

# THE LIBMCS

A STANDARD COMPLIANT & PRE-QUALIFIED  
MATHEMATICAL LIBRARY

---

Andoni Arregui\*, Fabian Schriever\*, Andreas Jung\*\*

2022-12-05

\* GTD GmbH, \*\* European Space Agency (ESA)



# TABLE OF CONTENTS

1. Motivation

2. Goal

3. Our Work

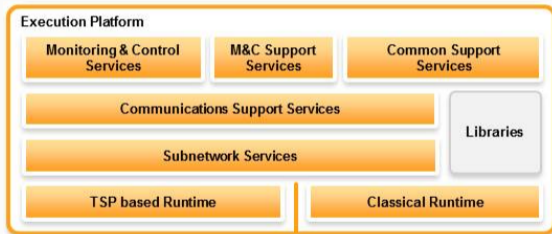
4. Further Steps

## **MOTIVATION**

---

## What does Mission Specific Space On-Board Software Need?

To be able to write a mission specific on-board application software we expect to be able to **rely** on a couple of **building blocks** or lower layers.

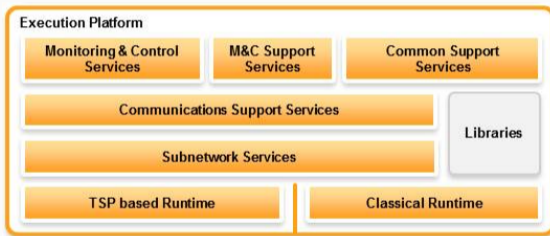


**Figure 1:** The SAVOIR On-Board Software Reference Architecture

# MINIMAL BUILDING-BLOCKS FOR ON-BOARD SOFTWARE

If we are not intending to write a bare-metal software we will at least need:

- An Operating System (e.g., RTEMS)
- A Standard Library (e.g., a standard C library)



## Else what?

We are almost not going to be able to do anything at all in our application software.

## What does Space On-Board Software Need?

- Standard compliant and
- ECSS Qualified Building blocks



## What do we lack?

But while we have pre-qualified Operating Systems such as EDISOFT RTEMS 4.8 and RTEMS SMP, we do not have a **pre-qualified Standard Library**.

## Bugs in Qualified Mathematical Libraries

Even DO-178 certifiable libraries present well identifiable bugs when it comes to mathematical computing.

Some libraries:

- Compute up to 38% of inexact results for  $\sin(x)$  in the range  $[0, \pi]$
- Compute up to 38% of inexact results for  $\cos(x)$  in the range  $[0, \pi/2]$
- Some libraries compute up to 50% of inexact results for  $\tan(x)$  in the range  $[0, \pi/2]$

While the LibmCS only produces around 3% of inexact results for these cases and all of them with a far better bounded error.

**GOAL**

---



## The LibmCS

To produce a **standard compliant library** for **critical systems** of the elementary **mathematical functions, qualified** to ECSS.

It is a POSIX (IEEE Std 1003.1-2017) compliant `libm` that:

- complies to IEEE-754 2019 (ISO 60559 – IEEE 754-2019)
- complies to ISO C18 (ISO/IEC 9899)
- complies to MISRA-C 2012
- is qualified to ECSS E-ST-40 and Q-ST-80 Category B

## The LibmCS

To produce a **standard compliant library** for **critical systems** of the elementary **mathematical functions, qualified** to ECSS.

In addition, its software architecture and algorithms shall be especially **designed** with **critical** and **embedded** systems in mind (GNC/AOCS systems and scientific payloads) .

And it should provide:

- **Good accuracy**, without the need to be fully accurate.
- Numerical **reproducibility** on different processor architectures.
- **Good WCET**, reasonable size, and complexity.

## OUR WORK

---

## Contractual Context

The work has been carried out under ESA Contract No. 4000130278/20/NL/AS since 2019.

