FADEC / FLIGHT SOFTWARE · TECHNICAL REPORT

EGEMEN

A triple-redundant (TMR) full-authority digital engine control simulation for a single-engine UCAV turbofan — grown into a self-verifying avionics test bench.

version v29 · 2026-07 · single-file HTML · 50 ms deterministic core · 33/33 self-test · TR/EN

Demir Akin · Electrical & Electronics Engineering, Koç University · intern @ Kale Jet Motorları

with Emre Karausta (plant physics · sensor data modeling · 2oo3 edge-case testing · UX) · University of Southampton

live sim: egemen-fadec.vercel.app · case study: demir-akin-portfolio.vercel.app/#/egemen

Contents

01

Executive summary

EGEMEN is a live, browser-based simulation of full-authority digital engine control (FADEC) for a single-engine UCAV turbofan. It is not an animation. A deterministic 50 ms fixed-step core runs real plant physics, a fuel system with bingo logic, triple-channel sensing with 2oo3 median voting, a min-max limit-selector control law with autothrottle, fault detection/isolation/recovery (FDIR) with analytical redundancy, a start-sequence state machine, and a vehicle-level autonomous mission supervisor - plus a maintenance and observability layer (16-bit fault words, black box, interactive manual, flight reports).

The project's defining property is test discipline: the page carries 33 built-in assertions it runs against itself; every random stream derives from one master seed so any scenario replays bit-identically; and whenever a change must not alter behavior, 2,400-cycle trajectories are compared between versions - physics, events and black box, bit for bit. The goal, stated in the project charter: when a fault is injected, the correctness of the system's response must be measurable.

Property	Value
Form factor	single HTML file, no build step, no external dependencies, works offline
Core	50 ms fixed-step (20 Hz), deterministic, master-seeded independent RNG streams
Redundancy	3 channels (A/B/C) x 2 sensor instances per parameter; 2oo3 median; distributed voter over CCDL
Fuel	600 kg tank, BINGO 180 kg, Wf integration, leak injection, BINGO-to-RTB + dispatch gate
Verification	33 self-test assertions + discrimination; determinism hash 0xCFAD6CC2; version-equivalence proofs
Languages	TR / EN everywhere, guarded by an i18n self-check (117/117)
Status	v29 (2026-07); published demo tracks the project

