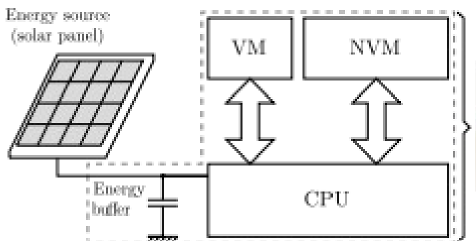
NOPAL: characterizing the energy aware approach for intermittent systems

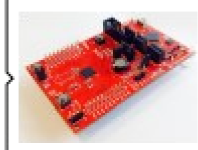
IETR-ASIC, Inria-PACAP, IRISA-GRANIT, LS2N-STR

NOPAL in a nutshell

Project facts:

- Follow-up action of project NOP (Safe and Efficient Intermittent Computing for a Batteryless IoT).
- One year, 70 K€, 12+6 man.month.





NOP explores the **energy-aware design** of autonomous batteryless IoT nodes. It has already produced promising results in terms of safety and efficiency. It is now time to challenge its assumptions.

Robustness of energy consumption models?

Energy-aware design requires to know the WCEC of programs.

Depending on the design level, we have explored several models :

- WCEC(task) = WCET(task) × Constant depending on the peripherals and subsystems used by the program.
- WCEC(insts) = **f**(instruction types), with various **f** : from linear function to ML models

Energy-aware design also requires to know the capacitance of the energy buffers.

In practice, these data may vary over time or from one device to another due, for example, to variability in the component manufacturing process, environmental conditions or ageing.

Questions

- What are the characteristics of this variability?
- What is its impact on the soundness of energy-aware design?
- Is it possible to derive safe (yet not too pessimistic) bounds that account for this variability?
- Or is it more efficient to integrate adaptation strategies in systems?

Full autonomy + embedded ML + long range radio?

Simple « RFID-like » systems do not benefit fully of an energy-aware design. More challenging systems are required to better showcase the benefits of the approach. Benchmarks found in the literature fall rather in the former category than in the latter one, so we propose to design a more challenging one.

Requirements

- End-to-end service, including sensing and communication.
- Non trivial long lasting computations, e.g., inference and classification.
- Modular design with optional parts, e.g., encryption, or anonymization.

Questions

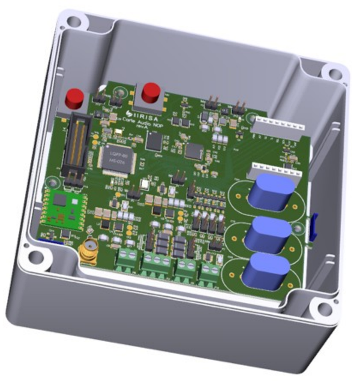
- How to minimize dependance on a specific hardware platform and/or software technology?
- Which metrics are relevant for assessing performance ? How to automate their extraction?
- Beside the « embedded ML » use case, should we also include other use cases, e.g. OTA update, or distributed computation?

Experimental platforms for NOPAL (from NOP)

BARD: batteryless IoT node built around off-the-shelf components, computes and sends acoustics indices in full autonomy.

NOP sensor node: batteryless IoT node integrating FPGA and external FRAM to execute ML workloads in full autonomy.

CNN toolchain: bridges the gap between high level description of DNN/CNN in Julia and embedded C code for batteryless IoT nodes.



Publications (from NOP)

[1] A. Bernabeu *et al.*, « *Cost-Optimal Timed Trace Synthesis for Scheduling of Intermittent Embedded Systems* ». In DEDS, vol 33, 2023.

[2] A. Bernabeu, « *Support d'exécution pour les systèmes intermittents* ». Thèse de doctorat, École Centrale Nantes, 2023.

[3] H. Reymond *et al.*, « *SCHEMATIC : Compile-time checkpoint placement and memory allocation for intermittent systems* ». In IEEE/ACM CGO, 2024.

[4] H. Reymond *et al.*, « *EarlyBird : Energy belongs to those who wake up early* ». In IEEE RTCSA, 2024. **Best paper award**

[5] H. Reymond, « *Modèle d'exécution conscient de l'énergie pour les systèmes intermittents* ». Thèse de doctorat, Université de Rennes, 2024.