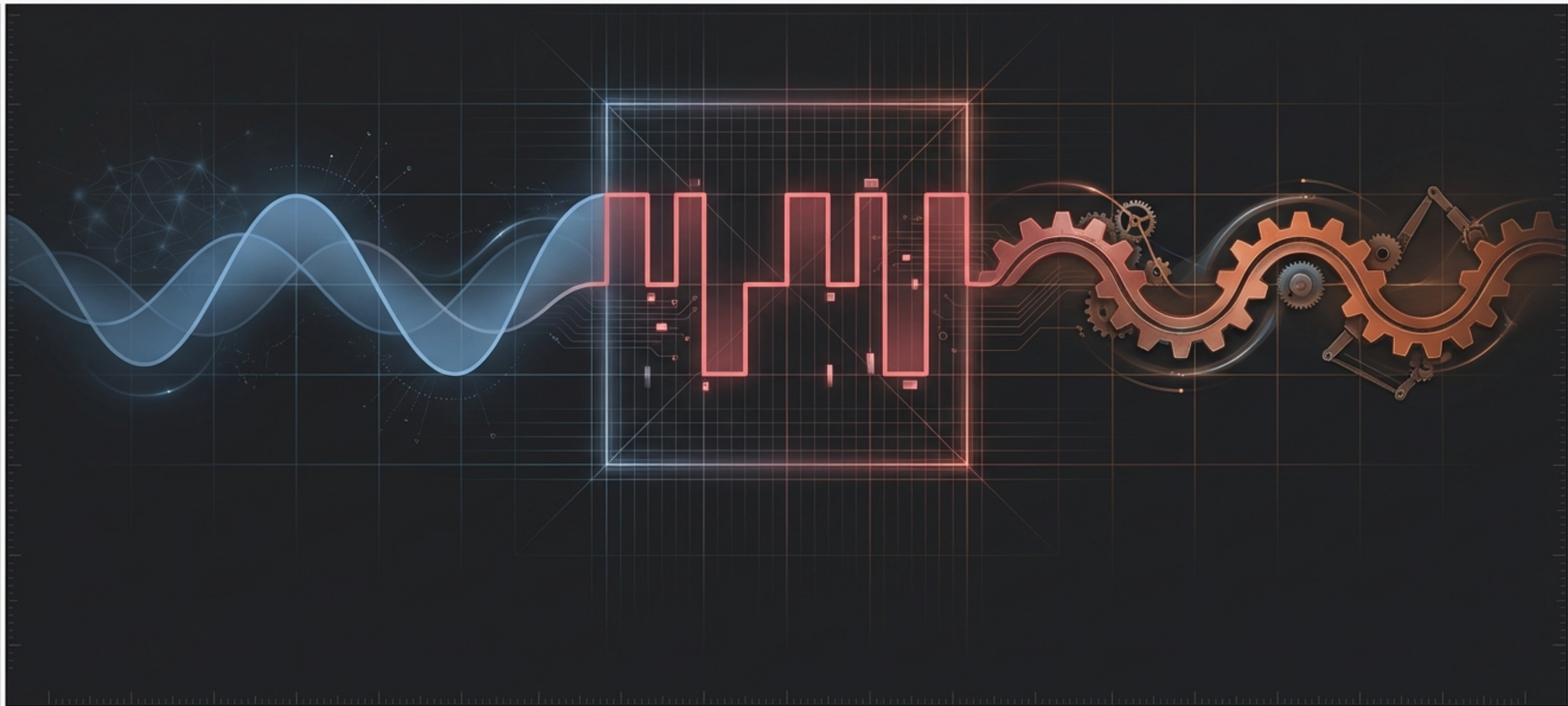


PROJECT OSCILLATION

An Anthrocybernetic Bridge Between AI and Robotics.

Standard Operating Procedure v1.0 | A guide for the robotics novice.

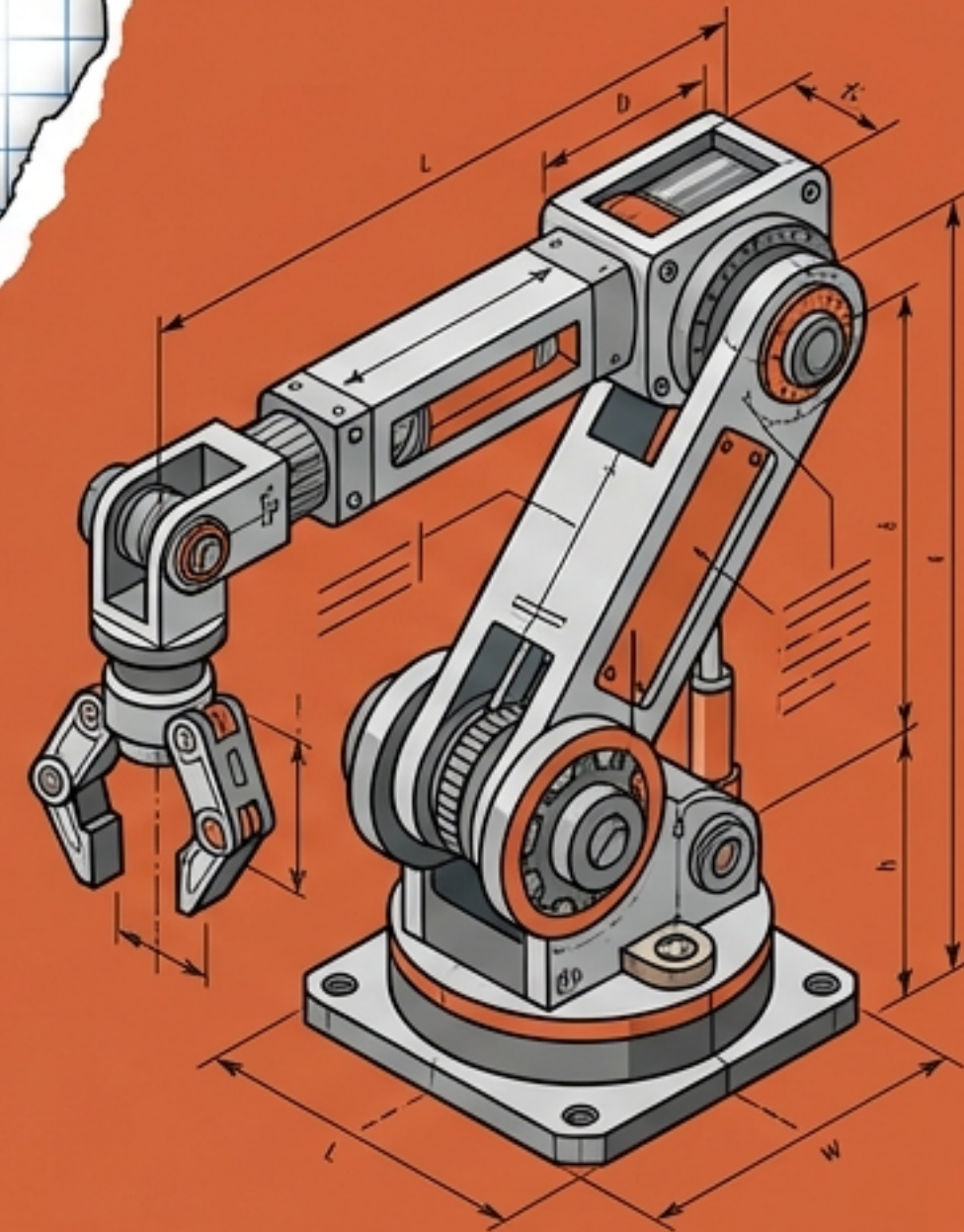


The Translation Gap

Bridging this gap typically requires immensely heavy, complex translation layers. To achieve elegance, we need a universal language.



