On-Device Spiking Neural Network Locomotion Learning on a €100 Quadruped: Sim-to-Real with Brain Persistence

Marc Hesse

Independent Researcher, Potsdam, Germany

mhflocke.com | github.com/MarcHesse/mhflocke

Abstract

This paper presents a complete Sim-to-Real pipeline for quadruped locomotion using biologically grounded spiking neural networks (SNNs) on a €100 Freenove Robot Dog Kit (FNK0050) with a Raspberry Pi 4. The system employs 232 Izhikevich neurons with reward-modulated spike-timing-dependent plasticity (R-STDP), a central pattern generator (CPG) for innate gait rhythm, and a cerebellar forward model for balance correction. Training occurs in MuJoCo simulation using a custom MJCF model of the Freenove hardware, achieving 8.2 m forward distance with zero falls in 50,000 steps. The trained brain transfers to real hardware via a Bridge architecture that maps SNN motor outputs to servo commands with real-time IMU feedback from an MPU6050 sensor. On-device learning enables the robot to reach actor competence 1.0 within 2,000 steps (40 seconds at 50 Hz). Brain persistence across sessions is demonstrated: a loaded brain achieves competence 1.0 from step 1, while a fresh brain requires 2,000 steps. Spectral analysis confirms the SNN produces independent motor patterns distinct from the CPG signal. The same architecture runs on the Unitree Go2 in simulation (45.15 ± 0.67 m, 10 seeds), demonstrating cross-embodiment transfer. All code is open source under Apache 2.0. This work extends the MH-FLOCKE framework described in Hesse (2026).

Keywords: spiking neural network, Izhikevich neuron, R-STDP, quadruped locomotion, Sim-to-Real, on-device learning, brain persistence, Raspberry Pi, Freenove, CPG, cerebellum

1. Introduction

Spiking neural networks (SNNs) offer a biologically plausible alternative to conventional deep reinforcement learning for robot control. While recent work has demonstrated SNN-based locomotion in simulation environments [1, 2, 3], the transfer to real hardware remains underexplored, particularly on affordable platforms accessible to independent researchers and educators.

Most existing SNN-robotics research operates exclusively in simulation [1, 3], uses custom neuromorphic hardware such as Intel Loihi or SpiNNaker [4, 5] that is not widely available, employs leaky integrate-and-fire (LIF) neurons that lack the rich dynamics of biological neurons [2], or performs inference only without on-device learning [6]. No prior work has demonstrated the full combination of: (a) biologically realistic Izhikevich spiking neurons, (b) reward-modulated STDP for on-device learning, (c) Sim-to-Real transfer, (d) brain persistence across sessions, and (e) real-time IMU feedback—all on a consumer-grade €100 robot kit.

This paper addresses this gap by presenting a complete Sim-to-Real pipeline using the MH-FLOCKE framework [7]. I demonstrate that a spiking neural network with 232 Izhikevich neurons can learn quadruped locomotion in MuJoCo simulation, transfer to a Freenove Robot Dog Kit (FNK0050) running on a Raspberry Pi 4, and continue learning on-device with real-time IMU feedback. The brain state persists across sessions, enabling cumulative learning. I provide spectral analysis of the SNN's motor output, an A/B test of brain persistence, and cross-embodiment results showing the same architecture running on the Unitree Go2 in simulation.

The Freenove-specific code (Bridge, Dashboard, MJCF model) will be released as part of the MH-FLOCKE GitHub repository (github.com/MarcHesse/mhflocke) under Apache 2.0.

2. Related Work

SNN-based locomotion control has been explored through several approaches. Rostro-González et al. [1] implemented a CPG using spiking neurons on an Arduino-based quadruped, but used fixed synaptic weights without on-device learning. Tang et al. [2] demonstrated a fully spiking RL framework for legged robots in Isaac Gym simulation but did not transfer to real hardware. Vandesompele et al. [3] used FORCE learning with spiking reservoirs for the Tigrillo quadruped in simulation. Bellegarda and Ijspeert [8] combined CPG with deep RL (CPG-RL) but used conventional ANNs rather than SNNs.

