Maseeh College of Engineering and Computer Science
PORTLAND STATE UNIVERSITY

General-Purpose Robotic Controller for Neuromorphic Neural Network Models

Zachary W. Clark-Williams[1], Alexander J. Hunt[2]

[1] Maseeh College of Engineering and Computer Science, Portland State University, Portland, Oregon
[2] Agile and Adaptive Robotics Lab, Portland State University, Portland, Oregon

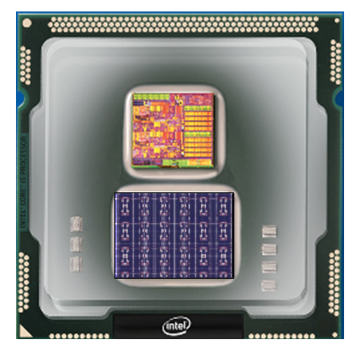
Semiconductor Research Corporation

Introduction

The fields of neuroscience and biomimetic robotics research, development, and system design is expanding rapidly. As the need for smarter controls increases the complexity of developed models and controls bring a greater need for testing and simulating. Increasingly complex models demand more from simulation and model testing. Today's software capabilities that are developed to handle such simulations and tests are growing in step with the hardware and robotic structural development, yet there is a gap in the integration of these two systems. There is a lack of systems that can load built neural control networks, process the simulation of created network and operate the robotic controls and sensor interfacing with speed that allows for real-time analysis and operation. This missing fast general-purpose controller is making research and development of biomimetic systems expensive not only in the microcontrol devices themselves, but in the time it takes to run a system, then scale results to a reasonable real world speeds. This complexity alone makes it unreasonable to expect new research students or enthusiasts to be able to get up to speed and start making research contribution in a reasonable amount of time.

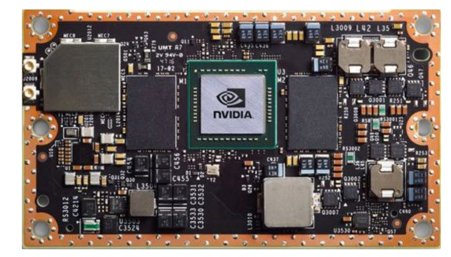
This research project is to investigate, design, build, and test a general purpose, parallel processing microcontroller for use in neural robot control. The controller will be centered around a multi-core SoC that may interface with a user's computer to load biological neuron simulations from animatlab. It will run the loaded neural calculations in parallel processes through a GPU/CPU SoC, neuromorphic processing unit, or FPGA SoC, which will output serial or parallel communication to 20+ motors, servos, solenoid valves, or other actuators. The microcontroller must also be able to receive 40+ sensor inputs for use in the control algorithms. The major goals of this research is to perform fast neural simulations and to optimize the communication between the processing unit and the robot's sensors and actuators. This will enable Portland State University's Agile and Adaptive Robotics Lab to investigate the neural control of balance and walking using robot systems in real time.

Processing Units to Choose From



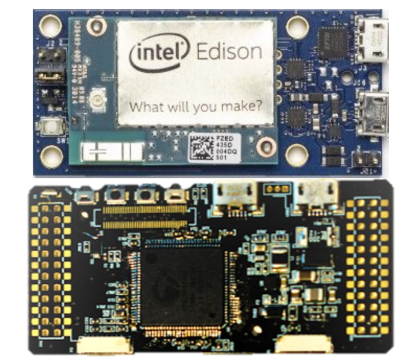
Intel Loihi

- Many-Core memresitive architecture for spiking neural net
- Highly configurable plastic synaptic controller
- 128 computing cores, each with 1024 artificial neurons
- 130,000 neurons
- 130 million configurable synaptic connections
- Fine-grain neuronal parallelism
- Processing unit is still in development stage
- Not easily accessible for all levels of researcher