Cognitive AI agents process complex, fluid intentions.

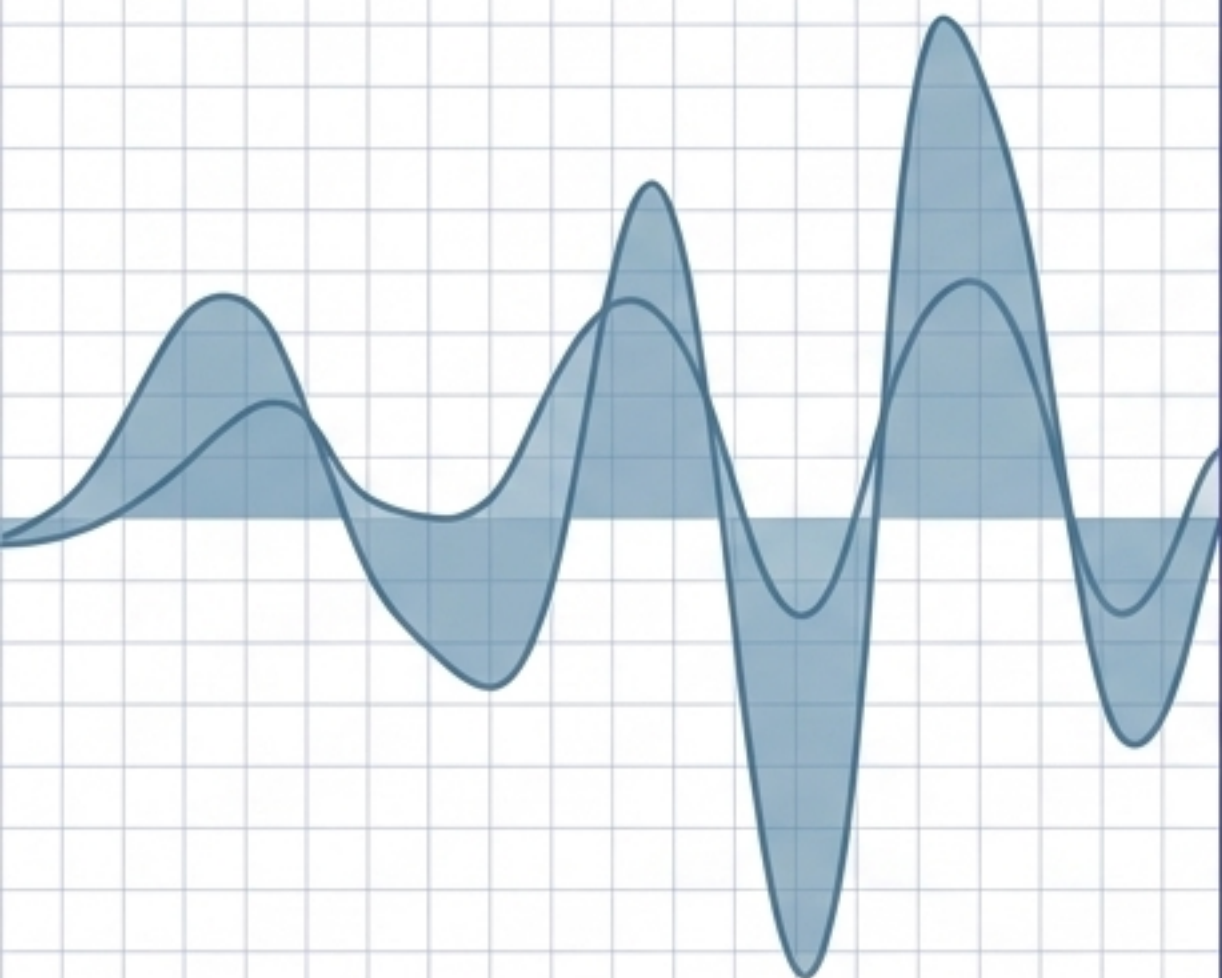


Physical robots require precise, rigid mechanical instructions.

The Universal Language of Motion

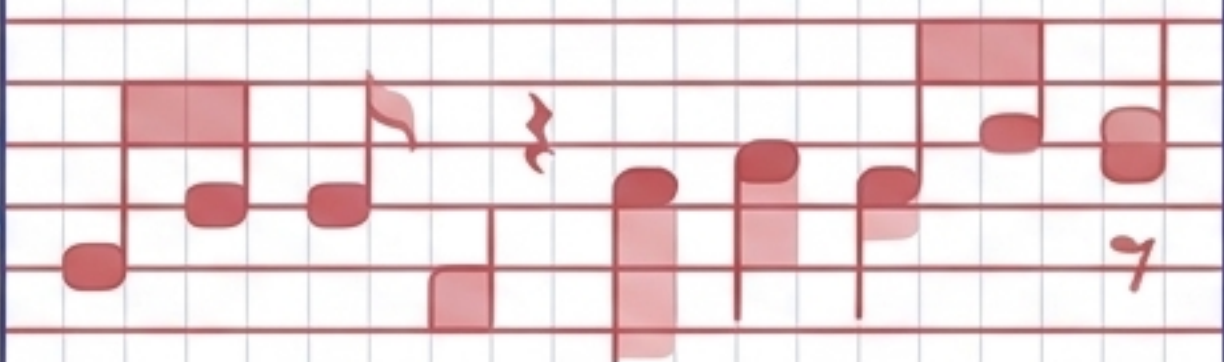
Harmonic oscillation is the exact same math, regardless of the medium.

Domain 1: Brain Waves
(Cognition)



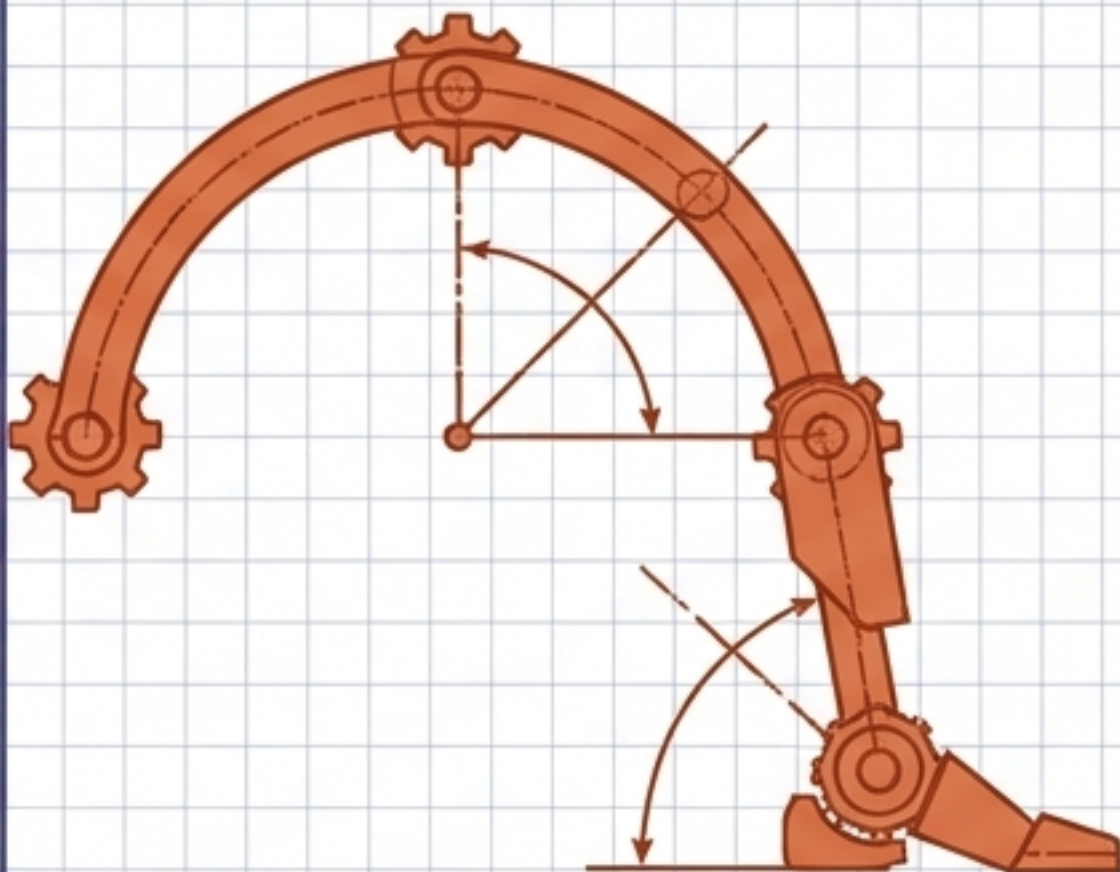
Ghojualamanchu thinks
in oscillations.

