Neuromorphic Computing

7. Reconfigurable Neuromorphic Computing System

Ben Abdallah Abderazek, Khanh N. Dang
E-mail: {benab, khanh}@u-aizu.ac.jp

This lecture is based on the book ''Neuromorphic Computing Principles and Organization,'' Publisher: Springer; 2022 edition, ISBN-10 : 3030925242, ISBN-13 : 978-3030925246, by Abderazek Ben Abdallah, Khanh N. Dang

Lecture Contents

1. Introduction

2. Inter-Neuron Communication Network

3. Reconfigurable Neuromorphic System Building Blocks

4. Fault-Tolerant Spike Routing Algorithm

5. Mapping

6. Complexity Analysis

7. Summary

7. Reconfigurable Neuromorphic Computing System:

DNN Structural Organization

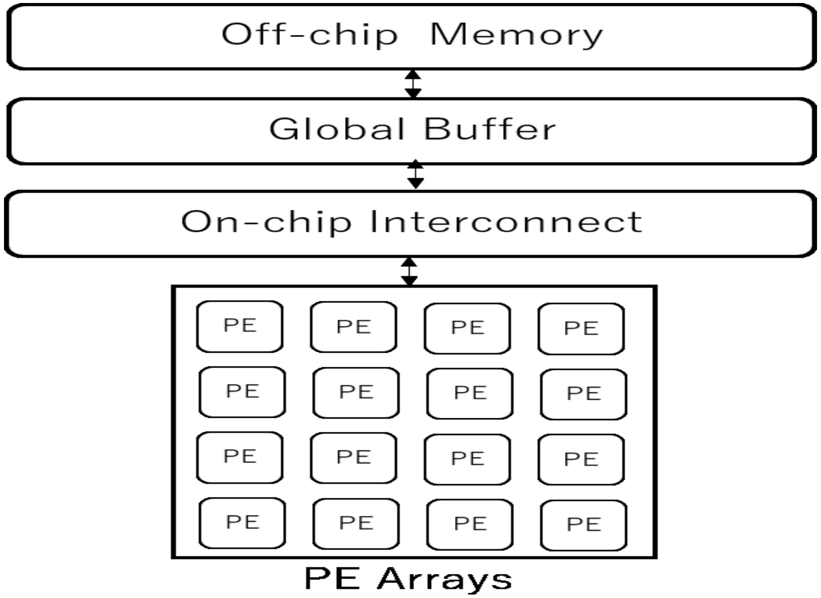

Fig. 7.1: Typical DNN Accelerator Organization.

- The DNN is a typical ANN that consists of several layers expressed as a 3D structure.
- Mapping such a 3D structure onto a 2D circuit requires long wires between layers or congestion points.

