Master Thesis
Bringing Modern C++ Concepts to RIOT OS

Embedded software development differs from other fields of software development in many regards. One of the most significant constraints is the limitation of available toolchains: Since devices of class 2 and below (using RFC 7228 terminology) have historically high diversity in terms of architectures used, available compilers options, language support, and toolchain maturity can greatly differ. In order to increase portability of code, embedded developers tend to be conservative when choosing programming languages, leading to the C language and specifically the legacy C99 standard often being widely used.

Modern programming concepts such as Scope-Bound Resource Management (SBRM) have the potential to avoid bugs especially in error handling, but are inherently incompatible with C. Similarly, the lack of strong type safety checks or a language specification free of undefined or implementation defined behavior increases the maintenance costs and reduce the code quality of C code compared to modern languages. For that reason, the increase in portability of code by choosing C as programming language comes with significant long-term costs during development, testing, and maintenance.

In recent years the ARM architecture however began to dominate the MCU market and most recently RISC-V was widely adopted by many MCU manufactures. Both platforms are well supported by popular and mature toolchains such as GCC and LLVM. With the plethora of MCU options for both ARM and RISC-V to choose from, choosing modern programming language no longer comes with the cost of significantly narrowing down the hardware options to run the code on. As a result, the benefits of other languages other than legacy C begin to outweigh the costs in reducing portability.

Among modern system programming languages suitable for Class 2 devices and below, the most obvious options are C++ and Rust. While Rust is of particular interest by eliminating any potential memory-safety issues and race conditions, C++ is easier to integrate with the existing C software and more mature. Hence, it is worthwhile to evaluate how the advantages of modern C++ APIs can be leveraged for existing C code basis and what overhead – if any – this adds compared to existing C interfaces.

Goals of the Thesis

The IoT Operating System RIOT [1] is a widely used large C code base with only a few native C++ APIs. It supports a wide range of architectures ranging from 8-bit AVR MCUs with only 2 KiB RAM over 16-bit MSP430 MCUs with up to 8 KiB RAM all the way up to 32-bit MIPS, Xtensa, RISC-V, and ARM Cortex M MCUs options ranging from 2 KiB to more than 512 KiB of RAM. For that reason, it is a suitable test case to evaluate the introduction of modern C++ APIs to large C code base in order to leverage the benefits of modern programming languages without having to rewrite all of the existing code.

Project type	Master Thesis
Duration	1 Term
Language(s)	English, German
Field	Computer Science

Contact	Marian Buschsieweke
E-Mail	buschsie@ovgu.de
Room	G29-314
Tel.	+49 391 67-52673

Task

- Provide or improve native C++ APIs for at least the following RIOT interface
  - The Thread API
  - The SPI API
  - The I^2C-API
  - The IRQ API (pairs of `irq_disable()` and `irq_restore()`)
  - Parts of the SOCK API
  - The Recursive Mutex API
  - The SAUL API
- Make use of language features and concepts of modern C++ where sensible, with at least one instance of each of the following
  - SBRM / RAI
  - template meta-programming
  - Object-oriented programming
- Evaluate the implemented interfaces regarding:
  - Does the use of modern programming languages and concepts introduce for the implemented interfaces overhead in terms of RAM, ROM, and CPU cycles? If so, by how much?
  - Briefly research the benefits of the use of modern C++ stated in literature and discuss to what degree the findings apply specifically to the context of embedded programming.
- Conclude by summarizing the trade-offs the use of modern C++ in RIOT are expected

References

[1] Emmanuel Baccelli and Oliver Hahm and Mesut Güneş and Matthias Wählisch and Thomas Schmidt. RIOT OS: Towards an OS for the Internet of Things. 32nd IEEE International Conference on Computer Communications (INFOCOM). 2013.

Project type	Master Thesis	Contact	Marian Buschsieweke
Duration	1 Term	E-Mail	buschsie@ovgu.de
Language(s)	English, German	Room	G29-314
Field	Computer Science	Tel.	+49 391 67-52673