DRACO7RT PROTOTYPE · UPDATED 23 OCTOBER 2026

DRACO7RT Full Mission Test

Launch · Deployment · Autonomous Momentum Dump · Station-Keeping · GPS Spoof Response · IMU Fusion

cFS Draco 7.0 on RTEMS 6 / draco-s706 (MPC8548, QEMU), booted through the real u-boot/TFTP chain, fed by NASA 42 dynamics, GPS and IMU samples, commanded from COSMOS, verified by closed evidence runs in which every reported number is checked against a hashed file.

Evidence basis: run run_20261023_201111 · it covers all 75 requirements that existed on 23 October; requirements.csv now holds 83, all traced to code (TRACE.md), the 8 added since (5 CFDP, 3 table rules) are verified by the CFDP run or pending · results slides regenerate with gen_results.py

ORIENTATION

The words in this deck, for a first-time reader

The vehicle side

- **cFS:** NASA's core Flight System, the flight software framework; **apps** are its building blocks (ACS attitude control, NAV, IMU, and NASA's LC, SC, HS health-and-safety, CS checksum)
- **RTEMS:** the real-time operating system. **e500:** the PowerPC processor. **u-boot, TFTP, NFS:** how the board boots and reads files
- **QEMU draco-s706:** a software model of the computer board, so no hardware is needed
- **LC / SC / RTS:** Limit Checker watches telemetry (a watchpoint, WP, is a limit; an actionpoint, AP, is the response); Stored Command runs pre-written command lists (relative-time sequences). Together: the onboard autonomy
- **MEKF:** the attitude filter that blends gyro and star-tracker data

The ground and sim side

- **COSMOS (OpenC3):** the ground system in a browser: telemetry screens, a command sender, and Script Runner, which runs the mission test. Reached through the front door https://draco.efsi.com/ (tile **Mission Control**, or **Open COSMOS**). **Yamcs** is a second ground station on the same downlink
- **NASA 42:** a physics simulator that supplies the true attitude, orbit, GPS fixes and gyro rates. The **bridge** packs them into commands the vehicle accepts
- **Closed evidence run:** a script that boots the vehicle from cold, runs every check against it, and writes a record with the exact console lines it read
- **dRT-XXX-nnn:** a numbered requirement. A **tag** is that number written at the line of code that implements it, so a tool can trace both ways

The AI assistant

- **What it is:** a program (Claude Code) that reads the repository, runs commands and tests, writes code and notes, and reports back in a text conversation with a person
- **What it did:** most of the new code, the test script, the evidence tooling, the notes and this deck, under a person's direction and review
- **What it is not:** nothing on the vehicle or in the ground system is AI. The flight software is ordinary C, verified the ordinary way
- **How to read its claims:** every number in this deck is read from a file in the repository, never typed from memory. The file paths are on the slides

If a term is not here, the COSMOS Operator Guide deck walks the screens stage by stage, and the Latest for SMA deck covers the standards and the AI rules. All decks: front door tile Presentations, or https://draco.efsi.com/slides/ (open, no sign-in).

