<?xml version="1.0" encoding="UTF-8" standalone="yes"?>

<w:document xmlns:w="http://schemas.openxmlformats.org/wordprocessingml/2006/main"
xmlns:r="http://schemas.openxmlformats.org/officeDocument/2006/relationships"
xmlns:m="http://schemas.openxmlformats.org/officeDocument/2006/math"
xmlns:v="urn:schemas-microsoft-com:vml"
xmlns:wp="http://schemas.openxmlformats.org/drawingml/2006/wordprocessingDrawing"
xmlns:a="http://schemas.openxmlformats.org/drawingml/2006/main"
xmlns:pic="http://schemas.openxmlformats.org/drawingml/2006/picture"
xmlns:w10="urn:schemas-microsoft-com:office:word"
xmlns:wne="http://schemas.microsoft.com/office/word/2006/wordml" xmlns:o="urn:schemas-microsoft-com:office:office" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

<w:body>

<w:p w:rsidR="000E" w:rsidRDefault="000E">

<w:r>

<w:t># EchoPulse vs NIST PQC KEMs – Performance and Resource Comparison (Document C8)</w:t>

</w:r>

</w:p>

<w:p w:rsidR="000E" w:rsidRDefault="000E"/>

<w:p w:rsidR="000E" w:rsidRDefault="000E">

<w:r>

<w:t>Version: 1.0</w:t>

</w:r>

</w:p>

<w:p w:rsidR="000E" w:rsidRDefault="000E">

<w:r>

<w:t>Date: May 11, 2025</w:t>

</w:r>

</w:p>

<w:p w:rsidR="000E" w:rsidRDefault="000E">

<w:r>

<w:t>Author: EchoPulse Initiative</w:t>

</w:r>

</w:p>

<w:p w:rsidR="000E" w:rsidRDefault="000E"/>

<w:p w:rsidR="000E" w:rsidRDefault="000E">

<w:r>

<w:t>## 1. Introduction</w:t>

</w:r>

</w:p>

<w:p w:rsidR="000E" w:rsidRDefault="000E">

<w:r>

<w:t>Lightweight Post-Quantum Cryptography (PQC) is crucial for securing embedded systems and resource-constrained devices, such as microcontrollers (MCUs), radio devices, and secure elements. These environments often have limited RAM, processing power, and energy budgets. This document compares EchoPulse, a novel symbolic KEM, with two leading NIST PQC KEM candidates:</w:t>

</w:r>

</w:p>

<w:p w:rsidR="000E" w:rsidRDefault="000E">

<w:r>

<w:t>- **EchoPulse:** A symbolic KEM designed for low RAM usage and inherent replay protection through graph mutation.</w:t>

</w:r>

</w:p>

<w:p w:rsidR="000E" w:rsidRDefault="000E">

<w:r>

<w:t>- **Kyber-512:** A widely accepted lattice-based KEM, offering strong security and relatively good performance.</w:t>

</w:r>

</w:p>

<w:p w:rsidR="000E" w:rsidRDefault="000E">

<w:r>

<w:t>- **FrodoKEM-640:** A code-based KEM, known for its conservative security approach but typically exhibiting higher resource requirements.</w:t>

</w:r>

</w:p>

<w:p w:rsidR="000E" w:rsidRDefault="000E"/>

<w:p w:rsidR="000E" w:rsidRDefault="000E">

<w:r>

<w:t>## 2. Comparison Table</w:t>

</w:r>

</w:p>

<w:tbl w:rsidR="000E" w:rsidTbl="000E">

<w:tblPr>

<w:tblStyle w:val="a7"/>

<w:tblW w:w="0" w:type="auto"/>

<w:tblLook w:val="04A0" w:firstRow="1" w:lastRow="0" w:firstColumn="1" w:lastColumn="0" w:noHBand="0" w:noVBand="1" w:evenHBand="0" w:oddHBand="1" w:evenVBand="0" w:oddVBand="0" w:firstRowFirstColumn="1" w:firstRowLastColumn="0" w:lastRowFirstColumn="0" w:lastRowLastColumn="0" w:firstColumnHBand="0" w:lastColumnHBand="0" w:firstColumnVBand="0" w:lastColumnVBand="0" w:topLeftCell="0" w:topRightCell="0" w:bottomLeftCell="0" w:bottomRightCell="0"/>