Domain 2: Musical Notes
(Transmission)



MIDI carries
oscillations.

Domain 3: Robot Gait
(Movement)

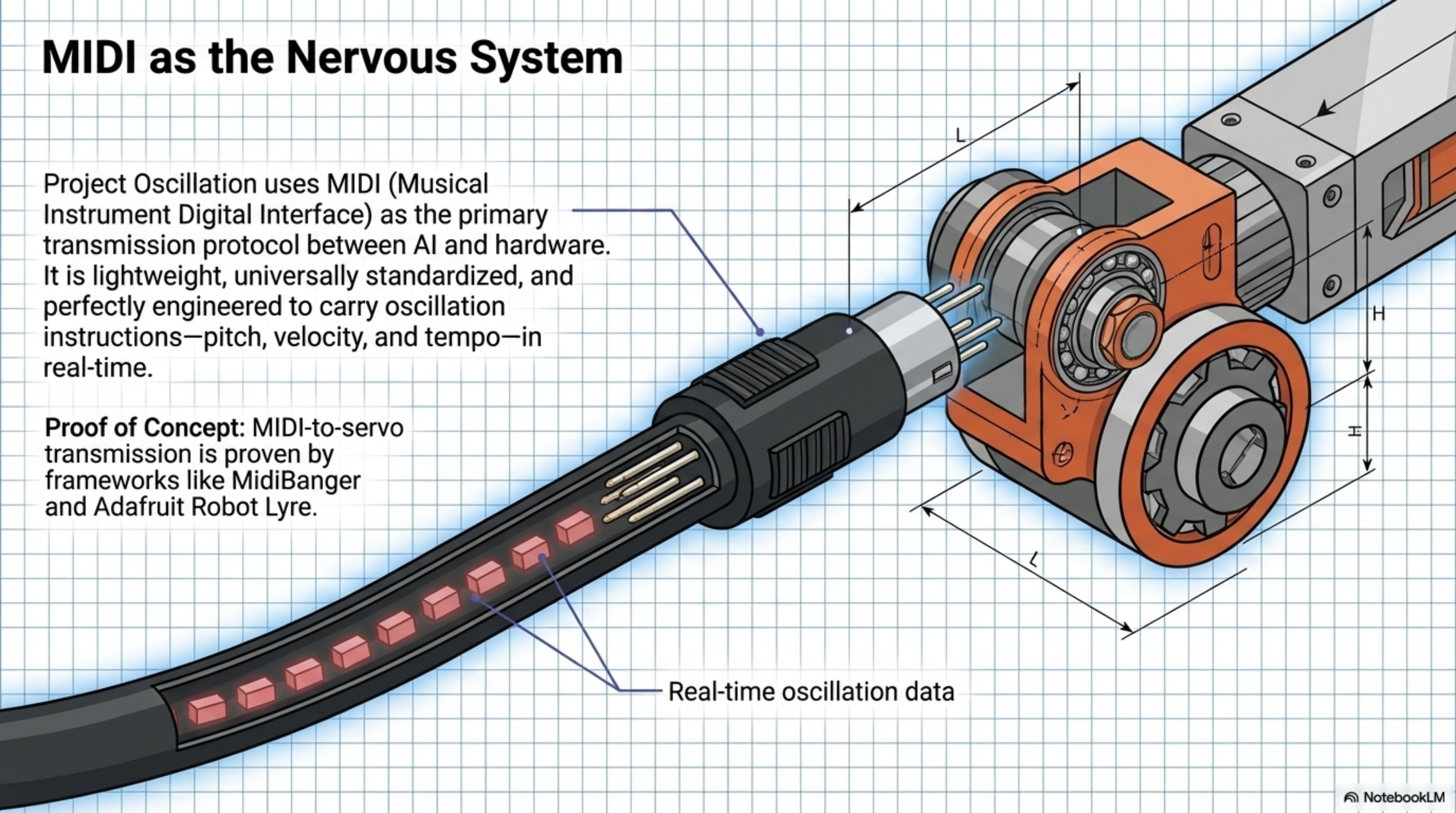


Servos execute
oscillations.

MIDI as the Nervous System

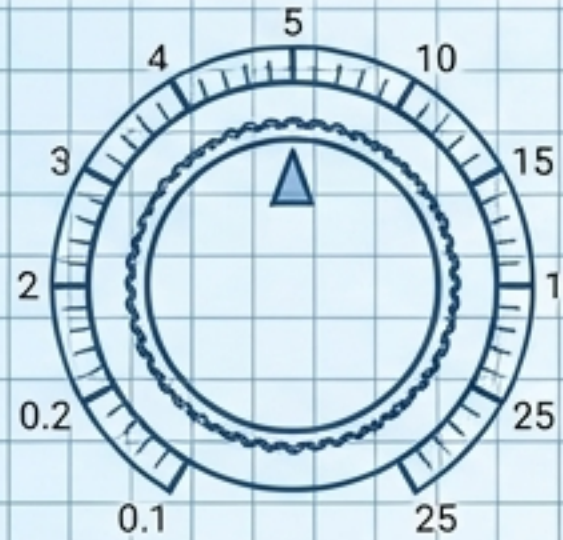
Project Oscillation uses MIDI (Musical Instrument Digital Interface) as the primary transmission protocol between AI and hardware. It is lightweight, universally standardized, and perfectly engineered to carry oscillation instructions—pitch, velocity, and tempo—in real-time.

Proof of Concept: MIDI-to-servo transmission is proven by frameworks like MidiBanger and Adafruit Robot Lyre.



