Evaluation of Rust usage in space applications by developing BSP and RTOS targeting SAMV71

1 Abstract

The main objective of the activity was to evaluate the usage of Rust programming language in space applications, by providing an RTOS targeting ARM Cortex-M7 SAMV71 microcontroller, a BSP (Board Support Package) and a Demonstration Application.

RTOS is implemented in the form of an executor instead of a classic scheduler. The scope of this project doesn't include preemption. This executor runs tasklets, which are fine-grained units of computation, that execute a processing step in a finite amount of time.

Basic functionality required to create a working system is provided – tasklet priorities, recurring and time-based execution, as well as communication facilities such as queues and events. Additionally, execution time statistics are reported to facilitate scheduling analyses and discovery of real-time related issues.

Creating a real time operating system validates Rust features mentioned in the section above in practice, evaluates Rust viability in space applications and additionally checks compatibility with ECSS software development process.

The main focus of the BSP part of the project was to provide a minimal set of functionalities for peripherals required to create the RTOS and interact with the board as well as example sensors.

In the second part of the activity, a small demonstration application software was developed. This demonstration provided input to a Lessons Learned report, describing the encountered issues, potential problem and improvement areas, usage recommendations and proposed way forward.