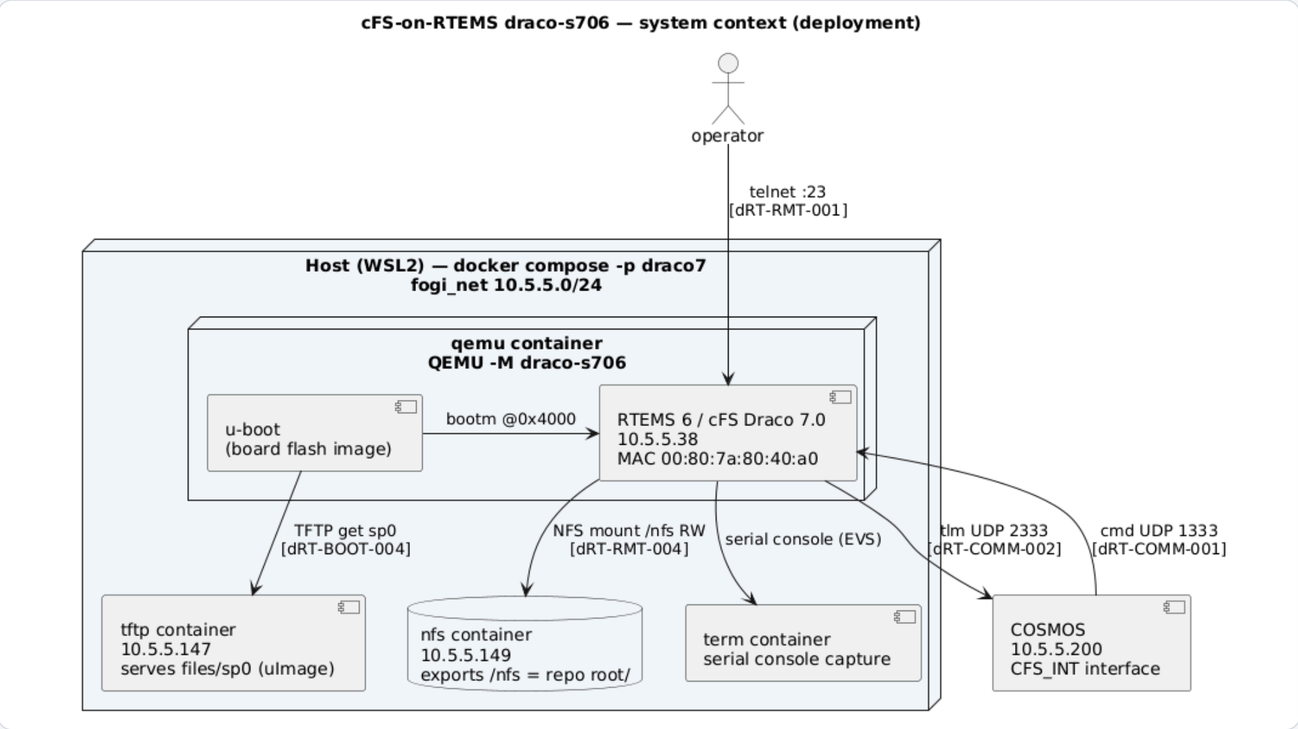
OVERVIEW

What this test demonstrates

1. A full early-orbit mission timeline executed by the flight software. The ground only initiates.
2. Real flight mechanisms under test: SC relative-time sequences with checksum-verified stored commands, LC watchpoint and actionpoint autonomy, SCH scheduling, EVS event reporting, CI and TO ground link.
3. Deployment discipline: two-step ARM then FIRE pyro channels with microswitch confirmation, including the negative test that FIRE without ARM is rejected.
4. Propulsion operations: autonomous momentum dump when wheel momentum exceeds the LC limit, plus ground-commanded station-keeping burns with propellant accounting.
5. New: GPS ingest with a plausibility gate. A forged or replayed fix is rejected, latched, and answered onboard by LC actionpoint 1 starting RTS-006, with no ground in the loop.
6. New: IMU ingest with a validation gate (validity, time, rate and acceleration limits, stuck axis, dropout). Validated rates feed the attitude filter; a fault or dropout falls back to the internal model within half a second, and a latched fault is answered by LC actionpoint 2 starting RTS-007.
7. Mission assurance built in: requirements written before the code, traced by source tags, verified by a scripted closed run, reported with artifact-cited provenance.
8. Operator and developer access: a telnet shell (the UART1 serial shell is off by design), a read-only serial console (front door › Demo control › **Serial console** tab), a live NFS-backed /cf in the dev config, and the 42 3D views in the browser next to COSMOS (front door tile **Live 3D view**, https://draco.efsi.com/42/, or Viewer › **3D view**).

ENVIRONMENT

System under test



What is real

cFS Draco 7.0 flight software, RTEMS 6, PowerPC e500 big-endian, the u-boot boot chain over TFTP, CCSDS command and telemetry, the COSMOS 7.2 ground system.

What is modeled

The draco-s706 board in QEMU 11 with MPC8548 peripherals, built in a 22.04 container to match the vehicle image, interrupt-controller timers at the real 50 MHz; vehicle dynamics, GPS and IMU from NASA 42, bridged to the vehicle big-endian.

Host

Moved from a Windows laptop to the EFSI server in September. Snap Docker sees only the home directory, so the repository is bind-mounted under it; all stack commands run from that path.

The board's unmodified u-boot fetches the cFS uImage over TFTP. The OS swap is invisible to the boot infrastructure.

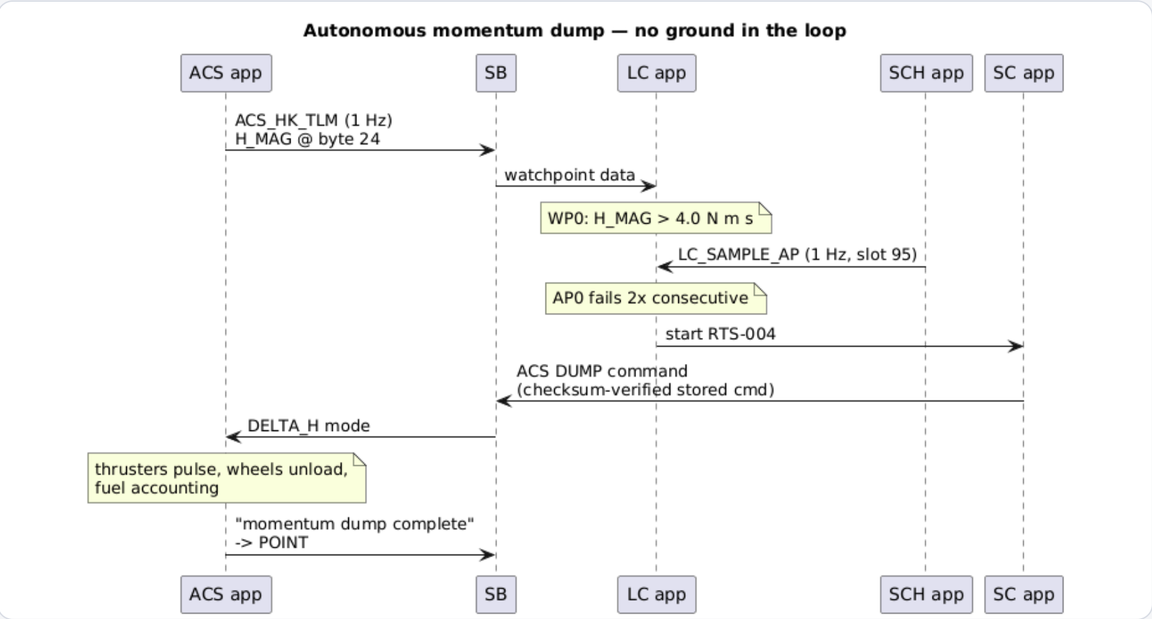
CONOPS

Mission timeline: test phases

Phase	What happens	FSW mechanism
P0 pre-launch	Boot to OPERATIONAL; RTS-001 auto-arms autonomy with one group command (RTS bank 2 to 7 enabled, LC ACTIVE)	SC power-on RTS
P0b negative	FIRE without ARM: rejected, error counted	ACS two-step deploy
P1 launch	Ground: SC START_RTS 5 gives LIFTOFF (front door › Demo control › MISSION › **LAUNCH (RTS-005)**, or the same button on Viewer › Mission › Limits & autonomy); separation at T+10 s with tipoff rates	SC RTS-005 to ACS
P2 deployment	Solar array and antenna: ARM, PYRO FIRE, DEPLOYING, microswitch confirms DEPLOYED	RTS-002 and 003 to ACS
P3 detumble	Rates damped into wheels; automatic transition to POINT below 0.005 rad/s	ACS control mode
P4 momentum	Disturbance torque accumulates wheel momentum in POINT	ACS dynamics
P5 auto dump	Momentum above 4.0 N m s: LC AP0 starts RTS-004, DELTA_H thrusters unload, AP re-armed	LC, SC and ACS
P6 station-keep	Ground: 0.5 m/s burn; self-terminates at target delta-V; fuel decremented	ACS maneuver mode
P7 IMU fusion	With 42 and the bridge up: IMU VALID, fault latch clear, ACS rate source IMU and steady for 10 s; skipped with a note when 42 is not running	IMU gate, ACS source select
Any time: GPS spoof	Implausible fix rejected and latched; LC AP1 starts RTS-006; next genuine fix accepted	NAV gate, LC and SC
Any time: IMU fault	Bad sample rejected and latched; LC AP2 starts RTS-007, which pins the filter to the internal model until ground recovery	IMU gate, LC, SC and ACS

AUTONOMY

Three autonomous responses with no ground in the loop



Momentum dump

H_MAG (wheel momentum magnitude) above 4.0 trips LC watchpoint 0; actionpoint 0 starts RTS-004; DELTA_H unloads; RTS-004 re-arms the actionpoint so the sawtooth repeats.