NVIDIA Tegra X1

- Graphics Processing Unit - NVIDIA Maxwell ™, 256 CUDA cores
- Central Processing Unit - Quad ARM® A57/2 MB L2
- Single board computer with surplus of communication buses, GPIO, camera interface connections pre-built-in
- Well established developer community support
- Linux system with easy programming and development atmosphere
- Easily accessible for all levels of researchers



Cypress PSOC5 and Intel Edison Mini Breakout Board

- Dual-core Intel Quark x86 CPU @ 400 MHz 1G DDR3 RAM, 4 GB EMMC Storage
- 32-bit ARM Cortex-M3 core @ 80 MHz
- 18 Digital GPIO, 18 Analog Digital GPIO
- FPGA's are highly configurable and fast for real-time operational control
- Inexpensive and accessible by all levels of researchers
- PSOC5 line of SOC's out of date as of Jan. 2014

HiSilicon Kirin 960

- ARM Mali-G71 MP8 GPU (900 MHz)
  4x Cortex-A73 (2.4 GHz) + 4x Cortex-A53 (1.8 GHz)
- Fast processing SoC with GPU-CPU architecture
- Approximately 100 GPIO, 2x camera buses
- Capable of parallel programming through openMP library
- Newly created, smaller development community but well supported by HiSilicon
- Expensive and not for sale to the general public

Decision Choice and Discussion

The presented devices are not only from a wide variety of companies and price ranges, but also from a range of functional operating architectures. This style of analysis allowed for the ability to understand which type of processing unit would fulfill the the requirements of the developing system. The requirements are from the view of functional operation, student research technical understanding and ramp-up time to be able to use the device, and if not only research students could operate it but also experienced researchers who are using older systems or more complex models.

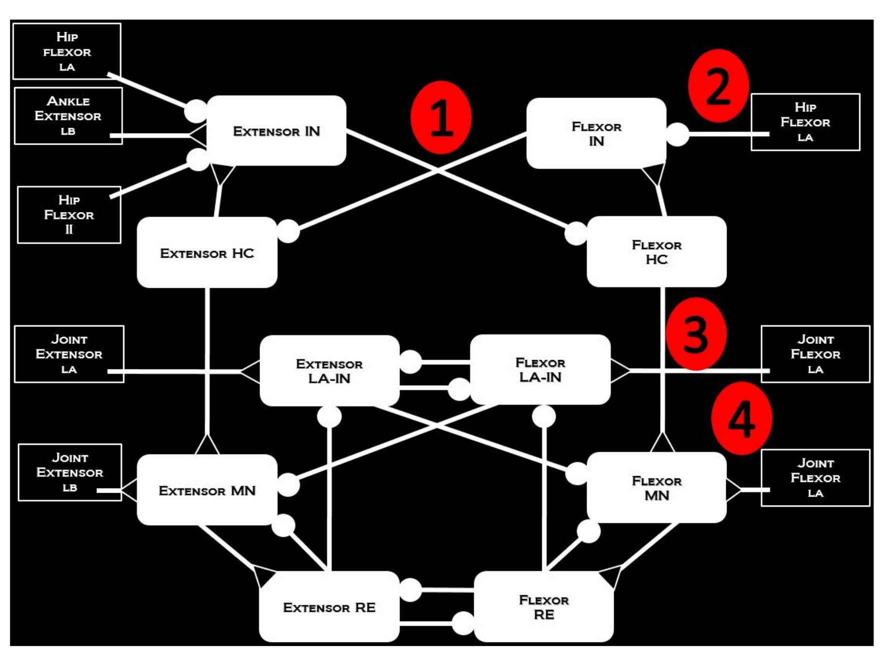
Requirements:

- Must be capable of running AnimatLab
- Fast processing speeds capable of real-time robotic control with spiking and non-spiking neural net models
- Parallel processing capability
- 100+ General Purpose Input Output interface ports, expandable if necessary
- Ease of use for new and experienced researchers alike
- Easily interfaceable and upgradable with capable of legacy as newer devices come out
- Inexpensive to moderately priced for unit, less than $300