- ESA aimed at a **properly qualified libm** that could easily be **reused** in all future missions as a building block, as projects qualified their mathematical libraries again and again.
- All the work has been carried out with great **support of the ESA** Technical Officer **Andreas Jung**.
- Precursor projects (MLFS) with ESA had been carried out by GTD since 2016.

## Newlib libm

We forked and re-engineered the Newlib libm 4.0.0

- The Newlib libm has a lot of **heritage** within the European space industry
- But:
  - the documentation lacks much information about the algorithms,
  - the coding style is heterogeneous, old, and not following any concrete standard, and
  - the library is very coupled to Newlib, which makes it difficult to understand what parts of the source-code are being compiled in.

### MLFS

We based on the knowledge gathered during the pre-qualification of the precursor library, the MLFS.

- The MLFS libm already provided an ECSS pre-qualified mathematical library.
- But:
  - it only contains a subset of the mathematical functions required by the standards,
  - its maintenance and extension is difficult, as its code-base heavily relies on the Newlib libm, and
  - its integration with other COTS software components can be challenging if these expect a standard compliant libm.

## LibmCS

The LibmCS is a completely re-engineered software.

- The complete source code has been **refactored** to allow:
  - the addition of improvements and alternative implementations
  - better long term maintenance
- **Static source code analysis**<sup>1</sup> has been carried out to:
  - avoid run-time errors
  - to achieve MISRA-C compliance
- The complete source code documentation has been rewritten and extended.
- We have been **contributing** our findings **back** to the **Newlib** community.

---

<sup>1</sup>Synopsys Coverity and Gimpel Software PC-lint were used

### LibmCS

The LibmCS is a standalone compilable mathematical library

- The LibmCS `libm` includes **all standard** required mathematical **functions** in single and double precision:
  - trigonometric, exponential, logarithmic, hyperbolic, remainder functions, &c.
  - totalling 75 functions per precision plus the complex number functions.
- **Corrections** carried out:
  - **Non-standard** compliant **behaviors** regarding **NaNs** (both sNaN and qNaN) and floating-point **exceptions** have been corrected (e.g., `pow`).
  - Other **numerical** non-standard behaviors have been corrected (e.g., `fmax`).
- The library is **compatible** with many **GCC** versions since 4.2.1 and also with recent **Clang** compilers.
- Is fully compatible with operating systems such as **RTEMS** and **GNU/Linux**.



## LibmCS

The LibmCS can be integrated in any environment where a standard C library can be integrated.

Additionally the LibmCS can be integrated into:

- MATLAB/Simulink developments for MIL, SIL, and PIL simulations through wrappers we provide.
- Ada based development environments via the provided bindings.

- Accuracy has been validated against the arbitrary precision library MPFR
- Many accuracy improvements have been made:
  - By fixing the algorithms ourselves
  - By porting better algorithms or fixes from FreeBSD
- Many other fixes have been ported from the precursor project MLFS.
- Identical accuracy results are obtained on x86-64 and LEON2.

Some examples:

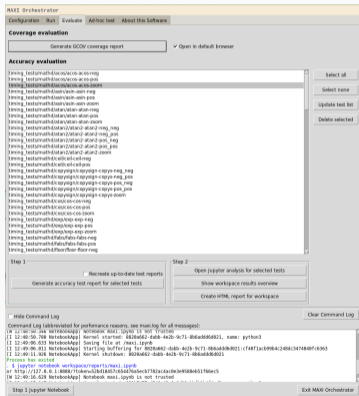
|                 | <b>Max. ULP<sup>a</sup> error</b> |               |
|-----------------|-----------------------------------|---------------|
| <b>function</b> | <b>Newlib<sup>b</sup></b>         | <b>LibmCS</b> |
| pow             | 636                               | 0.89          |
| powf            | 169                               | 1.00          |
| log10           | 2.08                              | 0.79          |
| log2            | 2.06                              | 0.90          |
| erfcf           | 63.9                              | 3.17          |

<sup>a</sup>Unit in the Last Place

<sup>b</sup>Results obtained by Paul Zimmermann  
(INRIA)