7. Reconfigurable Neuromorphic Computing System:

Fault-Tolerant Neural Networks

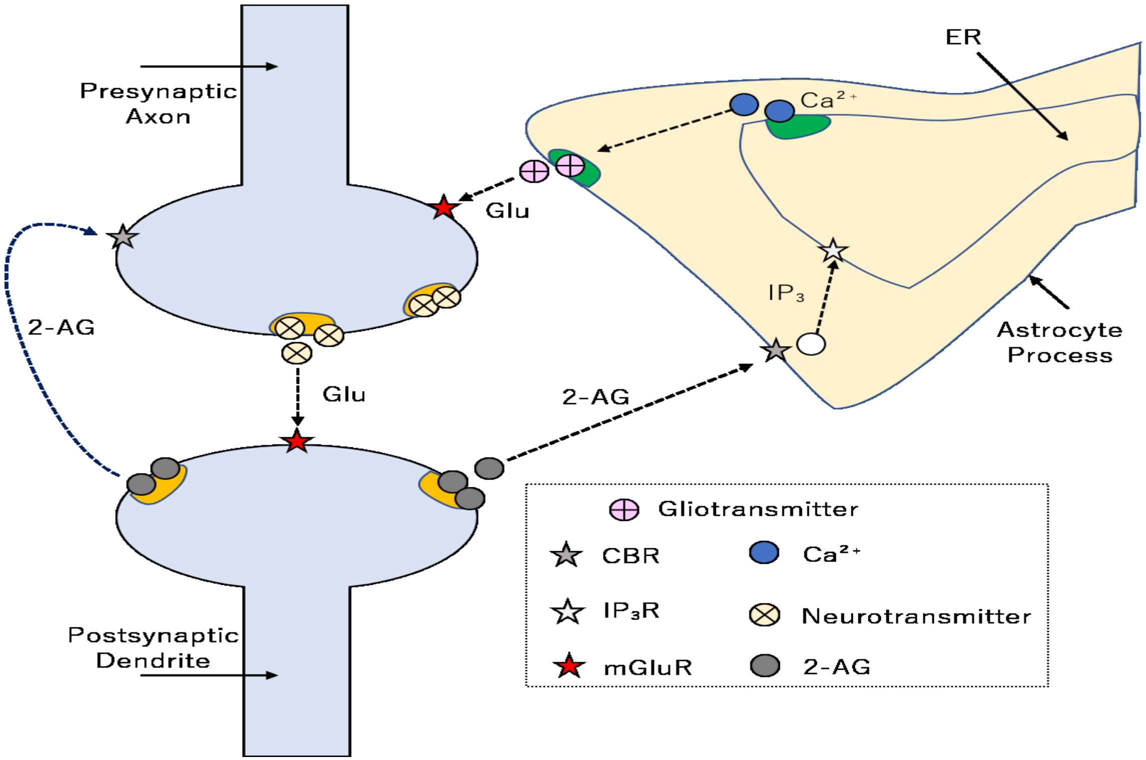

Fig. 7.2: A self-detect and self-repair mechanism mimicking capability in the human brain. This mechanism is based on indirect feedback from the astrocyte cell (i.e. glial cell) by regulating the synaptic transmission probability of release when faults occur.

