Formal Model and Notation for EchoPulse Key Encapsulation Mechanism (KEM)

This document formally defines the mathematical structures, operational procedures, and security game syntax for the EchoPulse Key Encapsulation Mechanism (KEM). It serves as a foundational reference for all subsequent security proofs and simulations, particularly for the IND-CCA2 reduction in the Random Oracle Model (ROM).

1. System Parameters

EchoPulse operates over a set of publicly defined parameters. These parameters are assumed to be globally known and securely established prior to any key encapsulation or decapsulation operations.

Symbol Alphabet (Σ): A finite set of $28=256$ distinct abstract byte-symbols. Each symbol $s \in \Sigma$ is represented as an 8-bit unsigned integer.
State Space (V): A finite set of $210=1024$ distinct states. Each state $v \in V$ is represented as a 10-bit unsigned integer.
Key Space (K): The set of all possible session keys, typically $\{0,1\}256$.
Payload Space (R): The set of all possible symbolic payloads, typically $\{0,1\}256$.
Ciphertext Space (C): The set of all possible ciphertexts generated by EchoPulse. A ciphertext ct is a tuple (venc,symbol_sequence), where $venc \in V$ and symbol_sequence is a list of symbols from Σ.
Final Key Length (LK): The length of the derived session key, $LK=256$ bits.

2. Symbolic Spaces and Functions

The core of EchoPulse's mechanics resides in its graph structure and dynamic functions.

Graph ($G(V,E)$): A directed graph where V is the set of states and $E \subseteq V \times V$ is the set of edges. The edges are defined by the transition function.
Transition Function (δ): A deterministic function $\delta: V \times \Sigma \to V$. For a given current state $v \in V$ and an input symbol $s \in \Sigma$, $\delta(v,s)$ uniquely determines the next state $v' \in V$. This function implicitly defines the edges E of the graph.
Mutation Function (μ): A deterministic function $\mu: N \to \text{Transform}(G)$. This function describes how the graph G (specifically, its transition function δ) changes over discrete time steps $t \in N$.
Let Gt denote the graph at time t. Then $Gt+1=\mu(Gt)$, meaning δ might change its mapping over time based on μ. The mutation schedule is assumed to be public and pre-defined.
Hash Function (H): A cryptographic hash function $H:\{0,1\}* \to \{0,1\}LK$ (e.g., SHA3-256). In security proofs, H is modeled as a Random Oracle.
For any input x, $H(x)$ is a random value if x has not been queried before.
For repeated queries x, $H(x)$ returns the same previously returned value.
The specific input format for key derivation is $venc \| r$, where venc is the final state and r is the encapsulated random payload.

3. Oracle Definitions

The security games are defined with respect to specific oracles available to the adversary.

Hash Oracle ($OH(x)$):
Input: $x \in \{0,1\}*$
Output: $y \in \{0,1\}LK$
avior: C maintains a list $Hqueries=\{(xi,yi)\}$. If x is in Hqueries (i.e., $x=xj$ for some j),