</w:tblPr>

<w:tblGrid>

<w:gridCol w:w="3026"/>

<w:gridCol w:w="2401"/>

<w:gridCol w:w="2401"/>

<w:gridCol w:w="2401"/>

</w:tblGrid>

<w:tr w:rsidR="000E" w:tblW="0" w:trHeightType="auto">

<w:tc w:rsidR="000E">
 <w:tcPr>
  <w:tcW w:w="3026" w:type="dxa"/>
 </w:tcPr>
 <w:p w:rsidR="000E" w:rsidRDefault="000E">
  <w:r>
   <w:t>Metric</w:t>
  </w:r>
 </w:p>
</w:tc>
<w:tc w:rsidR="000E">
 <w:tcPr>
  <w:tcW w:w="2401" w:type="dxa"/>
 </w:tcPr>
 <w:p w:rsidR="000E" w:rsidRDefault="000E">
  <w:r>
   <w:t>EchoPulse</w:t>
  </w:r>
 </w:p>
</w:tc>
<w:tc w:rsidR="000E">
 <w:tcPr>
  <w:tcW w:w="2401" w:type="dxa"/>
 </w:tcPr>
 <w:p w:rsidR="000E" w:rsidRDefault="000E">
  <w:r>
   <w:t>Kyber-512</w:t>

```
    </w:r>
   </w:p>
  </w:tc>
  <w:tc w:rsidR="000E">
   <w:tcPr>
    <w:tcW w:w="2401" w:type="dxa"/>
   </w:tcPr>
   <w:p w:rsidR="000E" w:rsidRDefault="000E">
    <w:r>
     <w:t>FrodoKEM-640</w:t>
    </w:r>
   </w:p>
  </w:tc>
 </w:tr>
 <w:tr w:rsidR="000E" w:tblW="0" w:trHeightType="auto">
  <w:tc w:rsidR="000E">
   <w:tcPr>
    <w:tcW w:w="3026" w:type="dxa"/>
   </w:tcPr>
   <w:p w:rsidR="000E" w:rsidRDefault="000E">
    <w:r>
     <w:t>RAM Usage</w:t>
    </w:r>
   </w:p>
  </w:tc>
  <w:tc w:rsidR="000E">
   <w:tcPr>
```

<w:tcW w:w="2401" w:type="dxa"/>
</w:tcPr>
<w:p w:rsidR="000E" w:rsidRDefault="000E">
<w:r>
<w:t>~8.4 KB</w:t>
</w:r>
</w:p>
</w:tc>
<w:tc w:rsidR="000E">
<w:tcPr>
<w:tcW w:w="2401" w:type="dxa"/>
</w:tcPr>
<w:p w:rsidR="000E" w:rsidRDefault="000E">
<w:r>
<w:t>~14–20 KB</w:t>
</w:r>
</w:p>
</w:tc>
<w:tc w:rsidR="000E">
<w:tcPr>
<w:tcW w:w="2401" w:type="dxa"/>
</w:tcPr>
<w:p w:rsidR="000E" w:rsidRDefault="000E">
<w:r>
<w:t>~40+ KB</w:t>
</w:r>
</w:p>

```
 </w:tc>
</w:tr>
<w:tr w:rsidR="000E" w:tblW="0" w:trHeightType="auto">
 <w:tc w:rsidR="000E">
  <w:tcPr>
   <w:tcW w:w="3026" w:type="dxa"/>
  </w:tcPr>
  <w:p w:rsidR="000E" w:rsidRDefault="000E">
   <w:r>
    <w:t>Encapsulation Time (M4F)</w:t>
   </w:r>
  </w:p>
 </w:tc>
 <w:tc w:rsidR="000E">
  <w:tcPr>
   <w:tcW w:w="2401" w:type="dxa"/>
  </w:tcPr>
  <w:p w:rsidR="000E" w:rsidRDefault="000E">
   <w:r>
    <w:t>~375 µs</w:t>
   </w:r>
  </w:p>
 </w:tc>
 <w:tc w:rsidR="000E">
  <w:tcPr>
   <w:tcW w:w="2401" w:type="dxa"/>
  </w:tcPr>
```

