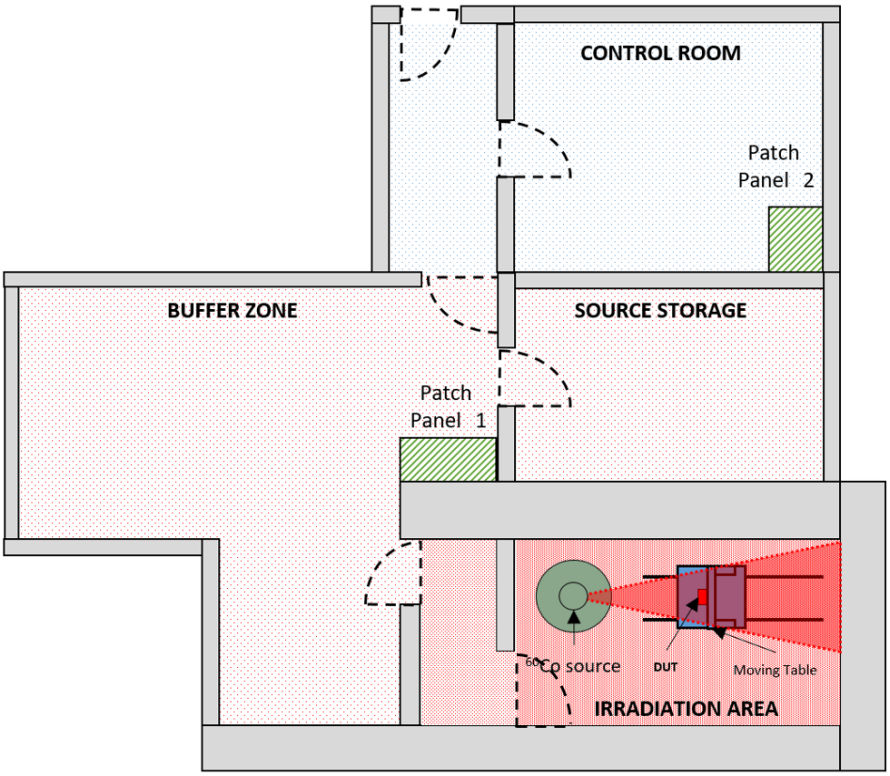


Figure 6.1: Layout of the CC60 facility

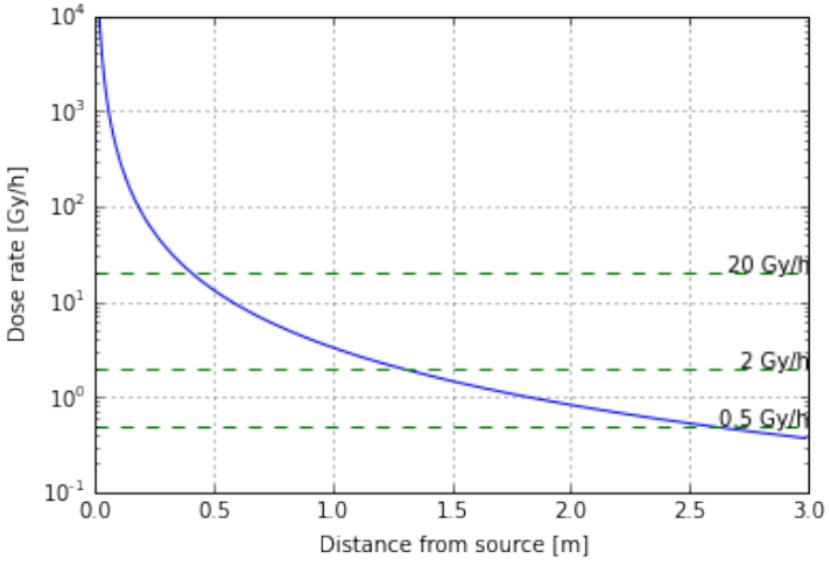


Figure 6.2: Calibration curve of the CC60 facility source

Figure 6.3: A view into the CC60 irradiation room. On the left, a 5-ton "sarcophagus" houses the 10 TBq gamma radiation source. The source is normally stored in the heavily shielded bottom part, and is pneumatically lifted to a height of 1.5 m when irradiation is being performed. A collimator is installed on the front (obscured in this picture) to shape the beam into 30° horizontally by 30° vertically.

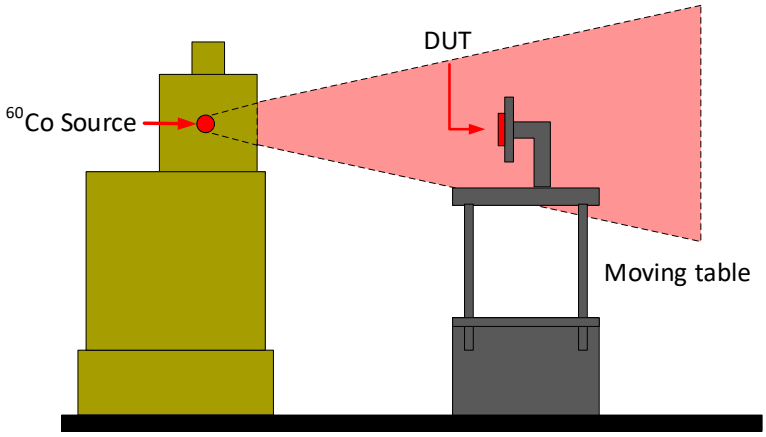


Figure 6.4: Side view of the CC60 irradiation setup

Often, USB connection to the DUT is desirable. However, the USB 2.0 specification limits cable length to 5 meters, which would be insufficient to make connection between the irradiation room and the control room. For this reason, a pair of active adapters is provided, enabling *tunneling* of up to 4 USB devices over an Ethernet cable. This significantly increases the possible distance. On the downstream computer, the USB devices are presented as if they were connected directly and no extra software drivers are required.

6.2 Dose irradiation tests of the BatMon

6.2.1 General setup

Two test campaigns have been carried out. The test was set up to mimic standard operation of the device, with some exceptions. An assembly of the main + sensor boards was used, however power was provided by a lab power supply to ensure constant monitoring and a defined current limit. In terms of software, the full functionality was enabled (measurement, transmission, logging), but the measurement cycle was shortened to collect more samples; for most of the test, the measurement period was 5 minutes.

