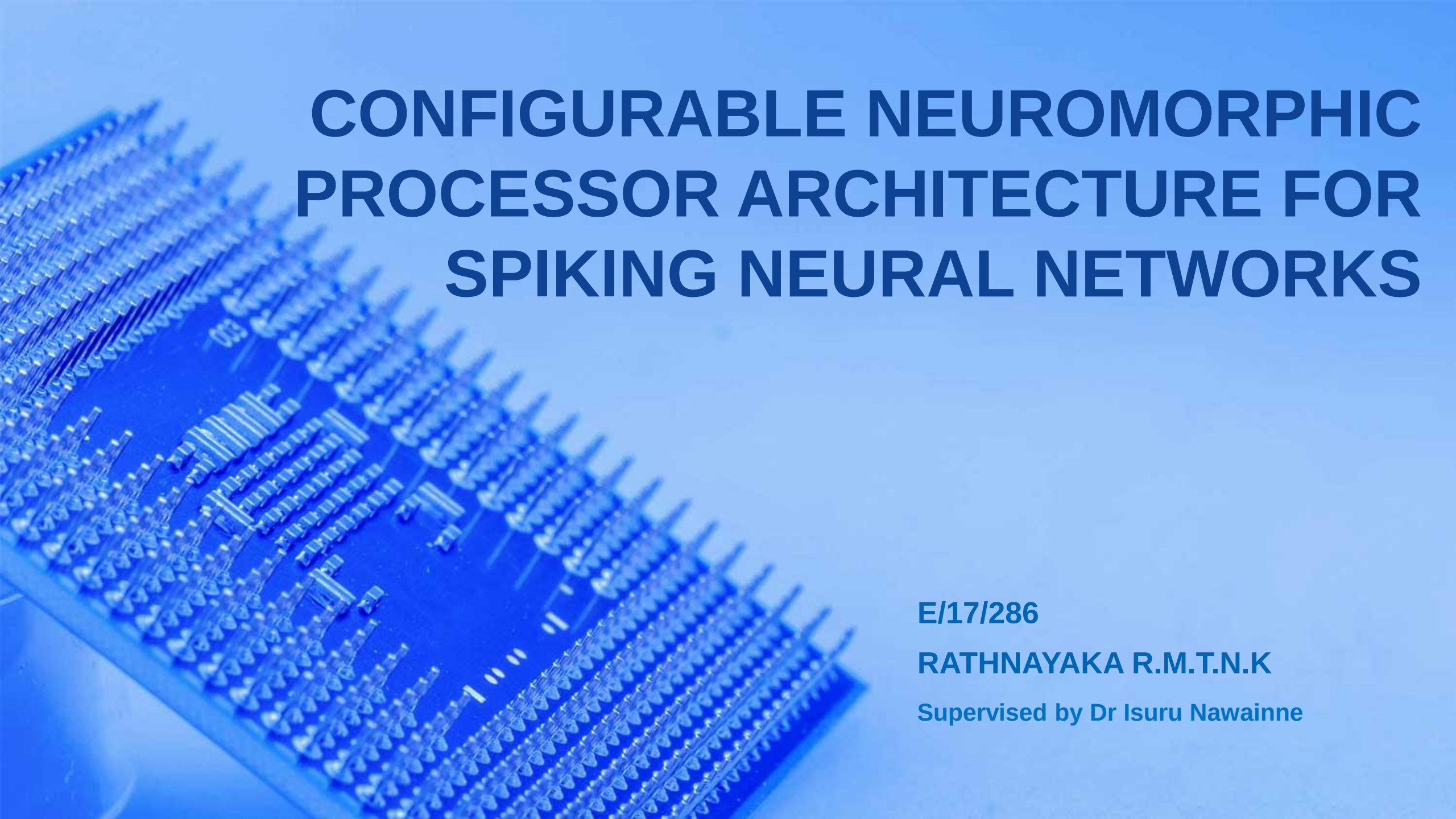