<w:p w:rsidR="000E" w:rsidRDefault="000E">
<w:r>
<w:t>~1000 µs</w:t>
</w:r>
</w:p>
</w:tc>
<w:tc w:rsidR="000E">
<w:tcPr>
<w:tcW w:w="2401" w:type="dxa"/>
</w:tcPr>
<w:p w:rsidR="000E" w:rsidRDefault="000E">
<w:r>
<w:t>~3000 µs</w:t>
</w:r>
</w:p>
</w:tc>
</w:tr>
<w:tr w:rsidR="000E" w:tblW="0" w:trHeightType="auto">
<w:tc w:rsidR="000E">
<w:tcPr>
<w:tcW w:w="3026" w:type="dxa"/>
</w:tcPr>
<w:p w:rsidR="000E" w:rsidRDefault="000E">
<w:r>
<w:t>Payload Size</w:t>
</w:r>
</w:p>

```
</w:tc>
<w:tc w:rsidR="000E">
 <w:tcPr>
  <w:tcW w:w="2401" w:type="dxa"/>
 </w:tcPr>
 <w:p w:rsidR="000E" w:rsidRDefault="000E">
  <w:r>
   <w:t>28 B</w:t>
  </w:r>
 </w:p>
</w:tc>
<w:tc w:rsidR="000E">
 <w:tcPr>
  <w:tcW w:w="2401" w:type="dxa"/>
 </w:tcPr>
 <w:p w:rsidR="000E" w:rsidRDefault="000E">
  <w:r>
   <w:t>~800 B</w:t>
  </w:r>
 </w:p>
</w:tc>
<w:tc w:rsidR="000E">
 <w:tcPr>
  <w:tcW w:w="2401" w:type="dxa"/>
 </w:tcPr>
 <w:p w:rsidR="000E" w:rsidRDefault="000E">
  <w:r>
```

<w:t>~960 B</w:t>
</w:r>
</w:p>
</w:tc>
</w:tr>
<w:tr w:rsidR="000E" w:tblW="0" w:trHeightType="auto">
<w:tc w:rsidR="000E">
<w:tcPr>
<w:tcW w:w="3026" w:type="dxa"/>
</w:tcPr>
<w:p w:rsidR="000E" w:rsidRDefault="000E">
<w:r>
<w:t>CPU Cycles (Encaps)</w:t>
</w:r>
</w:p>
</w:tc>
<w:tc w:rsidR="000E">
<w:tcPr>
<w:tcW w:w="2401" w:type="dxa"/>
</w:tcPr>
<w:p w:rsidR="000E" w:rsidRDefault="000E">
<w:r>
<w:t>~30,000</w:t>
</w:r>
</w:p>
</w:tc>
<w:tc w:rsidR="000E">

```
    <w:tcPr>
     <w:tcW w:w="2401" w:type="dxa"/>
    </w:tcPr>
    <w:p w:rsidR="000E" w:rsidRDefault="000E">
     <w:r>
      <w:t>~80,000</w:t>
     </w:r>
    </w:p>
   </w:tc>
   <w:tc w:rsidR="000E">
    <w:tcPr>
     <w:tcW w:w="2401" w:type="dxa"/>
    </w:tcPr>
    <w:p w:rsidR="000E" w:rsidRDefault="000E">
     <w:r>
      <w:t>~240,000</w:t>
     </w:r>
    </w:p>
   </w:tc>
  </w:tr>
  <w:tr w:rsidR="000E" w:tblW="0" w:trHeightType="auto">
   <w:tc w:rsidR="000E">
    <w:tcPr>
     <w:tcW w:w="3026" w:type="dxa"/>
    </w:tcPr>
    <w:p w:rsidR="000E" w:rsidRDefault="000E">
     <w:r>
```