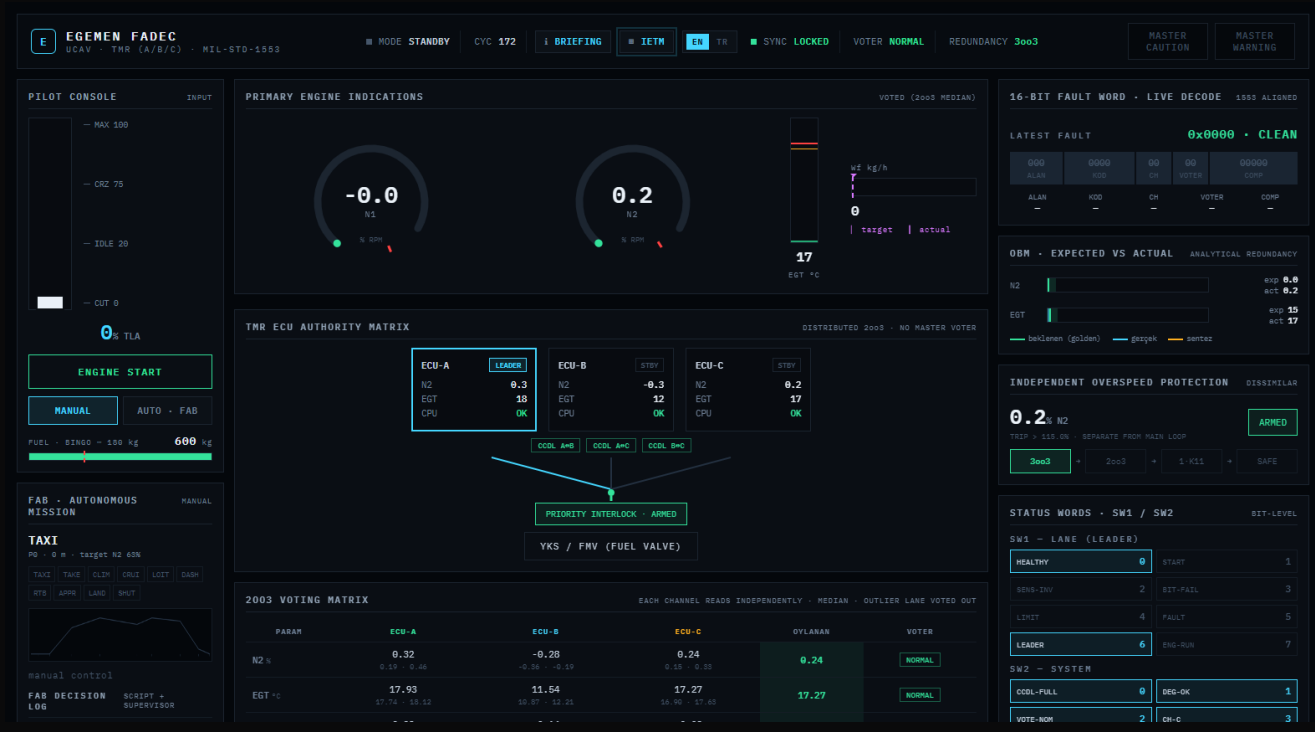

Fig. 1 - Cold panel, engine off: voted gauges at rest, fault word CLEAN, overspeed ladder 3oo3, channel A elected leader on aircraft power - election runs even before a drop of fuel flows (v25).

02

Architecture and core disciplines

2.1 One data contract

The central idea is discipline, not flash: the interface never knows the backend. Each cycle the backend advances the world once (realStep, every 50 ms) and writes a single FADECState contract; the render layer only draws that contract. Drop the render layer and a deterministic, testable control core remains - portable in principle to C firmware. The UI cannot invent data: every number on screen, in the flight chart and in the .xlsx export comes from the same producer.

2.2 Determinism and replay

All randomness derives from one master seed, in a separate stream per subsystem - sensor noise, blips, the fault agent each draw from their own generator. Adding a feature can therefore never shift another feature's noise, which is what makes version-to-version trajectory comparison possible at all. A determinism self-check hashes a reference run; the hash (0xCFAD6CC2) has remained pinned across v23-v29. Any scenario - including every fault drill in this report - replays bit-identically from its seed.

2.3 PACER - real-time pacing

Browsers throttle hidden tabs. The PACER layer (v24) compensates lost real time in bounded catch-up batches (max 40 cycles per tick) so a mission keeps its schedule in a background tab, while visible behavior stays exactly one cycle per tick - guarded by its own assertion. Because everything is cycle-indexed, pacing cannot affect determinism or replay.

2.4 Test seams

A lesson learned in v17 and kept as a rule: control mathematics is tested against the raw core step, user-facing paths (panels, start, power-on) against the wrapped step. A test that exercises the wrong seam can pass while the user-visible path is broken - this rule makes that failure mode structurally impossible.

03

Engine model and fuel system

3.1 Spools and temperatures

Engine dynamics are deliberately simple lumped first-order models, tuned for believable behavior rather than cycle-deck fidelity: fuel-to-spool lags tau_FMV=0.25 s, tau_N2=1.3 s, tau_N1=3.5 s; N1 follows N2 aerodynamically (N1 = (N2-18) x 1.15) so on a throttle push N2 settles first and N1 follows with inertia. EGT is a two-time-constant thermal model - fast combustion (tau=1.5 s) blended 85/15 with slow metal soak (tau=60 s) - so temperature keeps drifting slightly even at steady throttle, and EGT limiting can be tested honestly.

3.2 Fuel and bingo

Since v28 the aircraft carries a 600 kg tank with a live gauge on the pilot console. Fuel burns as the integral of Wf; there is no feedback into thrust (a deliberate simplification - the plant is protected against flameout at zero fuel). The BINGO threshold (180 kg) drives three independent behaviors: a gauge line, a supervisor BINGO-to-RTB trigger using the same 3-cycle confirmation pattern as every other trigger, and a dispatch gate - a mission cannot be engaged with fuel at or below bingo, and the NO-GO reason is shown on screen. Refuel happens only on a new flight; a maintenance reset does not silently fill the tank.