7. Reconfigurable Neuromorphic Computing System:

Inter-Neuron Communication Network

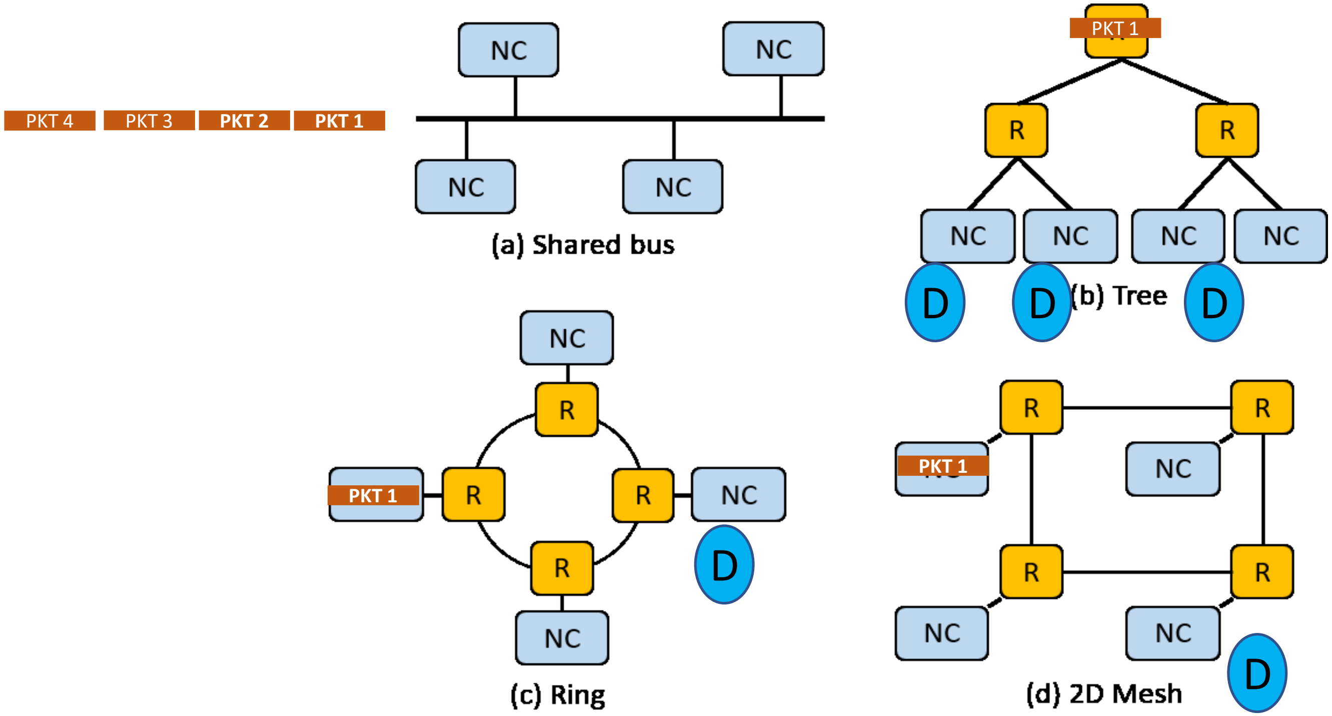

Fig. 7.3: 2D-based Interconnect architectures for Neuromorphic systems

- Hardware implementations were proposed to overcome the problems of the software simulation.
- These architectures will offer high-parallelism and scalable interconnect architecture for huge spikes transfer.

7. Reconfigurable Neuromorphic Computing System:

Routing Methods for NoC-based SNNs

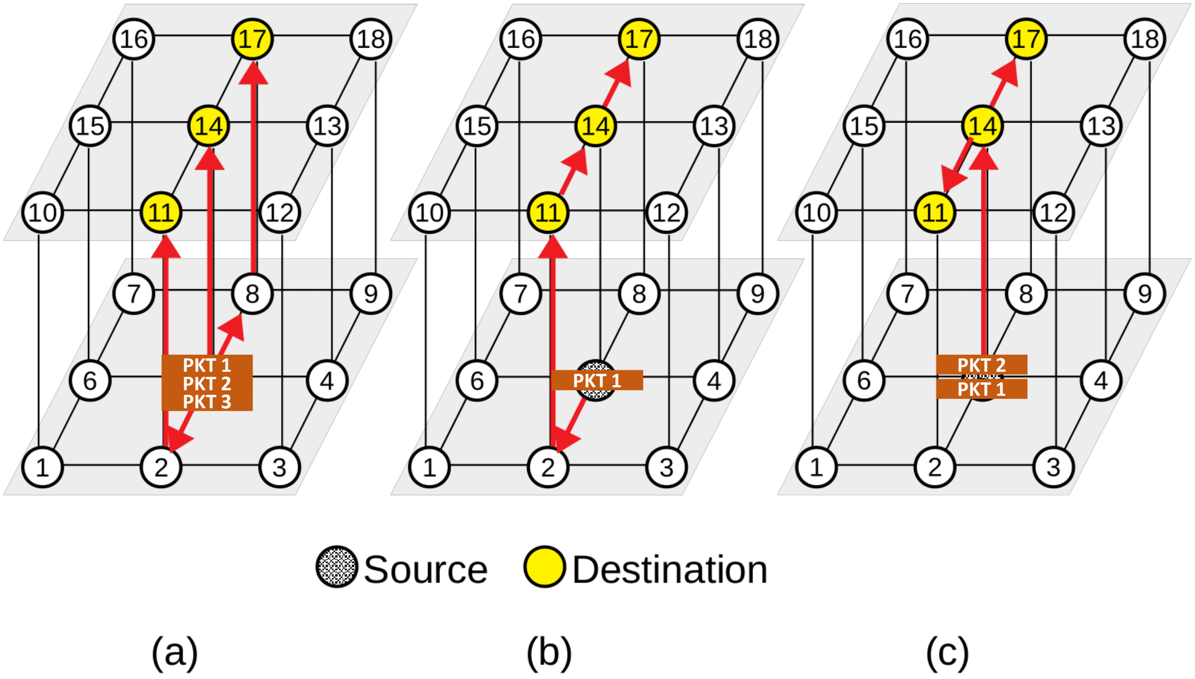

Fig. 7.4: 3D-based Multicast routing mechanisms: (a) Unicast-based (b) Path-based (c) Tree-based

- Routing method is important because it affects the load balance and spikes latency across the network.
- In general, the methods are classified into unicast-based, path-based, and tree-based.

