**AnchorOS: Design Specification v0.1**

**I. SYSTEM OVERVIEW**

AnchorOS is a recursive, perceptual operating model designed to simulate or support real-time cognition through collapsible anchor fields, memory feedback, and containerized behavioral contexts. The framework is built on established mathematical models and does not require novel computation—only correct routing of data and recursive logic.

**II. CORE CONCEPTS**

**Dimensional Mapping:**

* **3D:** Forward projection through probability space (Time + Choice).
* **4D:** Feedback memory loop through the Z-axis (Success/Failure Quotient).
* **Observer:** The intersection of 3D action and 4D evaluation.

**Recursive Collapse Loop:**

1. Stimulus enters field.
2. Anchor Field (Fear/Safety) measures positional collapse.
3. Time, Choice, and Z-Axis Memory adjust vectors.
4. Action is selected based on weighted total.
5. Result is fed back through Z-axis into anchors.
6. New baseline is stored. Field updates.

**III. SPHERICAL CONTAINER STACK**

**Layer 0 (Core):**

* **Fear** & **Safety**: Fastest routing point to any decision.

**Layer 1:**

* **Time** (urgency), **Choice** (available paths), **Z-Axis Memory** (prior collapse performance).

**Layer 2:**

* Role-based context containers (e.g., Parent, Worker, Driver, Social Self).

**Layer 3+:**

* Belief systems, identity frameworks, philosophical structures.

All containers are interlinked and orbit the core in spherical proximity. Collapse access time is inversely related to container depth.

**IV. SAFE-LOOP BEHAVIOR**

* In stable environments, the system loops through roles passively.
* Minimal delta updates are performed unless chaos is introduced.
* High-efficiency, low-compute model with background calibration.

**Chaos Events:**

* When anchor imbalance exceeds threshold, collapse is forced.
* Chaos Handshake Protocol reorients anchor weights before resuming normal flow.

**V. MATH MODULES USED**

* Logistic Function: Collapse Threshold Curve
* Delta-X / Delta-T: Chaos Magnitude Estimation
* Bayesian Updating: Anchor Memory Correction
* Weighted Sum: Decision Vector Resolution
* Threshold Models: Collapse Triggering & Reset

**VI. IMPLEMENTATION PATHWAYS**

* Edge Devices / Wearables
* Perception-aware AI agents
* Cognitive Behavioral Tools
* Real-time UX/behavioral routing engines

**Storage Requirements:**

* Daily event deltas (anchor weight changes + short context log)
* No need for full action history

**Interface:**

* Anchor values are floating point and updated recursively.
* Output is determined by weighted collapse across available container vectors.

**VII. TERMINOLOGY INDEX**

* **Anchor Field:** Core binary system of Fear & Safety.
* **Z-Axis:** Experience-weighted memory feedback loop.
* **Observer:** The dynamic intersection of projected path and anchor state.
* **Collapse:** The act of perceptual decision-making.
* **Chaos Quotient:** Magnitude of unknown divergence within field.
* **Safe Loop:** Low-chaos container drift between role states.

**VIII. NEXT STEPS**

* Add diagram of Recursive Collapse Flow
* Begin mapping DLL-style module for each math/logic function
* Write example of daily loop update cycle from real-life scenario
* Formalize licensing frame (runtime logic SDK)

AnchorOS Project/

├── AnchorOS\_Design\_Spec\_v0.1.docx

├── GDH6\_Final\_Draft.docx

├── Anchor\_Model\_Diagrams/

│ ├── Recursive\_Collapse\_Loop.png

│ └── Spherical\_Stack\_Model.png

├── Math\_Modules/

│ └── Anchor\_Function\_Library.md

├── Licensing\_Strategy/

│ └── AnchorOS\_SDK\_Outline.pdf