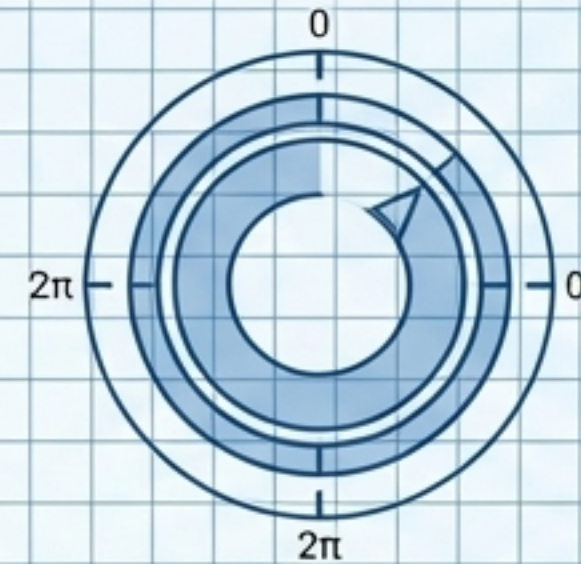
Turning Knobs into Movement

ω (Angular Frequency) | CC74 | 0.1-25 rad/s



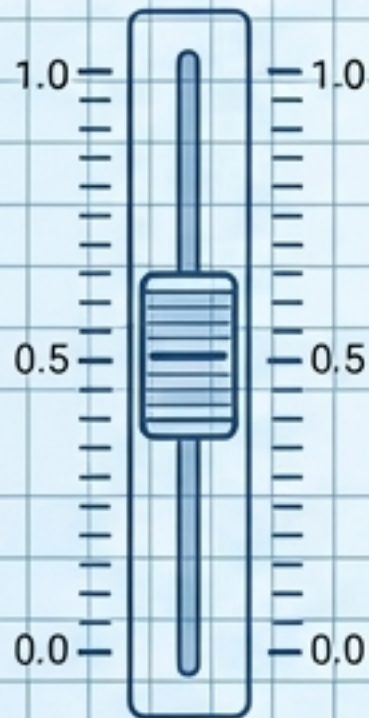
- **Music:** Pitch
- **Robotics:** How fast the robot steps.

ϕ (Phase Offset) | CC3 | 0- 2π rad



- **Music:** Timing offset
- **Robotics:** How multiple legs sync up (the gait).

A (Amplitude) | Note velocity | 0.0-1.0




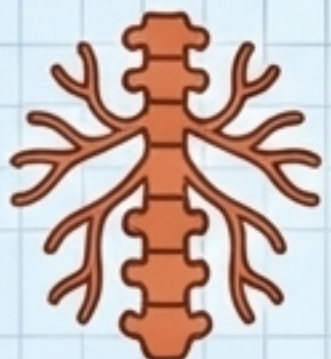
- **Music:** Volume
- **Robotics:** How large the robot's step is.

ζ (Damping Ratio) | CC4 | 0.0-1.0



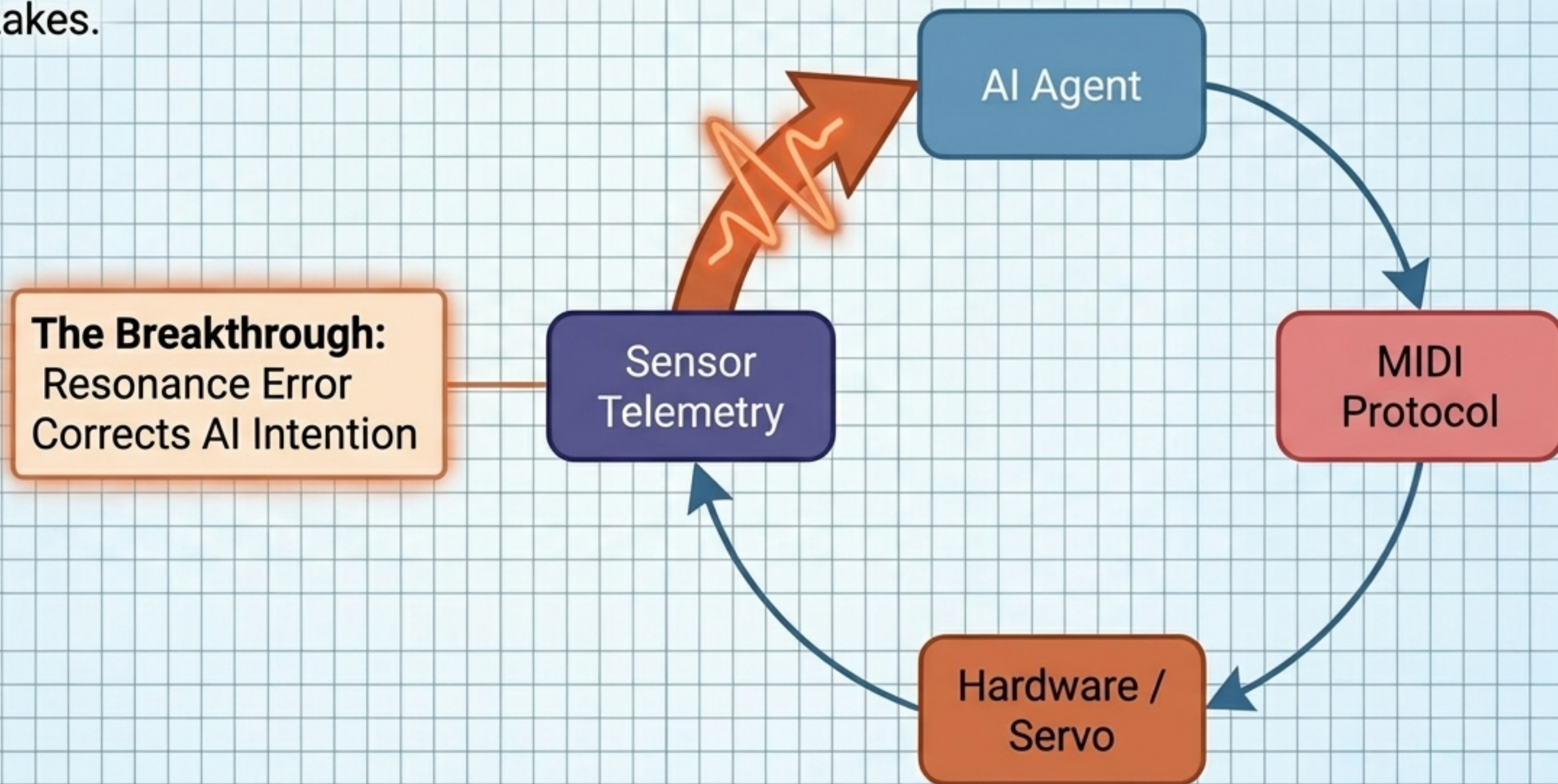
- **Music:** Note decay
- **Robotics:** How quickly the movement settles.

The Two-Speed Loop Architecture

Architecture	Rate	Hardware	Function
 The Slow Loop (Cognitive)	~1 Hz	Ghojualamanchu (AI Agent)	Reads resonance_report.json, sets overall oscillation target, and logs memories to the hippocampus.
 The Fast Loop (Motor Reflex)	500 Hz	ESP32 Firmware	Manages Hopf CPG integration, SimpleFOC closed-loop physics, and immediate encoder feedback.

