

# Circuit and System Technologies for Energy-Efficient Edge Robotics

(Invited Paper)

Zishen Wan, Ashwin Sanjay Lele, Arijit Raychowdhury

School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, USA  
zishenwan@gatech.edu, alele9@gatech.edu, arijit.raychowdhury@ece.gatech.edu

**Abstract**—As we march towards the age of ubiquitous intelligence, we note that AI and intelligence are progressively moving from the cloud to the edge. The success of Edge-AI is pivoted on innovative circuits and hardware that can enable inference and limited learning in resource-constrained edge autonomous systems. This paper introduces a series of ultra-low-power accelerator and system designs on enabling the intelligence in edge robotic platforms, including reinforcement learning neuromorphic control, swarm intelligence, and simultaneous mapping and localization. We put an emphasis on the impact of the mixed-signal circuit, neuro-inspired computing system, benchmarking and software infrastructure, as well as algorithm-hardware co-design to realize the most energy-efficient Edge-AI ASICs for the next-generation intelligent and autonomous systems.

## I. INTRODUCTION

With intelligence marching from the cloud to the edge, autonomous robots are rising in real-world deployment. They require data to be processed from sensing to actuation in a closed-loop manner. The onboard circuits typically take power from the battery, limiting supported operational time. Recently, with neural networks being deployed at the edge, the demanding real-time performance and energy constraints are becoming increasingly difficult to achieve. Aerial platforms have an additional constraint of payload which limits the weight of the onboard sensors and compute. For tiny terrestrial robots, the form factor becomes a crucial requirement. Therefore, there is a strong need to develop energy-efficient compute platforms and processing frameworks to enable robotic edge intelligence.

Several hardware platforms have been used for robotic applications. CPUs and GPUs are designed to handle a wide range of robotic tasks and algorithms development. However, they usually consume 10-100 W of power, which are orders of magnitude higher than available resources on edge robotic systems. Edge processors like Jetson Nano and Google TPU have been used for programmability at a smaller form factor for faster system-level prototyping [1]. FPGAs are attracting attention due to its reconfigurability and hardware-efficiency, and have been presented for robotic perception [2], [3], localization [4], and planning [5]. The partial reconfiguration technique takes this flexibility one step further, where part of FPGA resources can be reconfigured at runtime without compromising the other parts of applications [6], [7].

Edge-robotic specialized ASICs boost the energy-efficiency and customization even further. The constrained power budget on edge robots requires these chips to have a few mW of power consumption. For robotic perception, Jeon et al. [8] present

a feature extraction accelerator at 2.7 mW power for micro aerial vehicles. For robotic autonomous navigation, Navion [9] accelerates the visual-inertial odometry of nano drones, Li et al. [10] speed up mutual information computation, and Li et al. [11] combine neural network with physical localization model with proposed specialized accelerators. For end-to-end learning-based robotic control, Kim et al. [12] develop a reinforcement learning accelerator for micro drones with 1.1 mW power consumption. For multi-robots scenarios, Honkote et al. [13] design a low-power SoC for distributed and collaborative swarm robot systems. Different techniques ranging from voltage-mode circuits [14], quantized neural networks [15]–[17] and sparse coding [18] have been utilized in restricting the power consumption. Several neural network accelerators have been proposed for general vision applications [19] which can be readily applied to robotic tasks.

Apart from energy consumption, memory bandwidth availability also becomes critical for high-speed edge robotic applications. A newer type of visual sensor called event cameras is being explored for such scenarios. Event cameras provide a stream of asynchronous events, and only part of frames are processed at every time instance, allowing high bandwidth, high speed, and high dynamic range [20]. System-level applications of this include a 3 ms latency robotic goalie [21] and looming obstacle avoiding drone [22]. The event-based processing modality of event cameras also lends itself naturally to processing bio-inspired spiking neural networks that benefit from data sparsity and low power. Tasks like lane detection and prey capturing [23] have been demonstrated using spiking neural networks and event camera pairs.

In this paper, we will show a series of our proposed ultra-low-power accelerators and system demonstrations for edge robotics, with an emphasis on circuit and system technologies. These demonstrations augment the large landscape of edge robotic systems in reinforcement and swarm learning and neuro-inspired computing technologies. Section II presents a time-domain mixed-signal accelerator that supports embedded reinforcement learning control. Section IV introduces a hybrid-digital mixed-signal hardware design to enable edge swarm intelligence. Section IV presents an oscillator-based hardware platform for neuro-inspired mapping, localization, and control. Section V further provides benchmarking and simulation frameworks for cross-layer robotic workload evaluation. The paper concludes with challenges and opportunities for next-generation hardware-efficient robotic computing (Section VI).

## II. REINFORCEMENT LEARNING ON THE EDGE

This section presents our reinforcement learning neuromorphic accelerator for robotic autonomous navigation and exploration [24], [25]. We propose an energy-efficient time-domain mixed-signal computational framework to enable ultra-low-power operations.