<w:t>Mutation/Randomizer</w:t>
</w:r>
</w:p>
</w:tc>
<w:tc w:rsidR="000E">
<w:tcPr>
<w:tcW w:w="2401" w:type="dxa"/>
</w:tcPr>
<w:p w:rsidR="000E" w:rsidRDefault="000E">
<w:r>
<w:t>Yes (symbolic)</w:t>
</w:r>
</w:p>
</w:tc>
<w:tc w:rsidR="000E">
<w:tcPr>
<w:tcW w:w="2401" w:type="dxa"/>
</w:tcPr>
<w:p w:rsidR="000E" w:rsidRDefault="000E">
<w:r>
<w:t>No</w:t>
</w:r>
</w:p>
</w:tc>
<w:tc w:rsidR="000E">
<w:tcPr>
<w:tcW w:w="2401" w:type="dxa"/>

```
    </w:tcPr>
    <w:p w:rsidR="000E" w:rsidRDefault="000E">
     <w:r>
      <w:t>No</w:t>
     </w:r>
    </w:p>
   </w:tc>
  </w:tr>
  <w:tr w:rsidR="000E" w:tblW="0" w:trHeightType="auto">
   <w:tc w:rsidR="000E">
    <w:tcPr>
     <w:tcW w:w="3026" w:type="dxa"/>
    </w:tcPr>
    <w:p w:rsidR="000E" w:rsidRDefault="000E">
     <w:r>
      <w:t>Replay Protection</w:t>
     </w:r>
    </w:p>
   </w:tc>
   <w:tc w:rsidR="000E">
    <w:tcPr>
     <w:tcW w:w="2401" w:type="dxa"/>
    </w:tcPr>
    <w:p w:rsidR="000E" w:rsidRDefault="000E">
     <w:r>
      <w:t>Built-in (\u03BC)</w:t>
     </w:r>
```

```
    </w:p>
   </w:tc>
   <w:tc w:rsidR="000E">
    <w:tcPr>
     <w:tcW w:w="2401" w:type="dxa"/>
    </w:tcPr>
    <w:p w:rsidR="000E" w:w:rsidRDefault="000E">
     <w:r>
      <w:t>Not default</w:t>
     </w:r>
    </w:p>
   </w:tc>
   <w:tc w:rsidR="000E">
    <w:tcPr>
     <w:tcW w:w="2401" w:type="dxa"/>
    </w:tcPr>
    <w:p w:rsidR="000E" w:w:rsidRDefault="000E">
     <w:r>
      <w:t>Not default</w:t>
     </w:r>
    </w:p>
   </w:tc>
  </w:tr>
</w:tbl>
<w:p w:rsidR="000E" w:w:rsidRDefault="000E"/>
<w:p w:rsidR="000E" w:w:rsidRDefault="000E">
 <w:r>
```

<w:t>## 3. Graphical Comparison</w:t>

</w:r>

</w:p>

<w:p w:rsidR="000E" w:rsidRDefault="000E">

<w:r>

<w:t>A bar chart can effectively visualize the performance and resource differences between EchoPulse, Kyber-512, and FrodoKEM-640. Three separate bar charts are suggested:</w:t>

</w:r>

</w:p>

<w:p w:rsidR="000E" w:rsidRDefault="000E">

<w:r>

<w:t>- **RAM Usage (KB):** Each bar would represent the RAM footprint of each KEM. EchoPulse's bar should be significantly shorter than Kyber-512's and FrodoKEM-640's.</w:t>

</w:r>

</w:p>

<w:p w:rsidR="000E" w:rsidRDefault="000E">

<w:r>

<w:t>- **Encapsulation Time (µs):** This chart would compare the encapsulation speeds on a Cortex-M4F platform. EchoPulse should demonstrate a faster encapsulation time compared to the other two.</w:t>

</w:r>

</w:p>

<w:p w:rsidR="000E" w:rsidRDefault="000E">

<w:r>

<w:t>- **Payload Size (B):** This chart would illustrate the ciphertext size. EchoPulse has a significantly smaller payload size compared to Kyber-512 and FrodoKEM-640.</w:t>

</w:r>

</w:p>

<w:p w:rsidR="000E" w:rsidRDefault="000E">

<w:r>

<w:t>Each bar should be clearly labeled with the KEM's name (EchoPulse, Kyber-512, FrodoKEM-640) and color-coded for easy differentiation. Libraries like `matplotlib` or `plotly` in Python can be used to generate these charts. For example, a simple `matplotlib` implementation might involve:</w:t>

