

[Skip to content](#)

What is FrozenLake?

FrozenLake is a simple OpenAI Gym environment where an agent moves across a grid of frozen tiles trying not to fall into holes. But for your use case...

👉 **You don't care about the environment itself.**

You're just using the FrozenLake gym loop to insert and test hardware calls like your **1D convolution block**, as a "proof that the system works end to end."



Next Time: What We'll Do

We'll treat the FrozenLake-v1 agent's observations as **toy input** to your 1D conv block. For example:

1. Agent observes its state: `s = env.reset()`
2. You one-hot encode that to a 16-length vector
3. You run your conv block on that vector (via MyHDL or NumPy)
4. Output is your "simulated activation" — totally fake, but enough to test hardware
5. That output might drive a dummy "policy"



Why It's Useful

- ☒ Shows how to integrate hardware modules into a Python loop
- ☒ Tests the interface you'll later use for NSynth
- ☒ Gives you confidence in how to stream data, handle delays, and map to/from fixed

Oh that's why I'm confused. I thought it was a hw synthesizer or some sort of translation of code to W.



Search



Deep research



[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

[Skip to content](#)