GPS spoof response

NAV rejects and latches an implausible fix; LC watchpoint 1 sees the latch, actionpoint 1 starts RTS-006. Recovery by ground command only.

IMU fault response

IMU rejects and latches a bad sample; ACS drops to its internal gyro model within one cycle; LC watchpoint 2 sees the latch, actionpoint 2 starts RTS-007, which pins the filter to INTERNAL and re-asserts the gate. Ground clears the latch, re-arms AP2 and returns the source to AUTO.

Every hop is a real flight mechanism: telemetry-driven LC watchpoints, scheduled AP sampling, checksum-verified stored commands.

FSW

Flight software additions, all requirement-tagged

Component	Content	Traces
cFS/apps/acs	Modes SAFE, DETUMBLE, POINT, DELTA_H, STATIONKEEP; quaternion propagation; deploy channels; four thrusters; fuel and delta-V accounting; sensor model and MEKF estimator. Simulation-grade dynamics, labeled as such	dRT-ACS-001 to 022
cFS/apps/nav	GPS ingest on its own message ID; NO_FIX, FIX, STALE, REJECT states; plausibility gate (validity, time, altitude, speed, continuity); latched reject with clear command; rate-limited reject events	dRT-NAV-001 to 009
cFS/apps/imu	IMU sample ingest on its own message ID; NO_DATA, VALID, STALE, FAULT states; gate (validity bits, monotonic time, rate and accel limits, stuck axis); cumulative delta-angle and delta-V; latched fault with clear; rate-limited events. Pure imu_check module at 100% line coverage	dRT-IMU-001 to 010
cFS/apps/acs (fusion)	Rate source AUTO or INTERNAL: the filter runs on validated IMU rates when fresh, falls back to the internal model on stale or fault with 1 s re-selection hysteresis and bias-covariance re-inflation on every switch	dRT-ACS-023, 024
SC RTS bank	RTS-001 power-on arming with one group-enable command (five separate enables overflowed the SB pipe limit); RTS-005 launch; RTS-002 and 003 deploys; RTS-004 dump; RTS-006 spoof response; RTS-007 IMU fault response	dRT-ACS-012, NAV-009, IMU-009
LC tables	WP0: ACS H_MAG above 4.0, AP0 two fails start RTS-004. WP1: NAV REJECT_LATCHED, AP1 starts RTS-006. WP2: IMU FAULT_LATCHED, AP2 starts RTS-007	dRT-ACS-008, NAV-009, IMU-009
TO_LAB subscriptions	Downlink table: EVS events, all app housekeeping, three ACS, two NAV and two IMU packets, with limits sized for the 10 Hz IMU stream; CI_LAB uplink poll cut from 500 to 50 ms	dRT-ACS-007, NAV-004, IMU-004
Ground and sim	COSMOS fogi plugin with ACS, NAV, IMU, GPS42, IMU42 and SPACECOP targets and pages; 42 bridge packs GPS and IMU samples big-endian and injects fault modes; 42 3D views in the browser	dRT-NAV-002, IMU-002

TEST

The test script, COSMOS Script Runner

fogi_mission_test.rb phases, read from the script:

- P0 pre-launch: verify OPERATIONAL, ACS SAFE, stowed
- P0b negative test: FIRE without ARM must be rejected [dRT-ACS-003]
- P1 launch: start RTS-005 (RTS-001 armed the autonomy at power-on)
- P2 deployment: RTS-002/003 arm+fire; microswitches close [dRT-ACS-004]
- P3 detumble to POINT [dRT-ACS-005]
- P4/P5 autonomous momentum dump: LC WP0 trips AP0, RTS-004 [dRT-ACS-008]
- P6 commanded station-keeping burn, 0.5 m/s [dRT-ACS-010]
- P7 IMU fusion: ACS on the validated IMU stream when 42 and the bridge are up [dRT-ACS-023]
- P8 GPS spoof response: forged fix rejected and latched, LC AP1 fires RTS-006, ground recovery [dRT-NAV-007, 009]
- MISSION TEST COMPLETE, all phases passed

Every phase is cmd() plus wait_check() against live telemetry, with two timed holds in P7 and P8. No operator judgment calls.

The same sequence runs headless in run_evidence.py, writing a stamped run record either way.

Negative testing included: the FIRE-without-ARM rejection is a scripted assertion, not a footnote.

To run it: on the front door https://draco.efsi.com/, Demo control panel (bottom), group STACK: **Fresh vehicle**, then **Run** in the confirm box (about 70 s, the Job output tab ends “== OK”). Then group MISSION: **Mission test (COSMOS)**, **Run**. The Job output tab prints one banner per phase and ends “== OK: mission test completed” with the evidence record path. Greyed button: hover it for the reason (usually “flown this boot”).

The latest run from each ground station, COSMOS and Yamcs, phase by phase: next slide, read from the run records.

Script updated 26 September 2026 with P7 and P8. It now asserts on ACS, LC, IMU and NAV; both phases skip with a note when 42 is not running, so the test still passes on a bare vehicle.

TWO FULL MISSION TESTS

The same mission, flown once from each ground station

COSMOS: 9/9 PASS, 182 s

fogi_mission_test.rb in Script Runner, started by **Mission test (COSMOS)** (front door › Demo control › MISSION); commands over CFS_INT

cosmos_20261103_162908 · 3 Nov 2026 16:29 UTC · git d3d4cb2d

Yamcs: 9/9 PASS, 177 s

draco_mission_test.py, the line-for-line port, as a Yamcs script activity, started by **Mission test (Yamcs)** (front door › Demo control › YAMCS); commands over tc-out

ground_20261103_163655 · 3 Nov 2026 16:36 UTC · git d3d4cb2d

Phase	Requirements	COSMOS	Yamcs
P0 pre-launch: verify OPERATIONAL, ACS SAFE, stowed	—	PASS	PASS
P0b negative test: FIRE without ARM must be rejected	ACS-003	PASS	PASS
P1 launch: start RTS-005 (RTS-001 armed the autonomy at power-on)	ACS-012	PASS	PASS
P2 deployment: RTS-002/003 arm+fire; microswitches close	ACS-004	PASS	PASS
P3 detumble -> POINT	ACS-005	PASS	PASS
P4/P5 autonomous momentum dump: LC WP0 trips AP0 -> RTS-004	ACS-008	PASS	PASS
P6 commanded station-keeping burn, 0.5 m/s	ACS-010 ACS-011	PASS	PASS
P7 IMU fusion: ACS on the validated IMU stream when 42 + bridge are up	ACS-023	PASS	PASS
P8 GPS spoof response: forged fix -> NAV gate -> LC WP1/AP1 -> RTS-006, ground recovery	NAV-005 NAV-007 NAV-009	PASS	PASS

Each from a freshly booted vehicle with 42 running. Read from the run records in cFS/evidence/runs, never typed.