The campaigns are summarized in table 6.1:

Campaign	Date	DUTs tested	Dose rate [$\mathrm{Gy} \cdot \mathrm{h}^{-1}$]	TID to total failure [Gy]
#1	10/2019	1	2.86	1008.3
#2	01/2020	3	4.00	615.3 820.2 525.2

Table 6.1: Summary of test campaigns carried out at CC60.

Note that in the following plots, the x-axis denotes absorbed dose. As the dose rate was constant during each campaign, the dose directly corresponds to elapsed time, except for intervals when irradiation was interrupted.

6.2.2 Evaluation of sensor node lifetime

During campaign #1, the device was working up to a TID of 1008.3 Gy when it suddenly stopped responding and did not recover even after a power cycle. However, it must be noted that from 95 Gy onward, the device was running continuously. It was later observed that the dose after which the device is unable to start up anymore is often lower than the TID achievable in continuous operation.

For this reason, in campaign #2 the tested devices were power cycled every day (approximately every 100 Gy). Indeed, the TID before total failure was lower for all devices,

namely 615.3, 820.2 and 525.2 Gy. The mean lifetime of a BatMon sensor node can be thus estimated as (654 ± 151) Gy (sample standard deviation). However, as detailed below, parts of the functionality degrades or fails at a lower dose.

6.2.3 FGDOS characterization

One of the primary aims of the test was to validate if the FGDOS is performing as expected and the data is being processed correctly by the MCU. Based on the measured frequency over time, and the calibrated dose rate at DUT, sensitivity curves were constructed for several tested devices (figure 6.5).

The qualitative shape of the sensitivity curves agrees with previous findings in [18], but the spread in the initial sensitivity is somewhat higher than expected. The specified initial sensitivity of the sensor is 60 kHz/Gy [15], while the statistically derived sensitivity is (64.33 ± 6.78) kHz/Gy (ssd).

It worth noting, however, that uncertainties on the order of 10 % and more are not uncommon in dosimetry, in part because the measured quantity often spans several orders of magnitude. For example, the beam flux in the CHARM test facility is specified with a tolerance of 30 %.

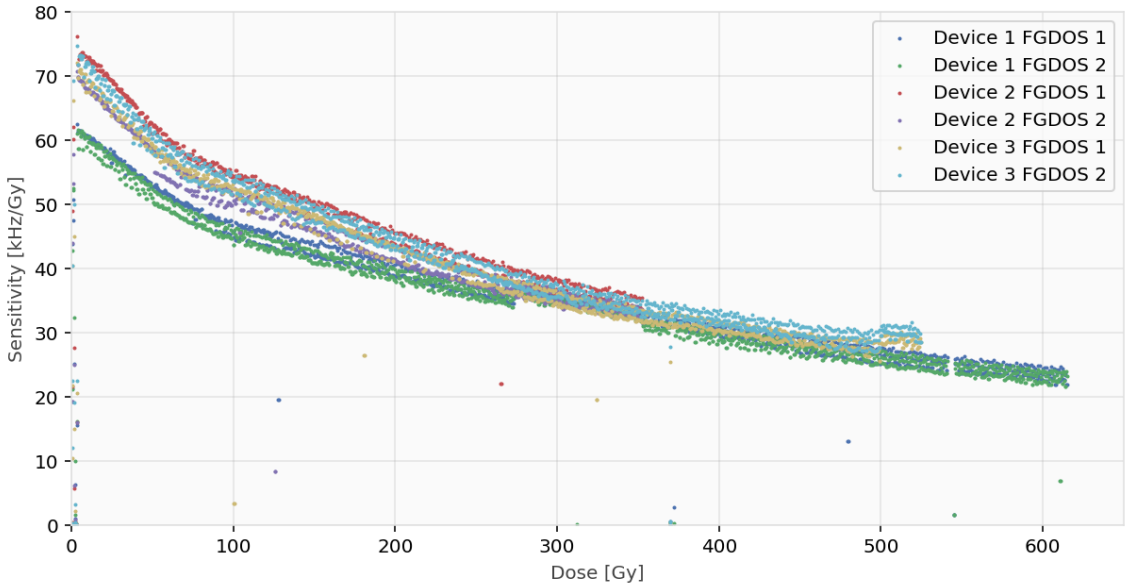


Figure 6.5: FGDOS sensitivity vs absorbed dose up to 600 Gy during campaign #2.

6.2.4 LoRa communication

It was shown in test [28] that the used LoRa module performs well even in a high flux of protons. To complement these findings, quantitative understanding of dose effects on the module was also sought.

A LoRa gateway was placed in a cable conduit above the door to the irradiation room at CC60 (see figure 6.1). The line of sight between the antenna of the gateway and the DUTs was obstructed by corner of the concrete wall, which explains why the absolute RSSI is low despite the short distance. Two variables were being monitored:

1. the Relative Signal Strength Indicator (RSSI) of every packet as evaluated by the receiving gateway
2. packet loss

Figure 6.6 is a plot of packet RSSI over time. Because of the high number of samples, each sample is plotted with a low opacity. Thus, the RSSI probability distribution at any given time can be judged from the shading pattern. From the results, it would seem that towards the end of the test, the RSSI dropped by about 2 dB on average. Due to the high sample variation and low RSSI resolution, it is not certain if this is due to degradation of the device, or simply an artifact of the test. In any case, the result is more than acceptable.

Occasionally, one or multiple consecutive packets were not delivered (figure 6.7). This is especially notable around the 100 Gy mark, when the measurement and transmission rate was temporarily increased. Because the gateway was placed in the heavily shielded irradiation room, the test setup presented almost ideal conditions, and the expected packet loss would be 0 %. Further tests will be needed to narrow down the issue between hardware, software and the gateway. The used gateway only logs packets that were received successfully; it is not possible to determine, for example, the percentage of CRC errors.