The Neural Circuit Architectural Priors approach [9] demonstrated that biologically inspired ANN architectures provide strong priors for quadruped locomotion, achieving good innate performance with fewer parameters than MLPs. The present work extends this philosophy to actual spiking neurons with R-STDP learning.

Neuromorphic hardware platforms such as Intel Loihi [4] and SpiNNaker [5] offer efficient SNN execution but require specialized hardware not available to most researchers. The approach presented here approach runs entirely on a standard Raspberry Pi 4, making it reproducible for under €200 total hardware cost.

3. System Architecture

The system architecture is described in detail in Hesse (2026) [7]. A brief summary of the components relevant to the Sim-to-Real transfer.

3.1 Spiking Neural Network

The SNN consists of 232 Izhikevich neurons (hardware deployment) organized in a feedforward topology. Izhikevich neurons were chosen over LIF models for their ability to reproduce diverse firing patterns observed in mammalian cortex (tonic spiking, phasic bursting, chattering) using only two coupled differential equations:

$$dv/dt = 0.04v^2 + 5v + 140 - u + I$$
$$du/dt = a(bv - u)$$

with reset condition: if $v > 30$ mV, then $v := c$ and $u := u + d$. Synaptic plasticity follows reward-modulated STDP (R-STDP), where synaptic weight changes are gated by a dopaminergic reward signal derived from locomotion performance.

3.2 Central Pattern Generator

A sinusoidal CPG provides innate diagonal-pair gait at 0.8 Hz with 12 mm stride length and 6 mm lift height. The CPG operates from the first step, providing stable locomotion before the SNN has learned. A competence gate linearly blends CPG and SNN outputs: $motor_{out} = (1 - \alpha) \times CPG + \alpha \times SNN$, where α (actor competence) increases as the SNN demonstrates stable walking. The floor value for CPG contribution is 40%, ensuring baseline stability.

3.3 Cerebellar Forward Model

A cerebellar module implements the Marr-Albus-Ito architecture with granule cells, Purkinje cells, and deep cerebellar nuclei. The cerebellum provides predictive balance corrections based on proprioceptive feedback, operating as a forward model that anticipates the consequences of motor commands.

4. Hardware Platform

4.1 Freenove Robot Dog Kit (FNK0050)

The Freenove FNK0050 is a commercially available quadruped robot kit priced at approximately €100. It features 12 degrees of freedom (3 per leg: hip yaw, hip pitch, knee), driven by SG90 micro servos (1.8 kg·cm torque) via a PCA9685 PWM driver over I2C. The onboard IMU (MPU6050, I2C address 0x68) provides 6-axis accelerometer/gyroscope data. Body dimensions are 136 × 76 mm with 55 mm leg segments and an approximate standing height of 99 mm. Total mass including the Raspberry Pi 4 is approximately 620 g.

Parameter	Value
Kit	Freenove FNK0050
Price	~€100 (excl. Raspberry Pi)
Compute	Raspberry Pi 4 (4 GB RAM)
DOF	12 (3 per leg) + 1 head (unused)
Servos	12× SG90 (1.8 kg·cm)
IMU	MPU6050 (I2C @ 0x68)
Servo Driver	PCA9685 (I2C @ 0x40)
Body	136 × 76 mm, ~620 g
Standing Height	~99 mm
Power	USB-C (5V 3A)

Table 1: Freenove FNK0050 hardware specifications.

4.2 MJCF Model

A custom MuJoCo MJCF model (v3.2) was created for the Freenove robot based on physical measurements and the manufacturer's assembly documentation. Key parameters include reference joint angles of -38° for hip pitch and 91° for knee flexion, derived from the Freenove inverse kinematics with a nominal foot position of (x=10, y=99, z=10) mm. The model includes 12 actuators (the head servo was excluded for locomotion training) and named foot contact geoms (FL, FR, RL, RR) for the foot contact sensor. Offscreen rendering resolution is 2560×1440 for video production.