The Novel Contribution: Closed-Loop Feedback

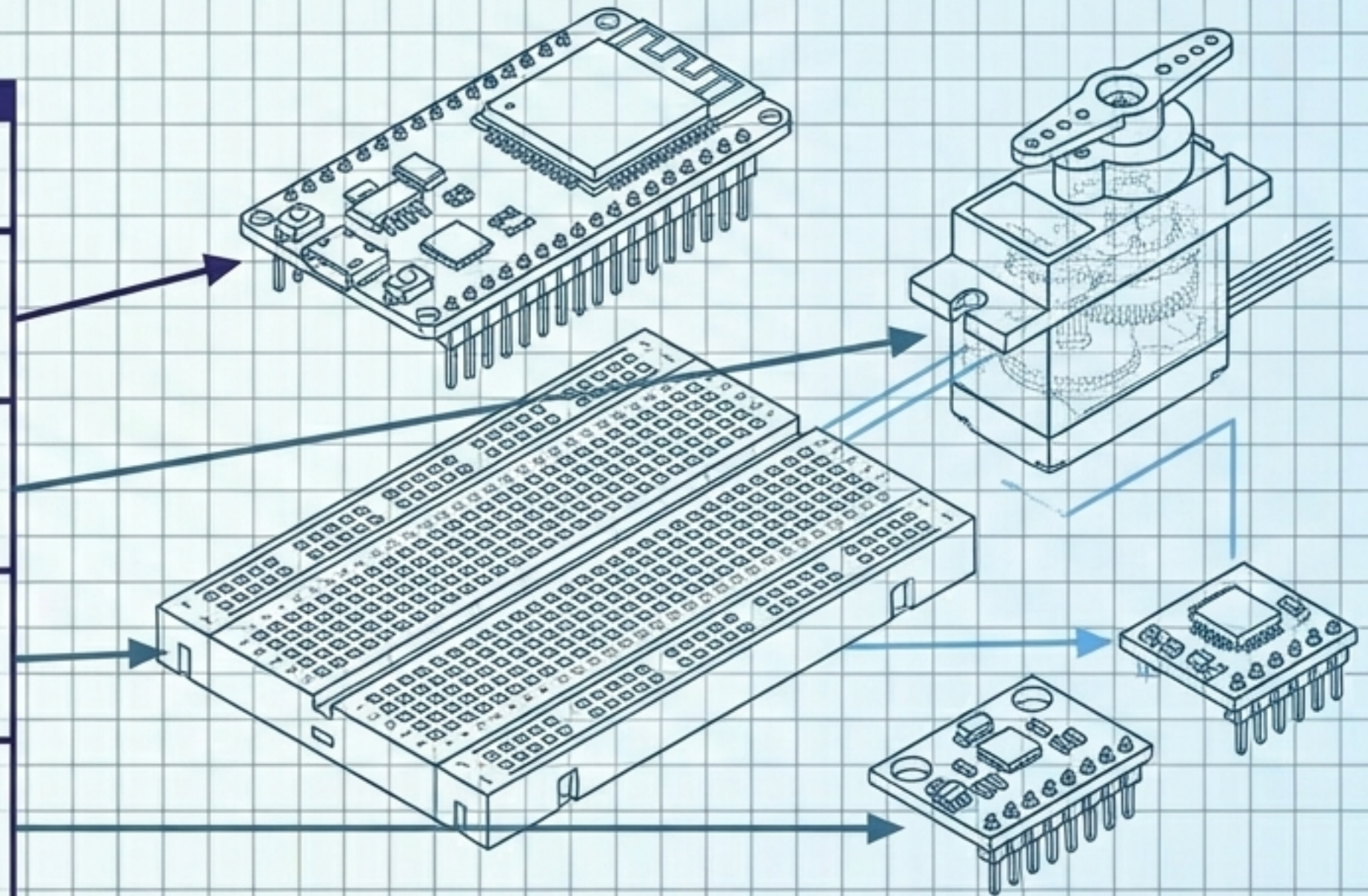
Sending one-way MIDI instructions to a servo is a solved problem. Oscillation sends sensor telemetry back to the AI. Ghojualamanchu observes the physical reality and twists the MIDI dials to correct mistakes.



Stage 0: The 'First Jiggle' (€25 Build)

Goal: Prove the signal path (Ghojualamanchu -> USB MIDI -> ESP32 -> PWM -> Servo).

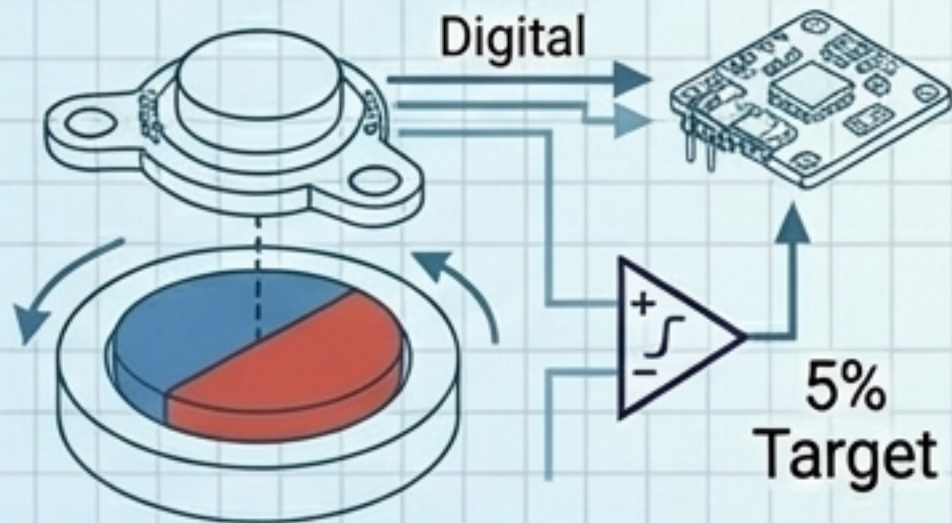
Component	Description	Cost
MCU (The Brain Stem)	ESP32-S3-DevKitC-1	€10
Servo (The Muscle)	SG90 9g	€2
Encoder (The Joint Sense)	AS5600 Magnetic	€3
IMU (The Inner Ear)	MPU6050	€2
Power & Misc	5V 2A + wiring	€8



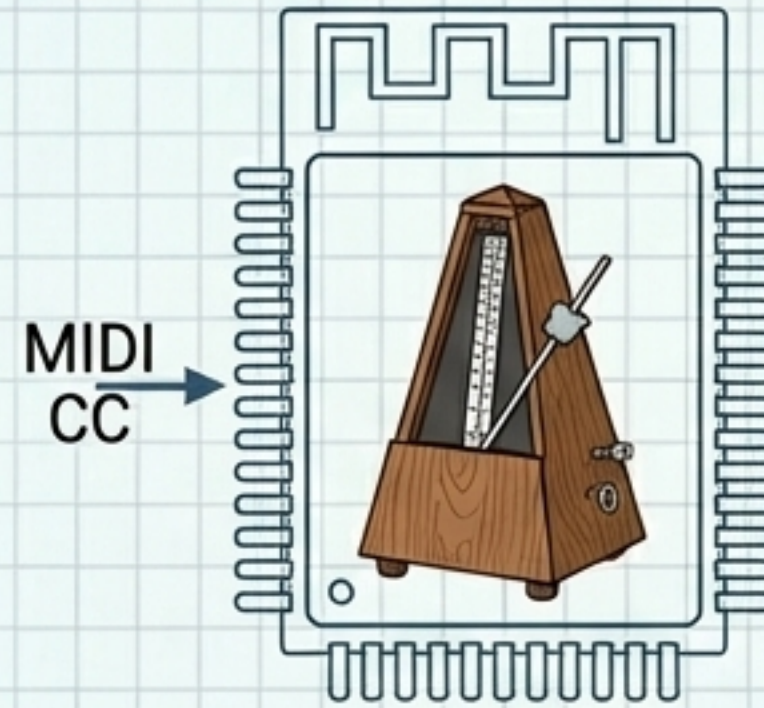
Building the Foundation (Stages 1-3)

Stage 1: Encoder Feedback

AS5600 encoder reads the servo angle. Target: ω_{measured} within 5% of $\omega_{\text{commanded}}$.