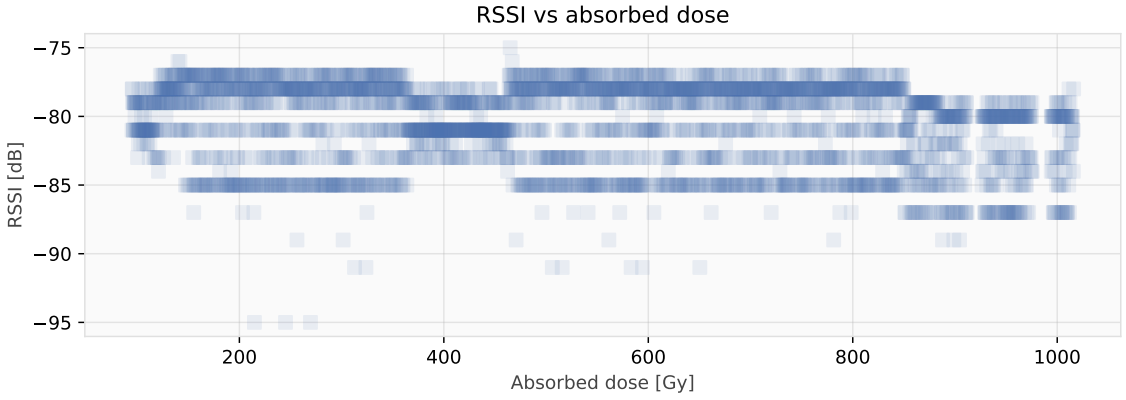


Figure 6.6: LoRa RSSI vs absorbed dose up to 1000 Gy during campaign #1.

6.2.5 Flash memory

When communication fails, it becomes the job of the flash memory to preserve the measured data. Every time a packet is composed for LoRa transmission, the same packet is saved into the flash memory. The written data is immediately read back and verified; if the writing was unsuccessful, it is retried up to 10 times in subsequent memory locations.

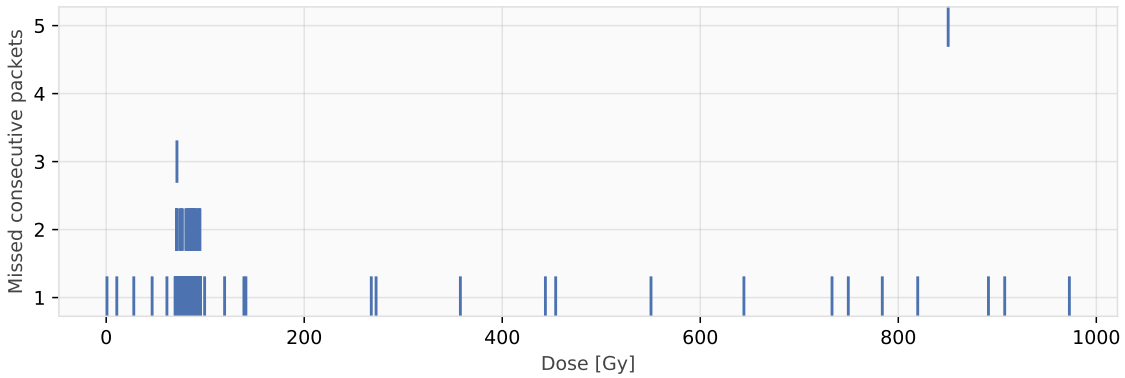


Figure 6.7: LoRa packet loss during campaign #1.

Figure 6.8 shows the number of write errors per packet; after 10 errors, further attempts are not made for that packet. As mentioned earlier, the devices were restarted several times during the campaign. On boot, the device attempts to detect the flash memory capacity; if this fails, no attempts are made to read or write any data. In the figure, this is indicated by a dashed vertical line.

On average, the dose before first failure was (428 ± 20) Gy.

While this test provides useful data, it is now understood to be insufficient; it will not detect a hypothetical failure mode whereby already written data was corrupted by radiation. To check this, the full contents of memory need to be read out and validated periodically (for example, every 50 or 100 Gy).

6.2.6 Summary

In the tests carried out, the device exceeded expectations by a wide margin. The number of tested devices was not sufficient to provide full confidence in the quantitative results, but important lessons have been learned (both regarding the test procedure and device failure modes) that will be useful for larger-scale qualification. It was found that the flash memory fails earlier than the rest of the system, and as a result, we recommend that a power gating transistor be added in the next hardware revision, to improve the lifetime of the memory.

