

SHD-CCP STANDARD: "EINSTEIN TILE" CALIBRATION SHEET

Protocol Version: 1.0 (H100-Native) **Target Architecture:** NVIDIA Hopper (SM90) **Topology:** 8x8 Bit-Field (64-bit) **Date:** November 2025

1. Executive Summary

The "**Einstein Tile**" is the standardized 2D bit-topology for the Spin-Half Dirac Compressed-Couple Packet (SHD-CCP). It is engineered to resolve the memory-bound bottlenecks inherent in aperiodic tiling simulations on high-performance GPU architectures.

By strictly segregating **Symbolic Control Data** to the packet edges (Halo Regions) and **Neural Payload Data** to the center, this layout enables NVIDIA H100 GPUs to utilize **Distributed Shared Memory (DSMEM)** for zero-latency neighbor synchronization while simultaneously leveraging the **Tensor Memory Accelerator (TMA)** for asynchronous bulk loading of internal state.

2. The 8x8 Grid Map

The following matrix represents the physical layout of the 64 bits in memory. This layout is non-negotiable for the "Einstein Tile" standard.

- **X-Axis:** Bits 0-7 (Columns)
- **Y-Axis:** Bits 0-7 (Rows)

Visual Layout

[NORTH HALO - DSMEM HOT ZONE]							
	C0	C1	C2	C3	C4	C5	C6 C7
R0	[SF SF SF SF P SP SP SP]						<-- Control Lane A (High Priority)
R1	[PL PL PL PL PL PL PL PL]						<-- Scaling Buffer
R2	[Q Q Q Q Q Q Q Q]						
R3	[Q Q Q Q Q Q Q Q]						<-- QUATERNION
R4	[Q Q Q Q Q Q Q Q]						CORE (TMA Block)
R5	[Q Q Q Q Q Q Q Q]						
R6	[PL PL PL PL PL PL PL PL]						<-- Scaling Buffer
R7	[F F F F F A A A]						<-- Control Lane B (Scheduling)
[SOUTH HALO - DSMEM HOT ZONE]							

Legend & Bit Allocation

Abbr	Field Name	Size	Color Code	Optimization Target
SF	Structural Form ID	4 bits	Violet	Priority 1: Edge stability & Tiling Logic. Determines geometric compatibility.
P	Parity Bit	1 bit	Gray	Priority 2: Error checking & Volatility. Introduces entropy.
SP	Spin Class ID	3 bits	Red	Dynamic update rules (rotation/behavior).
Q	Compressed Quat.	32 bits	Pink	TMA Target: Asynchronous Bulk Load. The neural payload.
PL	Payload Scaling	16 bits	Emerald	Buffer / Numerical Stability for FP8 operations.
F	Frequency ID	5 bits	Blue	Scheduling / Thread Priority hints.
A	Amplitude ID	3 bits	Amber	Magnitude / Diffusion constants.

3. Optimization Reasoning (H100 Architecture)

A. The "Sandwich" Protocol (DSMEM Optimization)

- **Feature:** NVIDIA H100 Distributed Shared Memory (DSMEM).
- **Constraint:** Crossing the boundary between Thread Blocks (Cores) is expensive if accessing Global Memory, but extremely fast (7x faster) if accessing the "Halo" via DSMEM.
- **Design:**
 - **Row 0 (North Edge)** holds the **Structural Form ID**. This is the primary key for aperiodic tessellation.
 - **Row 7 (South Edge)** holds **Frequency** and **Amplitude**, determining energy transfer between tiles.
- **Benefit:** The simulation can perform "Pre-Flight Checks" (neighbor compatibility) by reading *only* the top/bottom rows via DSMEM, without loading the heavy 32-bit Quaternion.

B. The Contiguous Core (TMA Optimization)

- **Feature:** Tensor Memory Accelerator (TMA).
- **Constraint:** TMA efficiency peaks when loading large, contiguous, aligned blocks of memory.

- **Design:**
 - Rows 2-5 contain the **32-bit Compressed Quaternion** as a solid 4x8 block.
 - **Benefit:** A single TMA descriptor can load this 32-byte region asynchronously. The compute thread issues the load command and immediately processes the light-weight Halo data (Form ID) while the TMA fetches the heavy neural data in the background. This effectively **hides the memory latency**.

C. The Buffer Zones (L2 Cache Optimization)

- **Design:** Rows 1 and 6 contain the **Payload Scaling Factor**.
- **Benefit:** These act as padding between the volatile Control Lanes and the stable Quaternion Core. If memory access strides are misaligned, fetching a Control Lane might accidentally pull in Row 1. Since Row 1 is merely integer scaling data, it minimizes the penalty of partial loads compared to the floating-point Quaternion.

4. Simulation Rules (The Einstein Protocol)

To successfully identify an "Einstein Tile" (a stable, non-repeating pattern), the simulation engine must strictly adhere to the following field-weighted transmission method.

Base Rule: Cellular Automata (B3/S23 - Born on 3, Survive on 2 or 3).

Field Weights (The "Stickiness" Factor): The simulation must apply weights to neighbors based on their field type (using DPX instructions on hardware):

1. **Structural Form ID (Weight: +2.0):**
 - These bits are "**Sticky.**" They represent the fundamental geometry. They are resistant to flipping, acting as the anchor for the aperiodic structure.
2. **Parity (Weight: +1.5):**
 - This bit is "**Volatile.**" It introduces necessary entropy to prevent the system from freezing into a simple periodic crystal.
3. **All Other Bits (Weight: +1.0):**
 - Standard interaction.

Calibration Goal: The system is considered "Calibrated" when:

1. The **Form ID** bits on the North Edge lock with the **Frequency** bits on the South Edge of the neighbor.
2. The **Quaternion Core** remains internally stable (survives).
3. The pattern does **not** repeat globally across the 32-core warp but maintains local connectivity.

5. Replication Steps (For Implementation)

To replicate this standard in code or hardware simulation:

1. **Initialize** a `Uint8Array(64)` (or equivalent 64-bit structure).
2. **Map Row 0 (Indices 0-7):**
 - Bits 0-3: Structural Form ID
 - Bit 4: Parity
 - Bits 5-7: Spin Class ID
3. **Map Row 7 (Indices 56-63):**
 - Bits 0-4: Frequency ID
 - Bits 5-7: Amplitude ID
4. **Map Rows 2-5 (Indices 16-47):**
 - Contiguous Compressed Quaternion (32 bits).
5. **Map Rows 1 & 6 (Indices 8-15, 48-55):**
 - Payload Scaling Factor.
6. **Execute** the simulation using the weighted neighbor scores defined in Section 4.

Authorized by: SHD-CCP Architecture Group