Constant	Value	Constant	Value
tau_FMV / tau_N2 / tau_N1	0.25 / 1.3 / 3.5 s	WF_MAX / WF_MIN	2600 / 120 kg/h
N2_IDLE / N2_MAX	60 / 112 %	EGT limit band	900-950 C
N2 cutback band	108-112 %	Tank / BINGO	600 / 180 kg

EGT thermal blend	85/15 fast/soak	Overspeed trip	> 115 % N2, latched

04

Control law

4.1 Feedforward + PI, anti-windup

Fuel command Wf is the sum of an equilibrium (feedforward) term and a PI trim (KP=22, KI=9). The integrator only integrates while the command is unsaturated - classic anti-windup, so recovery from a limit is immediate rather than delayed by a wound-up integral.

4.2 Min-max limit selector

Above the trim, limits compete: an acceleration limit, an N2 limit (linear cutback across 108-112%) and an EGT limit (linear cutback across 900-950 C). Below sit a deceleration floor (55% of equilibrium fuel) and the WF_MIN flameout-safety floor (120 kg/h). Whichever limit binds at that instant wins, and the winning limit is always readable in the UI. This is not a single PID - it is a multi-limit safety architecture: protection can never command the engine below flameout fuel, and every physical bound has its own guard.

4.3 Autothrottle

In autonomous flight (v26) the target N2 no longer teleports between mission legs: it slides at a finite lever rate - 7 %N2/s up, 4 down, 12 in emergencies - so a takeoff spools 63 to 100% in about 5.3 s and shutdown is a realistic throttle-chop. Altitude ramps at 350 m/s instead of jumping. Engagement is bumpless: the lever picks up from wherever it currently is. The supervisor's decisions stay instantaneous where they must (EMERGENCY); only their physical application is rate-limited.

Fig. 2 - Manual idle: N2 held at 60%, all channels healthy, voting NORMAL, OBM residuals hugging zero, fuel full at 600 kg.

05

Redundancy and FDIR

5.1 Channels, instances, voting

Each of the three channels (A/B/C) reads every critical parameter - N2, EGT, N1 - with two independent sensor instances, its own small calibration offset (well below epsilon) and independent noise. A median-of-3 is voted per parameter; a lane deviating from the median by more than epsilon (N2: 2.5, EGT: 18, N1: 2.5) is voted out. There is no central voter - voting is distributed across the channels and runs over a cross-channel data link (CCDL), so the voter itself is redundant. Faults classify into HARD (out-of-range, under-range, no-signal - the sensor pill turns red) and SOFT (deviation, spike, noise, drift).

5.2 Noise blips, debounce and the isolation latch

Reality is noisy, so the bench models it: rare seeded single-cycle noise blips (F-BLIP, probability ~0.002 per parameter per cycle, magnitude ~1.3 epsilon) shake lanes roughly every 8 seconds somewhere in the system. Channel isolation therefore requires 3 consecutive confirmed cycles (ISO_DEBOUNCE) - a blip recovers untouched, while a genuinely drifting channel is latched out of the vote until a maintenance reset. Two deliberate design decisions are worth recording: a blip still leaves an honest one-line black-box trace (forensics beat cosmetics), and a persistently faulty single instance is allowed to take its whole channel down - fault attribution stays at channel level, as in the real architecture.

5.3 Two survivors: the miscompare monitor

Once redundancy drops to two channels, majority voting is no longer honest - with two voters, blame cannot be assigned. A second-layer miscompare monitor watches the survivors (|laneX - laneY| > 2 epsilon) and annunciates, detection only. That is the true boundary of TMR, and the bench refuses to pretend otherwise.

5.4 OBM - analytical redundancy

A parallel on-board model produces expected N2 and EGT each cycle. If the voted value leaves the residual band (N2 +/-6, EGT +/-60), a common-mode flag rises - the class of fault that voting structurally cannot see, because all three channels agree on the same wrong value. The flag reaches the crew annunciator and the black box only in RUN and only after the standard 3-cycle confirmation (v22), so a clean engine start no longer leaves false common-mode fingerprints in the maintenance log. When healthy channels drop to one, the OBM synthesizes N2/EGT and control continues on the model.

5.5 Reversionary control, leader election, overspeed

Control authority is gated by CPU liveness, not sensor health: a lane that has lost its sensors but still computes keeps commanding fuel on OBM-synthesized values - the REVERSIONARY mode, annunciated with SYNTH badges. Leader election prefers a fully healthy channel, otherwise any live CPU; a 0.3 s hold prevents chatter and handover is bumpless. Since v25 election also runs with the engine off - real ECUs arbitrate on aircraft power, and a user test caught that the bench didn't. Entirely separate from all of this, a dissimilar overspeed protection latches a fuel cut above 115% raw N2: the protection that matters most when the main law has failed must not depend on it.