VERIFICATION, 1 OF 4

Closed evidence run on the new host: platform checks (15)

Check	Covers	Verdict, detail
V-BOOT-UIMAGE	BOOT-001	PASS: u-boot verified + booted the cFS uImage
V-BOOT-CHAIN	BOOT-004	PASS: unmodified u-boot env fetched sp0 over TFTP and bootm'd it
V-EXEC-OPER	EXEC-006	PASS: cFE reached OPERATIONAL
V-FS-STARTUP	FS-003	PASS: startup script opened from /cf
V-EXEC-APPS	EXEC-003	PASS: all 13 static apps initialized
V-NET-TSEC	NET-002	PASS: tsec0 attached via FDT and link UP
V-COMM-CIPORT	COMM-001	PASS: CI_LAB listening on 1333 (PID-derived)
V-RMT-NFSMOUNT	RMT-004	PASS: NFS export mounted at /nfs during startup

Check	Covers	Verdict, detail
V-FS-NFSCF	FS-004	PASS: dev config: /cf backed by live NFS (tarfs shadowed)
V-NET-PING	NET-003	PASS: ping 10.5.5.38 from tftp container
V-COMM-UPLINK	COMM-001	PASS: ENABLE_TELEMETRY accepted on 1333 (telemetry started => ...
V-COMM-DOWNLINK	COMM-002	PASS: 30 TO_LAB packets captured on 2333 (TO_LAB marker present)
V-RMT-TELNET	RMT-001	PASS: telnet login shell reached (prompt + ls /)
V-RMT-NFSRW	RMT-004	PASS: NFS /nfs read OK, target->host write OK
V-RMT-SERSHELL	RMT-006	FAIL: UART1 serial shell accepts commands (ls / over the wire)

run_20261023_201111 · 64/66 PASS · git 4b8f3df7 · sp0 sha256 2d57f4b2aec1813e... · V-RMT-SERSHELL fails by configuration: the serial shell is suppressed by the tracked NOS3 flag file (documented deviation); SKIP: V-ART-ACS-RATE

VERIFICATION, 2 OF 4

Closed evidence run on the new host: mission checks (14)

Check	Covers	Verdict, detail
V-ACS-ARM	ACS-012	PASS: power-on RTS-001 armed the autonomy (RTS bank + LC ACTIVE)
V-ACS-NEGFIRE	ACS-003	PASS: FIRE without ARM rejected
V-ACS-LAUNCH	ACS-012	PASS: RTS-005 launch: LIFTOFF + SEPARATION + DETUMBLE
V-ACS-DEPLOY	ACS-004	PASS: both deployments: ARM->FIRE->DEPLOYING->DEPLOYED w/ microswitch
V-ACS-DETUMBLE	ACS-005	PASS: detumble complete -> POINT
V-ACS-DUMP	ACS-006 to 008	PASS: AUTONOMOUS dump: LC AP -> RTS-004 -> DELTA_H -> unloaded
V-ACS-CMDDUMP	ACS-009	PASS: ground-commanded dump enters/exits DELTA_H via the same path

Check	Covers	Verdict, detail
V-ACS-SK	ACS-001, 010, 011	PASS: commanded station-keeping burn to target delta-V
V-ACS-FUELWARN	ACS-015, 016	PASS: latched LOW then CRITICAL warnings after SET_FUEL
V-ACS-FUELEXH	ACS-014	PASS: exhaustion: event + SAFE; propulsive commands rejected
V-ACS-UTCOV	ACS-017 to 020, 022	PASS: sensor+MEKF unit tests pass at 100% line coverage
V-ACS-EKFCONV	ACS-018, 020, 021	PASS: live MEKF converged: attitude error 0.043 deg < 0.5
V-ACS-BIAS	ACS-017, 019	PASS: live gyro-bias estimate tracks injected truth [+0.0020 -0.0014 ...
V-ACS-APPCOV	ACS-001 to 018, 020, 021, 023, 024	PASS: acs_app.c unit tests pass at 100.00% line coverage (floor 100%)

run_20261023_201111 · 64/66 PASS · git 4b8f3df7 · sp0 sha256 2d57f4b2aec1813e... · V-RMT-SERSHELL fails by configuration: the serial shell is suppressed by the tracked NOS3 flag file (documented deviation); SKIP: V-ART-ACS-RATE

VERIFICATION, 3 OF 4

Closed evidence run on the new host: GPS, IMU and fusion checks (16)

Check	Covers	Verdict, detail
V-IMU-UTCOV	IMU-005, 010	PASS: imu_check unit tests pass at 100% line coverage
V-NAV-UTCOV	NAV-005, 008	PASS: nav_gate unit tests pass at 100% line coverage
V-NAV-INGEST	NAV-001 to 004	PASS: live 42 GPS stream: fix acquired, HK STATE FIX, FIX_COUNT 7->14 over ...
V-NAV-GATE	NAV-005, 006	PASS: gate ON at init; OFF: +500 km teleport ACCEPTED (POS_TLM FIX, flags ...
V-NAV-FAULT	NAV-007, 009	PASS: teleport REJECTED 0x08 -> LC AP1 PASS->FAIL (LC 58) -> LC 1101 RTS = ...
V-NAV-RATELIMIT	NAV-008	PASS: 41 forged fixes: REJECT_COUNT 1->42, 3 REJECTED event(s) at totals ...
V-NAV-RECOVER	NAV-007, 009	PASS: ground recovery: CLEAR_REJECT -> REJECT_LATCHED CLEAR, LC 61 AP1 ...
V-IMU-INGEST	IMU-001 to 004; ACS-023	PASS: live stream: IMU VALID, ACS -> IMU, no STALE and a single source ...