</w:r>

</w:p>

<w:p w:rsidR="000E" w:rsidRDefault="000E">

<w:r>

<w:t>```python</w:t>

</w:r>

</w:p>

<w:p w:rsidR="000E" w:rsidRDefault="000E">

<w:r>

<w:t>import matplotlib.pyplot as plt</w:t>

</w:r>

</w:p>

<w:p w:rsidR="000E" w:rsidRDefault="000E">

<w:r>

<w:t/>

</w:r>

</w:p>

<w:p w:rsidR="000E" w:rsidRDefault="000E">

<w:r>

<w:t># Sample data (replace with actual values)</w:t>

</w:r>

</w:p>

<w:p w:rsidR="000E" w:rsidRDefault="000E">

<w:r>

<w:t>ram_usage = {'EchoPulse': 8.4, 'Kyber-512': 17, 'FrodoKEM-640': 45}</w:t>

</w:r>

</w:p>

<w:p w:rsidR="000E" w:rsidRDefault="000E">

<w:r>

<w:t>encaps_time = {'EchoPulse': 375, 'Kyber-512': 1000, 'FrodoKEM-640': 3000}</w:t>

</w:r>

</w:p>

<w:p w:rsidR="000E" w:rsidRDefault="000E">

<w:r>

<w:t>payload_size = {'EchoPulse': 28, 'Kyber-512': 800, 'FrodoKEM-640': 960}</w:t>

</w:r>

</w:p>

<w:p w:rsidR="000E" w:rsidRDefault="000E"/>

<w:p w:rsidR="000E" w:rsidRDefault="000E">

<w:r>

<w:t># Example: RAM Usage Bar Chart</w:t>

</w:r>

</w:p>

<w:p w:rsidR="000E" w:rsidRDefault="000E">

<w:r>

<w:t>plt.figure(figsize=(6, 4))</w:t>

</w:r>

</w:p>

<w:p w:rsidR="000E" w:rsidRDefault="000E">

<w:r>

<w:t>plt.bar(ram_usage.keys(), ram_usage.values(), color=['green', 'blue', 'red'])</w:t>

</w:r>
</w:p>
<w:p w:rsidR="000E" w:rsidRDefault="000E">
<w:r>
<w:t>plt.ylabel('RAM Usage (KB)')</w:t>
</w:r>
</w:p>
<w:p w:rsidR="000E" w:rsidRDefault="000E">
<w:r>
<w:t>plt.title('RAM Usage Comparison')</w:t>
</w:r>
</w:p>
<w:p w:rsidR="000E" w:rsidRDefault="000E">
<w:r>
<w:t>plt.tight_layout()</w:t>
</w:r>
</w:p>
<w:p w:rsidR="000E" w:rsidRDefault="000E">
<w:r>
<w:t>plt.savefig('ram_usage_comparison.png')</w:t>
</w:r>
</w:p>
<w:p w:rsidR="000E" w:rsidRDefault="000E">
<w:r>
<w:t>plt.show()</w:t>
</w:r>
</w:p>

<w:p w:rsidR="000E" w:rsidRDefault="000E">
<w:r>
<w:t>```</w:t>
</w:r>
</w:p>
<w:p w:rsidR="000E" w:rsidRDefault="000E"/>
<w:p w:rsidR="000E" w:rsidRDefault="000E">
<w:r>
<w:t>Similar code structures can be used to generate the other two bar charts.</w:t>
</w:r>
</w:p>
<w:p w:rsidR="000E" w:rsidRDefault="000E"/>
<w:p w:rsidR="000E" w:rsidRDefault="000E">
<w:r>
<w:t>## 4. Conclusion</w:t>
</w:r>
</w:p>
<w:p w:rsidR="000E" w:rsidRDefault="000E">
<w:r>
<w:t>EchoPulse presents a compelling alternative for post-quantum secure key exchange in low-resource environments. Its significantly lower RAM footprint, faster encapsulation time, and smaller payload size compared to Kyber-512 and FrodoKEM-640 make it particularly well-suited for MCUs, radio devices, and secure elements. Furthermore, the inherent replay protection offered by its symbolic mutation mechanism (\u03BC) and the entropy control provided by its path-based design are novel features that distinguish it from traditional algebraic KEMs.</w:t>
</w:r>
</w:p>
<w:p w:rsidR="000E" w:rsidRDefault="000E"/>
<w:p w:rsidR="000E" w:rsidRDefault="000E">

```
      <w:r>
        <w:t>*Document C8 — Comparative Visualization Layer — EchoPulse Initiative*</w:t>
      </w:r>
    </w:p>
  </w:body>
</w:document>
```