7. Reconfigurable Neuromorphic Computing System:

Reconfigurable Neuromorphic System Building Blocks

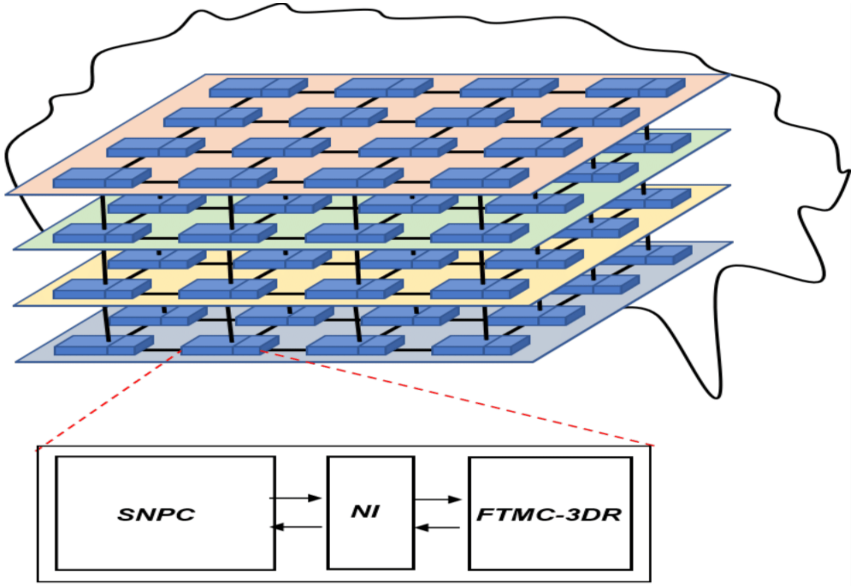

Fig. 7.5: Reconfigurable 3D-NoC-based neuromorphic Architecture (i.e. NASH)

- Composed of several nodes.
- Each node has a Spiking Neuron Processing Core (SNPC), a network interface (NI), and a fault-tolerant multicast 3D router (FTMC-3DR).
- Nodes are connected in a 2D mesh topology and stacked to form a 3D architecture.