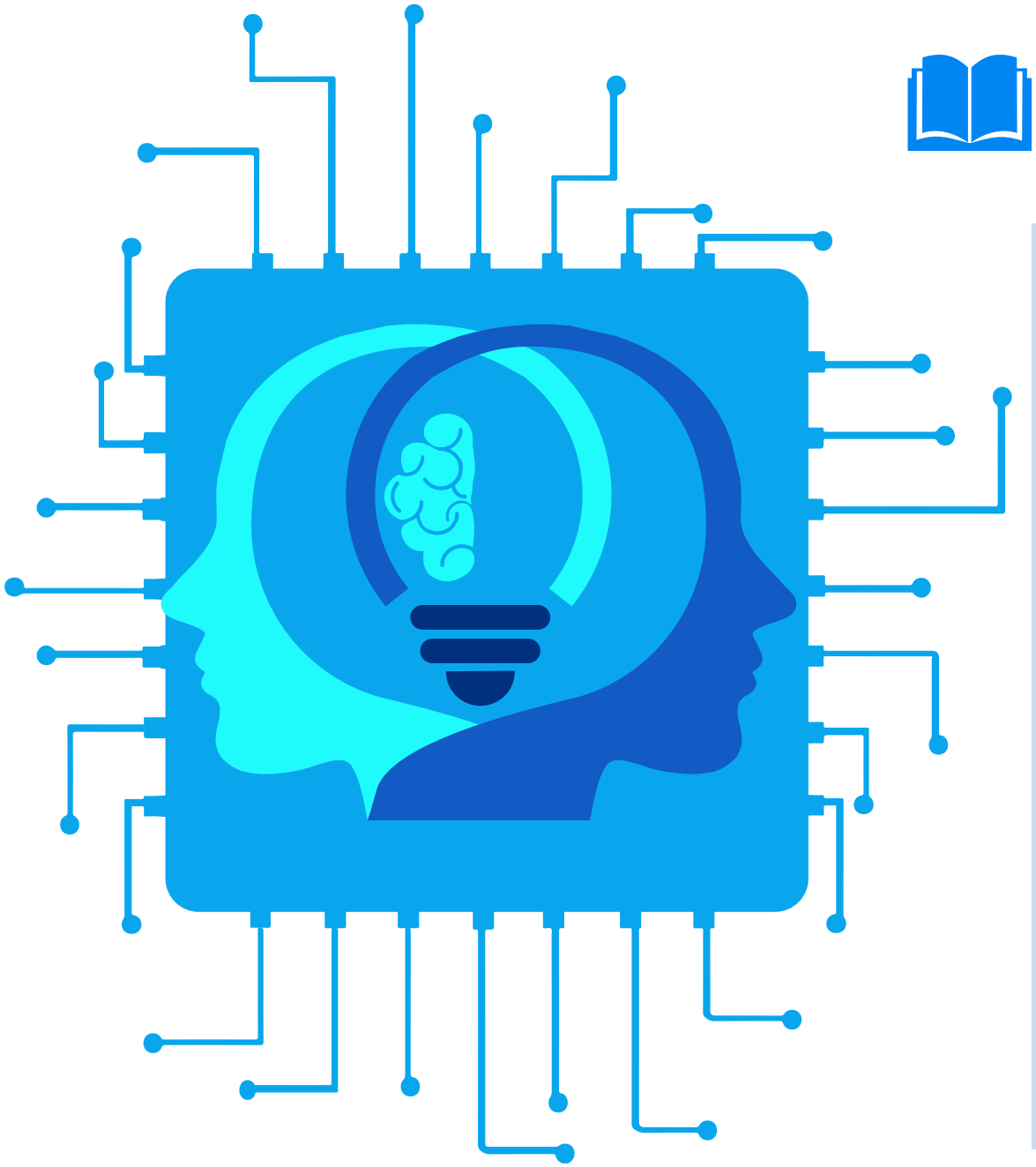


