INSTITUT NATIONAL
DES SCIENCES
APPLIQUÉES
LYON

Laboratoire CITI / INRIA

Research internship

Title: **Dynamic Allocation in Embedded Systems with Several Memory Types**

Location: CITI Lab, INSA-Lyon/INRIA

Keywords: Embedded Systems, Bare-Metal Programming, Memory Management

Contact: `guillaume.salagnac@insa-lyon.fr` — `http://perso.citi-lab.fr/gsalagnac`

Context

The general context for this work is the emergence of a new generation of System-on-Chip platforms based on non-volatile memory technologies. These technologies allow for designing very energy-efficient embedded systems, but they also require significant changes in terms of software programming [BCGL11]. In this work, we focus on the design and study of novel memory management techniques dedicated to such systems.

The emergence of non-volatile memory (e.g. PCM, MRAM, etc) promises to blur, and eventually remove, the traditional distinction between fast but volatile "working memory" (e.g. CMOS RAM) and persistent but slow "storage memory" (e.g. Flash). Moving away from the classical cache-based hierarchy, emerging architectures tend to adopt a heterogeneous organization where multiple banks of non-volatile memory are directly exposed to the processor. However each technology comes with specific limitations, like a high access latency or low write endurance. To ensure good execution performance, each piece of program data (i.e. code, variables, execution stack, etc) should be allocated to a bank with suitable characteristics [AH15].

Goal

In this project we focus on the issue of dynamic memory management, i.e. allocating, placing, and moving around, the various heap data structures used by the application program. In order to validate our ideas empirically, we developped a platform simulator which enables us to run embedded programs against various memory configurations (number of memory banks, bank capacity, read/write performance, etc). We are interested in the problem of object placement: at each `malloc()` request, the memory manager has to decide in which memory bank to allocate the new object. Our first results [DGM+18] focus on simple scenarios with just two banks: a "fast heap" and a "slow heap". Thus, the challenge consists in predicting which objects are going to be hot (i.e. intensely accessed during their lifetime) and which objects are going to be cold, and then in dispatching the allocation requests appropriately at runtime.

The goal of this internship is to extend this approach to memory configurations with multiple heaps. We want to go beyond the "fast vs slow" scenario, and study the same problem on more complex architectures. For instance, what if the memory banks cannot simply be ranked from "best performance" to "worst performance", but only form a partial order instead (e.g. one heap slower for writing, another slower for reading)? What information should be collected during the profiling stage? How to synthesize a practical placement strategy for multiple heap systems? One important issue is that of efficiency: the cost of deciding where to allocate an object should remain lower than the benefits of placing it correctly! Also, the placement strategy should be resilient to adverse circumstances like heap fragmentation causing allocations to fail. In the binary case, this situation leads to allocation fall-back: when an allocation in the better heap fails for some reason, the memory manager retries in the larger heap. But how to adapt this fall-back mechanism if we have more than two heaps?

Additional Info

Duration: 6 months (with legal compensation of approx. 600€/month)

Required Skills:

- Strong programming skills in both C (bare-metal apps) and C++ (SystemC/TLM simulator)
- Working knowledge of embedded architectures (CPU, pipeline, memory hierarchy, etc)

How to apply: send a CV to the contact address provided on the title page

References

[AH15] S. Alam and R. N. Horspool. “A Survey: Software-Managed On-Chip Memories”. In: *Computing and Informatics* 34.5 (2015), pp. 1168–1200.

[BCGL11] K. Bailey, L. Ceze, S. D. Gribble, and H. M. Levy. “Operating system implications of fast, cheap, non-volatile memory”. In: *HotOS 2011: 13th USENIX conference on Hot topics in Operating Systems*. 2011.

[DGM+18] T. Delizy, S. Gros, K. Marquet, M. Moy, T. Risset, and G. Salagnac. “Estimating the Impact of Architectural and Software Design Choices on Dynamic Allocation of Heterogeneous Memories”. In: *RSP’18, International Symposium on Rapid System Prototyping*. 2018.