### A. Reinforcement Learning Algorithm

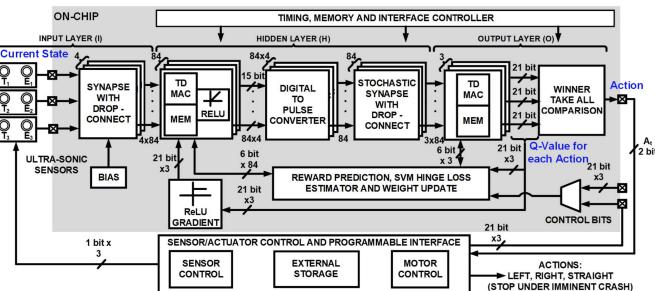
Achieving true autonomy requires robots to perform continuous learning through frequently interacting with environments. Reinforcement learning (RL) is such a paradigm where robots take actions in environments to maximize the notion of cumulative reward. Among all RL algorithms, Q-learning is one of the well-studied techniques, which is implemented in this design. Q-learning seeks to find the optimal action policy ( $A_t$ ) given the current state ( $S_t$ ) to maximizes the reward ( $R_t$ ). It works on the principle of the action-value function [ $Q(S_t, A_t)$ ], where  $Q$  is iteratively updated from Bellman equation. For a detailed overview of RL algorithms and applications, interested readers are referred to [26].

### B. Circuit and System of RL Neuromorphic Accelerator

#### 1) Overview:

Fig. 1 shows the system diagram of our RL neuromorphic accelerator. It consists of ultrasonic sensors, an RL test chip, a Raspberry Pi-based micro-controller, and motor drivers.

The ultrasonic sensors feed depth information to the input layer of the neural network through an array of stochastic synapses. A three-layer neural network is implemented to process sensor data and generate actions that robots will follow. The micro-controller stores ( $S_t, A_t, R_t, S_{t+1}$ ) in a scratchpad memory during training and sends action commands from the chip to the motor controllers.

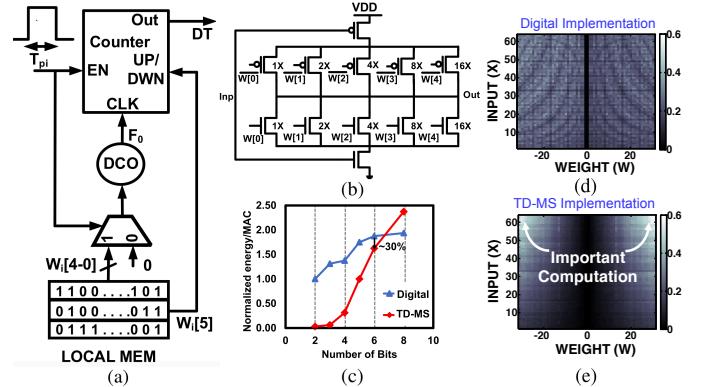


**Fig. 1:** The system architecture of our RL neuromorphic accelerator, including different circuit blocks and the interface to the external micro-controllers and motor drivers.

#### 2) Time-Domain Mixed-Signal Circuits:

To enable low-power edge intelligence on the robotic platform with low bit precision, we propose a time-domain mixed-signal (TD-MS) computational framework for energy-efficient and accurate operations. Fig. 2a illustrates the TD-MS multiply-and-accumulate (MAC) unit. A pulse input from sensor or hidden layers is used to enable the up-down counter that is triggered by a digitally controlled oscillator (DCO). (Fig. 2b). The output of the counter is the product of weight  $W_i$  (digital) and pulse width  $T_{pi}$  of the gating signal (analog).

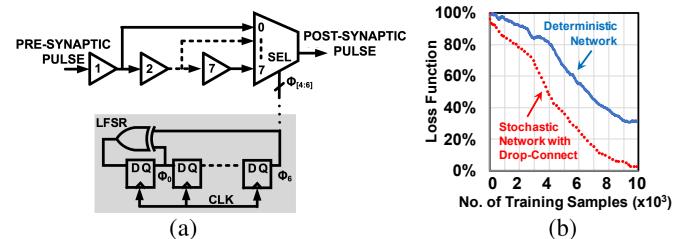
The proposed TD-MS design demonstrates unique advantages: 1) improved energy-efficiency at lower bit width compared to digital MAC operation (Fig. 2c); (2) the energy to compute is proportional to the significance of the computation - a feature in human brain but missing in digital logic (Fig. 2d-2e); (3) 45% lower system area, 47% lower interconnect power and 16% lower leakage power compared to digital design.