CONFIGURABLE NEUROMORPHIC PROCESSOR ARCHITECTURE FOR SPIKING NEURAL NETWORKS

E/17/286

RATHNAYAKA R.M.T.N.K

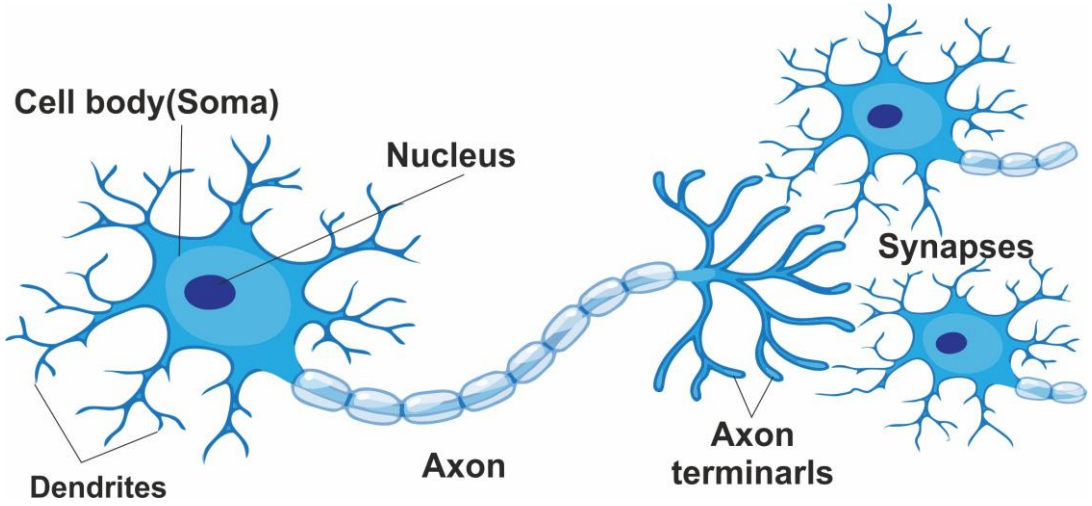
Supervised by Dr Isuru Nawainne

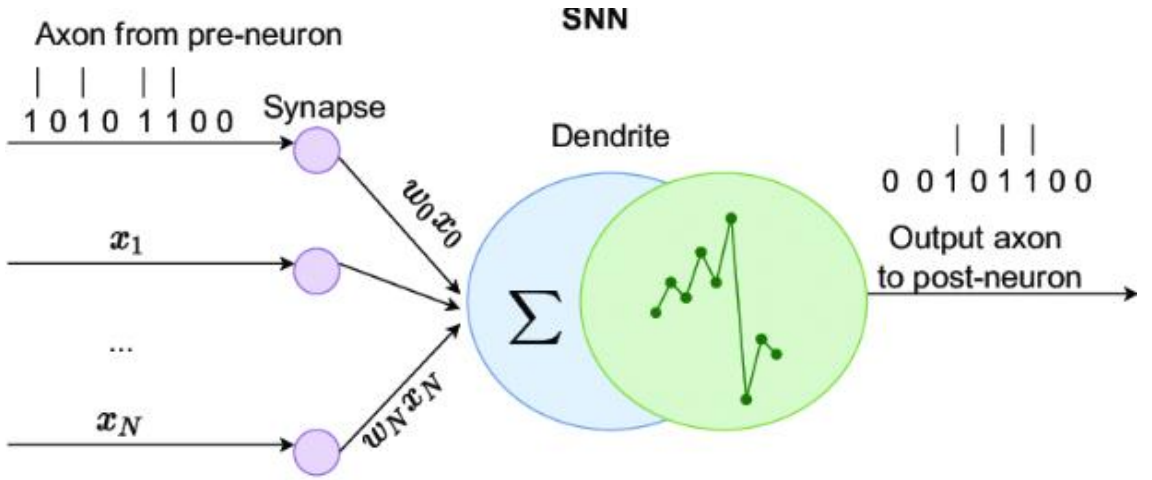


CONTENT

INTRODUCTION-SNN

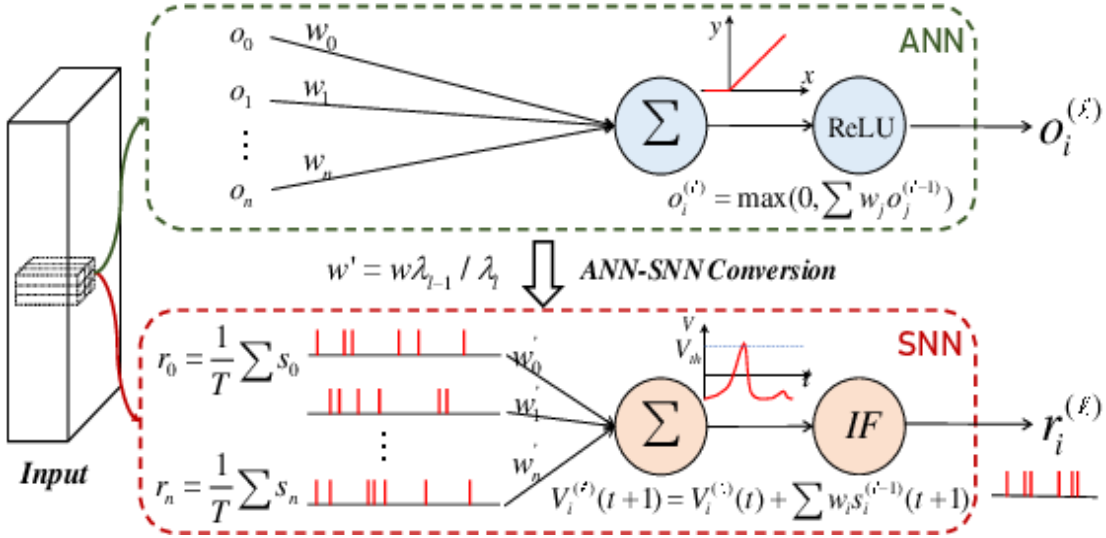
- Spiking neural networks (SNNs) are a type of neural network that simulate the behavior of biological neurons and synapses.
- The significance of SNNs lies in their ability to mimic the behavior of biological neural networks and in their potential for improving machine learning and artificial intelligence.
- The basic components of an SNN include neurons, synapses, and input/output spikes.

