International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Neuromorphic Computation

[1]Greeshma TS, [2]Veena S

[1]Student, Department of Electonics And Communication, S J C Institute of Technology, Chickballapur, Karnataka, India
[2]Assistant Professor (Guide), Department of Electonics And Communication, S J C Institute of Technology, Chickballapur, Karnataka, India.

ABSTRACT—

Although a variety of solutions for neuromorphic systems based on different hardware technology and software programming schemes, there has yet to be a common accepted one. Based on some recent findings in brain science, we propose a new design rule for developing a brain inspired computing system. We design and fabricate a neuromorphic chip, named 'Tianji' chip. A board multi-chip architecture-based PCB has been demonstrated. The detailed hardware implementation and software programming scheme are presented in this paper

1. Introduction

Neuromorphic computing combines computing fields such as machine learning and artificial intelligence with cutting-edge hardware development and materials science, as well as ideas from neuroscience. In its original incarnation, "neuromorphic" was used to refer to custom devices/chips that included analog components and mimicked biological neural activity [Mead1990]. Today, neuromorphic computing has broadened to include a wide variety of software and hardware components, as well as materials science, neuroscience, and computational neuroscience research. To accommodate the expansion of the field, we propose the following definition to describe the current state of neuromorphic computing:

Neuromorphic systems also have tended to emphasize temporal interactions; the operation of these systems tend to be event driven. Several properties of neuromorphic systems (including event-driven behavior) allow for low-power implementations, even in digital systems. The wide variety of characteristics of neuromorphic systems indicates that there are a large number of design choices that must be addressed by the community with input from neurophysiologists, computational neuroscientists, biologists, computer scientists, device engineers, circuit designers, and material scientists.

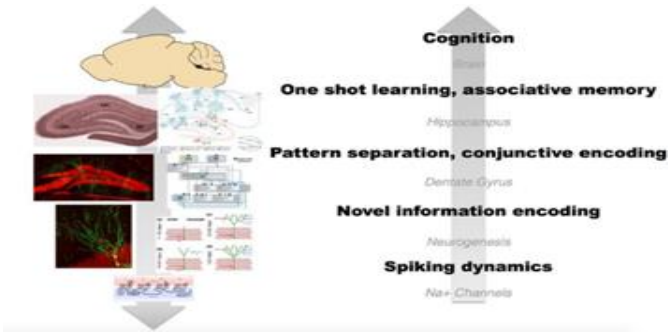

Fig: Levels of abstraction in biological brains and what functionality they may allow

2. Technology

One view of neuromorphic systems is that they represent one pole of a spectrum of repurposable computing platforms (Figure 1). On one end of that spectrum is the synchronous von Neumann architecture. The number of cores or computational units increases in moving across this spectrum, as does the asynchrony of the system.

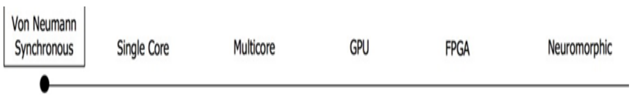

Fig 2: Spectrum of repurposable computing platforms

A second group of neuromorphic computing research is motivated by accelerating existing deep learning networks and training and thus is interested in building hardware that is customized specifically for certain types of neural networks (e.g., convolutional neural networks) and certain types of training algorithms (e.g., back-propagation).

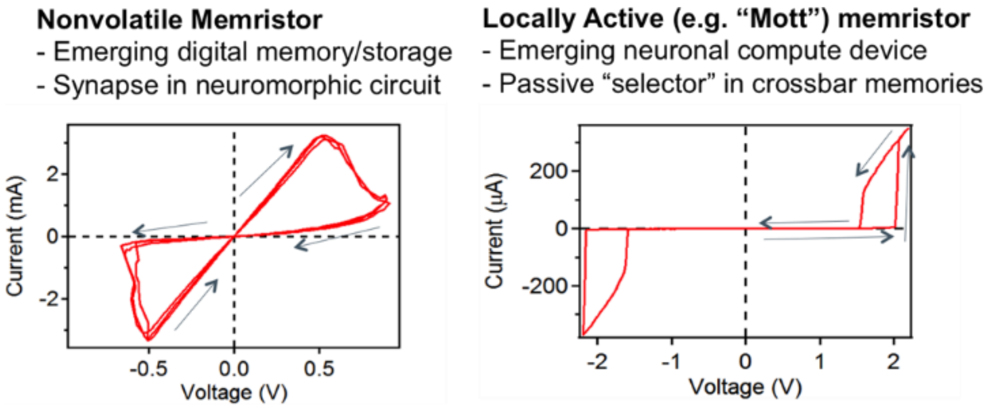

Fig 3: Two types of memristors that could be used in neuromorphic systems

The third and perhaps most common set of neuromorphic systems is motivated by developing efficient neurally inspired computational hardware systems, usually based on spiking and non-spiking neural networks. These systems may include digital or analog implementations of neurons, synapses, and perhaps other biologically inspired components. It is also worth noting that there are neuromorphic implementations using off-the-shelf commodities, such as field programmable gate arrays (FPGAs), which are useful as both prototypes systems and, because of their relative cost, have real-world value as well.