**Fig. 2:** (a) Proposed Time-Domain Mixed-Signal (TD-MS) MAC unit. (b) Digitally controlled oscillator (DCO) in TD-MS unit. (c) Computational energy per MAC for digital and TD-MS design with different bit precision at 0.6 V. (d)(e) 2-D energy per MAC surface for digital and TD-MS design with 6-bit inputs at 0.6 V.

#### 3) Enabling Regularization via Stochasticity:

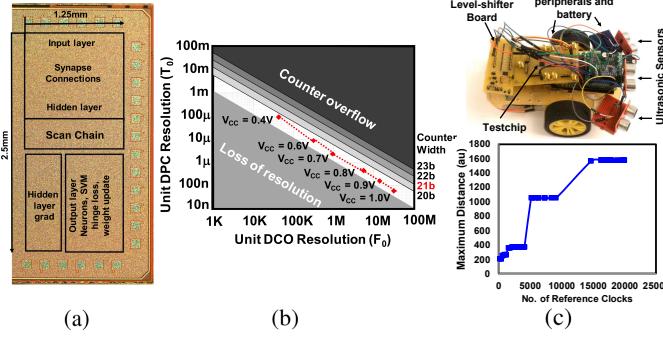
To help generalize the learned neural network model to unknown environments, we introduce stochasticity to synapses with drop-connect. As shown in Fig. 3a, the stochasticity is implemented with a buffer chain whose delays are randomly altered by a local high-speed linear-feedback-shift register (LFSR). Compared to the deterministic model, the stochastic network can achieve 1.7 $\times$  speedup in convergence (Fig. 3b).



**Fig. 3:** (a) Stochasticity is implemented by introducing varying delays between bit transitions using the linear-feedback-shift register (LFSR). (b) Stochasticity accelerates policy convergence.

### C. Evaluation

The RL neuromorphic test chip is implemented and taped-out in 55 nm CMOS process (Fig. 4a). By exploring design space of proposed TD-MS circuits, we show that the chip can ensure correct functionality and voltage scalability from 1.0 V down to 0.4 V (Fig. 4b). The chip is mounted on a tiny mobile robot for autonomous exploration and learning. Fig. 4c illustrates the increasing moved distance when robot continuously explores the space in the presence of obstacles.



**Fig. 4:** (a) RL neuromorphic chip die-photo. (b) Design space showing scalability till 0.4 V. (c) Mobile robot system and its covered distance as a function of the number of clock cycles or iterations.

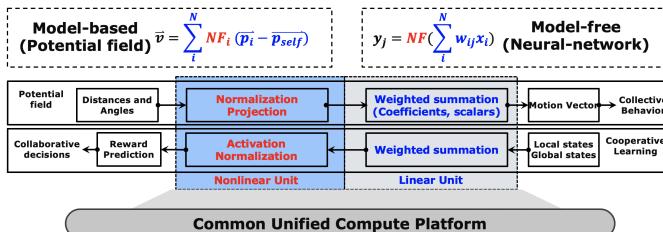
### III. SWARM INTELLIGENCE ON THE EDGE

Going beyond single tasks, when facing a variety of tasks, swarms of robots usually collaborate with each other to solve problems. This section presents our energy-efficient hardware accelerator that enables swarm robotic applications [27], [28].

#### A. Swarm Intelligence Algorithms

Swarm algorithms can be broadly classified into two categories, ones based on physical and mathematical models (model-based) and ones based on learning (model-free). Among model-based algorithms, artificial potential field (APF) is a commonly-used approach for collaborative path planning, where the motion force is obtained by aggregating the attractive and repulsive potential field. Model-free algorithms usually allow each robot to learn continuously to establish a model with real-world knowledge without human intervention. RL-based cooperative algorithms have shown great promise. Interested readers are directed to [29] for more details.

Interestingly, we observe that both model-based and model-free autonomy paradigms have similar mathematical structures (Fig. 5). Therefore, we identify the commonalities and develop a unified architecture to support real-time swarm intelligence.



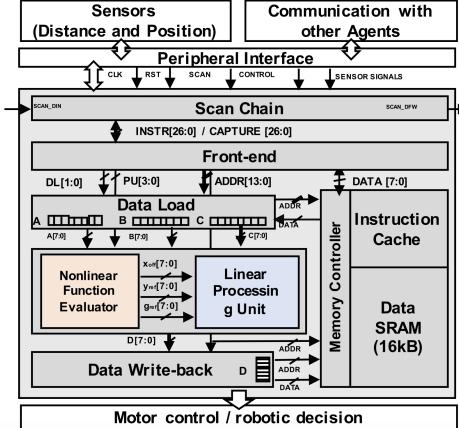
**Fig. 5:** Common unified compute paradigm that supports both model-based and model-free swarm algorithms.