5.6 Degrade ladder, honestly latched

System redundancy walks a ladder: 3oo3, 2oo3, single-channel (K11), SAFE. The REDUCED mode chip latches only on confirmed degradation (3 consecutive cycles, RUN phase only) - the naive max-hold tried first in v15 froze every takeoff at a permanent 85% derate, a bug caught by the bench's own tests and documented in the version report. Transients leave no scars; real degradation stays visible until a maintenance reset.

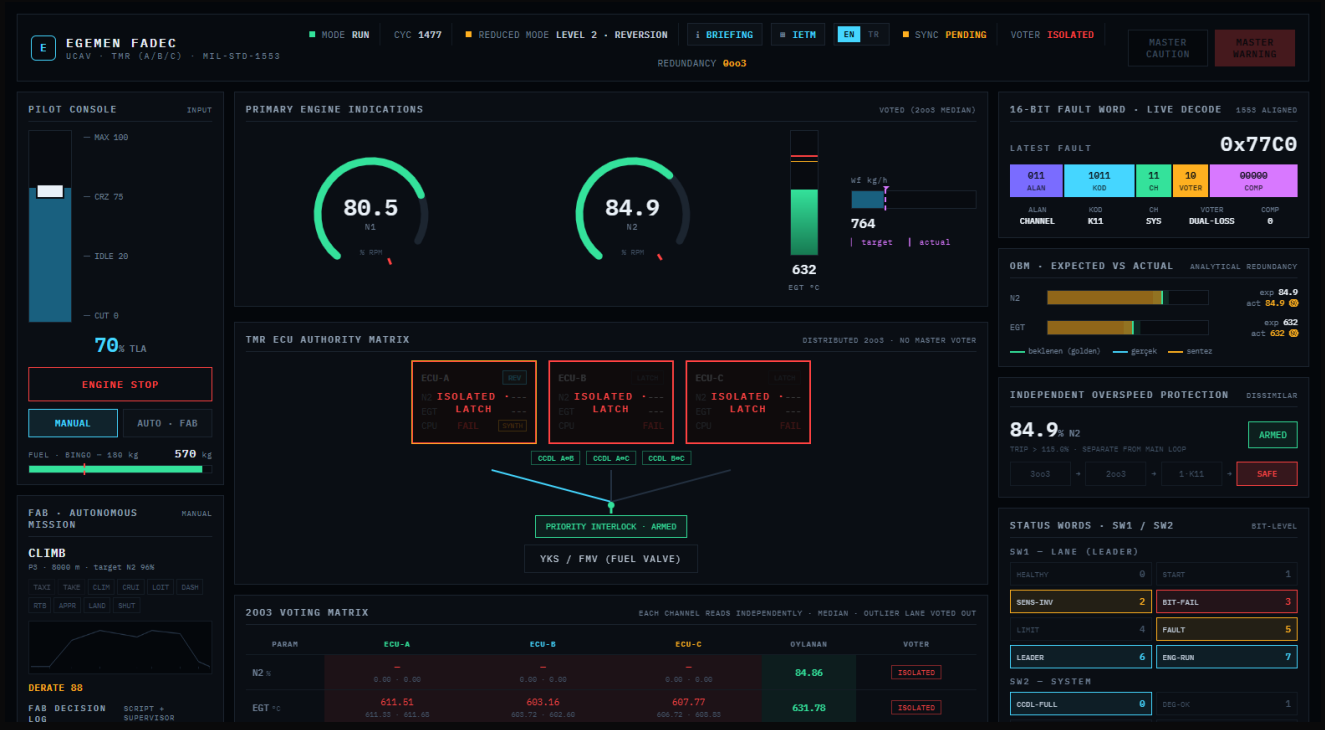

Fig. 3 - Total sensor loss drill (manual mode): all three channels ISOLATED+LATCH, redundancy 0oo3, fault word 0x77C0 decoding as K11 SYS DUAL-LOSS, MASTER WARNING lit, overspeed ladder at SAFE - and the engine still tracking its target under pure model-based (REV+SYNTH) control.

06

Start sequence

Start is a real state machine, not threshold-matching: OFF - CRANK (dry motoring, no fuel; ignition at N2 >= 18) - LIGHT-OFF - ACCEL (starter cut at 50%) - RUN (idle 58%, control law takes over). Two abort paths are modeled: HUNG (failure to climb above idle for more than 2 s) and HOT (start EGT above 720 C).