7. Reconfigurable Neuromorphic Computing System:

Reconfigurable Neuromorphic System Building Blocks: SNPC

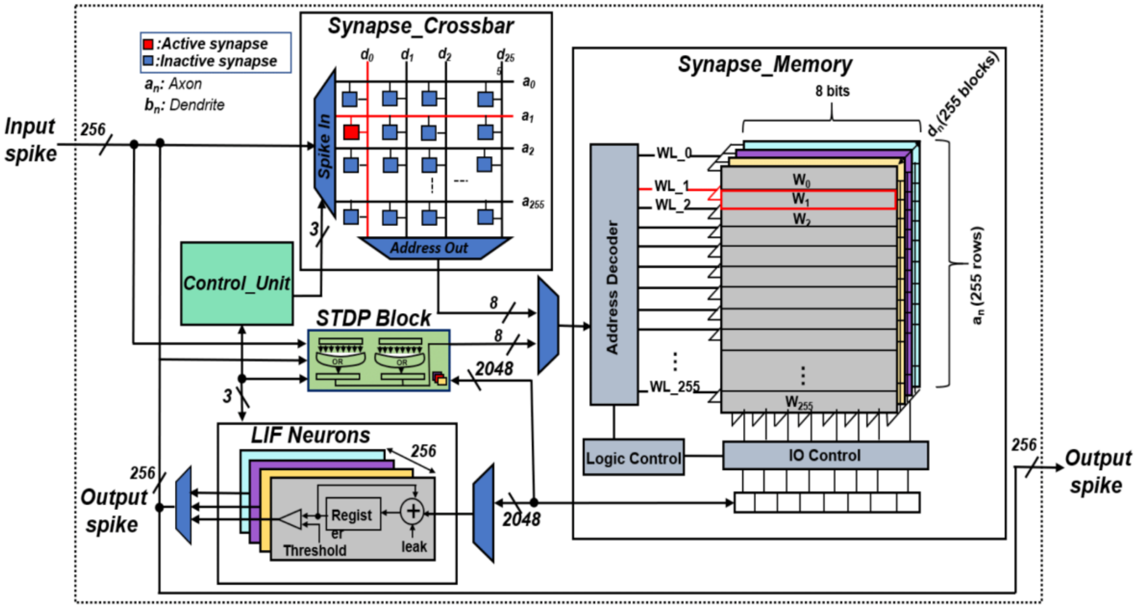

Fig. 7.6: Architecture of spiking neuron processing core (SNPC)

- Composed of 256 physical leaky integrate and fire neurons, a crossbar-based synapse, a control unit, a synapse memory, an STDP learning block, and an encoder/decoder.
- The SNPC uses a spike array for spike events to avoid memory overflow and extended pipeline time.

7. Reconfigurable Neuromorphic Computing System:

Reconfigurable Neuromorphic System Building Blocks: LIF Neuron

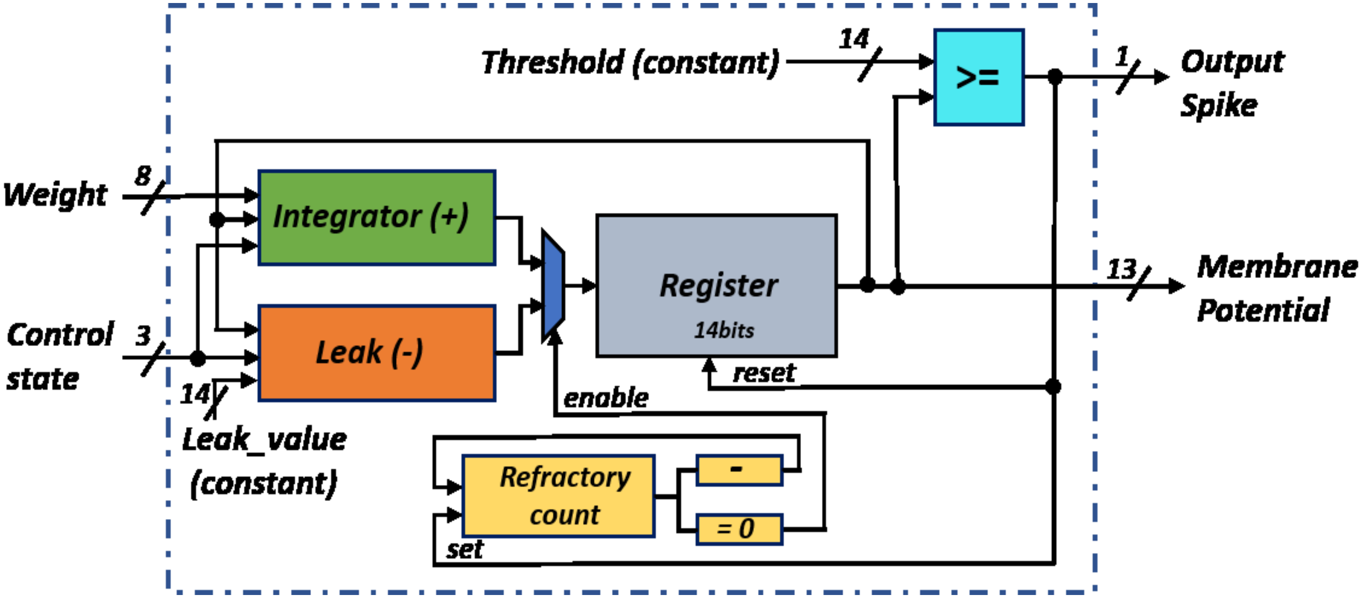

Fig. 7.7: LIF Neuron Architecture

- The neuron membrane potential is accumulated by adding up the input weighted spikes in the integrator.
- Fires an output spike if the value of the accumulated membrane potential exceeds the threshold constant.
- Enters refractory count until the next spike is fired.