With the above requirements taken into consideration and all the data from the options shown to the left, the unit that fits the best would be NVIDIA's Jetson TX1 single board computer with accompanying development kit. This device has a well established community of developers as well the supported development software of CUDA make the development of the robotic interfacing system easier for the continued growth of the whole computing device. The other consideration that made the Jetson TX1 a clear choice was its price, processing speed, legacy to the more capable device Jetson TX2, and its default operating system of Linux allows operation Animatlab directly on the computer

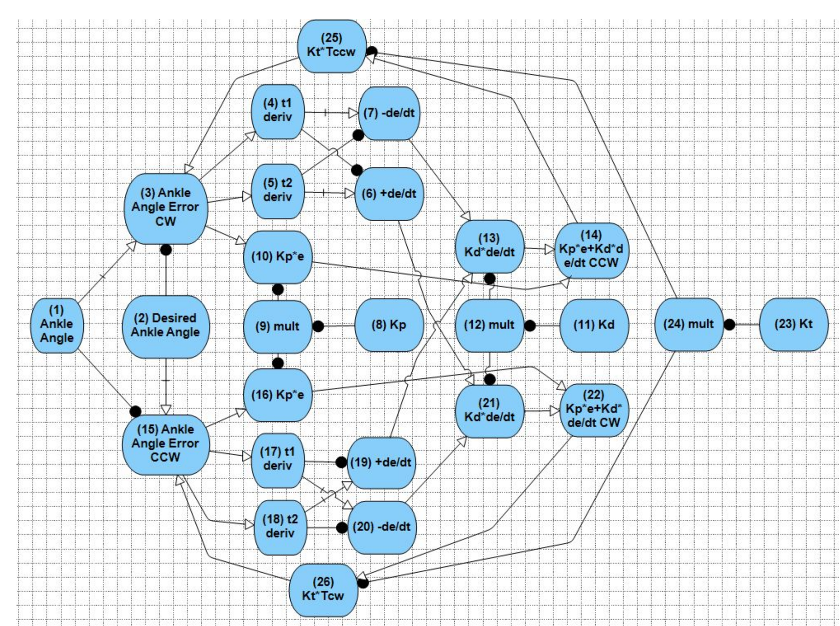
Robotic Neuromorphic Neural Net Controller Workflow and Implementation

Conceptualized Neuromorphic Neural Net Model



First the user will conceptually build the model, understanding the flow, control and system process they want to perform.

