INPUT / generation 2 — User manual

Library: `src/lib2/input/2/`
Edition: 2026-11-18, source/document review (documentation only)
Qualification: Wave 8 source/native parity accepted only; `INP_EDIT_R002` ABI, FK mask and caller RAM ownership remain open. Not drop-in for input/10e.

What this member does

Experimental editor generation with explicit function-key exit mask, retaining the bounded NUL-text editing model.

Public interfaces

Entry point or group	Inputs, outputs and behavior
`INP_EDIT_R002`	Same core descriptor `$10..$19` as input/10e, plus `$26` FK mask `0..0Fh`; `$0` exit category, `$8` terminating key, `$9` final length. Exact function-key policy requires separate consumer qualification.

How to integrate

Include `input.inc` once and the required keyboard/cursor providers. Prepare a complete editor descriptor including `$26`; `CAL INP_EDIT_R002`; interpret exit category and final text length. Use only in a client that explicitly tests FK-mask behavior.

Composition sketch (not an independently assembled client):

```
#INCLUDE (lib2/input/2/input.inc)
; Supply the dependencies, RAM, bank context and symbols described above.
        CAL     INP_EDIT_R002
```

Environment, dependencies and memory

UA14; private `INP_FKMASK` at `27EDh` in addition to Input editor scratch. Caller must ensure all memory, keyboard, cursor, stack and modes are compatible. Non-reentrant; broad clobbers.

Limits and evidence

Experimental interface; do not silently upgrade an input/10e consumer or mistake native binary parity for correctness of FK exits.

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: `input.inc`.
- Further contracts/source notes: ../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.