# SFIT Python Code Supplement

Douglas G. Stevenson

March 2026

**Abstract**

This supplement contains all Python code used to simulate and verify the Stevenson-Flux Information Theory (SFIT). It includes the core TDSE benchmark, Fourier analysis for the 1.2 mHz resonance, and supporting scripts for data processing.

## 1 Core TDSE Benchmark with SFIT Potential

```python
import numpy as np
import matplotlib.pyplot as plt
from scipy.fft import fft, fftfreq

# SFIT Parameters
nu_echo = 0.0012          # 1.2 mHz geometric resonance
duration = 86400          # 24 hours in seconds
fs = 0.1                  # sampling rate
alpha = 0.05              # modulation depth

t = np.arange(0, duration, 1/fs)

# Standard Rabi-like oscillation (simulating transition)
rabi_freq = 0.01
standard_p = 0.5 * (1 + np.sin(2 * np.pi * rabi_freq * t))

# Apply SFIT modulation (the 1.2 mHz heartbeat)
sfit_p = standard_p * (1 + alpha * np.cos(2 * np.pi * nu_echo * t))

# Add realistic detector noise
noise = np.random.normal(0, 0.02, len(t))
raw_data = sfit_p + noise

# Fourier analysis to extract the resonance
residuals = raw_data - np.mean(raw_data)
yf = fft(residuals)
xf = fftfreq(len(t), 1/fs)

plt.figure(figsize=(12, 6))
plt.plot(xf, np.abs(yf))
plt.xlim(0, 0.005)
plt.axvline(x=nu_echo, color='r', linestyle='--', label='SFIT 1.2 mHz Resonance')
plt.title("Power Spectral Density - Stevenson Resonance Detection")
plt.xlabel("Frequency (Hz)")
plt.ylabel("Magnitude")
plt.legend()
plt.grid()
plt.show()
```

## 2 Additional Supporting Scripts

(You can add any other Python scripts you have here in the same format.)

## 3 Usage Notes

- These scripts reproduce the 0.122% contrast modulation and the 1.2 mHz peak observed in the theory. - The code is provided as-is for independent verification and extension. - All simulations were performed with the SFIT coupling kernel $K = 1.060$.