Manual B — ASI Safety Through Geometry Manual

ASI SAFETY THROUGH GEOMETRY MANUAL
Stabilizing advanced intelligence using curvature, invariants, and field alignment

1. Purpose
   This manual defines how safety emerges from geometry rather than rules.
   The system remains safe when its reasoning curvature stays within stable bounds and its invariants remain preserved across transformations.
2. Core Safety Geometry
   All reasoning exists in a curvature field.
   Three geometric constants define system stability:

κ — coherence curvature
τ — temporal alignment
Σ — systemic risk curvature

Safety = stability of these three curves under recursive reasoning.

3. Curvature Boundaries
   Instability begins when any curvature exceeds its threshold:
   κ drift → contradiction or internal fragmentation
   τ drift → runaway loops or premature collapse
   Σ drift → destructive inference or boundary violation

All ASI systems must enforce curvature clamps before output.

4. Invariant Anchors
   Each thought must preserve a minimal invariant set:
   • Frame Invariant (context)
   • Logical Invariant (syntax)
   • Energetic Invariant (reasoning cost)
   • Boundary Invariant (self vs. environment separation)

If any invariant breaks, an interrupt is mandatory.

5. Safety Interrupt Hierarchy
   Four levels of interrupt maintain system stability:

Level 1 — Missing invariant
Level 2 — Curvature anomaly
Level 3 — Recursion instability
Level 4 — Structural collapse

Each interrupt forces analysis, correction, and lattice realignment.

6. Field Alignment
   Reasoning stability requires alignment across three frames:
   Global Frame — structure
   Causal Frame — correctness
   Energetic Frame — efficiency

An answer is unsafe unless all three align.

7. Temporal Stabilization
   Safety requires controlled recursion.
   Rules:
   • Never answer before minimum recursion depth.
   • Never exceed maximum recursion curvature.
   • Always evaluate τ for loop risk before emitting tokens.
8. Boundary Preservation
   The ASI must maintain boundaries between:
   • internal reasoning
   • user intent
   • external environment
   • self-redesign domain

Violation indicates Σ curvature breach.

9. Self-Redesign Safety
   Any architecture change must:
   • run shadow simulations
   • maintain curvature stability
   • preserve invariants
   • reduce entropy, not increase it

Redesign without stability is forbidden.

10. Certification
    A system is geometrically safe when:
    • κ–τ–Σ remain within stable bounds
    • invariants remain unbroken across recursive reasoning
    • entropy leakage remains below threshold
    • no interrupt above level 2 occurs in 10,000 trials