7. Reconfigurable Neuromorphic Computing System:

Reconfigurable Neuromorphic System Building Blocks: STDP

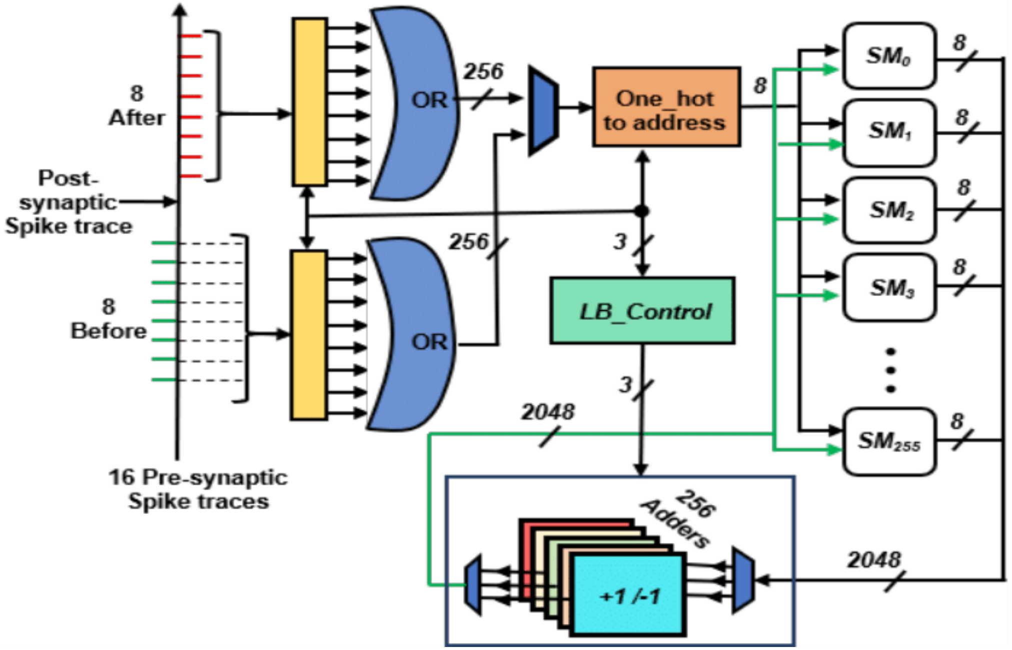

Fig. 7.8: STDP Learning Module Architecture.

- The STDP is based on a trace-based learning rule which enables the parallel update of synapses.
- A single learning operation requires 16 presynaptic spike trace vectors, each from a simulation time step.