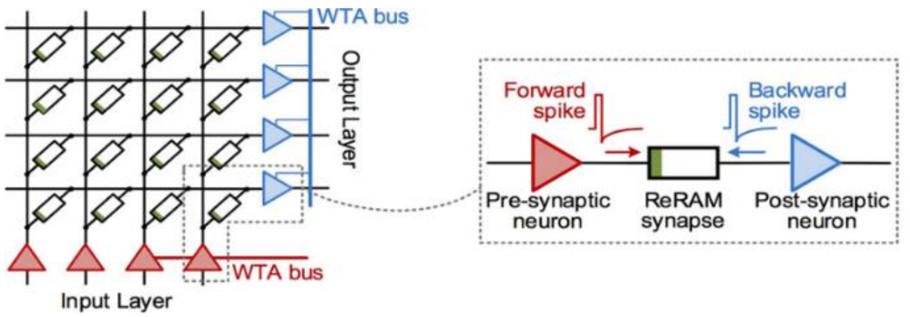

Fig 4: ReRAM used as synapses in a crossbar array

First, the neuromorphic computer itself is no longer made up of a separate memory unit and CPU; it is instead made of different computational building blocks, each of which is likely contain some combination of memory and processing. Although it is possible for there to be a programming language that is used by a programmer, it is more likely that the neuromorphic computer configuration for a particular application is determined by a training/learning algorithm that may be run either off-line and off-chip, or run on the chip itself. There could still be a compiler to turn the programming language or output from the training algorithm into configuration commands for the neuromorphic computer. Finally, the application is likely communicating with the neuromorphic computer using spikes (assuming the neuromorphic computer is representing a spiking neural network) or some other simple message type, in which case an input/output converter will likely be necessary.

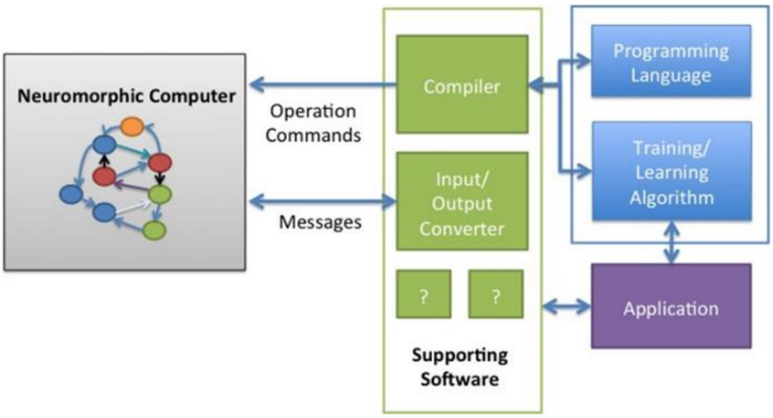

Fig 5: A potential neuromorphic architecture from the computer science perspective.

3. Advantages

Neuromorphic devices represent an attempt to mimic aspects of the brain's architecture and dynamics with the aim of replicating its hallmark functional capabilities in terms of computational power, robust learning and energy efficiency.

Neuromorphic computing compared to traditional approaches are energy efficiency, execution speed, robustness against local failures and the ability to learn.

4. Applications

1. Medicine.
2. Large-Scale Operation & Product Customization.
3. Artificial Intelligence.
4. Imaging.

5. Conclusion

To conclude, neuromorphic computing will bring forth the untouched capabilities of AI and will set a revolutionary example in the coming years. The objective of neuromorphic computing is to make computers behave like a human brain and work along the lines of the human nervous system, and neuromorphic computing posits the engineering of computers in a way that comprises millions of artificial silicon neurons enabled to transfer electric spikes from one another.

References

1. C. Mead, Neuromorphic electronic systems, Proceedings of theIEEE, vol. 78, no. 10, pp. 1629–1636, Oct 1990.
2. D. Monroe, Neuromorphic computing gets ready for the (really) bigtime, Communications of the ACM, vol. 57, no. 6, pp. 13–15, 2014.
3. J. Von Neumann and R. Kurzweil, The computer and the brain. YaleUniversity Press, 2012.
4. A, M. Turing, Computing machinery and intelligence, Mind, vol. 59,no. 236, pp. 433–460, 1950.
5. A. F. Murray and A. V. Smith, Asynchronous vlsi neural networksusing pulse-stream arithmetic, Solid-State Circuits, IEEE Journal of,vol. 23, no. 3, pp. 688–697, 1988.
6. F. Blayo and P. Hurat, A vlsi systolic array dedicated to hopfieldneural network, in VLSI for Artificial Intelligence. Springer, 1989,pp. 255–264.
7. F. Salam, A model of neural circuits for programmable vlsi im-plementation, in Circuits and Systems, 1989., IEEE InternationalSymposium on. IEEE, 1989, pp. 849–851.
8. S. Bibyk, M. Ismail, T. Borgstrom, K. Adkins, R. Kaul, N. Khachab, and S. Dupuie, Current-mode neural network building blocks foranalog mos vlsi, in Circuits and Systems, 1990., IEEE InternationalSymposium on. IEEE, 1990, pp. 3283–3285.
9. F. Distante, M. Sami, and G. S. Gajani, A general configurablearchitecture for wsi implementation for neural nets, in Wafer ScaleIntegration, 1990. Proceedings.,[2nd] International Conference on.IEEE, 1990, pp. 116–123.
10. [J. B. Burr, Digital neural network implementations, Neural net-works, concepts, applications, and implementations, vol. 3, pp. 237–285, 1991.