## MAXI

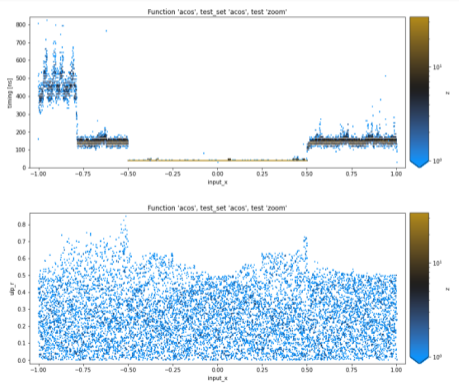
Is the test-suite application accompanying the LibmCS.



- Python app running in GNU/Linux
- Test specification via YAML files
- Runs over 4.000 unit tests on the target platform
- Runs over 1.000 single value validation tests including all special value cases
- Runs ca. 250 Million accuracy tests on the target platform

## MAXI

Is the test-suite application accompanying the LibmCS.



- Generates reports for:
  - unit tests results
  - unit tests structural coverage
  - validation tests results
  - accuracy results
  - function execution time results
- Enables *ad-hoc* numerical testing of functions

## Reports Data

Reports data is generated in HDF5 and in HTML for convenience.

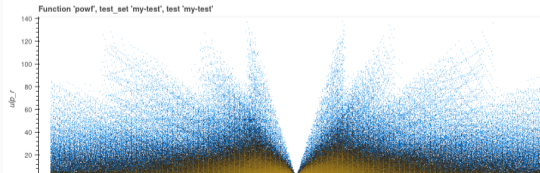
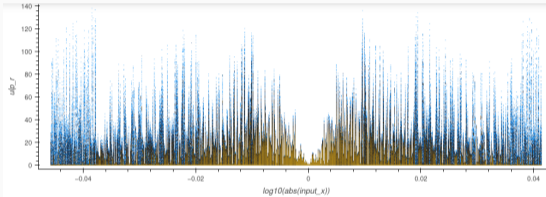
| Procedure | Tests   | Correctly Rounded | Rounding Errors | Incorrect Calculation | Max(ULP Error)     | Max(Relative Error)    | Count >= |
|-----------|---------|-------------------|-----------------|-----------------------|--------------------|------------------------|----------|
| acoe      | 1985730 | 1851107           | 0               | 134623                | 0.8765321072663197 | 2.2204453264077553e-16 | 0        |
| acoeff    | 1975390 | 1838314           | 0               | 137076                | 0.878975355767335  | 9.908110754831793e-08  | 0        |
| acosh     | 976099  | 787207            | 0               | 188892                | 1.2937213013690982 | 2.220445507254318e-16  | 0        |
| acoshf    | 976095  | 751459            | 0               | 224636                | 1.8315634630332807 | 1.2216484192779917e-07 | 3        |
| asin      | 1913882 | 1852999           | 0               | 60883                 | 0.8888279292758835 | 2.2204632321733271e-16 | 0        |
| asinf     | 1921896 | 1859562           | 0               | 62334                 | 0.8911380174268004 | 9.973882508635057e-08  | 0        |
| asinh     | 1051256 | 951971            | 0               | 99285                 | 1.4867979808320595 | 2.2204307213289743e-16 | 0        |
| asinhf    | 1918523 | 1689867           | 0               | 228656                | 1.358693636733727  | 1.258961751501552e-07  | 2        |
| atan      | 1005133 | 1004927           | 0               | 206                   | 0.7131745117735095 | 2.181909297945967e-16  | 0        |
| atan2     | 4657325 | 3944070           | 0               | 713255                | 1.4879911083121078 | 2.220440511464833e-16  | 0        |
| atan2f    | 5077929 | 4410483           | 0               | 667446                | 1.4118383101855179 | 1.0                    | 71402    |
| atanf     | 958193  | 958845            | 0               | 1348                  | 0.7624165636971508 | 7.456462900675268e-08  | 0        |
| atanh     | 1966688 | 1954258           | 0               | 12430                 | 1.5369668785870692 | 2.474323864409681e-16  | 3        |
| atanhf    | 1829726 | 1789598           | 0               | 40128                 | 1.5219120488577629 | 1.182973287239012e-07  | 0        |
| cbct      | 1071256 | 982542            | 0               | 88714                 | 0.6667553444889124 | 2.22043989287517e-16   | 0        |