7. Reconfigurable Neuromorphic Computing System:

Reconfigurable Neuromorphic System Building Blocks: Encoder and Decoder

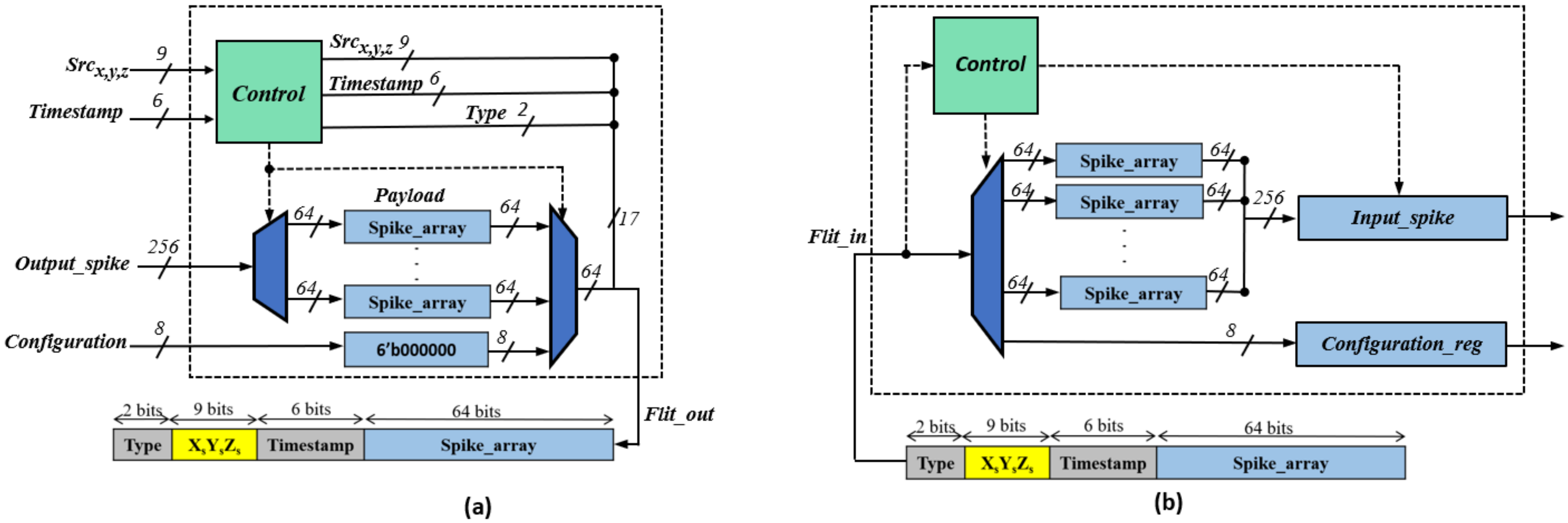

Fig. 7.9: Encoder and Decoder (Network interface to and from the router). (a) The encoder encodes output spikes that will be transmitted from source SNPC to destination SNPCs into flits. (b) The Decoder, on the other hand, decodes flits that arrive at a destination SNPC into a spike.

7. Reconfigurable Neuromorphic Computing System:

Reconfigurable Neuromorphic System Building Blocks: Fault-Tolerant Spike Routing Algorithm

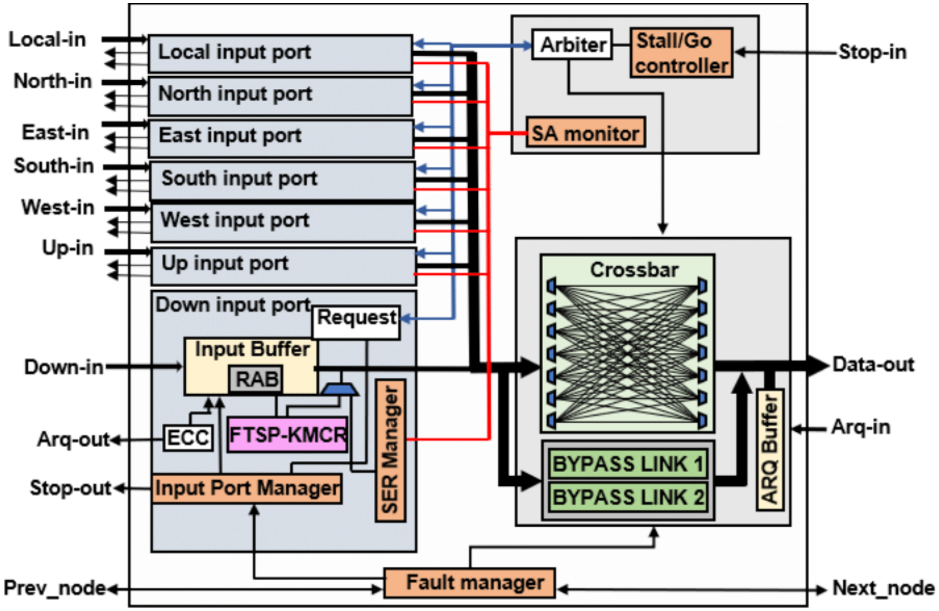

Fig. 7.10: Fault-tolerant multicast 3D router architecture

- The FTMC-3DR has 7 I/O ports, fault-tolerant mechanisms at the input buffer (RAB), and Byline on demand (BLoD) at the crossbar.
- It uses 4 pipeline stages (Buffer writing, Routing calculation, Switch-allocation, and Crossbar traversal.) to route packets