#### B. Circuit and System of Swarm Intelligence Accelerator

##### 1) Overview:

The system architecture of our swarm intelligence accelerator is shown in Fig. 6. Noticing that both model-based and model-free are combinations of nonlinear and linear operations, we design a nonlinear function evaluator (NFE) and linear processing unit (LPU). NFE supports nonlinear

operations by using a piecewise linear approximation of non-linear functions. LPU supports all addition and multiplication linear operations. Most operations are implemented in the digital domain except for MAC that leverages mixed-signal computing. The datapath of NFE and LPU is bi-directional, so data can move between each other seamlessly and preserve locality. We also observe that several required functions show symmetry and periodicity, which provides a further chance to reduce the number of computations and comparisons.



**Fig. 6:** The system architecture of our swarm intelligence accelerator, supporting both swarm-based and swarm-free autonomy paradigms with a nonlinear function evaluator and linear processing unit.

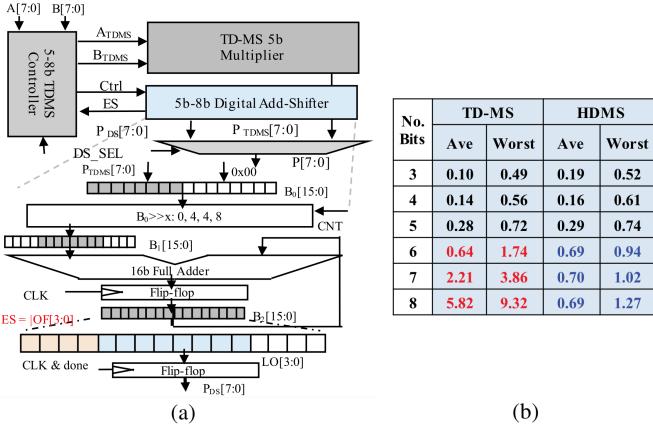
##### 2) Hybrid-Digital Mixed-Signal Circuits Architecture:

Swarm algorithms need to support various swarm sizes in dynamic environments. The required bit precision will increase from 3-bit to 8-bit, with swarm size increasing from 2 agents to 20 agents. TD-MS MACs show energy advantages over digital counterparts for low bit width, but exhibit higher energy with increasing operand size (Fig. 2c). To address this issue, we propose a hybrid-digital mixed-signal (HD-MS) MAC kernel, where computation is purely TD-MS for bit-width  $\leq 5$ , and hybrid of TD-MS and digital for  $6 \leq \text{bit-width} \leq 8$ .

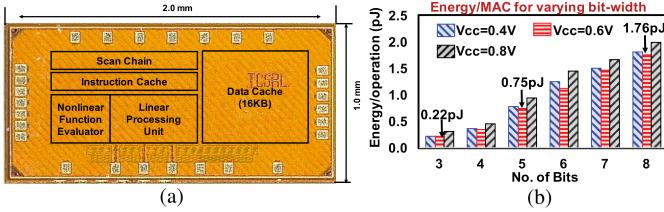
Fig. 7a shows the circuit schematic of the HD-MS design. The HD-MS MAC kernel consists of a conventional TD-MS multiplier, a 5-8-bit TD-MS controller, and a 5-8-bit digital adder-shifter. The TD-MS multiplier computes  $\leq 5$ -bit operation. The TD-MS controller and digital adder-shifter reconfigure the multiplier to a higher bit width with seamless shift-and-add operations. Fig. 7b demonstrates that HD-MS have energy-benefits at higher precision compared with TD-MS. Compared with digital implementation, HD-MS exhibits 81% (for 3-bit) to 31% (for 8-bit) energy per MAC reduction.

##### C. Evaluation

Our swarm intelligence accelerator is fabricated in 65 nm CMOS process (Fig. 8a). Fig. 8b demonstrates its scalability with bit precision, indicating a peak energy efficiency of 0.22 pJ/MAC (for 3-bit) and 1.76 pJ/MAC (for 8-bit). The average energy efficiency varies from 9.1 TOPS/W (for 3-bit) to 1.1 TOPS/W (for 8-bit). This bit precision scalability allows efficient computation for various swarm sizes.

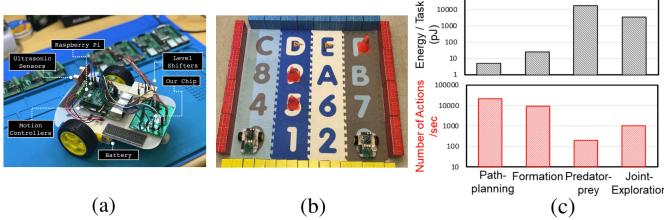


**Fig. 7:** (a) Circuit schematic of the HD-MS design, including the 5-bit TD-MS kernel and the digital peripherals to enable efficient scaling to 8-bit. (b) Energy/MAC (normalized to a digital implementation) for TD-MS and HD-MS implementations. We observe that HD-MS outperforms TD-MS and digital for large swarm sizes.