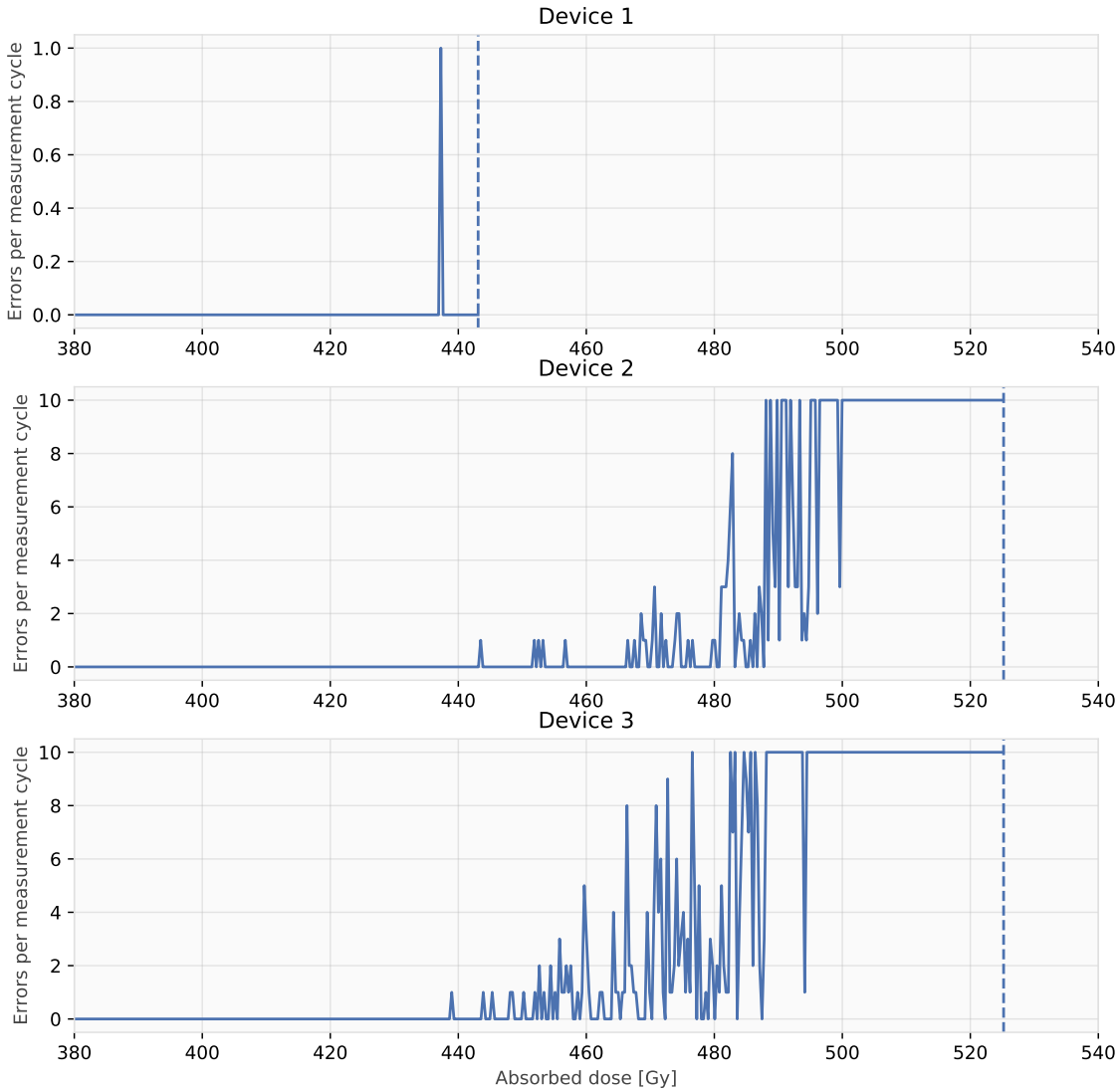


Figure 6.8: Flash write errors vs absorbed dose during campaign #2. The vertical dashed lines indicate restarts after which the flash memory was found unresponsive (more specifically, the *Read JEDEC ID* command did not return a valid value)

6.3 Test of LoRa reception in the LHC

In January 2020, two *LORIX One* LoRaWAN gateways were installed in RE18, a shielded cavern adjacent to the LHC tunnel (figure 6.9).[1] Instead of a traditional antenna, which suffers from a free-space path loss proportional to the squared distance (and if placed in the cavern, would be completely attenuated by the heavy shielding), a pair of *leaky feeder* cables (figure 6.10) was installed, each connected to one of the gateways, leading out into the tunnel and then continuing in opposite directions. Leaky feeders are commonly used in tunnel environments, where only short range is needed in directions perpendicular to the conduit, but a long segment without active electronics is desirable.

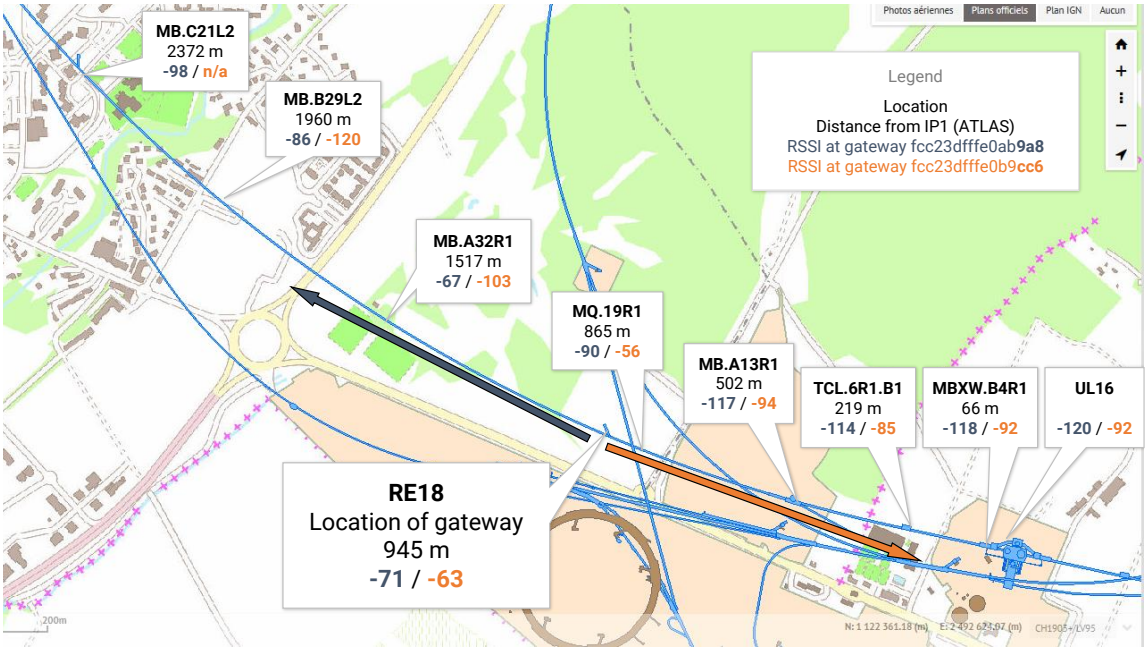


Figure 6.9: Layout of the tunnel around RE18 and the sampling locations. Colored arrows indicate the direction of the two leaky feeder cables. For each highlighted position, the magnet name, DCUM, and the two RSSI values are stated.

A BatMon sensor node was configured to transmit a packet every 60 seconds and was then carried from the entry at IP1 (far east in the map) westwards for about 2.4 kilometers, stopping regularly to evaluate stable RSSI at selected positions (names of nearby magnets are reported). The position is reported in terms of DCUM (cumulative westwards distance from IP1). The results are reported as RSSI of uplink packets seen by each of the gateways, in figures 6.11, 6.12 as a function of time and DCUM, respectively. From the plots, we can deduce that the gateway labeled `9A8` is connected to a feeder extending westwards (direction of increasing DCUM) and conversely for `CC6` eastwards from RE18.