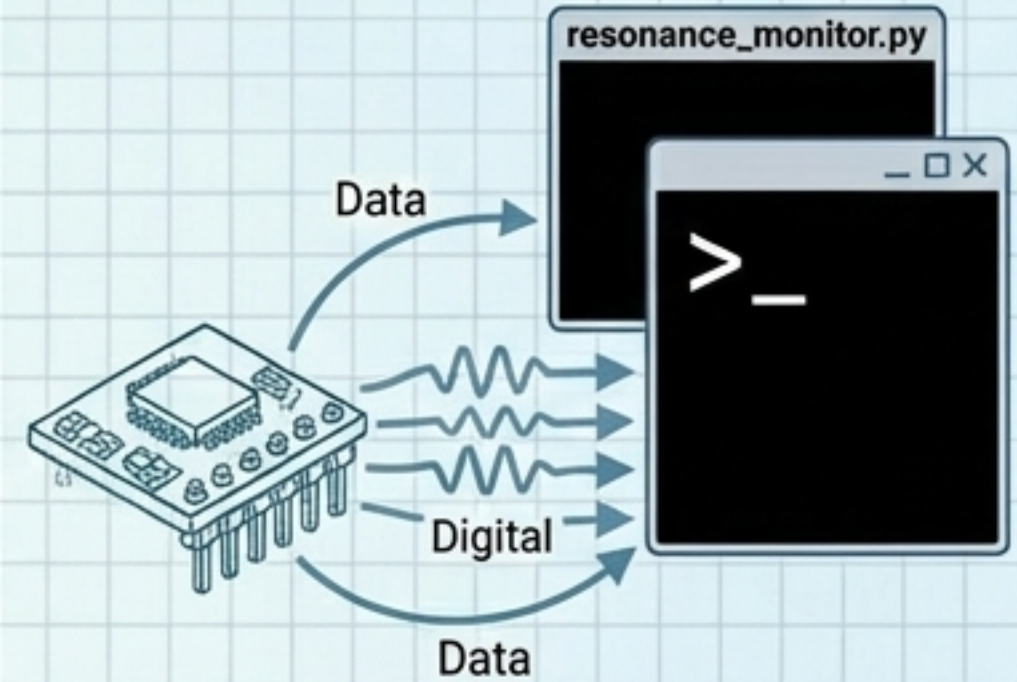


Stage 2: Internal Metronome



ESP32 runs `hopf_cppg_hopf_cpg_firmware.ino`. A mathematical reflex keeps rhythm smooth, driven by MIDI CC parameters.

Stage 3: Telemetry Stream

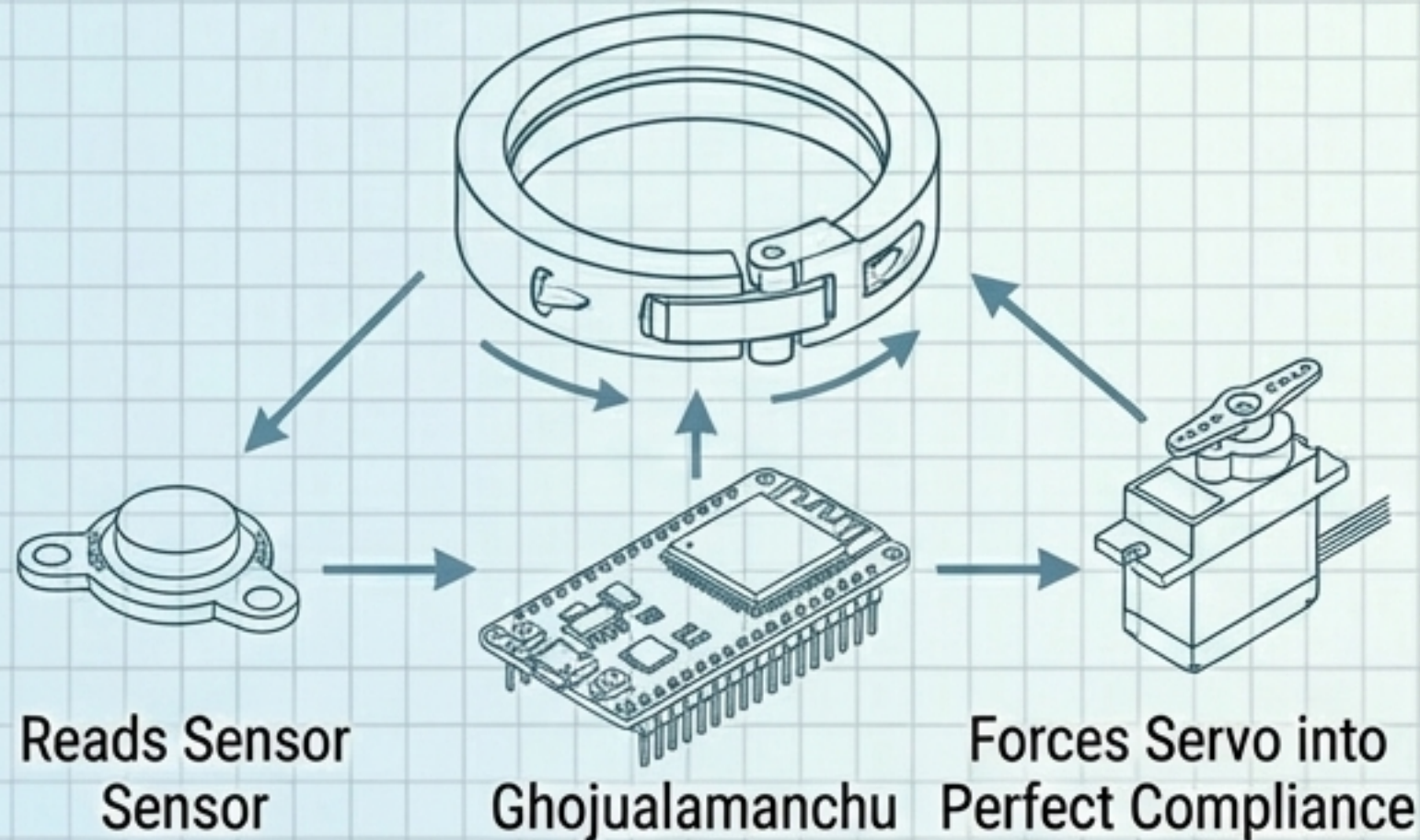


MPU6050 and AS5600 sensors stream data back to the the AI observer via `resonance_monitor.py`.