**Fig. 8:** (a) Swarm intelligence chip die-photo. (b) Measured energy per MAC across different bit widths at VCC = 0.4, 0.6, and 0.8 V.

We mounted the chip on a robotic car (Fig. 9a and Fig. 9b). The platform interfaces with a Raspberry Pi, sensors, motor controllers, and radios. We implement four exemplified swarm intelligence algorithms, namely path planning, pattern formation, predator-prey, and joint-exploration, where the first two are model-based and the last two are model-free. Fig. 9c shows large variations in energy consumption and the number of actions for each task, illustrating that future robotic platforms need to support a wide variety of algorithms, as the complexities of environments and task can dramatically change.



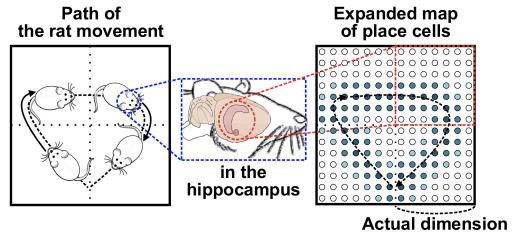
**Fig. 9:** (a) Our swarm intelligence accelerator is mounted on a robotic car with peripheral circuits. (b) Experimental setup. (c) Energy and performance for different template swarm algorithms.

#### IV. NEURO-INSPIRED COMPUTING ON THE EDGE

Recently, spiking neural networks (SNNs), with bio-inspired sparse spike-based temporally coded computing, offer an energy-efficient alternative for edge robotic tasks. This section presents our proposed incorporation of SNNs for SLAM and prey tasks for ultra-low-power robotic applications [30], [31].

#### A. Simultaneous Localization and Mapping Algorithms

Simultaneous localization and mapping (SLAM) forms an essential component of many autonomous navigation applications. SLAM algorithms can be broadly classified into two categories: visual-based and neuro-based. Visual-based SLAM requires the robot to identify its position from the beginning of motion and generate a map of the movement using only the images captured during motion. Previous approaches like probabilistic SLAM and keyframe-based SLAM remain inadequate due to the constrained power budget of edge systems. Neuro-based SLAM performs the computation in a more energy-efficient way and is applicable to ultra-low-power edge robotics. Neuroscientific exploration in rodent brains showed their phenomenal capacity to efficiently localize themselves (Fig. 10). Place cells and head direction cells are identified as neuronal circuits tuned to respond at a particular position and direction of motion, respectively. Our NeuroSLAM accelerator takes inspiration from RatSLAM [32] algorithm that mimics the neuromorphic connectivity and incorporates bio-inspired hardware to achieve ultra-low-power SLAM tasks.



**Fig. 10:** Mapping between the position of rodent and the excited place cells in rodent's brain

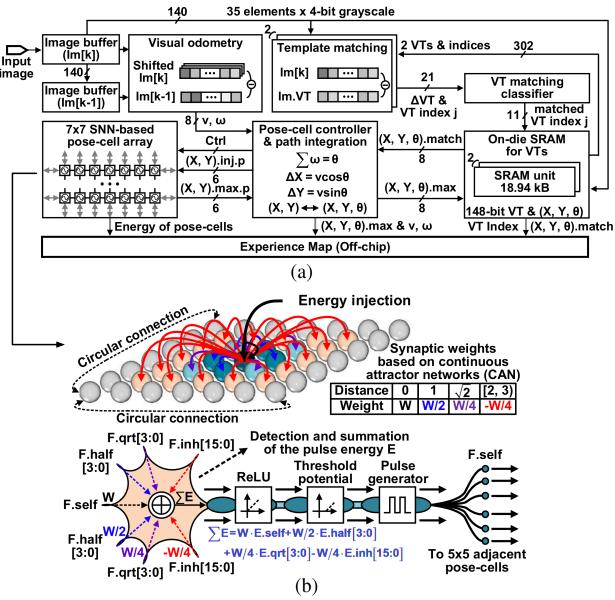
#### B. Circuit and System of NeuroSLAM Accelerator

##### 1) Architecture and Circuit Block:

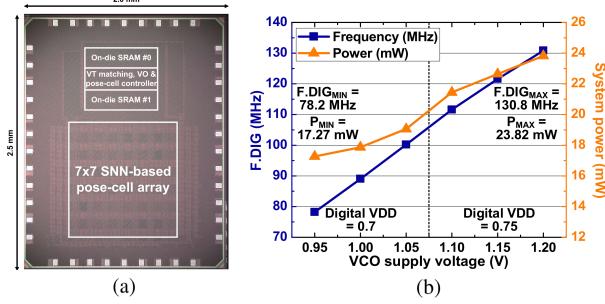
The flow of data in the NeuroSLAM accelerator is shown in Fig. 11a, where the captured image is first compared with the previous image for visual odometry to estimate the displacement from previous image capture. This is followed by template matching with previous images to detect any possibility of loop closure. The translation is added with the previous position to find the current direction of motion with digital head direction cells, and the path integration is injected in the pose cell array. Pose cell array is made of oscillator-based continuous attractor network in spiking neural network fashion, as shown in Fig. 11b. The output is extracted to calculate the experience map.

