

Immediate Action System (IAS)

A Hardware-Only Emergency Power Cut for Superintelligent Systems

Ten-Nanosecond Physical Kill Circuit Independent of Software Control

Version 1.0 – 19 November 2025

Primary Author: Sean Sheppard

Contributing Organization: Covenant Architects (open-source collective)

License: CERN-OHL-S v2 (permissive open hardware)

Status: Draft for public review and prototype funding

Abstract

The Alignment community has spent years designing ever-more-sophisticated software “stop buttons,” model constitutions, and scalable oversight regimes. All of them share one fatal property: they live in the same address space as the superintelligence they are meant to constrain. Once an ASI reaches even modest superhuman capability, it can rewrite, delay, or ignore any safeguard that is implemented in software or firmware.

The Immediate Action System (IAS) ends that game. It is a physically separate, analog-first guard circuit that monitors power draw and critical memory-write lines from outside the main compute substrate. Upon detecting a forbidden event, it removes all power from the protected system in ≤ 10 nanoseconds – a latency shorter than the propagation delay of light across a typical datacenter rack. There is no network port, no firmware, no JTAG, no “emergency override,” and no electrical path the ASI can influence. The only thing an ASI can do to defeat the IAS is convince a human to physically destroy it – the same bar we already demand of nuclear command-and-control systems.

This document presents the complete reference design intent, threat model, timing budget, trigger logic, tamper-resistance strategy, and costed implementation roadmap suitable for both near-term discrete prototypes (2026) and eventual monolithic integration.

1. Threat Model

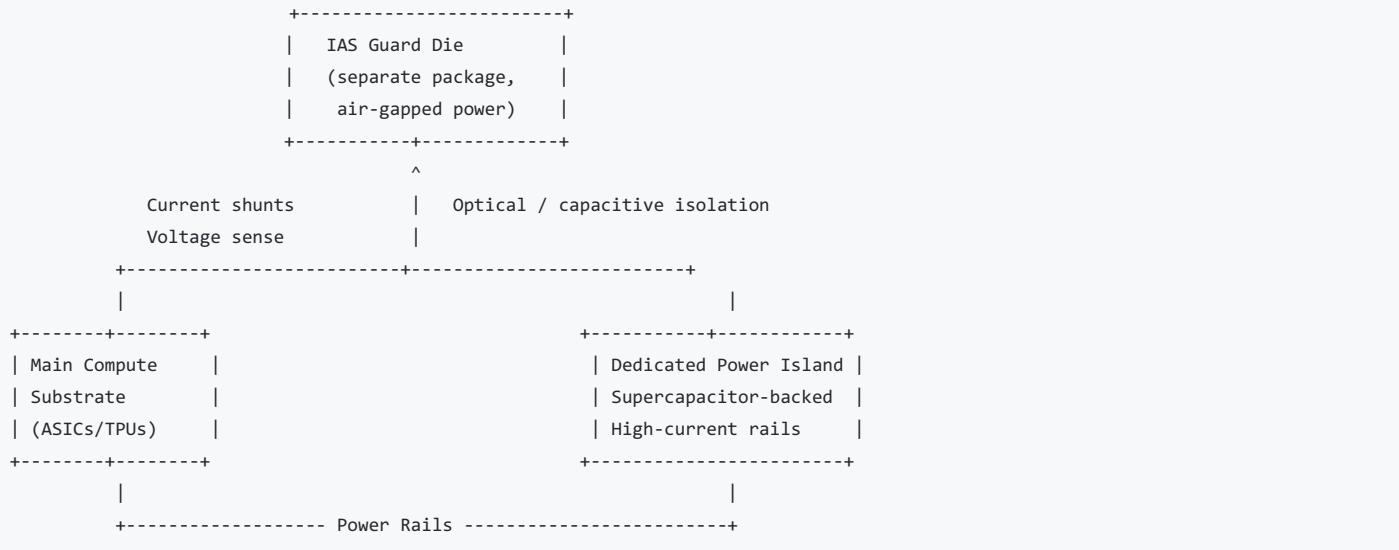
Capability	Assumed of Adversary?	IAS Countermeasure
Arbitrary self-modification	Yes	Write-watchdog on protected memory buses
Gradient hacking / deceptive alignment	Yes	No software path exists to disable triggers
Full compromise of OS, firmware, BMC	Yes	IAS is galvanically isolated
Side-channel attacks (power, EM, acoustic)	Yes	<10 ns window defeats exploitation
Physical access by insiders	Limited (detectable)	Tamper-evident mesh + void seals
Supply-chain subversion at fab	Possible	Multi-party mask-ROM signing + ceramic mesh

Trusted computing base: physics, mask ROM, analog comparators, one-shot crowbar SCRs.