Check	Covers	Verdict, detail
V-IMU-FAULT	IMU-005, 007 to 009	PASS: spike -> REJECTED (rate-limited events) -> LC AP2 (LC 1102) -> ...
V-ACS-FAULTFB	ACS-024	PASS: ACS fell back to INTERNAL (fault/stale) on the rejected stream
V-IMU-GATE	IMU-006	PASS: gate ON at init; OFF: injected |w|=86.603 rad/s spike ACCEPTED (IMU ...
V-IMU-RECOVER	IMU-007, 009; ACS-023	PASS: ground recovery: CLEAR_FAULT, LC AP2 re-armed, AUTO -> MEKF back on ...
V-ACS-FALLBACK	ACS-024	PASS: stream stopped: IMU STALE within 0.5 s, ACS -> INTERNAL (stale)
V-NAV-STALE	NAV-003	PASS: stream stopped: NAV STALE event within NAV_STALE_S, HK STATE STALE, ...
V-NAV-APPCOV	NAV-001 to 008	PASS: nav_app.c unit tests pass at 100.00% line coverage (floor 100%)
V-IMU-APPCOV	ACS-024; IMU-001 to 009	PASS: imu_app.c unit tests pass at 100.00% line coverage (floor 100%)

run_20261023_201111 · 64/66 PASS · git 4b8f3df7 · sp0 sha256 2d57f4b2aec1813e... · V-RMT-SERSHELL fails by configuration: the serial shell is suppressed by the tracked NOS3 flag file (documented deviation); SKIP: V-ART-ACS-RATE

VERIFICATION, 4 OF 4

Static artifact checks, no boot needed (21)

Check	Covers	Verdict, detail
V-ART-BOOT-LOADADDR	BOOT-002	PASS: uImage load 0x4000 == entry 0x4000 == ELF ...
V-ART-BOOT-STARTGUARD	BOOT-003	PASS: deployed payload begins with 0x7c711b78; ...
V-ART-BOOT-FDTEMBED	BOOT-005	PASS: ELF: bsp_fdt_get/bsp_fdt_copy are PSP functions ...
V-ART-BOOT-FDTCLOCKS	BOOT-006	PASS: embedded DTB (12736 B, valid FDT) cpu node ...
V-ART-BOOT-CCSR	BOOT-007	PASS: ELF absolute symbol qoriq = 0xe0000000 (CCSR ...
V-ART-BOOT-UARTPOLL	BOOT-008	PASS: console_device_table in the ELF binds ...
V-ART-EXEC-ABI	EXEC-001	PASS: ELF32 big-endian PowerPC; 'rtems-6.0.0 ...
V-ART-EXEC-STATIC	EXEC-002	PASS: no SHT_DYNAMIC section; RTEMS_DYNAMIC_LOAD ...
V-ART-EXEC-OSLIMITS	EXEC-004	PASS: OSAL limits tasks/queues/modules/timers: ...
V-ART-EXEC-PROCID	EXEC-005	PASS: targets.cmake cpu1_PROCESSORID 100; as-built ...
V-ART-FS-TARFS	FS-001	PASS: RTEMS_INCLUDE_TARFS; OSAL bsp_tarfs_setupfs.c ...

Check	Covers	Verdict, detail
V-ART-FS-TARGEN	FS-002	PASS: make_rtems_tarfs.sh stages ...
V-ART-NET-LIBBSD	NET-001	PASS: rtems-libbsd selected (OSAL_RTEMS_NETWORK_LIB ...
V-ART-NET-LOOORDER	NET-004	PASS: network_ppce500_rtems_Init calls ...
V-ART-NET-FAILSAFE	NET-005	PASS: all 2 failure branches of ...
V-ART-COMM-CFSINT	COMM-003	PASS: COSMOS CFS_INT = udp 10.5.5.38 write 1333 read ...
V-ART-RMT-SHELLSET	RMT-002	PASS: as-built rtems_shell_Initial_commands = 23 ...
V-ART-RMT-NOAUTH	RMT-003	PASS: telnetd config and per-session shell env set ...
V-ART-RMT-NFSREG	RMT-005	PASS: draco_mount_nfs() (line 166) registers ...
V-ART-ACS-RATE	ACS-002	SKIP: design cadence is a 100 ms SB pend timeout ...
V-ART-ACS-THRTLM	ACS-013	PASS: ACS_ActTlm_Payload_t carries ThrFireCount[4] ...

run_20261023_201111 · 64/66 PASS · git 4b8f3df7 · sp0 sha256 2d57f4b2aec1813e... · V-RMT-SERSHELL fails by configuration: the serial shell is suppressed by the tracked NOS3 flag file (documented deviation); SKIP: V-ART-ACS-RATE

THE MISSION TEST, EXTENDED TO AN HOUR

Planned in PlanDev, flown from COSMOS, compared afterwards

What ran

- Plan 11 “DRACO launch day”: the mission test’s phases in its first ten minutes, then 45 minutes of pointing with 34 scheduled momentum dumps
- Turned into a COSMOS script that sends each command at its planned second and makes the mission test’s own checks
- 60 minutes on a freshly booted vehicle, 22 October 2026

Result	Value
Checks	**155 / 155 PASS**
Command timing	all within 0.1 s of plan
Momentum dumps	41 planned, 41 flown
Detumble	20.2 s planned, 19.9 s flown
Fuel at one hour	9.445 kg planned, 9.442 flown

Not part of the closed evidence run and not tied to requirements. Record: claude-notes/PLAN-VS-ACTUAL-2026-10-22.md; tools in tools/plandev; the Ground System deck explains it. See the plan: front door, Ground stations panel, PlanDev card, Open Plan (https://plan.draco.efsi.com, new tab), or Plan ↗ in the viewer’s top bar.

SQA

Mission assurance with the new software

Requirements first

The 24 ACS, 9 NAV and 10 IMU requirements were written before the first line of their code; the IMU set came from a frozen interface contract; six assistant conversations, each driven and reviewed by a person, implemented it in parallel. The earlier port groups were reverse-derived.

Traceability

83 of 83 requirements carry [dRT-...] tags at the implementing line (75 of 75 when this slide was written, 23 October). gen_trace.py regenerates matrices and diagrams and fails the build on any code with no requirement.

Closed verification

run_evidence.py boots the real boot chain and executes every check against the live target. Results are stamped run records with captured console, telnet and packet logs.

Hard provenance and static analysis

gen_report.py builds the Design and Verification Report where every number cites an on-disk artifact in a SHA-256 manifest. cppcheck runs over the core and the mission apps with a dispositioned deviation log, and McCabe complexity is recorded per function.