4.3 Bridge Architecture (v2.5)

The Bridge translates between the MH-FLOCKE brain architecture and the Freenove hardware. It operates at 50 Hz (~25 ms per step including IMU reading) and performs the following per-step operations: (1) read IMU pitch/roll/yaw from MPU6050, (2) encode sensory input as 18-channel tensor (12 joint positions + 4 IMU channels + 2 CPG phase), (3) run SNN inference (232 neurons, 5,154 synapses), (4) blend CPG and SNN outputs via competence gate, (5) convert motor commands to servo angles via inverse kinematics, (6) write to PCA9685 servo driver. IMU data feeds directly into the SNN input layer (channels 14–17) and modulates the reward signal with a 20% stability bonus based on the upright metric.

Leg	Hip Yaw	Hip Pitch	Knee
Front Left (FL)	Ch 4	Ch 3	Ch 2
Rear Left (RL)	Ch 7	Ch 6	Ch 5
Rear Right (RR)	Ch 8	Ch 9	Ch 10
Front Right (FR)	Ch 11	Ch 12	Ch 13

Table 2: PCA9685 channel mapping for the Freenove robot.

5. Sim-to-Real Transfer

The Sim-to-Real transfer follows a three-stage pipeline: (1) train the SNN brain in MuJoCo simulation using the Freenove MJCF model, (2) export the trained brain state (synaptic weights, neuron parameters) as a checkpoint file, (3) load the checkpoint on the Raspberry Pi and run inference with real sensor feedback.

Importantly, the SNN continues learning on the real hardware. The Bridge architecture provides the same reward signal structure as the simulation (forward velocity + upright stability), enabling R-STDP to refine synaptic weights based on real-world dynamics. This is distinct from approaches that freeze network weights after simulation training.

The primary challenges in the transfer were: (a) matching the MJCF joint reference angles to the physical servo calibration, (b) handling the sim-to-real gap in servo dynamics (the SG90 servos have lower torque and slower response than the simulated actuators), and (c) integrating real IMU data into the reward computation. Challenge (a) was resolved by deriving reference angles from the Freenove inverse kinematics code. Challenge (b) was mitigated by the CPG's 40% floor contribution, which provides stable gait even when the SNN output is imprecise. Challenge (c) was addressed by mapping MPU6050 pitch/roll readings to the same upright metric used in simulation.

6. Experiments

6.1 Simulation Results

The SNN was trained for 50,000 steps in MuJoCo using the Freenove MJCF model on flat terrain. Training was performed on a standard desktop PC (no GPU required). Results are summarized in Table 3.

Metric	Value
Steps	50,000
Forward Distance	8.222 m
Falls	0
Upright Streak	50,000 (never fell)
Actor Competence	0.847
Final CPG Weight	48%
Neurons	4,650
Step Time	179 ms (6 sps)
PCI	0.166

Table 3: Simulation results (Freenove MJCF, 50k steps).

6.2 Hardware Deployment

The trained brain was deployed on the Freenove robot via the Bridge v2.5 architecture. The SNN ran at 50 Hz with 232 neurons and 5,154 synapses. IMU feedback from the MPU6050 provided real pitch/roll data at each step. The robot achieved actor competence 1.0 and maintained stable walking across multiple sessions. Table 4 shows representative IMU readings during walking.

Condition	Pitch (°)	Roll (°)	Upright
Typical walking	+3.4	+2.6	0.90
During step	-7.9	-2.5	0.82
Near-stumble	-4.3	-14.3	0.67
Perfect balance	+0.1	+0.1	1.00

Table 4: Representative IMU readings during hardware walking.

At step 650 during session 3, a near-stumble event was recorded with roll of -14.3° and upright metric dropping to 0.67. The actor competence briefly decreased from 1.000 to 0.996 before recovering at the next step. This demonstrates the system's robustness: the CPG floor (40%) maintained gait stability while the

SNN adjusted.

6.3 Brain Persistence: A/B Test

To verify that brain persistence provides meaningful benefit, I conducted an A/B test on the physical hardware. Test A started with a fresh brain (randomized synaptic weights). Test B loaded a brain with 31,684 accumulated steps across 8 prior sessions.