AnimatLab Generated Neuromorphic Neural Net Model

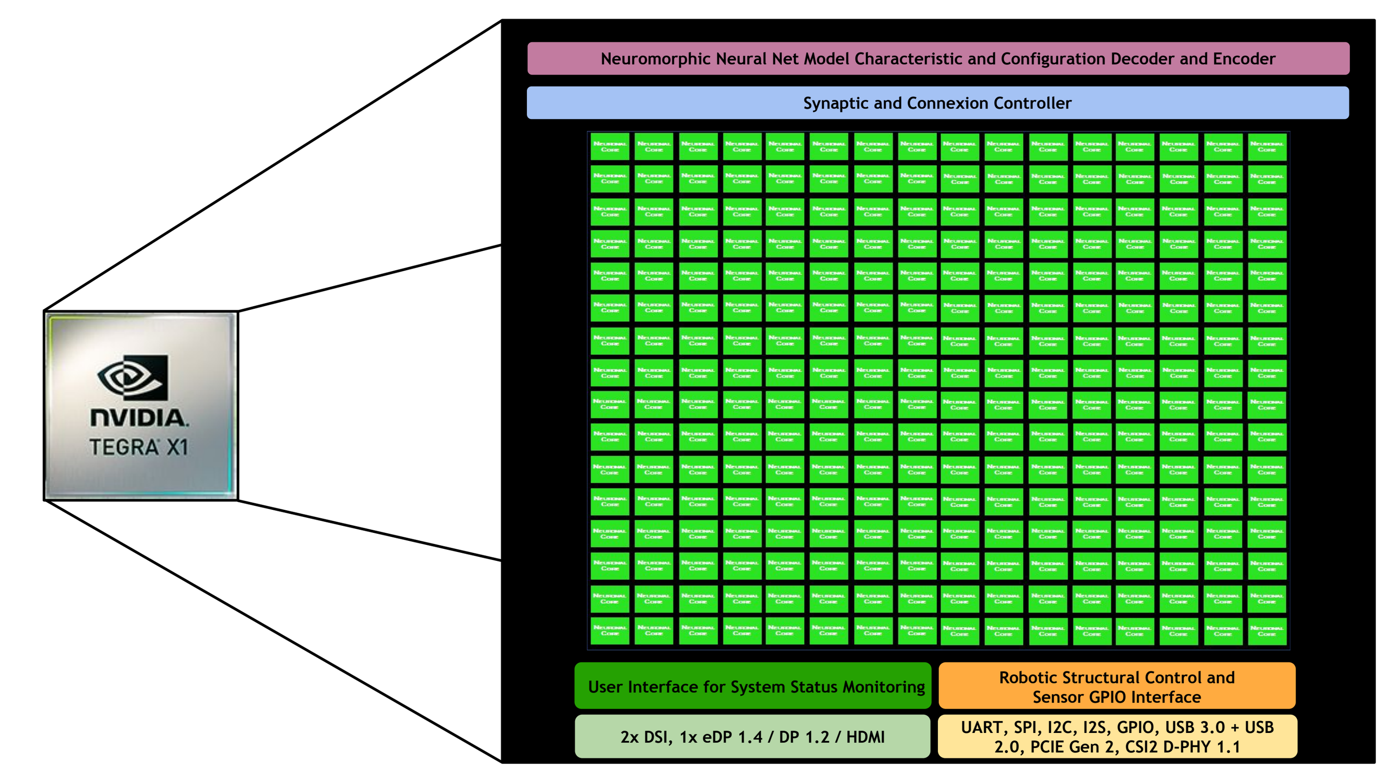


All connections will be direct, through the built model in Animatlab.

AnimatLab Generated Neuromorphic Neural Net Characteristic Configuration File

The model characteristic and configuration will be passed via configuration file to the parse and build program.

Neuromorphic Neural Net Realized in Parallel Core Architecture of the NVIDIA Tegra X1



A parser, decoder and assignment script then parses the configuration file and decodes the model, assigning neurons to separate cores. All neurons are set to be run in a parallel environment to allow fast and biologically mimicked operation. The environment can then be monitored by the robotic controller connexions and synaptic channels during operation time. The monitored processes will be fed back to a user interface where they can monitor the operation and data as the robot operates its actions.

Process Post Operating SBC Choice

Once the choice of utilizing the NVIDIA Jetson TX1 development board and kit was finalized the workflow process was designed to be as non-disruptive to the systems in place as possible. The switch to using the new board and system should be as easy as possible so that researchers from all levels of experience could easily transition to this system. The real change comes from running all systems on the Jetson TX1 instead of linking two application program interfaces through a virtual port and then outputting that information to the robotic controller via ethernet.

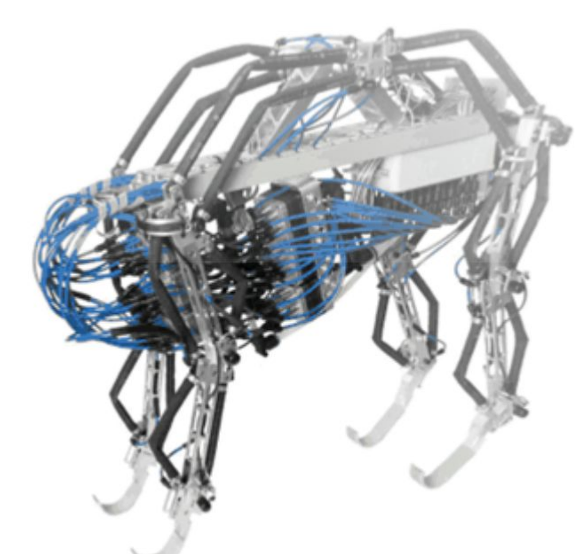
- The new system will focus on upgrading the performance of the processing speed, running as close to real-time as possible.
- The Jetson TX1 will be stripped of all unused applications to stream line and focus all power toward running the generated models and robotic controls.
- As the models grow in complexity and the need for a camera becomes evident, the operating software will be built to run custom computer vision models built through neuromorphic neural net models.
- Once the system has been full built and vetted it will be available for other universities, research teams and other entities to use it in their studies as well.