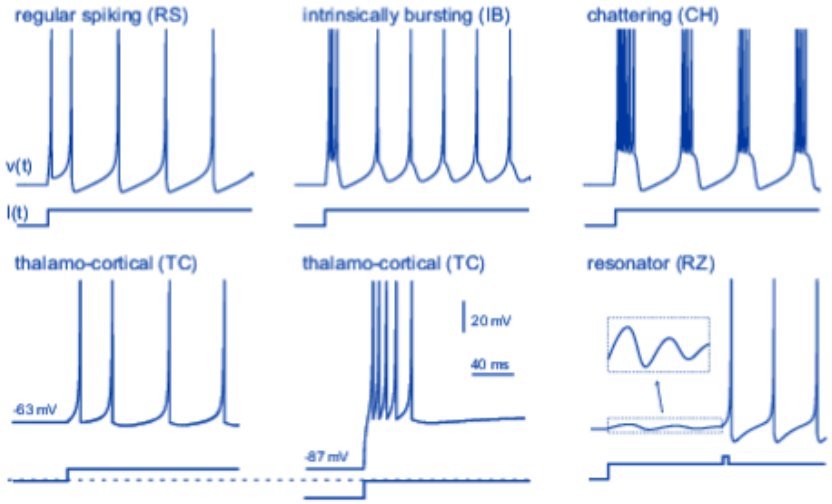




INTRODUCTION-SNN

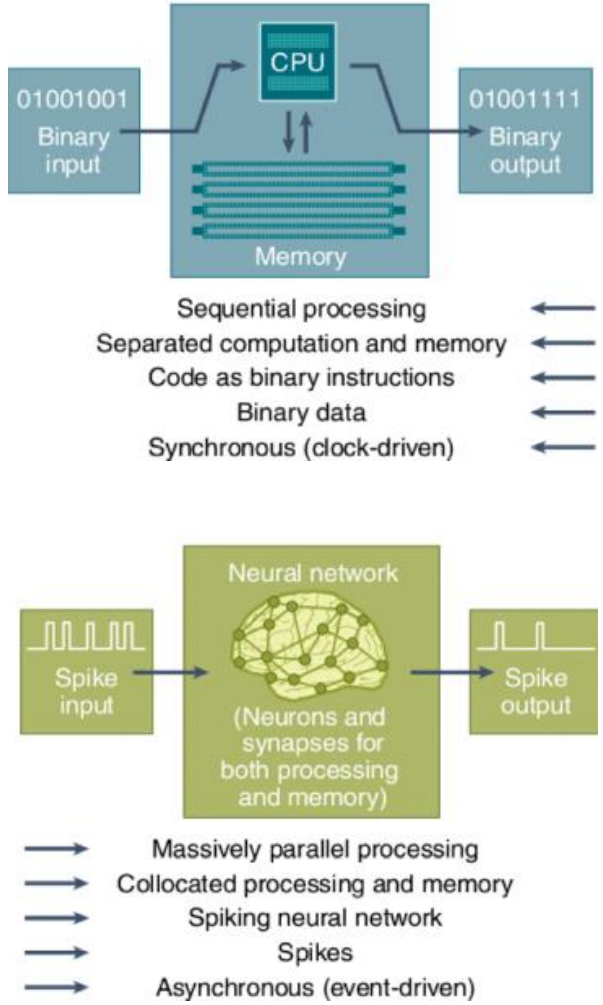
- Unlike traditional artificial neural networks (ANNs), which use abstract mathematical models, SNNs process information through the timing and frequency of spikes.
- SNNs can be used for a variety of applications, such as speech recognition, image processing, and robotics.
- Compared to ANNs, SNNs offer potential advantages such as better energy efficiency and higher accuracy.