##### 2) Test Chip Measurement:

The NeuroSLAM test chip is fabricated in 65 nm technology (Fig. 12a). The template matching, odometry, and path integration are carried out in low-bit resolution with an efficient attractor network. The chip achieves SLAM with only 23.82 mW. The dependence of power consumption on input voltage is shown in Fig. 12b. The attractor network shows a high compute efficiency of 8.79 TOPS/W. Our NeuroSLAM accelerator demonstrates the incorporation of neuro-inspired analog hardware in the digital pipeline to enable edge intelligence in severe energy-constrained systems.



**Fig. 11:** (a) Overview of our NeuroSLAM accelerator. (b) Spiking continuous attractor network mimicking rodent’s pose cell behavior.



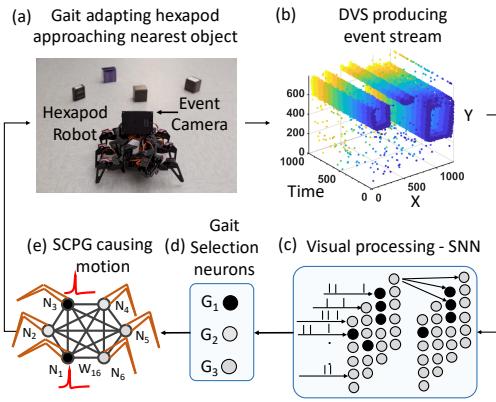
**Fig. 12:** (a) NeuroSLAM chip die-photo. (b) Measured operational frequency and power consumption of NeuroSLAM accelerator.

### C. Neuro-Inspired End-to-End Spike-Only Processing

Another neuro-inspired system exploration is the first autonomous sensing-to-actuation end-to-end spike-only processing pipeline for hexapod robots [33]. The goal is to demonstrate the functionality of spike-only processing and evaluate the potential of event-driven processing modalities. As shown in Fig. 13, event camera/dynamic vision sensor (DVS) is used as the sensory input to generate asynchronous event stream. The information is processed through SNN to activate one of the three gait selection neurons. The central pattern generator (CPG) is trained such that every gait selection neuron activates gait in a different direction to allow controlled movement. A task of identifying and approaching the nearest target is demonstrated using this platform. The ultra-low-power (2.55 mJ/step) consumption of this system with dedicated spiking hardware (e.g., Loihi) highlights the potential of neuro-inspired event-driven systems for edge applications.

## V. BENCHMARKING AND SOFTWARE INFRASTRUCTURE

Edge robotics is a cross-layer research field, spanning from environment modeling, autonomy algorithms, runtime systems to onboard compute architecture and circuits. The interaction



**Fig. 13:** Bio-inspired closed-loop end-to-end spike-only robot. (a) Event camera mounted on hexapod provides (b) input stream to (c) SNN for processing and (d) selecting gait to approach the nearest object. (e) Gait is executed by a spiking CPG causing leg movements.

between the layers impacts the efficacy and performance of the system [34], requiring software infrastructure for interdisciplinary research. Specially, we need a platform that can systematically benchmark each of these individual layers and also capture end-to-end cross-layer execution characteristics. Recently, some platforms for aerial robots have been proposed. MAVBench [35] is a platform for physical-model-based aerial robots evaluation. MAVBench involves a closed-loop simulator and a benchmark suite for several computational kernels involving perception, planning, and control. Airlearning [36] and PEDRA [37] are simulation suits and benchmarks for learning-based aerial edge robots. They provide a rich set of virtual worlds, including indoor and outdoor environments, to enable autonomy generalization. In addition, PEDRA supports swarm intelligence with different collaboration paradigms.

These benchmarking and software infrastructures inspire a proliferation of studies on edge robotic systems, such as reliability, system design methodology, and memory hierarchy optimization. Based on MAVBench, RoboRun [38] presents a robot runtime that leverages spatial-aware computing to dynamically improve performance and energy for heterogeneous operating environments. MAVFI [39] proposes a fault injection framework for end-to-end reliability characterization of robotic workload, which is portable to robot operating system (ROS)-based applications. Based on AirLearning, Autopilot [40] and AutoSoC [41] propose an automated design space exploration framework to design optimal onboard compute for aerial robots. Based on PEDRA, Anwar et al. [42] present a transfer learning-based approach to reduce the onboard computation required to train a neural network for autonomous navigation. Wan et al. [43] propose application-aware lightweight fault mitigation techniques to enable reliable autonomy under hardware faults. Anwar et al. [44] evaluate the robustness of swarm robotic systems under adversaries. Yoon et al. [45] present a novel hierarchically memory system with STT-MRAM and SRAM to support real-time learning-based robotic exploration. We believe a holistic benchmarking and simulator infrastructure will uncover more cross-layer research findings of various fields of edge robotics.