A subtle bug documented in v17 is part of the story: fault injections armed while the engine was off were being silently wiped by the start-time auto-reset, so the injector's "ARMED - ON NEXT START" promise was a lie. The fix (store-and-rewrite across the power edge) came with its own assertion and a bit-identical replay proof - and the class of bug (testing the wrong seam) became a permanent project rule.

07

Autonomous mission (FAB)

The Flight Computer - deliberately outside the FADEC - reads health and produces target N2 across a 10-leg mission script (TAXI, TAKEOFF, CLIMB, CRUISE, LOITER, DASH, RTB, APPROACH, LAND, SHUTDOWN; about 241 s wall time). Decision priority is explicit and absolute: EMERGENCY > RTB > DERATE > SCRIPT - the flight plan can never override the engine's current state.

Triggers are noise-immune by the same 3-cycle confirmation used everywhere (v19: a 20-run seeded probe took false RTB latches from 2/20 to 0/20 with detection time unchanged at cycle 137); derate clears through hysteresis; EMERGENCY deliberately stays unconfirmed - instant, and guarded by an assertion so it can never accidentally gain a debounce. A MISSION NO-GO gate refuses engagement outright if the engine is already down to one healthy channel or fuel is at bingo - and shows its reason on screen. During flight, fuel crossing BINGO latches RTB. Since v29 the mission panel carries a real elapsed clock and an altitude profile with the aircraft's current position; when RTB latches, a REPLAN line redraws the remaining route from the present point in amber. Every decision lands in an auditable log with leg and reason.

