Deck Calculus — Reader Card (One-Page Cheat Sheet)

Principle

One 52-card deck encodes any finite state by (1) card identity (rank×suit), (2) stack order (top→bottom), (3) orientation (face-up/down), (4) piles (partitions), (5) visible cuts. Two differently colored decks encode any finite interaction by align/zip/overlay—no new moves, only labels.

Canonical Dictionary (example)

Rank: A=0,…,K=12 • Suits: ♣=control, ♠=data, ♦=meta, ♥=guard • Red/Black parity: invariant lane • Position: index • Face-up=1 (active), Face-down=0 (dormant). Publish once in the ledger.

Universal Moves → Abstract Effects

Cut: rotate/split • Reverse: mirror/pal test • Riffle: stable merge • Overhand: block move • Deal into n piles: hashing/bucketing • Collect piles: concatenate • Flip: bit toggle • Transpose piles: matrix transpose • Zip two decks: function graph • Overlay: OR/XOR by rule.

Single-Deck State (readback)

1) Read position-indexed cards (sequence). 2) Read face (bit). 3) Read suit/rank (type/value). 4) Read piles (partitions). 5) Respect cuts (sectioning). Output: deterministic state vector.

Two-Deck Interaction (readback)

Align Blue (domain) with Red (codomain). Row i encodes pair (x_i, f(x_i)). For relations, lay Blue across, Red down; a card at (i,j) marks (x_i,y_j)∈R. Transitions: Blue=t, Red=t+1.

4-Bit Commit

P: reverse preserves pattern? M: face flip + suit swap preserves rule? Δ: last change was local? S: declared bound tightened? + 4-nybble hash from visible top cards. Record in ledger.

Eight Standard Views (coverage)

1) By suit • 2) By rank class • 3) By parity • 4) By position mod 4 • 5) Pile histogram • 6) Face-up banding • 7) Reversed • 8) Mirrored.

Start

Pick dictionary → Encode state → Encode mapping (two decks) → Run 8-view audit → Compute 4-bit → Photograph & ledger.