Metric	Test A (Fresh)	Test B (Loaded)
Initial Competence	0.000	1.000
Initial CPG Weight	89%	40%
Steps to Comp. 1.0	2,000 (40 s)	1 (instant)
Peak Firing Rate	0.112 (step 400)	0.155 (step 350)
Mean DA Signal	0.68	0.67
Duration	60 s (2,243 steps)	60 s (2,244 steps)
Prior Experience	None (randomized)	31,684 steps (8 sessions)
Accumulated After Test	2,243 steps	33,928 steps (9 sessions)

Table 5: A/B test results for brain persistence on real hardware.

The results demonstrate unambiguous benefit of brain persistence. The loaded brain (Test B) achieved competence 1.0 from the first step with CPG already at its floor value of 40%, indicating that the SNN had fully taken over locomotion control. The fresh brain (Test A) required 2,000 steps (40 seconds at 50 Hz) to reach the same competence level, with CPG weight gradually decreasing from 89% to 40%. The higher peak firing rate in Test B (0.155 vs 0.112) suggests that the loaded brain's synaptic weights encode more active firing patterns.

6.4 SNN Signal Analysis

To determine whether the SNN produces independent motor patterns or merely copies the CPG signal, I performed spectral analysis of the joint position trajectories from the 50,000-step simulation run, comparing early-phase (steps 1,000–5,000, CPG at 90%) with late-phase (steps 35,000–49,000, CPG at ~45%) motor output.

Metric	Early (CPG 90%)	Late (CPG ~45%)	Interpretation
Spectral Entropy	3.87 bits	6.86 bits	SNN adds frequency content
Fundamental Energy	43.1%	0.2%	CPG sine wave replaced
Periodicity Dev.	0.123	0.040	More regular (conservative)
Joint Variance	0.016	0.005	Smaller movements
Diagonal Asymmetry	0.101	0.067	More symmetric

Table 6: Spectral analysis of SNN vs CPG motor output.

The spectral entropy increase from 3.87 to 6.86 bits and the near-complete elimination of CPG fundamental energy (43.1% $\rightarrow$ 0.2%) confirm that the SNN produces qualitatively different motor patterns from the CPG. However, the decrease in joint variance (0.016 $\rightarrow$ 0.005) and periodicity deviation (0.123 $\rightarrow$ 0.040) indicate that the SNN learns **conservative dampening** rather than active gait improvement. The SNN makes movements smaller and more regular than the CPG alone.

This finding is biologically plausible: young animals learning to walk initially adopt cautious, small-amplitude movements before developing confident, large-amplitude gaits [10]. The conservative strategy may reflect an optimal response to the reward signal, which penalizes instability but does not explicitly reward speed or stride length. Future work will investigate reward shaping that incentivizes more dynamic locomotion.

6.5 Cross-Embodiment Transfer

The same MH-FLOCKE architecture was previously evaluated on the Unitree Go2 quadruped in MuJoCo simulation [7]. A 10-seed ablation study at 50,000 steps on hilly terrain yielded:

Configuration	Distance (m)	Std Dev
SNN + Cerebellum (B1)	45.15	±0.67
Full System (C1)	45.15	±0.67
PPO Baseline	12.83	±7.78

Table 7: Cross-embodiment results on Unitree Go2 (from [7]).

The MH-FLOCKE architecture outperformed PPO by 3.5× on the Go2 with 11.6× lower variance across seeds. The same architecture, without modification to the SNN topology, CPG structure, or reward function, was then deployed on the Freenove robot with a different MJCF model, different number of neurons (232 vs 4,650), and different servo hardware. This demonstrates that the biological priors—CPG for innate rhythm, R-STDP for learning, cerebellar model for balance—generalize across robot morphologies.

7. Discussion

This work demonstrates a complete Sim-to-Real pipeline for SNN-based locomotion on affordable hardware. Several aspects merit discussion.

Conservative dampening. The SNN learns to reduce joint amplitude rather than increase locomotion speed. While this may appear suboptimal, it reflects a rational strategy under the current reward structure: the upright stability component dominates the reward, encouraging the SNN to prioritize not falling over forward progress. Biological locomotion development follows a similar trajectory, with young animals initially prioritizing stability over speed [10]. Future work will explore multi-phase reward curricula that gradually shift emphasis from stability to velocity.

Brain persistence. The A/B test provides clear evidence that synaptic weights encode generalizable locomotion knowledge that transfers across sessions. The loaded brain's instant competence demonstrates that R-STDP weight updates are not merely session-specific adaptations but represent accumulated motor learning. This is analogous to long-term potentiation in biological systems.