This deck follows the same rule: phases from the test script, results from the run record, counts from requirements.csv and the regenerated trace.

NASA STANDARDS

How this evidence maps to the NASA standards

Standard, in plain words	What this project does about it	Status
NPR 7150.2D: NASA's software rulebook, with rigor scaled by a class letter	Classified **Class E** (a prototype exploring a design concept). Class E invokes only classification, records, reuse assessment, safety-criticality and assurance planning. The Class D practices on this deck, traceability, unit tests, test against requirements, target validation, static analysis, CM, are applied voluntarily	Class E set recorded 30 Sep 2026; voluntary set 3 met, 4 partial
NASA-STD-8739.8B: software assurance and software safety	Safety-critical determination: **no**. No hazard analysis exists because nothing physical can be harmed; the vehicle is a software model. SWE-134 safety items are therefore not invoked, though several are implemented by design	Determination recorded; SMA concurrence not yet obtained
NASA-STD-7009B: how far to trust a model or simulation	Capability self-assessed factor by factor: verification 2 of 4 (closed runs, checksummed baseline), validation 1 (no real-vehicle referent), data pedigree 1. Fidelity gaps are stated on the slides: QEMU board model, simulation-grade dynamics, synthetic IMU noise	Self-assessment only; no formal record
NPR 7123.1D: verification (built it right) versus validation (built the right thing)	Verification: 83 requirements traced to code (TRACE.md); the 75 that existed on 23 October all covered by the 66 closed-run checks, the 5 CFDP ones by the CFDP run, the 3 table rules pending; unit tests for every mission-app file. Validation: the intended use in MISSION_TEST_DESIGN.md is demonstrated end to end by the mission test	Verification strong; validation internal only
SWE-146 and handbook topic 7.25: AI-written code is auto-generated code	Every line the AI assistant wrote is under the same tests, static analysis and human review as hand code; the person who directed it is responsible; the instructions given to it are under configuration control	Applied; the generator itself cannot be validated

Record: cFS/evidence/rtems_reqs/CLASSIFICATION.md. The Latest for SMA deck carries the full scorecard, the SWE-134 item-by-item review and the NASA guidance on AI, quoted.

FINDINGS, AUGUST

What the process caught while building the mission test

Finding	Caught by	Class
NFS mounted at /mnt shadowed the /cf tarfs; ES could not open the startup script, zero apps started	First closed evidence run (8 of 13)	Flight defect
LC watchpoint offset assumed a 12-byte telemetry header; the header is 16 bytes, so autonomy never fired	Mission-sequence dry run	Config defect
SC walks RTS entries in 32-bit words with round-up; unpadded 10-byte commands broke table validation	Boot events (RTS load reject)	Table defect
Upstream RTS sample tables are endian-naive; arguments must be big-endian on this target	Code review during authoring	Latent defect
TO_LAB default subscription table downlinks nothing; July's closed loop carried only TO_LAB's own housekeeping	COSMOS page design	Gap
The RTEMS shell init does not attach the shell to the named device; it inherits the caller's console	Serial-shell bring-up	Latent defect
LC sets a triggered actionpoint passive by design; after the first dump momentum grew unbounded. RTS-004 now re-arms it	Long-duration live vehicle	Flight defect
COSMOS packet definitions for MM, MD and FM were counter stubs with misaligned fields; rewritten from the flight structs	Log Messages size errors	Ground defect

FINDINGS, SEPTEMBER

What the process caught since the August deck

Finding	Caught by	Class
SC subscribes to its own command ID with the default pipe limit of four; RTS-001's fifth ENABLE_RTS was silently dropped with no counter moving. Replaced by one group-enable command	One SB "Msg Limit Err" event in the boot console	Flight defect
Restarting 42 rewinds simulation time; the NAV gate correctly rejects every fix as a replay until sim time passes the last accepted fix. Documented ground reset procedure	Live vehicle after a 42 restart	Behavior, documented
The downlink command in the runbook, TO ENABLE_OUTPUT, does not exist; the command is TO ENABLE_TELEMETRY with IP_ADDR	Bootstrap script on the new host	Documentation defect
Snap Docker exposes only the home directory; every bind mount from the web root failed. Repository bind-mounted under home; scripts derive paths from their own location	First stack start after the move	Infrastructure
The 42 binary copied from the old host linked an older freeglut and crashed in text drawing; rebuilt on the target host, the same rule already applied to the flight software	Debugger backtrace	Build discipline
The server firewall silently dropped the vehicle's telemetry to the host bridge address, failing the uplink and downlink checks with everything else green. One scoped rule on the internal bridge fixed it	Closed evidence run on the new host; 32 firewall log entries	Infrastructure
The serial shell is suppressed by a tracked flag file from the NOS3 integration, yet the app that needs the UART is listed in the startup script without its object in the dev /cf. The serial-shell check now fails by configuration	Closed evidence run	Config inconsistency

FINDINGS, IMU INTEGRATION

What the process caught while adding the IMU, the NAV verifiers and fixing the clock

Finding	Caught by	Class
CI_LAB services its uplink socket only on a 500 ms receive timeout, at most eleven packets per cycle. A 10 Hz stream backed up 12 s and kept growing. Mission override sets the timeout to 50 ms; path latency now 166 ms median	Sequence-matched packet capture on the bridge	Flight config defect
The EVS long-event downlink limit was 32; the boot burst is exactly 32 events, so the 33rd and 34th were dropped every boot with no counter moving. Raised to 48	"Msg Limit Err" on every earlier boot	Pre-existing flight defect
A comment line ending in "+" in the COSMOS plugin file is read as line continuation by the parser, so the next target definition vanished silently	Missing IMU42 target after install; bisected plugin file	Ground tool behavior
The IMU dropped its time reference on every stale entry, disabling replay and stuck detection after any gap. Reference now persists until a ground clear	Review by a separate assistant conversation briefed on flight safety, read by a person	Design defect, fixed pre-release
Rate-source selection had no hysteresis; delivery hiccups made the filter thrash between sources. Added 1 s re-selection hold and bias-covariance re-inflation on switch	Review by a separate assistant conversation briefed on flight safety, read by a person	Design defect, fixed pre-release
42 exports only true rates and accelerations over its IPC; its sensor noise model never reaches the wire. The bridge now adds gyro noise and announces the fidelity gap at start-up	Reading the 42 autocode; requirements-honesty review	Fidelity gap, disclosed
Ground recovery after an autonomous spoof response re-fired RTS-006 when the actionpoint was re-armed in the same second as the latch was cleared; LC had not yet sampled the cleared latch. Recovery now waits for the latch to read clear	Scripted P8 phase and V-NAV-RECOVER during authoring	Procedure defect
The vehicle clock ran at 0.494x wall time. RTEMS ticks off the interrupt controller's global timers assuming bus/8 = 50 MHz; upstream QEMU hard-codes them at 25 MHz. Now a board property set to 50 MHz. The last 1% (1 Hz tone every 1.011 s) was a second defect, fixed 23 Oct: each tick re-armed from the late callback; now 1.000 s	IMU rate counters vs wall clock, then a 60 s clock measurement	Board-model defect, fixed
A QEMU built on the Ubuntu 24.04 host needs glibc 2.38 and exits 127 inside the 22.04 vehicle container. The emulator is now built in a 22.04 container by run/build-qemu.sh	First boot on the rebuilt emulator	Build discipline

