



D5.8

Micro-ROS benchmarking and validation tools Release - Final

Grant agreement no.	780785
Project acronym	OFERA (micro-ROS)
Project full title	Open Framework for Embedded Robot Applications
Deliverable number	D5.8
Deliverable name	Micro-ROS benchmarking and validation tools Release - Final
Date	December 2021
Dissemination level	public
Workpackage and task	5.8
Author	Alexandre Malki (Łukasiewicz-PIAP), Tomasz Kołcon (Łukasiewicz-PIAP)
Contributors	Mateusz Maciaś (Łukasiewicz-PIAP)
Keywords	Microcontroller, NuttX, Benchmarking
Abstract	This deliverable summarizes works in Task 5.3: Benchmark tooling for application developers.



Contents

1	Summary	2
2	Overview	2
2.1	Introduction to Benchmarking	3
3	Source code	3
4	Hardware used	4
5	Configuration	4
6	Example	4
7	Other concepts and tools	4
8	Conclusions	4
	References	4

1 Summary

This deliverable summarizes the works in Task 5.3: Benchmark tooling for application developers. It is a description of final release of benchmarking tools. Previous developments described in D5.6 are continued here. The tools developed in the scope of task 5.3 are used internally as well as released as open source packages.

This document will first provide links to the released software and their respective documentation. In addition, A revised and general description of the created tool and it's architecture.

Term	Definition
ROS	Robot Operating System
MCB	Multi-connectors board
SWO	Single Wire Output
SWD	Serial Wire Debug
JTAG	Joint Test Action Group (also name of interface)
ITM	Instrumentation (or Instruction) Trace Macrocell
OS	Operating System
RTOS	Real Time Operating System
HW	HardWare
IP	Internet Protocol
TCP	Transmission Control Protocol
RAM	Random Access Memory
6LoWPAN	IPv6 over Low-Power Wireless Personal Area Networks.
I/O	Input/Output
ETM	Embedded Trace Macrocell
JSON	JavaScript Object Notation
AST	Abstract Syntax Tree
TFA	Trace Framework Abstraction
SB	Shadow Builder
KPI	Key Performance Indicators

2 Overview

This document provides description of the changes from D5.6 [1]. Additional works were performed in connection with the extension of the OFERA project by 1 year. It also links all required source repositories.

Below are links to repositories with a description of the concept, architecture and a repository with benchmarking tools:

- <https://micro.ros.org/docs/concepts/benchmarking/concept/> [2],
- <https://micro.ros.org/docs/tutorials/advanced/benchmarking/> [3],
- https://github.com/micro-ROS/benchmarking_shadow-builder/ [4].

2.1 Introduction to Benchmarking

As this deliverable is only an extension of D5.6, we will not duplicate the description of benchmarking tools. Full description is in the D5.6 [1]. The architecture has not changed since last year, only bugs have been fixed and support for the Zephyr OS has been added.

3 Source code

Comparing to the previous deliverable some links were changed. The current ones are listed below.

- Shadow Builder:
 - Git repository: https://github.com/micro-ROS/benchmarking_shadow-builder [5]
 - Path relative to the repository: `./shadow-builder`
- Trace Framework Abstraction Core:
 - Git repository: https://github.com/micro-ROS/benchmarking_shadow-builder [5]
 - Path relative to the repository: `./tfa_core`
- Trace Framework Abstraction Plugins:
 - Git repository: https://github.com/micro-ROS/benchmarking_shadow-builder [5]
 - Path relative to the repository: `./tfa-plugin`
- Tutorial on how to write a Trace Framework Abstraction Plugin:
 - Git repository: https://github.com/micro-ROS/benchmarking_shadow-builder/tree/master/tutorial/01_create_plugin_time [4]
- Patched NuttX:
 - Git repository: <https://github.com/micro-ROS/NuttX> [6].
- Patched NuttX applications:
 - Git repository: https://github.com/micro-ROS/nuttx_apps [7].
- Zephyr OS using Common Tracing Format:
 - Git repository: <https://github.com/zephyrproject-rtos/zephyr> [8].
 - Documentation: <https://docs.zephyrproject.org/1.14.1/guides/tracing/ctf.html> [9].
- Multi-Connectors Board:
 - Git repository: <https://github.com/micro-ROS/mcb> [10].
- Other benchmark tools over Serial Wire Debug:
 - Git repository: <https://github.com/microROS/benchmarking> [11].