Computational efficiency. The SNN with 232 neurons runs at 50 Hz on a Raspberry Pi 4 without any optimization (pure Python with PyTorch). The 25 ms step time leaves headroom for additional sensory processing. For comparison, PPO-based approaches typically require GPU-accelerated simulation for training and inference on more capable hardware.

Limitations. The current evaluation lacks several important metrics: forward velocity comparison between SNN-active and CPG-only gaits on real hardware, energy consumption measurements, and multi-trial statistical analysis on the physical robot. The single-seed simulation result (8.2 m) for the Freenove is less robust than the 10-seed Go2 ablation. Additionally, the SNN's low firing rate (0.004–0.030) raises questions about whether all neurons contribute meaningfully to locomotion control.

8. Conclusion and Future Work

This paper has demonstrated that a biologically grounded spiking neural network can learn quadruped locomotion in simulation and transfer to real hardware on a €100 robot kit. The system achieves stable walking with on-device learning at 50 Hz, brain persistence across sessions, and cross-embodiment generalization between the Unitree Go2 and the Freenove FNK0050. All code will be available as part of the

MH-FLOCKE open-source project (github.com/MarcHesse/mhflocke, Apache 2.0).

Future work includes: (1) integration of the TF-Luna LiDAR sensor (0.1–8 m, I2C) for obstacle detection and avoidance, enabling the SNN to demonstrate reactive behavior beyond locomotion; (2) longer training sessions to investigate whether the conservative dampening behavior evolves into more dynamic gaits; (3) disturbance compensation experiments to test SNN recovery from external perturbations; (4) multi-seed hardware evaluation for statistical robustness; and (5) a second paper focused on the cognitive architecture modules (episodic memory, drives, metacognition) described in [7].

9. Ethical Statement

This research is intended for peaceful applications including education, agriculture, inspection, and assistive robotics. The author opposes the use of autonomous locomotion technology in weapons systems. The open-source release is made under Apache 2.0 to maximize accessibility for researchers and educators worldwide.

AI assistance disclosure: The author's native language is German. AI language tools (Anthropic Claude) were used to assist with English-language drafting and editing of this manuscript. All scientific content, experimental design, implementation, and analysis are solely the work of the author.

10. Supplementary Material

A demonstration video showing the MuJoCo simulation training and real hardware deployment is available at: https://www.youtube.com/watch?v=7iN8tB2xLHI

References

[1] H. Rostro-González et al., "Quadrupedal robot locomotion: A biologically inspired approach and its hardware implementation," *Front. Neurosci.*, 2015.

[2] H. Tang et al., "Fully spiking neural network for legged robots," *arXiv:2310.05022*, 2023.

[3] A. Vandesompele et al., "Closed loop control of a compliant quadruped with spiking neural networks," *BNAIC/BeneLearn*, 2019.

[4] M. Davies et al., "Loihi: A neuromorphic manycore processor with on-chip learning," *IEEE Micro*, vol. 38, no. 1, pp. 82–99, 2018.

[5] S. Furber et al., "The SpiNNaker project," *Proc. IEEE*, vol. 102, no. 5, pp. 652–665, 2014.

[6] J. Eshraghian et al., "Training spiking neural networks using lessons from deep learning," *Proc. IEEE*, 2023.

[7] M. Hesse, "MH-FLOCKE: Biologically grounded embodied cognition for quadruped locomotion learning," *Zenodo*, DOI: 10.5281/zenodo.19336894, 2026.

[8] G. Bellegarda and A. Ijspeert, "CPG-RL: Learning central pattern generators for quadruped locomotion," *IEEE RA-L*, vol. 7, no. 4, pp. 12547–12554, 2022.

[9] N. X. Bhattasali et al., "Neural circuit architectural priors for quadruped locomotion," *arXiv:2410.07174*, 2024.

[10] J. M. Blázquez and C. Dobrzecka, "Alteration in the pattern of locomotion following a partial movement restraint in puppies," *Acta Neurobiol. Exp.*, vol. 49, pp. 39–46, 1989.