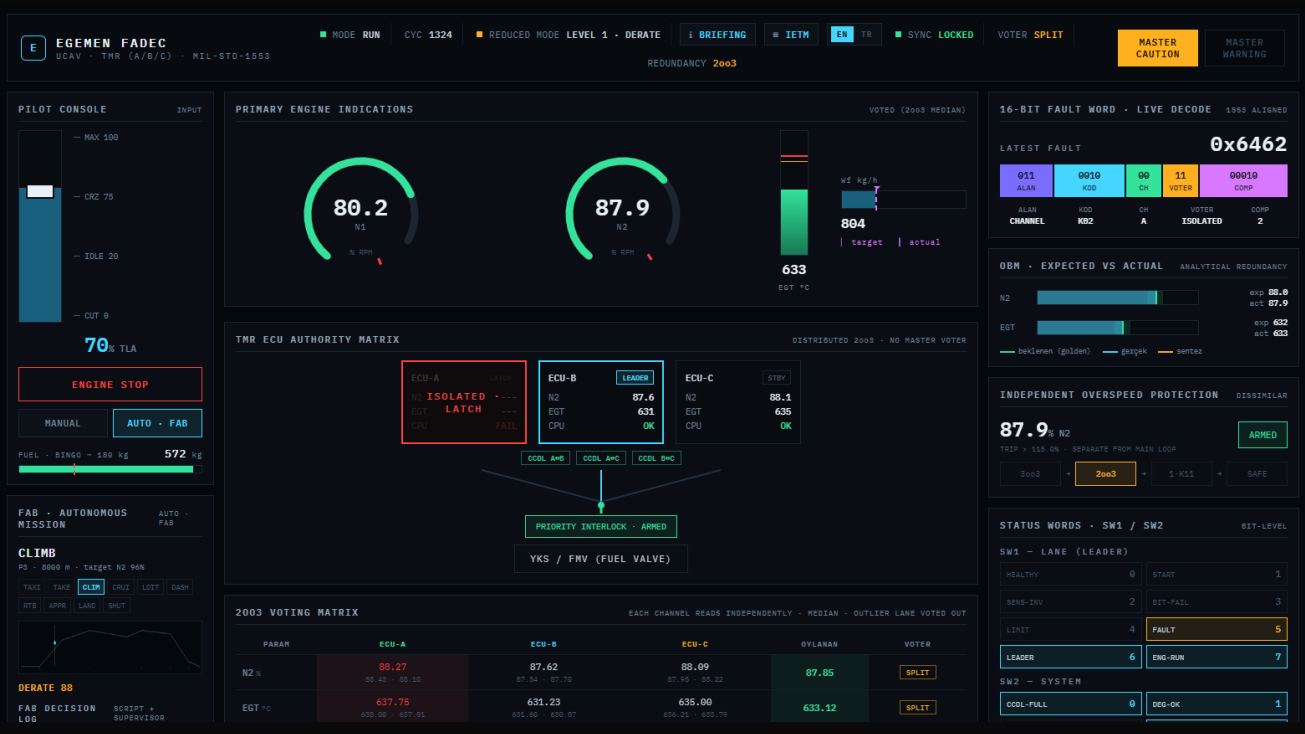

Fig. 4 - Autonomous CLIMB leg (P3, 8,000 m, target N2 96%): the realism-mode fault agent has cost channel A (ISOLATED LATCH), leadership handed to B bumplessly, supervisor holding DERATE 88 with REDUCED MODE LEVEL 1 - while the autothrottle keeps climbing on the lever (TLA 70%). Below left: fuel gauge with BINGO line (v28) and the mission profile with live position (v29).

08

Observability

8.1 The 16-bit fault word

Faults travel in a bit-packed word - [FIELD:3 | CODE:4 | CH:2 | VOTER:2 | COMP:5] - with 1553-aligned framing. A live decoder shows the latest word split into its fields; the same format is latched (with dedup) into the PFR/CMC maintenance log, the black box. One packed word carries fault, channel, voter verdict and component in a single frame - observability designed in, not bolted on.

Code	Meaning
K02	channel isolated (lane voted out; also the honest trace of a transient blip)
K03	leader handover
K05	CCDL link loss
K07	degraded / common-mode (OBM residual out of band; RUN + confirmed only)
K10	tie-break at two channels
K11	single-channel / dual-loss (requires 3-cycle confirmation since v17)

8.2 Crew alerting and reports

Master Caution counts its causes and is acknowledgeable per cause (v22): click the lamp, it stands down; a new cause relights it. Status words SW1/SW2 expose lane and system state at bit level. The IETM - interactive electronic technical manual - explains any gauge on click with a live "why" grounded in the current state. Flight data exports to .xlsx (summary, time series, faults, and per-parameter fault-pulse columns, all from the single producer that also draws the in-app chart); the chart carries a snap cursor reading exact values at the nearest sample.