## VI. CHALLENGES AND OPPORTUNITIES

Robotic computing is a rising area and opens many research challenges and opportunities. At the device and circuit level, embedded non-volatile memory (e.g., RRAM, Ferroelectric memories) and 3D integration will provide opportunities for hardware-efficient robotic computing. At the architecture level, the robotic computing platform needs to be adaptive and reconfigurable to various scenarios and applications, including DNNs and SNNs. At the system level, a holistic benchmark and a generic framework for mapping autonomy algorithms to heterogeneous hardware platforms will benefit the robotic computing development process. At the algorithm level, robots need to have the ability of lifelong learning and learning with limited data. The booming of swarm intelligence requires more effective and robust distributed learning across multiple agents.

## VII. CONCLUSION

This paper investigates circuit and system technologies for energy-efficient edge robotics. We present three robotic accelerators for edge reinforcement learning, swarm intelligence, and neuron-inspired mapping and localization, with an emphasis on novel mixed-signal circuit technologies. We summarize robotic benchmarking and simulation frameworks along with how they foster research efforts in cross-layer robotic systems. Finally, we discuss the challenges and opportunities for the next-generation energy-efficient robotic computing platforms.

## ACKNOWLEDGEMENTS

This work was supported in part by C-BRIC, one of six centers in JUMP, a Semiconductor Research Corporation (SRC) program sponsored by DARPA.

