

## R\_open: Rift Stage-3 Hex Encoder & Resolution Trident

//! SemVerX PolyGatic Registry & Runtime //! Implements: Extended semantic versioning (major.minor.patch(channel)) //! Resolution: Tri-node BiDAG (X:Upload, Y:Runtime, Z:Backup)

### 1. Overview

R\_open is a multi-language implementation of the **Rift Stage-3 Hex Encoder**. It facilitates the translation of 1-D binary signals into 3-D duplex streams using a **2→1 sparse duplex** strategy.

#### The Resolution Trident

The system operates on three primary modules of resolution:

- Tri-node BiDAG:** Manages state across Upload (X), Runtime (Y), and Backup (Z) nodes using Eulerian and Hamiltonian strategies.
- FilterFlash Coherence Gating:** Ensures signal integrity during the duplex transformation.
- Observer-Mediated Recovery:** A rate-limited pattern (5-10 updates/sec) that monitors the Red-Black AVL tree for fault recovery.

### 2. Technical Specs

Feature	Implementation
<b>Encoding</b>	2→1 Sparse Duplex (A/B Polarity)
<b>Pruning</b>	Red-Black AVL ( $\frac{1}{2} \log n$ auxiliary space)
<b>Coherence</b>	FilterFlash Functor Gating
<b>Versioning</b>	SemVerX (Major.Minor.Patch.Channel)

### 3. Build & Integration

#### C (MinGW/GCC)

```
gcc -DRIFT_OPEN MAIN -O2 main.c -o ropen_c.exe
./ropen_c.exe input.bin A
```

#### C++ (Modern C++17)

```
g++ -std=c++17 main.cpp -o ropen_cpp.exe  
.rlopen_cpp.exe input.bin B
```

## C# (.NET SDK)

```
dotnet run -- input.bin A
```

## 4. SemVerX Logic

Versions are categorized by channels:

- **Legacy**: Backward compatibility maintenance.
- **Experimental**: Fast-tracking FilterFlash functors.
- **Stable**: Production-ready BiDAG nodes.
- **LTS**: Long-term support for industrial RIFT applications.

*“Structure is a signal. Polarity is a strategy. ROPEN is the experiment.”*