The test-suite reports:

- Show the **pass/fail** status of all unit and validation tests
- Provide **structural coverage** data of the test runs on target (based on gcov)
- Present the maximum **ULP** and **relative error** measures
- Enable further assessment via generated Jupyter Notebooks

## Reports Data

Reports data is generated in HDF5 and in HTML for convenience.



The test-suite reports:

- Show the **pass/fail** status of all unit and validation tests
- Provide **structural coverage** data of the test runs on target (based on gcov)
- Present the maximum **ULP** and **relative error** measures
- Enable further assessment via generated Jupyter Notebooks

## ECSS Category B

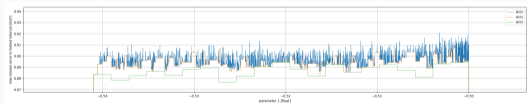
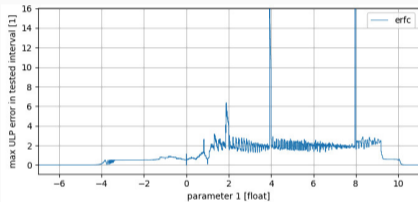
The LibmCS has been pre-qualified as an ECSS Category B software

- **Full ECSS** software engineering and product assurance **documentation** required by E-ST-40 and Q-ST-80 has been produced.
- **Full compatibility** with (but not only) **SPARC V8** processors.
- The pre-qualification evidences have been produced on:
  - LEON2 (AT697-FT SPARC V8 processor)
  - LEON4 (N2X SPARC V8 processor)
  - x86-64
- **Full structural coverage** including **MC/DC** has been achieved.
- **GCC** 4.2.1, 4.9.3, 10.3.1, and 12.2.0 have been used for the qualification tests.
- **RTEMS** 4.8, 4.11, 6 SMP and GNU/Linux have been used as operating systems.

## Independent Software Verification and Validation (ISVV)

The LibmCS **ISVV**, required for a category B software, has been carried out by ESA

- Complementary **static source code analysis** has been done (Synopsys Coverity)
- Complementary accuracy checks have been carried out:
  - Executing **extensive accuracy tests** in parallel on multicore x86-64 processors.
  - With a complementary approach to the qualification tests by searching for the maximum error.





## LibmCS is free for everyone

The **LibmCS** has a BSD-like license and is **free** to download and **use** within free and proprietary software.

- Full source code **publicly available** on **GitLab**: <https://gitlab.com/gtd-gmbh/libmcs>
- The Software User Manual and Software Design Document are also hosted on GitLab
- Full ECSS pre-qualification **documentation** available:
  - Freely for ESA member states
  - else available via GTD GmbH
- Full delta-**Qualification Kit**, including the **Test-Suite**, available:
  - **Freely** for **ESA member states**
  - else **available** via **GTD GmbH**

## Project specific delta-qualifications

The LibmCS enables project specific **delta-qualifications** with a **minimized effort**.

A project using the LibmCS needs to follow the provided **Qualification Guideline** to:

- Execute the unit tests on target
- Execute the validation tests on target
- Execute the timing tests on target
- Assess hardware-software integration aspects
- Fill the provided **Qualification Template**

## **FURTHER STEPS**

---

### ECSS Category A

- It has been qualified to Category B with the highest requirements
- Optional requirements such as full MC/DC have been already completed
- The LibmCS only lacks object to source code traceability to achieve Category A level.
- This upgrade will be addressed in an upcoming study.

### DO-178C

- A preliminary assessment has been carried out to define the gaps to a DO-178C certifiability.
- Aspects such as tool qualification need to be covered.