U32 / generation 1 — User manual

Library: `src/lib2/u32/1/`
Edition: 2026-11-18, source/document review (documentation only)
Qualification: CLARITAS Wave 6 curated source/native parity accepted; donor Lab/Chapter application witnesses are separate from general library-call verification.

What this member does

Provide reusable ROM-independent unsigned 32-bit arithmetic, checked arithmetic and decimal conversion primitives.

Public interfaces

Entry point or group	Inputs, outputs and behavior
`UADD32 / USUB32 / UMUL32`	`$15..$18` first u32, `$5..$8` second; wraps modulo 2^32; result `$15..$18`.
`UDIV32NZ / UDIV3216NZ`	Nonzero divisor in `$5..$8` or `$5/$6`; quotient `$15..$18`, remainder `$10..$13`.
`UMOD32NZ / UMOD3216NZ`	Nonzero divisor; remainder returned in `$15..$18`.
`UADD32CHK / USUB32CHK / UMUL32CHK`	Checked operations returning `$4=0` valid, 1 range; overflow behavior depends on method.
`UDECAPP`	Append one numeric digit `$5` to value `$15..$18`; `$4=1` on overflow without modifying input.
`UPARSE32DEC`	IX ASCII, `$12` length; `$4=0` success, 1 overflow, 2 nondecimal, 3 empty; output `$15..$18`.
`UFMT32DEC`	Value `$15..$18`, IX >=11-byte destination; emits NUL-decimal string and digit count `$14`.

How to integrate

Include only needed `.inc` implementation(s) plus their documented dependencies. Set register operands and `CAL` the exact public label. For checked calls consume result only if `$4=0`; for divisor routines guard against zero. For decimal formatter supply at least 11 writable bytes.

Composition sketch (not an independently assembled client):

```
        ; $15..$18 and $5..$8 loaded as unsigned 32-bit values
        CAL     UADD32
        ; $15..$18 = modulo-2^32 sum
```

Environment, dependencies and memory

HD61700 register groups little-endian low word first. For division divisor must be nonzero. For decimal conversion valid IX bank, `$SX/$SY` and stack; exact preconditions vary by include header. No mandatory universal RAM pool for core operations.

Limits and evidence

`UMUL32` destroys multiplier `$5..$8`; ordinary add/sub/mul wrap. `UADD32CHK` has a wrapped intermediate on overflow and must be discarded. `UPARSE32DEC` leaves partial output unspecified on failure. See individual source headers.

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: `add_u32_u32.inc`, `sub_u32_u32.inc`, `mul_u32_u32.inc`, `divmod_u32_u32.inc`, `divmod_u32_u16.inc`, `mod_u32_u32.inc`, `mod_u32_u16.inc`, `checked_add_sub_u32.inc`, `checked_mul_u32.inc`, `parse_u32_dec.inc`, `format_u32_dec.inc`, `dec_append_u32.inc`.
- Further contracts/source notes: Each selected .inc source header. 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.