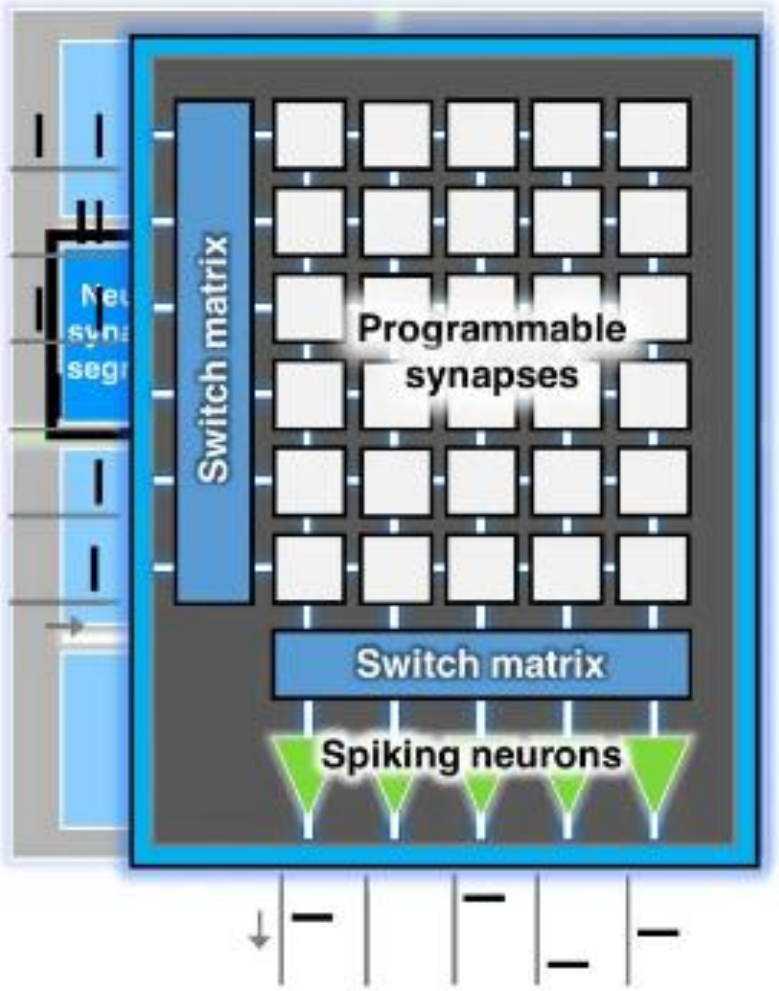
Neuromorphic computing

- Neuromorphic computing is an approach to computing that uses hardware and software inspired by the principles of the human brain.
- The goal of neuromorphic computing is to develop computers that can perform complex tasks with energy efficiency and robustness similar to that of biological neural networks.
- Compared to traditional von Neumann architecture, neuromorphic computing offers potential advantages such as lower power consumption, faster processing, and more efficient use of memory



Neuromorphic computing

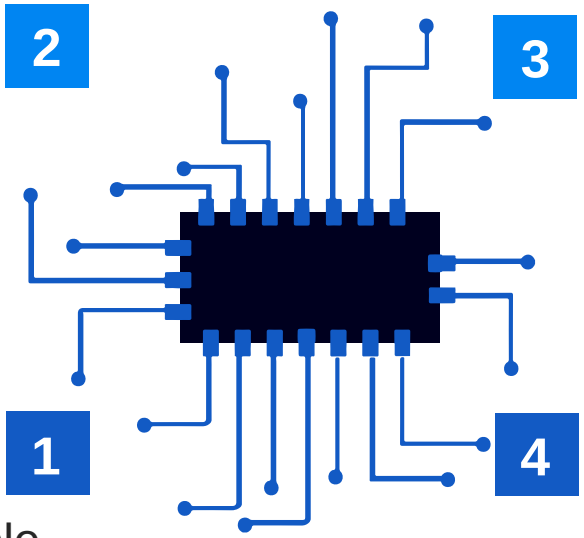
- Neuromorphic computing can be implemented using digital, analog, or mixed-signal circuits, as well as with specialized hardware such as memristors.
- Neuromorphic computing has potential applications in areas such as machine learning, robotics, and brain-computer interfaces.
- The significance of neuromorphic computing lies in its potential to revolutionize computing by enabling machines to process information in ways that are more natural and human-like.



LITERATURE REVIEW

Existing spiking neural network designs can be difficult to use for non-experts.

While there are existing hardware designs and specific architecture or accelerators for SNNs, there is a gap in the literature regarding the development of configurable neuromorphic processor architectures based on RISC-V.

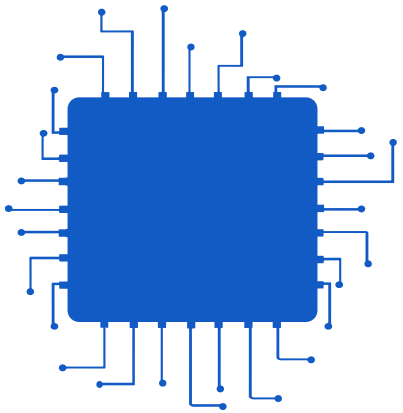


Gap by developing a configurable neuromorphic processor architecture that uses RISC-V and is easy to use for programmers of various skill levels.

Effectiveness of using RISC-V for developing configurable neuromorphic architectures and provides a platform accessible to a wider range of users and researchers.

AIM

- The aim of our research is to develop a configurable neuromorphic processor architecture for spiking neural networks that uses the RISC-V instruction set architecture, and to demonstrate that architecture is both high-performing and low-power, making it an attractive solution for a range of applications.



OBJECTIVES

1

To design and implement a Configurable neuromorphic processor architecture for spiking neural networks.

2

To implement a RV32IM pipelined CPU in Verilog as a starting point for the design.
.

3

To complete the current RISC-v NOC (Network on Chip) FPGA implementation for SNNs and integrate it into the processor architecture.
.