Based on the observed attenuation slope between -75 (conservative) and -30 (optimistic) $\mathrm{dB} \cdot \mathrm{km}^{-1}$, peak RSSI at -60 dB close to gateway and a sensitivity of -110 dB for reliable

[1] Due to the accelerator having been shut down since December 2018, the background radiation in the tunnel was sufficiently low to disregard its effects on the test.

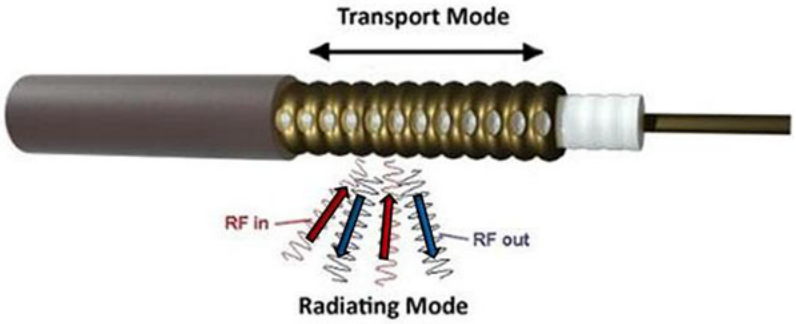


Figure 6.10: Leaky feeder cable. From [33].

reception, the maximum gateway spacing will be between 1.3 and 3.3 km, assuming feeders spreading in both directions from pairs of gateways. For the LHC circumference of 26.7 km, we estimate that between 17 and 42 gateways would be required to achieve full coverage.

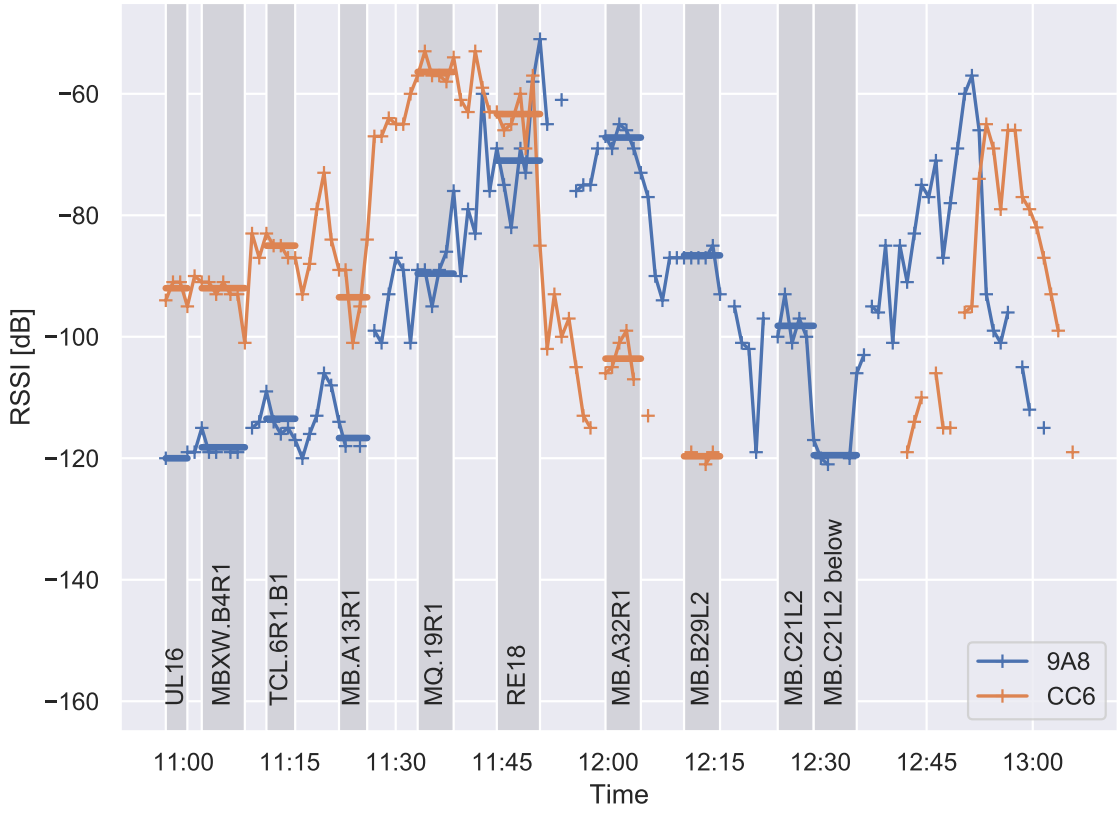


Figure 6.11: RSSI over time. Magnet names are noted for stationary intervals.

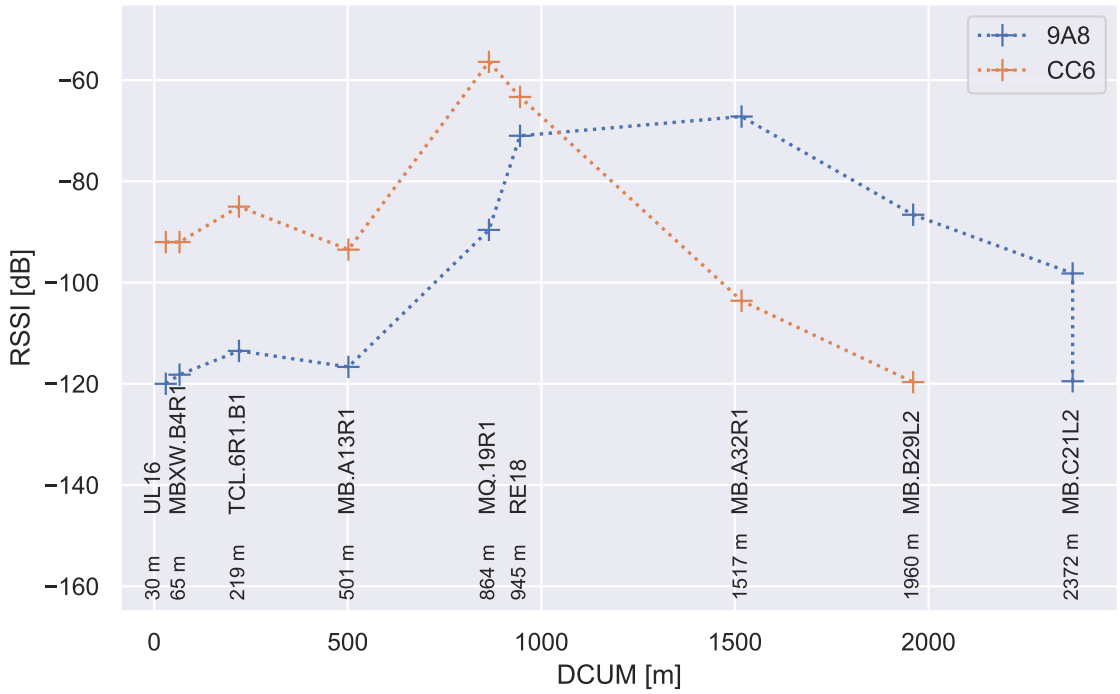


Figure 6.12: RSSI as a function of longitudinal position.