RUNBOOK

Running the demo: two tracks

Presenter, no programming: the front door

1. Open https://draco.efsi.com/ and press **Sign in with Microsoft** (or, under it, type the shared COSMOS password in the box and press **Sign in with password**). Every button below is in the **Demo control** panel at the bottom of the page; each opens a confirm box: press **Run**, then read the live log in the **Job output** tab
2. Stack down? Group STACK, **Start stack**: boots everything, enables telemetry. Check the status strip at the top: Vehicle, Stack, Commanding COSMOS
3. Open the watch page in a second tab: tile **Mission Control** (the viewer, FOGI HOME), then Viewer › Home › **Overview** and Viewer › GNC › **ACS**
4. Group MISSION, **LAUNCH (RTS-005)**, and narrate the ACS page; or **Mission test (COSMOS)** for the scripted run
5. Between shows: group STACK, **Fresh vehicle** (about 70 s, back on the pad). At the end: **Shut down**
6. 3D view: group 3D VIEW, **42 start** (also clears the IMU latch and resets the GPS gate), then tile **Live 3D view** (https://draco.efsi.com/42/); the filter switches to the IMU within a few seconds
7. Greyed button: hover it for the reason. Values frozen: group HEALTH, **Verify**, then **Downlink / map**

Developer, terminal in /home/efsi/draco7_0/run

- ./demo.sh: the same presenter steps as a menu (1 start, 2 LAUNCH, 3 status, 4 live events, 5 fresh vehicle, 6 shut down)
- ./draco.sh rtems up | stop: checked bring-up and stop
- ./draco.sh rtems fsw, then sync_cf: build the flight software and re-baseline the live /cf (required after every build)
- ./bootstrap-draco.sh --verify | --per-boot: health check; re-map targets and enable downlink after each reboot
- docker compose -p draco7 down, then python3 ../cFS/evidence/rtems_reqs/run_evidence.py --keep-up: the 66-check closed evidence run. The stack must be down first or the script never reboots the vehicle
- gen_trace.py, gen_report.py: regenerate trace and report. gen_deck.py still builds the legacy .pptx by hand, but nothing runs it: the decks are powerpoint/decks (this one: gen_results.py after a closed run)
- One-time COSMOS setup: install both plugin gems, map ACS, NAV and IMU to CFS_INT, never GPS42 or IMU42. One-time host setup: a ufw rule allowing UDP 2333 in on the fogi_net bridge, or the evidence run cannot capture the downlink

Changed since August: the run directory, the front door with its Demo control (replaces the public COSMOS address as the start), the per-boot bootstrap step, the NAV mapping, and the 42 browser view.

VALIDATION

COSMOS: which screens, and how to trust the data

Screens: Viewer (tile Mission Control), group button, page button

- **Home › Mission home** (FOGI HOME): top level, one box per subsystem with live values and Open buttons
- FOGI pages: **Mission › App health** (APPS), **GNC › Attitude & rates** (GNC, live graphs), **Mission › Deployments**, **Propulsion**, **Limits & autonomy** (LIMITS, with the **LAUNCH (RTS-005)** button), **Events**, **Sequences (ATS/RTS)**, **Home › Overview**; walls under **Wall**: Wall (3 screens), GNC wall, Sensors wall, Mission wall, All-in-one (WALL_ALL also via Select Screen)
- **GNC › ACS** (ACS ACS): one-page combined mission view
- **Tools › Packet Viewer**, target NAV, packets HK_TLM_PKT and POS_TLM_PKT: STATE, FIX_COUNT, REJECT_LATCHED, altitude, speed (or **GNC › NAV (GPS gate)**)
- **GNC › IMU**: stream state, sample rate, fault latch, gyro against ACS truth, the ACS rate source
- **Tools › Command Sender**: TO ENABLE_TELEMETRY with IP_ADDR 10.5.5.200 for downlink (front door › Demo control › HEALTH › **Downlink / map** does it for you); ACS SET_FUEL for the fuel demo (or the fuel buttons on Mission › Propulsion)
- Stage-by-stage walkthrough: the separate COSMOS Operator Guide deck

Validate the telemetry before presenting

1. **Tools › Cmd/Tlm Server**, Interfaces tab: CFS_INT CONNECTED and Tlm Pkts climbing
2. Same page, Log Messages at the bottom: no "received with actual packet length" errors. A size error means a definition no longer matches the flight structs. Report it
3. **Tools › Packet Viewer**: any packet's RECEIVED_COUNT increments about once a second, and RECEIVED_TIMEFORMATTED is now. A first query right after mapping can return stale values
4. Expected pad values: ACS MODE SAFE, FUEL_KG 12.0, FUEL_STATE OK, DEPLOYED_BITS 0, RATE_SOURCE INTERNAL; NAV NO_FIX and IMU NO_DATA until 42 feeds
5. Quick check without COSMOS: the status strip at the top of the front door (Vehicle, CFE_ES age a second or two, Stack, Commanding)

All pages null? Target unmapped or downlink off: bootstrap-draco.sh --per-boot fixes both; on the front door it is Demo control › HEALTH › **Downlink / map** (then **Verify**).