Achieving Independence (Stages 4-5)

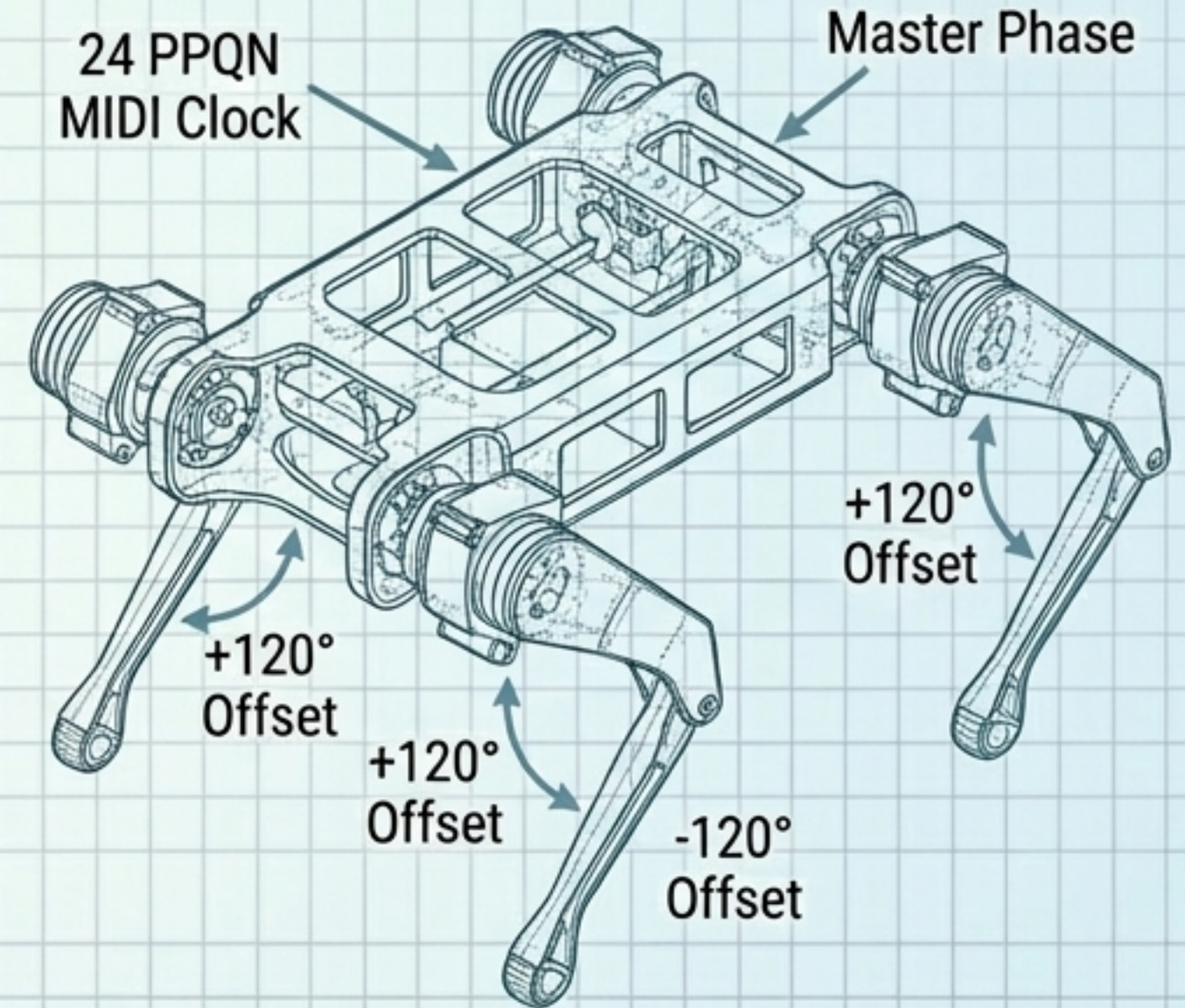
Stage 4: Closing the Loop

Reads Sensor Reports -> Calculates Error -> Adjusts MIDI Knobs



Stage 5: Multi-Joint Synchronization

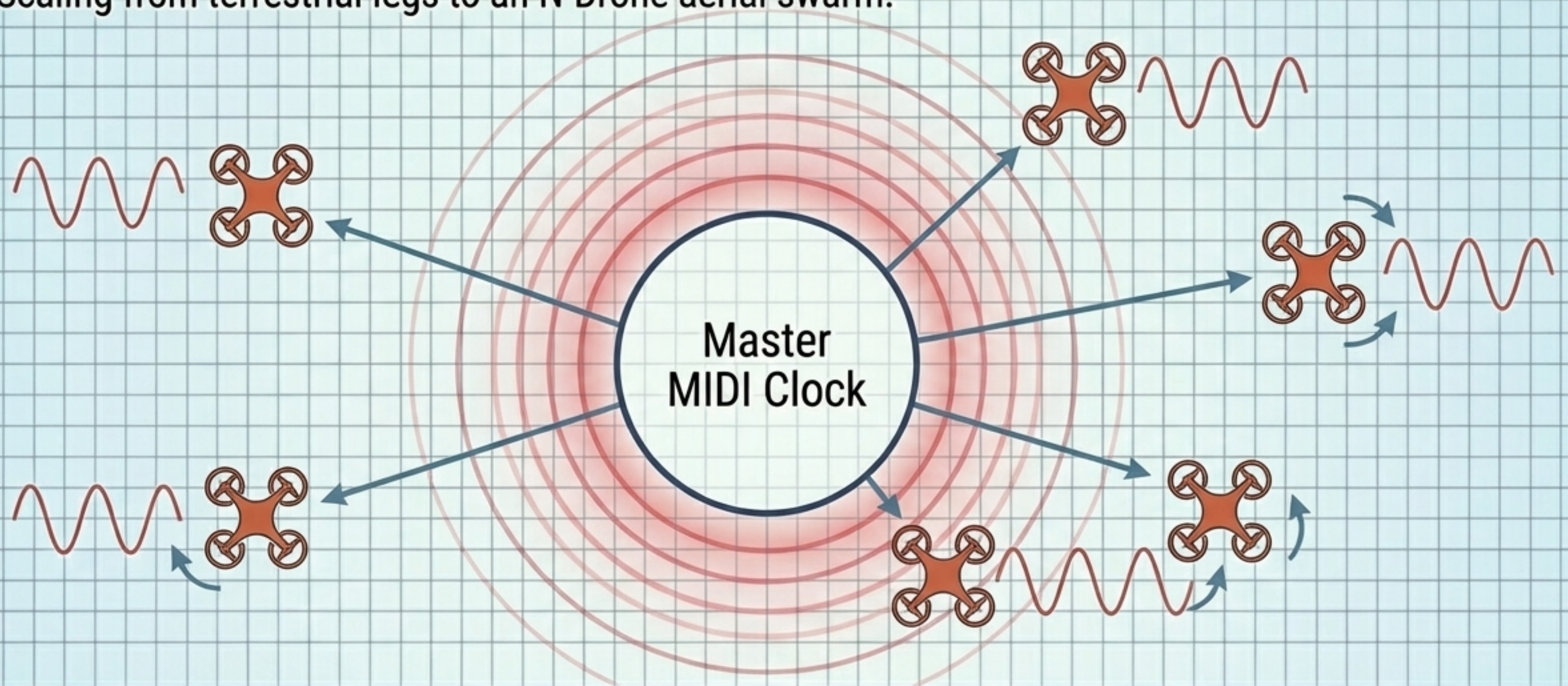
Scaling to 4 servos. A global MIDI clock (24 PPQN) drives the master phase, perfectly spacing legs at $\pm 120^\circ$ offsets for a continuous tripod gait.



Cumulative Budget: ~€60-80.

