MATHSOFT / generation 2 — User manual

Library: `src/lib2/mathsoft/2/`
Edition: 2026-11-18, source/document review (documentation only)
Qualification: Latest observed generation but NOT PROMOTED. Candidate-only source split; `fn/` and engine composition native parity and independent runtime acceptance are not established.

What this member does

Copy-only, source-equivalent restructuring of Math 1.1 into 17 function includes, shared engines and intentionally overlaid scratch declarations.

Public interfaces

Entry point or group	Inputs, outputs and behavior
`M11_ADD / M11_SUB / M11_MUL`	Binary nine-byte B9 operations using `M_IN`, `M_A_INB`, `M_OUT`; status in `$0`.
`M11_DIV / M11_SQRT / M11_EXP`	Division, square root, exponential; donor-specific B9 input/output buffers.
`M11_LGT / M11_LN`	Base-10 / natural logarithm from shared log engine.
`M11_SIN / M11_COS / M11_TAN`	Trigonometry from shared trig engine.
`M11_SINH / M11_COSH / M11_TANH`	Hyperbolic functions from shared engine.
`M11_ASINH / M11_ACOSH / M11_ATANH`	Inverse hyperbolic functions from shared engine.

How to integrate

For a **source-equivalent validation composition**, use `FUNCTIONS.md` include order. Example SIN: `scratch/trig_scratch.inc`, `engine/trig_engine.inc`, `fn/sin.inc`, each exactly once. Donor r019 source and an accepted ROM remain control witnesses; this split is not a drop-in promoted upgrade.

Composition sketch (not an independently assembled client):

```
#INCLUDE (lib2/mathsoft/2/scratch/trig_scratch.inc)
#INCLUDE (lib2/mathsoft/2/engine/trig_engine.inc)
#INCLUDE (lib2/mathsoft/2/fn/sin.inc)
        ; Caller provides M_IN, M_OUT, UA14 and valid stack
        CAL     M11_SIN
```

Environment, dependencies and memory

Nine-byte B9 buffers `M_IN`, `M_A_INB`, `M_OUT` supplied by caller; UA14, `$SX=0`, `$SY=1`, valid stack. Shared scratch addresses intentionally overlap both math engines and B9 multiplication. Do not nest calls. No universal pointer ABI or complete register preservation established.

Limits and evidence

Although the donor/r019 and v2 layouts have historical matching ROM digest, the later v3 `fn/` layout was not natively built in its source record. Before use, qualify actual include closure and ROM hash in the owning client.

This documentation does not introduce a new build or runtime PASS. Source equivalence, native assembler parity, emulator execution and physical-machine evidence remain distinct.

Actual source and further reference

- This exact member folder in Google Drive
- Source modules: `fn/*.inc (17 public wrappers)`, `engine/*.inc (shared numerical engines)`, `scratch/*.inc (shared UA14 scratch)`, `add.inc through invhyp_core.inc (donor-exact full owners)`.
- Further contracts/source notes: API.md, FUNCTIONS.md, RAM.md, README.md. These may include dated donor paths or older acceptance statements.
- For a new consumer, pin exact source files and hashes, verify full include closure, inspect register/bank/resource conflicts and test the resulting application on each claimed target.

Praeteritum ad futurum servamus.