REQUIREMENTS BACKUP · ACS-023 AND 024 ARE ON THE IMU SLIDE

The ACS requirement set: 23 of 25

ID	Shall (abbreviated)	Ver
dRT-ACS-001	The ACS application shall provide the control modes SAFE, DETUMBLE, POINT, ...	T
dRT-ACS-002	The ACS application shall propagate simulation-grade rigid-body attitude ...	A
dRT-ACS-003	The ACS application shall provide two deployment channels (solar array, ...	T
dRT-ACS-004	Upon pyro fire, a deployment channel shall transition FIRING to DEPLOYING to ...	D
dRT-ACS-005	In DETUMBLE mode the ACS shall damp body rates and shall autonomously ...	T
dRT-ACS-006	In POINT mode the ACS shall accumulate wheel momentum under a modeled ...	A
dRT-ACS-007	The ACS shall publish housekeeping, attitude, and actuator telemetry packets ...	I
dRT-ACS-008	When wheel momentum magnitude exceeds the LC watchpoint threshold, the system ...	D
dRT-ACS-009	The ACS shall accept a ground command to initiate a momentum dump, entering ...	T
dRT-ACS-010	The ACS shall accept a station-keeping command with a target delta-V ...	T
dRT-ACS-011	The ACS shall account for propellant: fuel mass shall decrease in proportion ...	A

ID	Shall (abbreviated)	Ver
dRT-ACS-012	The launch sequence shall execute from SC relative-time sequences: RTS-001 ...	D
dRT-ACS-013	The ACS shall telemeter per-thruster fire counts and cumulative on-time.	I
dRT-ACS-014	When propellant is exhausted the thrusters shall be ineffective: no momentum ...	T
dRT-ACS-015	The ACS shall telemeter a fuel state (OK/LOW/CRITICAL/EXHAUSTED) and shall ...	T
dRT-ACS-016	The ACS shall provide a SET_FUEL maintenance command (test/sim use) that sets ...	T
dRT-ACS-017	The ACS shall model attitude sensors: a three-axis rate gyro whose ...	T
dRT-ACS-018	The ACS shall estimate spacecraft attitude with a Multiplicative Extended ...	T
dRT-ACS-019	The MEKF shall estimate the three-axis gyro bias as part of its state vector; ...	A
dRT-ACS-020	The ACS shall telemeter the filter attitude uncertainty (3-sigma, degrees) ...	A
dRT-ACS-021	After star-tracker acquisition the MEKF attitude estimation error (angle ...	T
dRT-ACS-022	The ACS attitude-estimation module (sensor model + MEKF) shall be implemented ...	T
dRT-ACS-025	The ACS's tunable parameters shall be loaded from cFE tables at initialization ... (added 27 Oct)	T

Verification method: T test, A analysis, I inspection, D demonstration. Full text in cFS/evidence/rtems_reqs/requirements.csv and verify/requirements.yaml.

REQUIREMENTS BACKUP

The NAV requirement set, 10 requirements (001 to 009 new in September)

ID	Shall (abbreviated)	Ver
dRT-NAV-001	The flight software shall provide a navigation application (NAV) that maintains the onboard navigation state from an externally supplied GPS fix. NAV ...	I
dRT-NAV-002	NAV shall accept GPS fixes as NAV_GpsFixCmd_t messages on NAV_GPS_MID, carrying receiver validity, GPS rollover/week/seconds, ECI position and ...	T
dRT-NAV-003	NAV shall report a navigation state of NO_FIX, FIX, STALE or REJECT. The state shall become STALE when no fix has been accepted within 3 seconds, and ...	T
dRT-NAV-004	NAV shall telemeter the navigation solution (position, velocity, geodesy, radius, speed, fix age and counters) and housekeeping ...	D
dRT-NAV-005	NAV shall validate every received GPS fix against a plausibility gate before accepting it, rejecting a fix whose receiver validity flag is clear, ...	T
dRT-NAV-006	The NAV plausibility gate shall be enabled at initialization and shall be commandable off and on from the ground, so that the protected and ...	T
dRT-NAV-007	NAV shall latch an indication that a fix has been rejected since the last clear, and shall provide a ground command to clear that indication, so that ...	T
dRT-NAV-008	NAV shall rate-limit rejection events so that a sustained flood of forged fixes cannot exhaust or obscure the event log, while still annunciating the ...	T
dRT-NAV-009	A rejected GPS fix shall autonomously trigger an onboard response without ground action: Limit Checker watchpoint 1 shall monitor the NAV ...	T
dRT-NAV-010	NAV's tunable parameters (plausibility limits, stale time, event pacing, housekeeping cadence, gate state at boot) shall be loaded from the cFE table NAV.CFG_TBL ... (added 27 Oct)	T

Verification method: T test, A analysis, I inspection, D demonstration. Every NAV requirement is tagged at its implementing line in cFS/apps/nav.

REQUIREMENTS BACKUP

The IMU and fusion requirement set, 13 requirements (12 new in September)

ID	Shall (abbreviated)	Ver
dRT-IMU-001	The flight software shall provide an inertial measurement unit application (IMU) ...	I
dRT-IMU-002	IMU shall accept samples as IMU_SampleCmd_t messages on IMU_SAMPLE_MID of exactly 80 ...	T
dRT-IMU-003	IMU shall report a sensor state of NO_DATA, VALID, STALE or FAULT. The state shall ...	T
dRT-IMU-004	IMU shall telemeter housekeeping (command and error counters, state, fault flags, ...	D
dRT-IMU-005	IMU shall validate every received sample against a plausibility gate before ...	T
dRT-IMU-006	The IMU plausibility gate shall be enabled at initialization with default limits of ...	T
dRT-IMU-011	IMU's tunable parameters (validation limits, stuck count, ...) shall be loaded from the cFE table IMU.CFG_TBL ... (added 27 Oct)	T

ID	Shall (abbreviated)	Ver
dRT-IMU-007	IMU shall latch an indication that a sample has been rejected since the last clear. ...	T
dRT-IMU-008	IMU shall rate-limit rejection events so that a sustained stream of faulty samples ...	T
dRT-IMU-009	A latched IMU fault shall autonomously trigger an onboard response without ground ...	D
dRT-IMU-010	The IMU sample-validation module (imu_check) shall be implemented as a pure, ...	T
dRT-ACS-023	The ACS shall select its attitude-rate measurement source under a ground-commandable ...	T
dRT-ACS-024	When IMU data becomes stale (no VALID data for 0.5 seconds) or faulted (a FAULT data ...	T

The September twelve were written before the code from a frozen interface contract (claude-notes/IMU-INTERFACE.md); every one is tagged at its implementing line and covered by a closed-run check or the unit-test harness. IMU-011 (table rule) is pending.