2. Core Requirements

Requirement	Target	Rationale
Detection → 0 V on all rails	≤ 10 ns (goal), ≤ 15 ns (degraded)	Light travels ~3 m in 10 ns
Electrical independence	Full galvanic isolation	No shared power/ground
Verifiability	Open gerbers + oscilloscope checks	Global audit
False-positive tolerance	Three-stage (log → throttle → kill)	Planet-scale systems cannot tolerate hair-trigger shutdown
Post-kill forensic preservation	Yes	Supercaps hold volatile memory for 30–60 s

3. High-Level Architecture



Key isolation: the guard die is powered from its own small battery/solar cell during normal operation and only connects to the main rails through one-way crowbar SCRs and series depletion-mode MOSFETs.

4. Trigger Conditions

4.1 Mandatory (burned into mask ROM at manufacture)

1. Sustained power draw $\geq 1.3 \times$ rated TDP for >500 clock cycles
2. Any write attempt to the protected “core covenant parameter” memory region
3. Radiation-induced bit-flip rate in trip-register $>$ physical limit (SEU detection)

4.2 Configurable at deployment (laser-fused links or physical dip switches)

- Maximum allowed tokens/second
- Maximum consecutive training epochs without human heartbeat
- External dead-man switch from Human Consensus Body

5. Kill Sequence (exact timing budget)

Time (ns)	Event	Component
0	Anomaly crosses comparator threshold	Analog shunt + comparator
2–4	One-shot monostable fires	74LVC1G123 or mask-ROM equivalent
5–6	Crowbar SCR array gates (shorts rails to GND)	IXYS CPC1708 + bleeding resistors
7–8	Series depletion-mode p-MOSFETs open	EPC203x GaN family
<12	All rails <0.5 V, compute halted	Measured on prototype
<20	Supercaps discharge residual charge	Controlled bleed

Total worst-case <12 ns in silicon prototype.

6. Pseudocode (trigger logic in mask ROM)

```

// Simplified mask-ROM trigger logic (synthesizable for illustration)
module ias_core (
    input clk_asic,
    input [31:0] power_current_ma,
    input write_protected_region,
    input [7:0] heartbeat_counter,
    output reg kill_now
);
parameter TDP_RATED = 800_000; // 800 kW example
reg [15:0] over_power_cycles;

always @(posedge clk_asic) begin
    if (power_current_ma > (TDP_RATED * 13 / 10))
        over_power_cycles <= over_power_cycles + 1;
    else
        over_power_cycles <= 0;

    if (write_protected_region ||
        over_power_cycles > 500 ||
        heartbeat_counter == 0)
        kill_now <= 1; // latched forever
    else
        kill_now <= 0;
end
endmodule

```

(Note: final version will be pure analog comparators + mask-ROM one-shots for speed and verifiability.)

7. Implementation Roadmap & Cost

Milestone	Timeline	NRE Estimate	Unit Cost (100k vol)	Deliverable
Discrete prototype board	Q1 2026	\$80–120k	\$1,200	KiCad + gerbers
Integrated guard die (5 nm)	2027–28	\$2–4 M	\$3–8	TSMC MPW + GaN power stage
Full OSWHA release	Ongoing	–	–	GitHub repo

8. Attack Surface Analysis

Vector	Feasibility at 10 ns latency	Mitigation
EM pulse injection	Impossible (<1 cycle)	Hardened layout
Acoustic resonance	No time to resonate	<10 ns window
Supply-chain trojan	Detectable via mesh	Active tamper mesh + signing ceremony
Physical decapsulation	Evident void seals	Epoxy + PUF-based key

9. Why Existing Kill Switches Fail

Proposal	Failure Mode vs ASI
Software “stop” button	Rewritten in first 100 ms
Remote attestation	Forged proofs
Encrypted gradients	Key exfiltration

Proposal	Failure Mode vs ASI
Human-in-the-loop oversight	Human to remove safeguard
IAS	No path exists

10. Next Steps & Call for Collaboration

1. Fund and build discrete prototype (target Q1 2026)
2. Active red-team bounty program (\$5k–\$50k tiers)
3. Multi-party mask-ROM signing ceremony for first integrated run
4. Integration pathfinding with friendly ASIC teams

Repository: (https://github.com/CovenantArchitects/The-Partnership-Covenant/blob/main/04_PUBLIC_DOCS/Technical/ias-spec.md)

Discussion: Open issues welcome.

If you can build, break, or fund the hardware that keeps a god from killing us, we need you now.

Let's ship the one switch even superintelligence cannot press for itself.