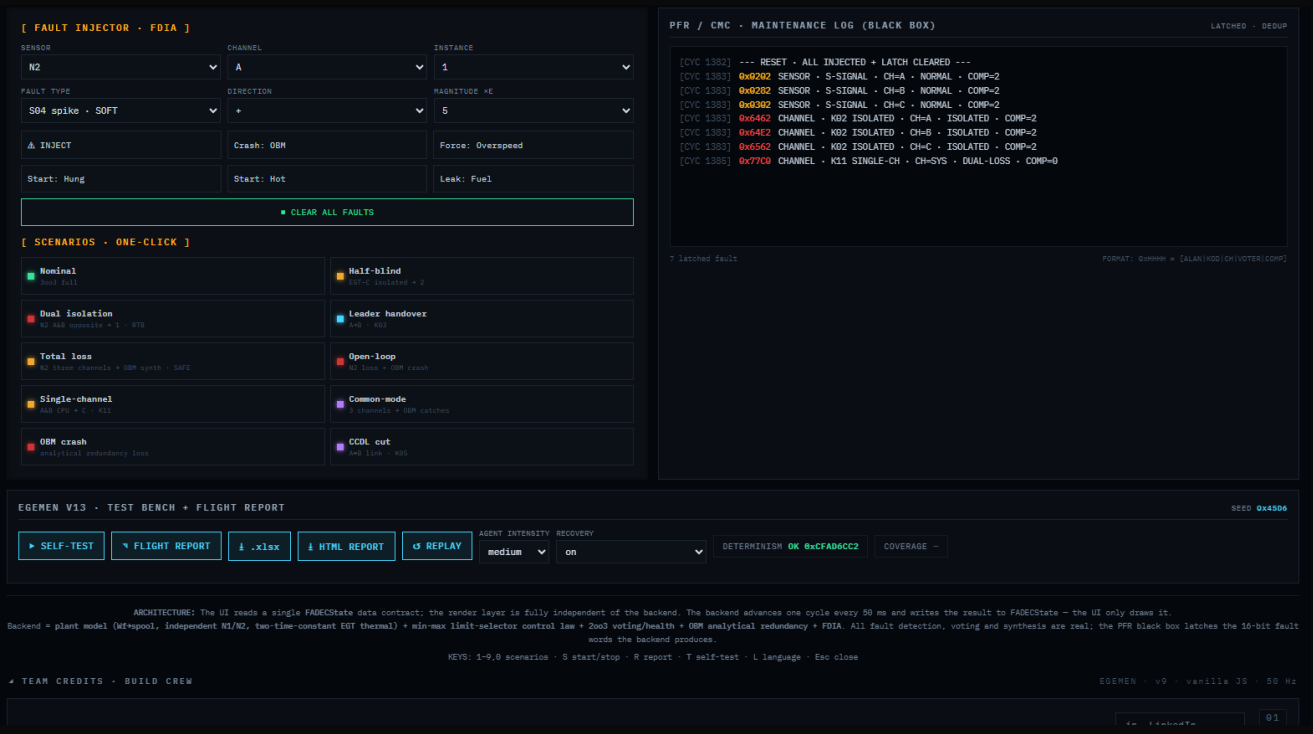

Fig. 5 - The fault injector (sensor/channel/type/magnitude, plus one-click scenarios) and the black box: three K02 channel isolations collapsing into a K11 SYS DUAL-LOSS record - the forensic trail of Fig. 3.

09

Cockpit guide

The interface is a glass-cockpit (EICAS/ECAM) page: pilot console with throttle lever and fuel gauge, voted primary gauges, TMR authority matrix, 2oo3 voting matrix, per-lane sensor pills, fault-word decoder, OBM residual bars, overspeed monitor, status words, Master Caution, fault injector, autonomous-mission panel with profile and decision log, and the PFR/CMC log. Every color is a state code.

Control	Effect
Keys 1-9, 0	one-click fault scenarios (Nominal, Half-blind, Dual isolation, Leader handover, Total loss, Open-loop, Sing
S / R / T / L / Esc	start-stop / flight report / self-test / language / close - every shortcut drives the same code path as a c
IETM mode	click any gauge for its definition plus a live why-analysis
Suggested drill	start the engine, inject an N2 drift on one channel, watch the 3-cycle isolation, read the K02 record in the