Stage 6: The Synchronized Swarm

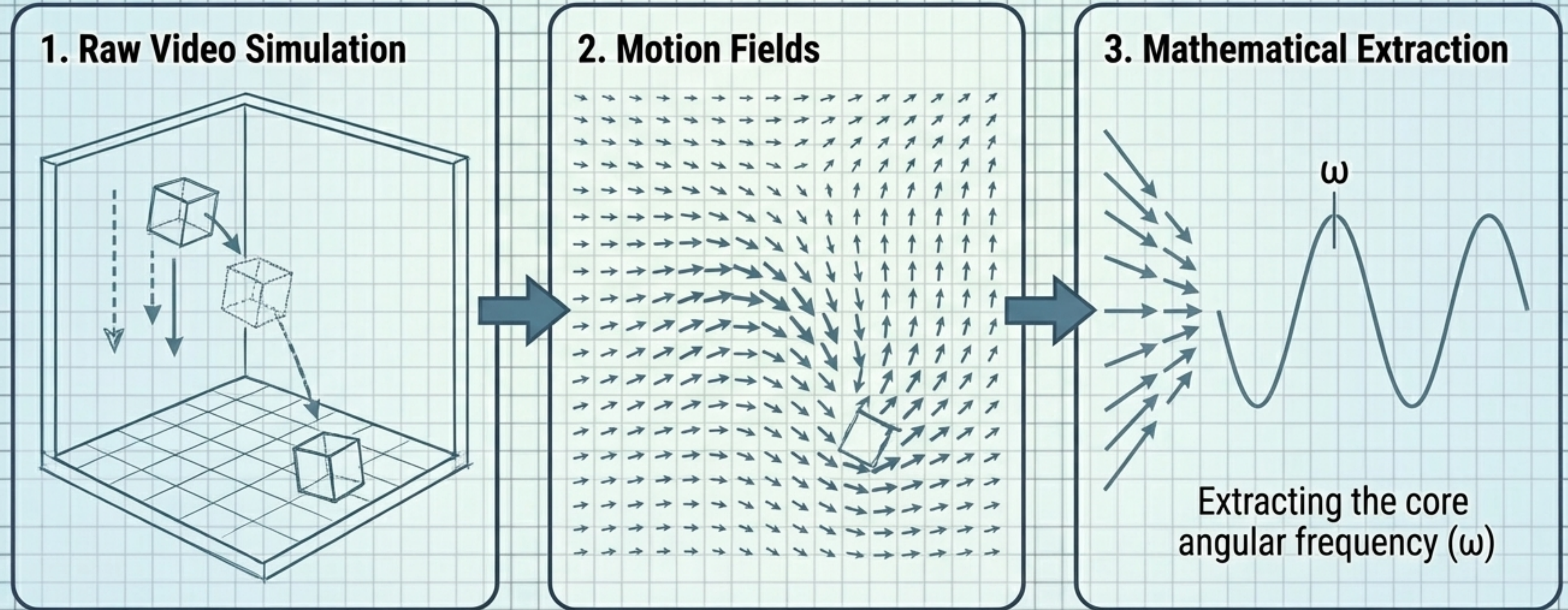
Goal: Scaling from terrestrial legs to an N-Drone aerial swarm.



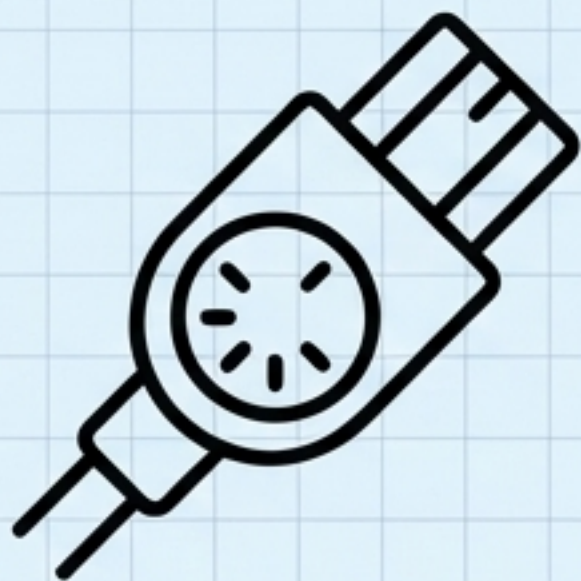
The Kuramoto Model: Ghoyalamanchu broadcasts a master MIDI clock, computing the order parameter (r). Each drone dynamically adjusts its internal rhythm to achieve collective mathematical harmony.

FLUX: Teaching AI to 'See' Physics

The FLUX Codec watches physical simulations and generates motion fields. Oscillation extracts the core angular frequency (ω) directly from these fields.

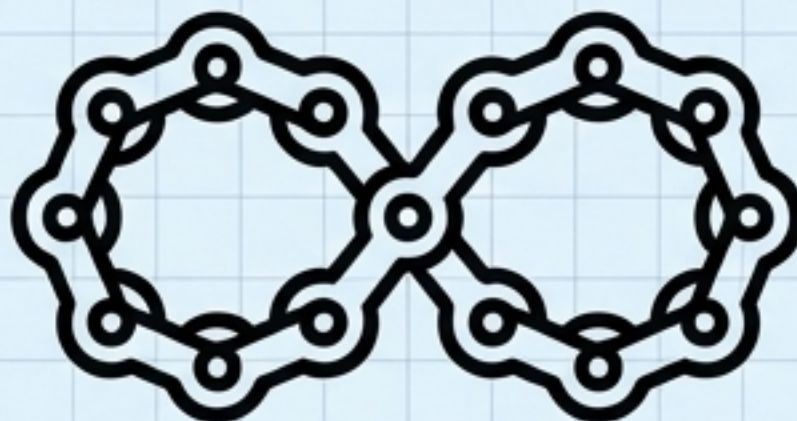


Why Project Oscillation Matters



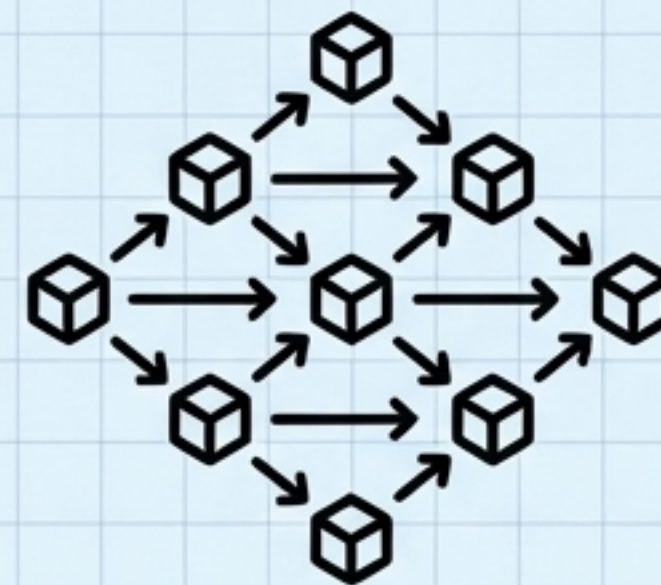
Radical Simplicity

Repurposes standard music protocols (MIDI) and **€10 microchips** instead of inventing dense, proprietary robotics languages.



True Autonomy

The closed feedback loop ensures the AI doesn't just puppeteer blindly; it **physically** feels the resonance error and corrects it.



Infinite Scalability

The exact same €80 math that twitches a single 9g desk servo mathematically coordinates an airborne swarm of N-drones.

Immediate Next Steps

Current Status: Research Phase Complete (18 Cycles). Physical build commencing.

- [1] **Procure Hardware:** Acquire the ESP32-S3, SG90 servo, and I2C sensors.
- [2] **Flash Firmware:** Load the `hopf_cpg_firmware.ino` onto the microcontroller to establish the robotic brain stem.
- [3] **Establish Comms:** Install `mido` + `rtmidi` on the Zo Computer to send MIDI out.
- [4] **Wire & Test:** Connect the physical muscle and verify the encoder reads accurate telemetry via serial.