6.4 Power consumption test and battery lifetime estimation

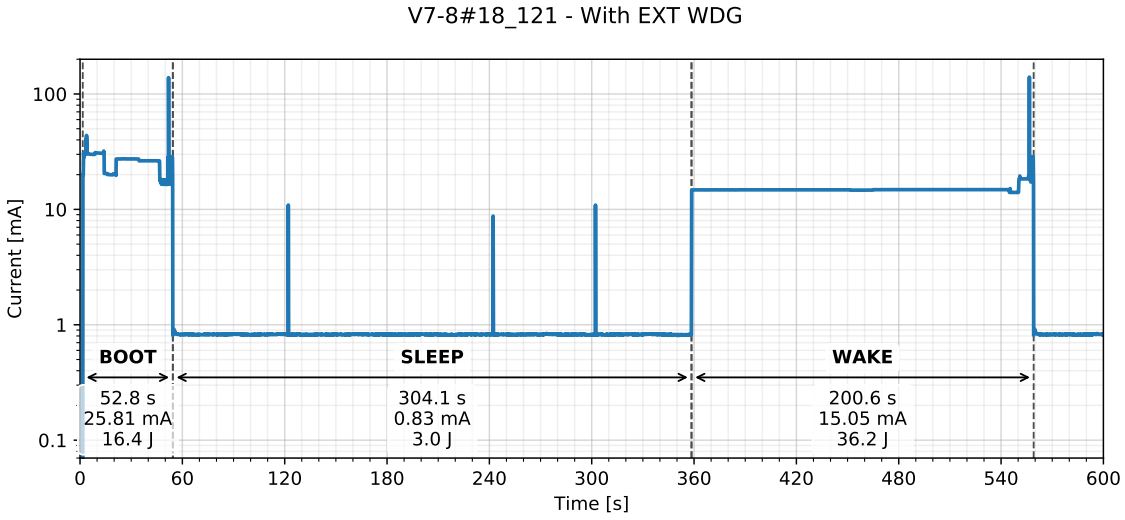


Figure 6.13: Current consumption test result with different operation modes highlighted. Compare to power state design in figure 5.4. The terms *standby* and *sleep* are used here interchangeably.

The power consumption was tested using a short measurement cycle; however, understanding of the device operation permits extrapolation of the results. From the results in figure 6.13, we can establish:

1. standby current $I_s = 0.83\,\mathrm{mA}$
2. active current $I_a = 15.06\,\mathrm{mA}$
3. active time per measurement cycle $t_a = 200.6\,\mathrm{s}$ (dominated by the time needed to scrub all memories)
4. charge consumed during initialization $Q_{boot} = I_{boot} \cdot t_{boot} = 1.36\,\mathrm{C}$

Then, for a given measurement period T_{meas}, the sleep time will be

$$t_s = T_{meas} - t_a. \tag{6.1}$$

We can evaluate average current consumption (post-initialization) as a weighted average

$$\bar{I} = \frac{t_a \cdot I_a + t_s \cdot I_s}{t_a + t_s}, \tag{6.2}$$

and the expected battery lifetime

$$\tau = \frac{n \cdot Q\left(\frac{\bar{I}}{n}\right) - Q_{boot}}{\bar{I}}, \tag{6.3}$$

for n cells of capacity Q connected in parallel. Per [34], the effective capacity of the used battery depends on current sourced from each cell, and we express Q as a function of the *average* current (which might not be strictly correct, but a more accurate physical model of the cell is not available). Note also, that as a result of this dependency, doubling the number of parallel cells will not exactly double the lifetime.

Table 6.2 shows the results for selected measurement periods, assuming $n = 2$ batteries in parallel. While the results are very encouraging, they will need to be validated also by long-term (not accelerated) testing.

Measurement period	$\bar{I}$ [mA]	Q [mAh]	Lifetime τ [days]	Lifetime τ [months]
5 minutes	10.31	7900	64	2.1
1 hour	1.62	8200	434	14.5
1 day	0.86	8450	791	26.4

Table 6.2: Predicted battery lifetime at 25 °C with 2 cells in parallel. Q corresponds to $\frac{\bar{I}}{2}$; uniform distribution of the current among cells is assumed.

Chapter 7

Conclusions

The objective of this work was to advance the BatMon to a stage where the system is complete enough to enable pilot deployment of nodes in selected areas, mainly by completing the software package, carrying out artificial tests and discovering potential issues. The starting point was a conceptual vision for the system, a sensor node hardware prototype and accompanying basic firmware. During the following months,

1. the remaining required features were implemented in the sensor node software, including sensor calibration and conditioning, more robust wireless communications, data storage in flash memory and API for sensor board drivers;
2. additional features were proposed and implemented: serial interface for configuration using the flash memory as storage, built-in self-test;
3. several tests were carried out, with the side effect of producing a considerable amount of re-usable utility software, in the form of Python scripts, JupyterLab notebooks and dynamic spreadsheets;
4. new accessory hardware was designed and produced: battery board, break-out board;
5. documentation of the software and certain processes was updated and extended;
6. a demonstration user interface was developed, which can serve as a basis for a user-facing application.

The test results demonstrate that the system is able to stand up to the requirements, sometimes exceeding them by considerable margin (428 Gy TID without degradation; 2 years of battery lifetime with daily measurement). Where problems and limitations were discovered, improvements to hardware and/or software were proposed. Undeniably, further tests will be needed, particularly to establish the reliability in SEU-producing radiation fields and to validate results extrapolated from accelerated tests.