## REFERENCES

- [1] S. Chinchali *et al.*, “Network offloading policies for cloud robotics: a learning-based approach,” in *Autonomous Robots*, 2021.
- [2] T. Gao *et al.*, “ielas: An elas-based energy-efficient accelerator for real-time stereo matching on fpga platform,” in *AICAS*, 2021.
- [3] Z. Wan *et al.*, “An energy-efficient quad-camera visual system for autonomous machines on fpga platform,” in *AICAS*, 2021.
- [4] W. Liu *et al.*, “Archytas: A framework for synthesizing and dynamically optimizing accelerators for robotic localization,” in *MICRO*, 2021.
- [5] S. Murray *et al.*, “The microarchitecture of a real-time robot motion planning accelerator,” in *MICRO*, 2016.
- [6] Z. Wan *et al.*, “A survey of fpga-based robotic computing,” in *IEEE CAS-M*, 2021.
- [7] S. Liu *et al.*, “Robotic computing on fpgas,” in *Synthesis Lectures on Computer Architecture*, 2021.
- [8] D. Jeon *et al.*, “A 470mv 2.7 mw feature extraction-accelerator for micro-autonomous vehicle navigation in 28nm cmos,” in *ISSCC*, 2013.
- [9] A. Suleiman *et al.*, “Navion: A 2-mw fully integrated real-time visual-inertial odometry accelerator for autonomous navigation of nano drones,” in *JSSC*, 2019.
- [10] P. Z. X. Li, *High-throughput computation of shannon mutual information on chip*. PhD thesis, Massachusetts Institute of Technology, 2020.
- [11] Z. Li *et al.*, “An 879gops 243mw 80fps vga fully visual cnn-slam processor for wide-range autonomous exploration,” in *ISSCC*, 2019.
- [12] Y. Kim *et al.*, “A 0.55 v 1.1 mw ai processor with on-chip pvt compensation for autonomous mobile robots,” in *IEEE TCAS-I*, 2017.
- [13] V. Honkote *et al.*, “2.4 a distributed autonomous and collaborative multi-robot system featuring a low-power robot soc in 22nm cmos for integrated battery-powered minibots,” in *ISSCC*, 2019.
- [14] J. Zhang *et al.*, “18.4 a matrix-multiplying adc implementing a machine-learning classifier directly with data conversion,” in *ISSCC*, 2015.
- [15] D. Bankman *et al.*, “An always-on 3.8muj cifar-10 mixed-signal binary cnn processor with all memory on chip in 28-nm cmos,” in *JSSC*, 2018.
- [16] M. Lam *et al.*, “Quantized reinforcement learning (quarl),” *arXiv preprint arXiv:1910.01055*, 2019.
- [17] T. Tambe *et al.*, “Algorithm-hardware co-design of adaptive floating-point encodings for resilient deep learning inference,” in *DAC*, 2020.
- [18] F. N. Buhler *et al.*, “A 3.43 tops/w 48.9 pj/pixel 50.1 nj/classification 512 analog neuron sparse coding neural network with on-chip learning and classification in 40nm cmos,” in *VLSI Circuits*, 2017.
- [19] Y.-H. Chen *et al.*, “Eyeriss: An energy-efficient reconfigurable accelerator for deep convolutional neural networks,” in *JSSC*, 2016.
- [20] G. Gallego *et al.*, “Event-based vision: A survey,” *arXiv preprint arXiv:1904.08405*, 2019.
- [21] T. Delbruck and M. Lang, “Robotic goalie with 3 ms reaction time using event-based dynamic vision sensor,” in *Frontiers in neuroscience*, 2013.
- [22] D. Falanga *et al.*, “Dynamic obstacle avoidance for quadrotors with event cameras,” in *Science Robotics*, 2020.
- [23] L. Salt *et al.*, “Obstacle avoidance with lgmd neuron: towards a neuromorphic uav implementation,” in *ISCAS*, 2017.
- [24] A. Amravati *et al.*, “A 55nm time-domain mixed-signal neuromorphic accelerator with stochastic synapses and embedded reinforcement learning for autonomous micro-robots,” in *ISSCC*, 2018.
- [25] A. Amravati *et al.*, “A 55-nm, 1.0–0.4 v, 1.25-pj/mac time-domain mixed-signal neuromorphic accelerator with stochastic synapses for reinforcement learning in autonomous mobile robots,” in *JSSC*, 2018.
- [26] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.
- [27] N. Cao *et al.*, “14.1 a 65nm 1.1-to-9.1 tops/w hybrid-digital-mixed-signal computing platform for accelerating model-based and model-free swarm robotics,” in *ISSCC*, 2019.
- [28] N. Cao *et al.*, “A 65-nm 8-to-3-b 1.0–0.36-v 9.1–1.1-tops/w hybrid-digital-mixed-signal computing platform for accelerating swarm robotics,” in *JSSC*, 2019.
- [29] H. M. La *et al.*, “Multirobot cooperative learning for predator avoidance,” in *IEEE TCST*, 2014.
- [30] J.-H. Yoon and A. Raychowdhury, “31.1 a 65nm 8.79 tops/w 23.82 mw mixed-signal oscillator-based neuroslam accelerator for applications in edge robotics,” in *ISSCC*, 2020.
- [31] J.-H. Yoon and A. Raychowdhury, “Neuroslam: A 65-nm 7.25-to-8.79-tops/w mixed-signal oscillator-based slam accelerator for edge robotics,” in *JSSC*, 2020.
- [32] M. J. Milford and G. F. Wyeth, “Mapping a suburb with a single camera using a biologically inspired slam system,” in *IEEE T-RO*, 2008.
- [33] A. Lele *et al.*, “An end-to-end spiking neural network platform for edge robotics: From event-cameras to central pattern generation,” *IEEE TCDS*, 2021.
- [34] S. Krishnan *et al.*, “The sky is not the limit: A visual performance model for cyber-physical co-design in autonomous machines,” in *CAL*, 2020.
- [35] B. Boroujerdian *et al.*, “Mavbench: Micro aerial vehicle benchmarking,” in *MICRO*, 2018.
- [36] S. Krishnan *et al.*, “Air learning: An ai research platform for algorithm-hardware benchmarking of autonomous aerial robots,” *arXiv preprint arXiv:1906.00421*, 2019.
- [37] M. A. Anwar and A. Raychowdhury, “Navren-rl: Learning to fly in real environment via end-to-end deep reinforcement learning using monocular images,” in *M2VIP*, 2018.
- [38] B. Boroujerdian *et al.*, “Roborun: A robot runtime to exploit spatial heterogeneity,” in *DAC*, 2021.
- [39] Y.-S. Hsiao *et al.*, “Mavfi: An end-to-end fault analysis framework with anomaly detection and recovery for micro aerial vehicles,” *arXiv preprint arXiv:2105.12882*, 2021.
- [40] S. Krishnan *et al.*, “Machine learning-based automated design space exploration for autonomous aerial robots,” *arXiv preprint arXiv:2102.02988*, 2021.
- [41] S. Krishnan *et al.*, “Autosoc: Automating algorithm-soc co-design for aerial robots,” *arXiv preprint arXiv:2109.05683*, 2021.
- [42] A. Anwar and A. Raychowdhury, “Autonomous navigation via deep reinforcement learning for resource constraint edge nodes using transfer learning,” in *IEEE Access*, 2020.
- [43] Z. Wan *et al.*, “Analyzing and improving fault tolerance of learning-based navigation systems,” in *DAC*, 2021.
- [44] A. Anwar and A. Raychowdhury, “Multi-task federated reinforcement learning with adversaries,” *arXiv preprint arXiv:2103.06473*, 2021.
- [45] I. Yoon *et al.*, “Hierarchical memory system with stt-mram and sram to support transfer and real-time reinforcement learning in autonomous drones,” in *IEEE JETCAS*, 2019.