4

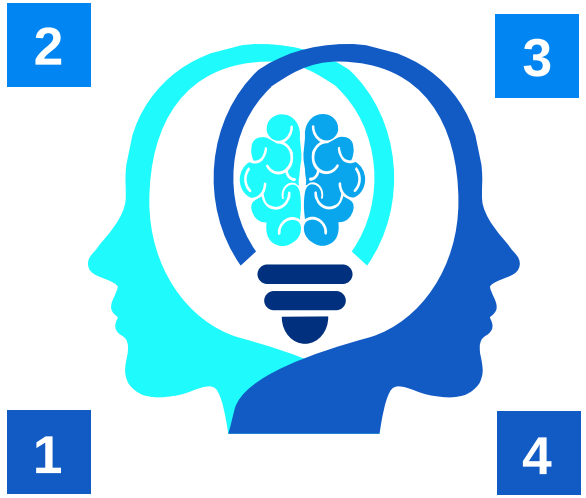
To create a test SNN application to verify the functionality and performance of the processor architecture.
.

5

To evaluate the power consumption and speed of the configurable neuromorphic processor architecture and compare it with existing solutions in the literature

LEARNING OUTCOMES

- To become familiar with the RISC-V instruction set architecture and its implementation in hardware
- To gain experience in developing and testing SNN applications.



- To gain a deep understanding of the principles and concepts of spiking neural networks and neuromorphic computing.
- To understand the trade-offs between power consumption, performance, and area in processor architecture design.

METHODOLOGY

- Our research methodology consists of two main stages:

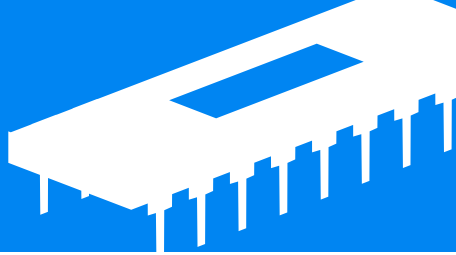
.

  - Implementing a RV32IM pipelined CPU in Verilog as a starting point for the design.
  - Completing the current RISC-v NOC (Network on Chip) FPGA implementation for spiking neural networks (SNNs) and integrating it into the processor architecture

- For the first stage, we will use Verilog as the hardware description language to implement the RV32IM pipelined CPU. This will serve as a foundation for the design and will allow us to build upon existing RISC-v architecture.

- For the second stage, we will use FPGA to implement the NOC for SNNs and integrate it into the processor architecture. This will enable us to develop a configurable neuromorphic processor architecture for spiking neural networks.

TIMELINE



Activity	Time (Week)																												
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1. Initial Preparations																														
1.1 Project Selection																														
1.2 Discussions with superviser																														
2. Project Planning																														
2.1 Preparing of the project proporsal																														
2.2 Preparing presentation																														
2.3 Project proporsal presentation																														
3. Project Tasks																														
3.1 Building RV32IM core with verilog																														
3.2 Put core in to FPGA																														
3.3 RISC-v NOC (Network on Chip) FPGA implementation for SNN																														
3.4 SNN application to verify the functionality and performance																														
3.5 evaluate the power consumption and speed of the configurable																														
7. Finalizing the Project																														
7.1 Preparing Final report and presentation																														

REFFRENCES

1. RISC-V Based Network on Chip Architecture for Spiking Neuron Processing -Heshan Dissanayake Buddhi Perera Dinindu Thilakarathne Department of Computer Engineering University of Peradeniya

2. Izhikevich, E. M. (2004). Which Model to Use for Cortical Spiking Neurons? IEEE Transactions on Neural Networks, 15(5), 1063-1070. https://doi.org/10.1109/TNN.2004.832719

3. Izhikevich, E. M. (2003). Simple Model of Spiking Neurons. IEEE Transactions on Neural Networks, 14(6), 1569-1572. https://doi.org/10.1109/TNN.2003.820440


THANK YOU