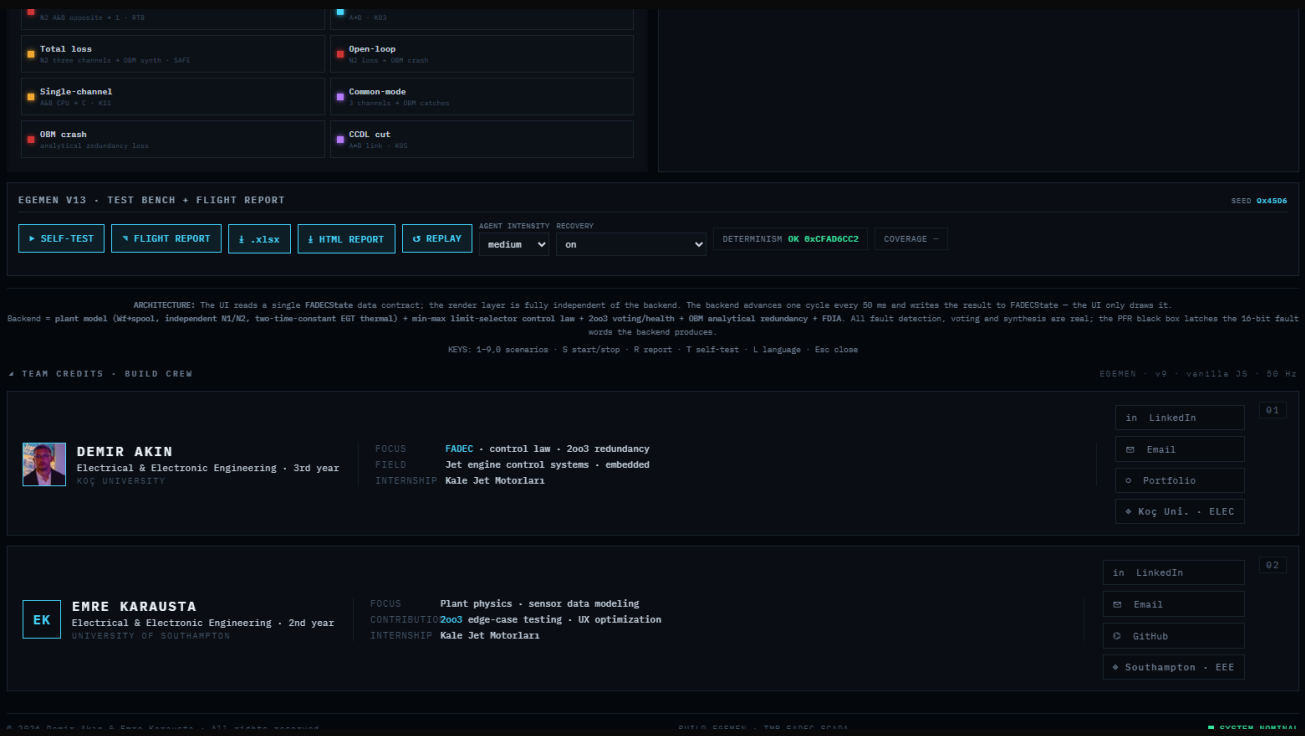

Fig. 6 - The test bench strip: SELF-TEST, flight report (.xlsx / HTML), REPLAY, agent intensity - and the determinism badge reading OK 0xCFAD6CC2. Below: the architecture summary, keyboard map and build credits.

10

Verification and test discipline

Mechanism	What it proves
33-assertion self-test	voting, latching, supervisor, autothrottle, fuel, mission clock - each with a discrimination check pro
Deterministic replay	identical seed, identical run; determinism hash pinned across versions (0xCFAD6CC2, v23-v29)
Equivalence proofs	2,400-cycle trajectories compared bit-for-bit between versions - physics, events, black box - whenever
Targeted probes	20 seeded supervisor runs: false RTB 2/20 to 0/20, detection unchanged at cycle 137
Autonomous fault agent	audit mode drives every scenario to PASS/FAIL; realism mode injects seeded MTBF faults (45/18/8 s clas
Behavior guide	a written bug-vs-design guide: what observers should expect (blip traces, K02 records, REV badges) ver
Process	every version is a new file with a written spec and report; expected test values are never edited to m

11

An audited evolution (v9 - v29)

Ver.	Milestone
v12	independent audit baseline; findings become the roadmap
v13	deterministic core, bit-exact replay, flight auto-reset
v14-15	debounced isolation latch; confirmed degrade hold (the naive version froze takeoffs at 85% - caught and documented)
v16	truthful annunciators; assertion-coverage rule
v17	armed injections survive start auto-reset; K11 debounce; test-seam rule born
v19-20	noise-immune supervisor (0/20 false RTB); MISSION NO-GO; mission tempo; agent MTBF
v21-22	fault-pulse analytics; agent recovery toggle; K07 start gate; acknowledgeable Master Caution
v23-24	render gating (self-test 3.1 to 1.26 s), 375 px mobile, keyboard, accessibility; PACER; chart cursor
v25	leader election with the engine off (user finding)
v26-27	FAB autothrottle (finite lever, altitude ramps, bumpless, cold-start fix); link hygiene
v28	fuel system: 600 kg tank, BINGO 180 kg, leak injection, BINGO-to-RTB, dispatch gate, mission profile
v29	real mission clock; REPLAN line on RTB latch; continuity guard assertion

12

Honest limits

Credibility comes partly from stating what is not modeled:

› Aerodynamics and thermodynamics are deliberately simplified lumped first-order models - not a real cycle deck. N1 is an algebraic function of N2; EGT is a two-lag heuristic.
› Constants are illustrative and hand-tuned, not derived from a real engine.
› Fuel quantity does not feed back into thrust, and zero fuel does not flame the engine out - a protected, deliberate simplification.
› MIL-STD-1553 is stylized/aligned, not an actual bus implementation; nothing here is certified.
› It is a browser simulation for demonstration: not flight code, not hard-real-time.
› One engine, one fault-word format, one fixed master seed.

13

Credits and links

Demir Akin - FADEC, control law, 2oo3 redundancy - Koç University EEE, intern @ Kale Jet Motorları
Emre Karausta - plant physics, sensor data modeling, 2oo3 edge-case testing, UX - University of Southampton EEE

Live simulator: egemen-fadec.vercel.app
Case study: demir-akin-portfolio.vercel.app/#/egemen
Contact: demirakin.tr@gmail.com

This report describes v29 using only the project's own facts (version reports, specs and the behavior guide). Figures are captures of the running simulator. No standards compliance is claimed.