4 Hardware used

Hardware setup is the same and described in the previous deliverable [1].

5 Configuration

Configuration is the same and described in the previous deliverable [1].

6 Example

Examples are well described in the previous deliverable [1]. Due to the fact that nothing has changed, we will not duplicate the content either.

7 Other concepts and tools

This year, the focus was on improvements to the existing solution in order to improve reliability. Zephyr OS support has been runned and the Common trace interface on default example. Indeed, the Zephyr always had the CTF facilities available in its tools.

The application can easily traced for benchmarked in the Zephyr OS. As a matter of fact, the Zephyr OS first implementation of Common Trace Format was re-used and ported within the NuttX RTOS.

More information is available from the Zephyr official documentation portal <https://docs.zephyrproject.org/1.14.1/guides/tracing/ctf.html> [9].

The results from the trace can be interpreted by the same tool that was used for the benchmarking of NuttX (i.e. babeltrace). The use of the Python interface allows to easily create different types of benchmarks depending on the trace output produced by an instrumented code that is complying with the Common Trace Format standard.

8 Conclusions

The tools described in this paper have been used successfully to conduct benchmarks. Thanks to them, extremely useful comparative results of the Dashing and Galactic micro-ROS versions were obtained [12]. By making the tools open-source, we hope they will be further developed by the community. The task of creating benchmark tools under the OFERA project has been completed.

References

[1] Łukasiewicz-PIAP, 'Micro-ROS benchmarking and validation tools Release - initial'. [Online]. Available: http://www.ofera.eu/storage/deliverables/M36/OFERA_57_D56_Micro-ROS_benchmarking_and_validation_tools_Release_-_Initial.pdf

- [2] A. M. OFERA, 'Benchmarking concepts'. [Online]. Available: <https://micro-ros.github.io/docs/concepts/benchmarking/>
- [3] A. M. OFERA, 'Benchmarking with the Shadow Builder'. [Online]. Available: <https://micro.ros.org/docs/tutorials/advanced/benchmarking/>
- [4] Łukasiewicz-PIAP, 'Shadow Builder Tutorial'. [Online]. Available: https://github.com/micro-ROS/benchmarking_shadow-builder/tree/master/tutorial/01_create_plugin_time
- [5] Łukasiewicz-PIAP, 'Shadow Builder'. [Online]. Available: https://github.com/micro-ROS/benchmarking_shadow-builder
- [6] Łukasiewicz-PIAP, 'Nuttx Instrumented'. [Online]. Available: <https://github.com/micro-ROS/Nuttx>
- [7] Łukasiewicz-PIAP, 'NuutX application Instrumented'. [Online]. Available: https://github.com/micro-ROS/nuttx_apps
- [8] Zephyr-Community, 'Zephyr repository'. [Online]. Available: <https://github.com/zephyrproject-rtos/zephyr>
- [9] Zephyr-Community, 'Zephyr Common Trace Format documentation'. [Online]. Available: <https://docs.zephyrproject.org/1.14.1/guides/tracing/ctf.html>
- [10] Łukasiewicz-PIAP, 'MCB'. [Online]. Available: <https://github.com/micro-ROS/mcb>
- [11] Łukasiewicz-PIAP, 'Micro-ROS benchmarking tools'. [Online]. Available: <https://github.com/micro-ROS/benchmarking>
- [12] Łukasiewicz-PIAP, 'Micro-ROS benchmarks - Extension'. [Online]. Available: http://www.ofera.eu/storage/deliverables/M48/OFERA_93_D5.7_Micro-ROS_benchmarks_-_Extension.pdf