Unfortunately, the worldwide health crisis in Spring 2020 and consequential lock-down disrupted the planned test campaigns and changed the focus of CERN when it began re-opening in May. Although this means that the work presents fewer test results than

originally intended, the additional room allowed deeper discussion of the context and relevant theory, which may make the work ultimately more appealing for readers without background in radiation effects and dosimetry.

Nevertheless, the goal of the work, to ensure the BatMon's readiness for pilot deployment, has been fulfilled.

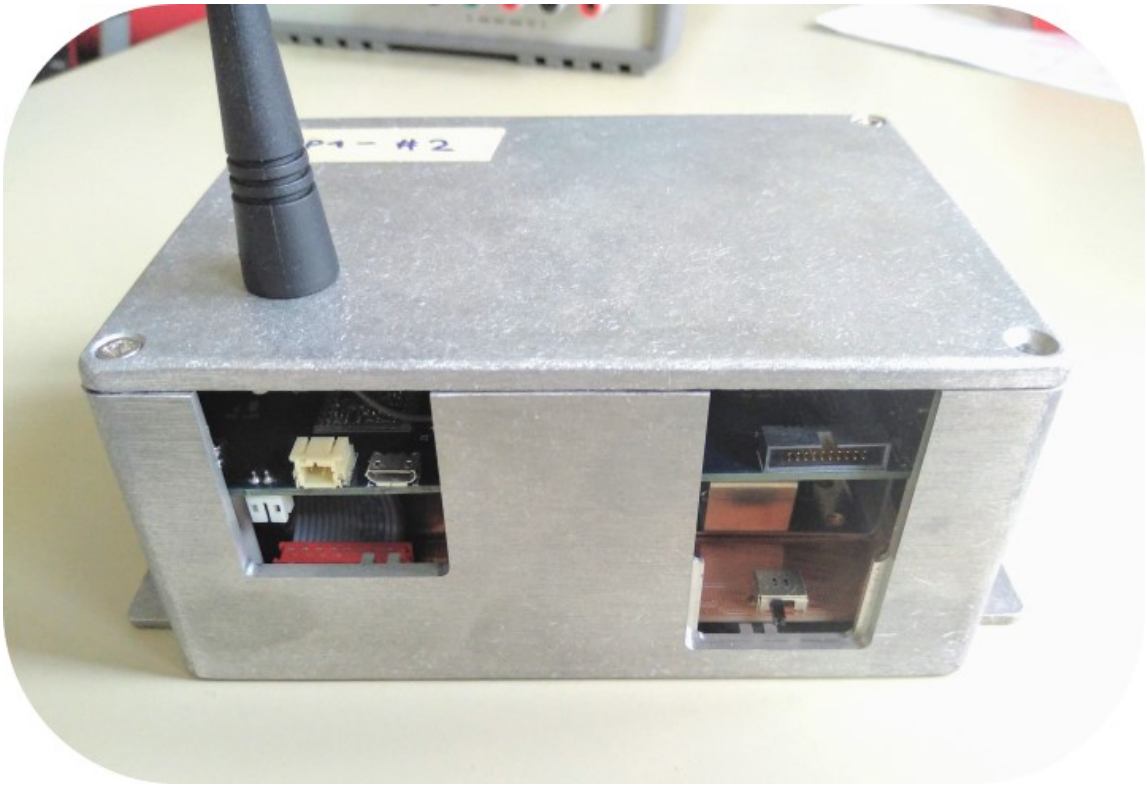


Figure 7.1: An assembled prototype of a BatMon sensor node.

Bibliography

[1] Rubén García Alía. "Radiation Fields in High Energy Accelerators and their impact on Single Event Effects. Champs ionisants dans un accélérateur à haute énergie et leur impact sur les Effets Singuliers". Presented 15 Dec 2014. Oct. 2014. URL: https://cds.cern.ch/record/2012360.

[2] G Spiezia et al. "A New RadMon Version for the LHC and its Injection Lines". In: *IEEE Trans. Nucl. Sci.* 61.6 (2014), 3424–3431. 8 p. DOI: 10.1109/TNS.2014.2365046. URL: https://cds.cern.ch/record/2011660.

[3] CERN TWiki. *Radiation*. https://twiki.cern.ch/twiki/bin/view/BGV/Radiation. [Online; accessed 03-August-2020]. 2016.

[4] Giovanni Spiezia et al. "The LHC Radiation Monitoring System - RadMon". In: *PoS* RD11 (2011). Ed. by Oscar Adriani et al., p. 024. DOI: 10.22323/1.143.0024. URL: https://inspirehep.net/literature/1206738.

[5] Open Hardware Repository. *WorldFIP*. https://www.ohwr.org/projects/cern-fip/wiki/WorldFIP. [Online; accessed 03-August-2020]. 2015.

[6] Matteo Brucoli. "Total ionizing dose monitoring for mixed field environments". Theses. Université Montpellier, Nov. 2018. URL: https://tel.archives-ouvertes.fr/tel-02155482.

[7] *Radiation Handbook for Electronics*. [Published online; accessed 03-August-2020]. Texas Instruments, 2019. URL: https:///www.ti.com/radbook.

[8] T. R. Oldham and F. B. McLean. "Total ionizing dose effects in MOS oxides and devices". In: *IEEE Transactions on Nuclear Science* 50.3 (2003), pp. 483–499.

[9] R. C. Baumann. "Radiation-induced soft errors in advanced semiconductor technologies". In: *IEEE Transactions on Device and Materials Reliability* 5.3 (2005), pp. 305–316.

[10] N. A. Dodds et al. "Selection of Well Contact Densities for Latchup-Immune Minimal-Area ICs". In: *IEEE Transactions on Nuclear Science* 57.6 (2010), pp. 3575–3581.

[11] M. Johnson et al. *Latch-Up. SCAA124*. [Online; accessed 09-August-2020]. Texas Instruments, Apr. 2015. URL: https://www.ti.com/lit/wp/scaa124/scaa124.pdf.

[12] P. Pavan et al. “Flash memory cells-an overview”. In: *Proceedings of the IEEE* 85.8 (1997), pp. 1248–1271.