This puppy-bot, image courtesy of Alexander Hunt, is an example of the type of robots this control system and workflow is intended to be used with.

Synonymous robots to be built in Portland State University's Agile and Adaptive Robotics Lab, Dr. Hunt and his team will utilize the control system to achieve lower power consumption so that the robots need less batteries, therefore can be lighter and less prone to overheating.

A variety of assorted sensors for the developed robots to use when learning and reacting to its surrounding environment, meaning calculations, interfacing, and computational speeds are paramount for system operations.

Ideally this developed controller will make the development and control of such robots easier and more accessible for everyone.



Discussion & Continued Work

Generating this system will allow for a more accessible platform for biomimicry robotics development, neuroscientific research, and many more fields to progress through simulation and testing allowing for greater steps in the progression of science, technology, engineering, and mathematics.

- This project brought forth many questions in the capabilities of newer architecture such as the neuromorphic processing units and how they will be able to compete against the evey advancing current GPU/CPU SoC's.
- Integrating a memresister architecture into the current SoC's would allow for far more controlled and easier integration of simulations models to the real robotics platforms.
- Monitoring the performance upgrades that come from the integration of these systems will be the determining factor of whether this will be the correct path, GPU SoC's, or if the field should pivot to working on remote systems like AWS or BrainScaleS collaboration with the European Union creating the Human Brain Project.

Acknowledgements

- SRC Education Alliance award number 2009-UR-2032G
- Maseeh College of Engineering and Computer Science
- Agile and Adaptive Robotics Laboratory
- PhD. Douglas Hall

References

A. J. Hunt, N. S. Szczecinski, E. Andrada, M. Fischer, and R. D. Quinn, "Using Animal Data and Neural Dynamics to Reverse Engineer a Neuromechanical Rat Model," in *Biomimetic and Biohybrid Systems*, S. P. Wilson, P. F. M. J. Verschure, A. Mura, and T. J. Prescott, Eds. Springer International Publishing, 2015, pp. 211–222.

M. Davies *et al.*, "Loihi: A Neuromorphic Manycore Processor with On-Chip Learning," in *IEEE Micro*, vol. 38, no. 1, pp. 82-99, January/February 2018. doi: 10.1109/MM.2018.112130359

G. A. Elliott and J. H. Anderson, "Real-World Constraints of GPUs in Real-Time Systems," *2011 IEEE 17th International Conference on Embedded and Real-Time Computing Systems and Applications*, Toyama, 2011, pp. 48-54. doi: 10.1109/RTCSA.2011.46

J. Wolf *et al.*, "RTOS Support for Parallel Execution of Hard Real-Time Applications on the MERASA Multi-core Processor," *2010 13th IEEE International Symposium on Object/Component/Service-Oriented Real-Time Distributed Computing*, Carmona, Seville, 2010, pp. 193-201. doi: 10.1109/ISORC.2010.31

N. Otterness *et al.*, "An Evaluation of the NVIDIA TX1 for Supporting Real-Time Computer-Vision Workloads," *2017 IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS)*, Pittsburgh, PA, 2017, pp. 353-364. doi: 10.1109/RTAS.2017.3

AGILE AND ADAPTIVE ROBOTICS LAB

Undergraduate Research & Mentoring Program