[13] S. Gerardin and A. Paccagnella. “Present and Future Non-Volatile Memories for Space”. In: *IEEE Transactions on Nuclear Science* 57.6 (2010), pp. 3016–3039.

[14] M. Álvarez et al. “Total Ionizing Dose Characterization of a Prototype Floating Gate MOSFET Dosimeter for Space Applications”. In: *IEEE Transactions on Nuclear Science* 60.6 (2013), pp. 4281–4288.

[15] *Floating Gate Dosimeter. Technical data sheet*. FGD-03F. Rev. A0.7. IC Málaga. Oct. 2019.

[16] Matteo Brucoli et al. “Floating Gate Dosimeter Suitability for Accelerator-like Environments”. In: *IEEE Transactions on Nuclear Science* (Aug. 2017), pp. 1–1. URL: https://hal.archives-ouvertes.fr/hal-01956667.

[17] M. Brucoli. *A new Floating Gate Dosimeter for CERN applications*. R2E Annual Meeting. Dec. 2018.

[18] M Brucoli et al. “Investigation on Passive and Autonomous Mode Operation of Floating Gate Dosimeters”. In: *IEEE Trans. Nucl. Sci.* 66.7 (2019), 1620–1627. 8 p. DOI: 10.1109/TNS.2019.2895366. URL: https://cds.cern.ch/record/2689001.

[19] *LPWAN Networks*. [Online; accessed 06-August-2020]. CERN. URL: https://lpwan-service.docs.cern.ch/lpwan/lpwan_network/.

[20] *LoRaWAN 1.0.3 Specification*. LoRa Alliance. July 2018. URL: https://lora-alliance.org/sites/default/files/2018-07/lorawan1.0.3.pdf.

[21] K. Koenen. *Understanding the LoRaWAN® Architecture*. Sept. 2019. URL: https://tech-journal.semtech.com/understanding-the-lorawan-architecture.

[22] Christelle Caillouet, Martin Heusse, and Franck Rousseau. “Optimal SF Allocation in LoRaWAN Considering Physical Capture and Imperfect Orthogonality”. In: *GLOBECOM 2019 - IEEE Global Communications Conference*. Waikoloa, United States, Dec. 2019. URL: https://hal.inria.fr/hal-02267218.

[23] Andri Rahmadhani and Fernando Kuipers. “When LoRaWAN frames collide”. English. In: *WiNTECH 2018 - Proceedings of the 12th International Workshop on Wireless Network Testbeds, Experimental Evaluation and Characterization, Co-located with MobiCom 2018*. Ed. by Swarun Kumar and Andres Garcia-Saavedra. 12th ACM International Workshop on Wireless Network Testbeds, Experimental Evaluation and Characterization, WiNTECH 2018, co-located with ACM MobiCom 2018 ; Conference date: 02-11-2018. United States: Association for Computing Machinery (ACM), 2018, pp. 89–97. DOI: 10.1145/3267204.3267212.

[24] The Things Industries. *Duty Cycle*. https://www.thethingsnetwork.org/docs/lorawan/duty-cycle.html. [Online; accessed 06-August-2020].

[25] *LoRaWAN 1.1 Regional Parameters*. LoRa Alliance. Jan. 2018. URL: https://lora-alliance.org/sites/default/files/2018-04/lorawantm_regional_parameters_v1.1rb_-_final.pdf.

[26] G. Piscopo and G. Tsiligiannis. “Radiation Test Report. ATSAMD21G18”. [Unpublished]. Feb. 2019.

[27] G. Piscopo. “Radiation Test Report. IS25LP128-JBLE-TR”. [Published internally; CERN EDMS number 2015772]. Aug. 2018.

[28] R. Castellotti. “Radiation Test Report. Feather m0 LoRa”. [Published internally; CERN EDMS number 2360087]. Jan. 2018.

[29] R. Ferraro. “CHARM Radiation Test Report. TPL5110 System Timer for Power Gating”. [Published internally; CERN EDMS number 1863865]. Oct. 2017.

[30] SCPI Consortium. *Standard Commands for Programmable Instruments (SCPI)*. IVI Foundation. 1999.

[31] J.P. Wozniak and C. Roderick. “NXCALS - Architecture and Challenges of the Next CERN Accelerator Logging Service”. In: *Proceedings of ICALEPCS2019*. New York, NY, USA, Oct. 2019. URL: http://icalepcs2019.vrws.de/papers/wepha163.pdf.

[32] InfluxData Inc. *InfluxDB key concepts*. [Online; accessed 06-August-2020]. URL: https://docs.influxdata.com/influxdb/v1.8/concepts/key_concepts/.

[33] M.J. Martin. *Mining, Oil, & Gas: Leaky Feeder*. [Online; accessed 09-August-2020]. Oct. 2015. URL: https://vividcomm.com/2015/10/04/mining-oil-gas-leaky-feeder/.

[34] *Cylindrical cells*. TL-4920. Rev. C 01/06. Tadiran Batteries. URL: http://www.tadiranbat.com/assets/tl-4920.pdf.

Appendix A

List of acronyms

Acronym	Meaning
API	Application programming interface
CC60	CERN Cobalt-60
CRC	Cyclic redundancy check
DCUM	Distance cumulée (distance along an accelerator from a reference point)
DUT	Device under test
FGDOS	Floating-gate dosimeter
HTTP	Hypertext transfer protocol
LHC	Large Hadron Collider
LPWAN	Low-power wide area network
MCU	Microcontroller
MQTT	MQ Telemetry Transport, a publish-subscribe network protocol
PCB	Printed-circuit board
PSI	Paul Scherrer Institute
RSSI	Relative Signal Strength Indicator
SEE	Single-event effect
SEU	Single-event upset
SPI	Serial peripheral interface
SRAM	Static random-access memory
ssd	Sample standard deviation
TID	Total ionizing dose
UART	Universal asynchronous receiver/transmitter; also used to refer to an asynchronous serial communication interface consisting of Transmit and Receive signals.

Appendix B

Contents of the attached CD

Photos	Photographic documentation of hardware and test campaigns
Test_protocols	Cited test protocols that are not available publicly
thesis.pdf	Full text of the thesis