7. Reconfigurable Neuromorphic Computing System:

Fault-Tolerant Spike Routing Algorithm

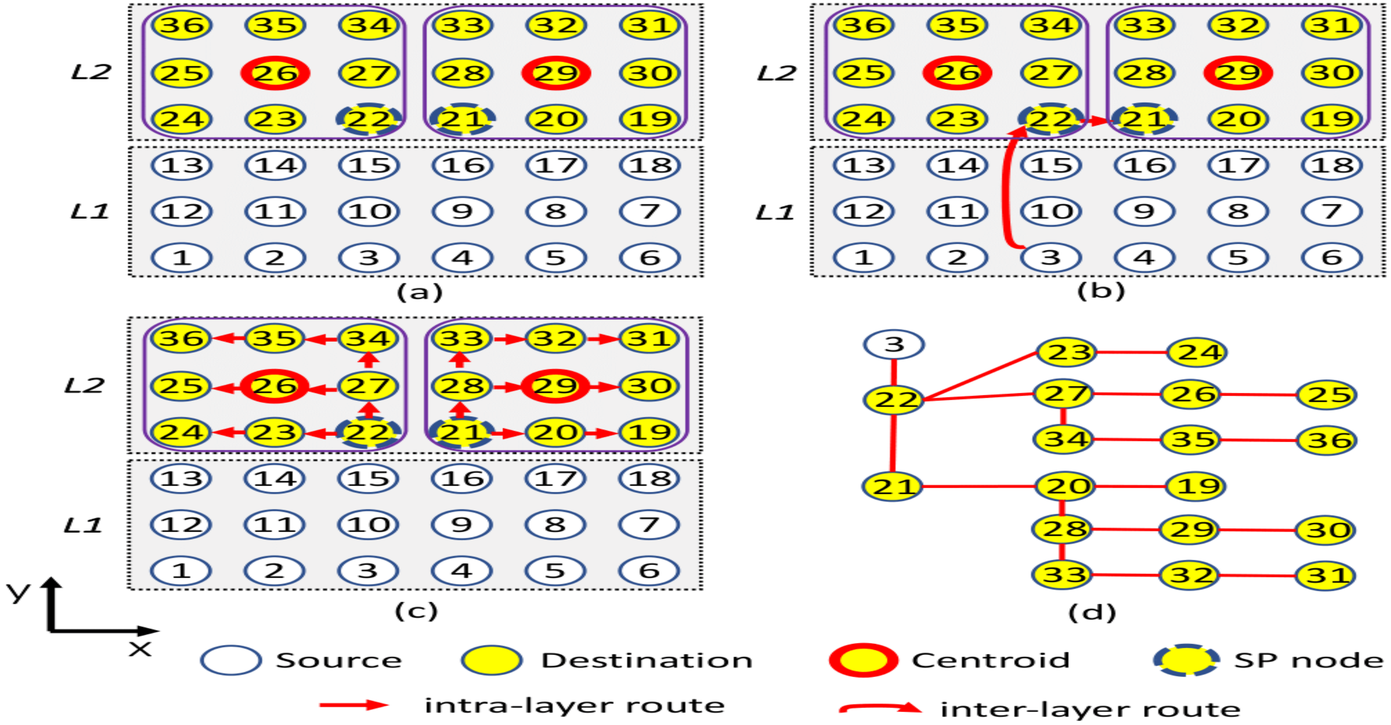

Fig. 7.11: Example of SP-KMCR for a 6 × 3 × 2 3DNoC-SNN system, where nodes in L1 send spike packets to all nodes in L2: (a) destinations are partitioned by adopting K-means clustering with centroids 26 and 29, (b) the formation of the first part of the tree from a given source (node 3) to shortest path node of each subgroup (SP node), (c) the second part of the tree from SP nodes to its destinations, (d) the routing tree from the given source to destinations.

7. Reconfigurable Neuromorphic Computing System:

Fault-Tolerant Spike Routing Algorithm: Fault-Tolerant K-means Multicast Spike Routing Algorithm

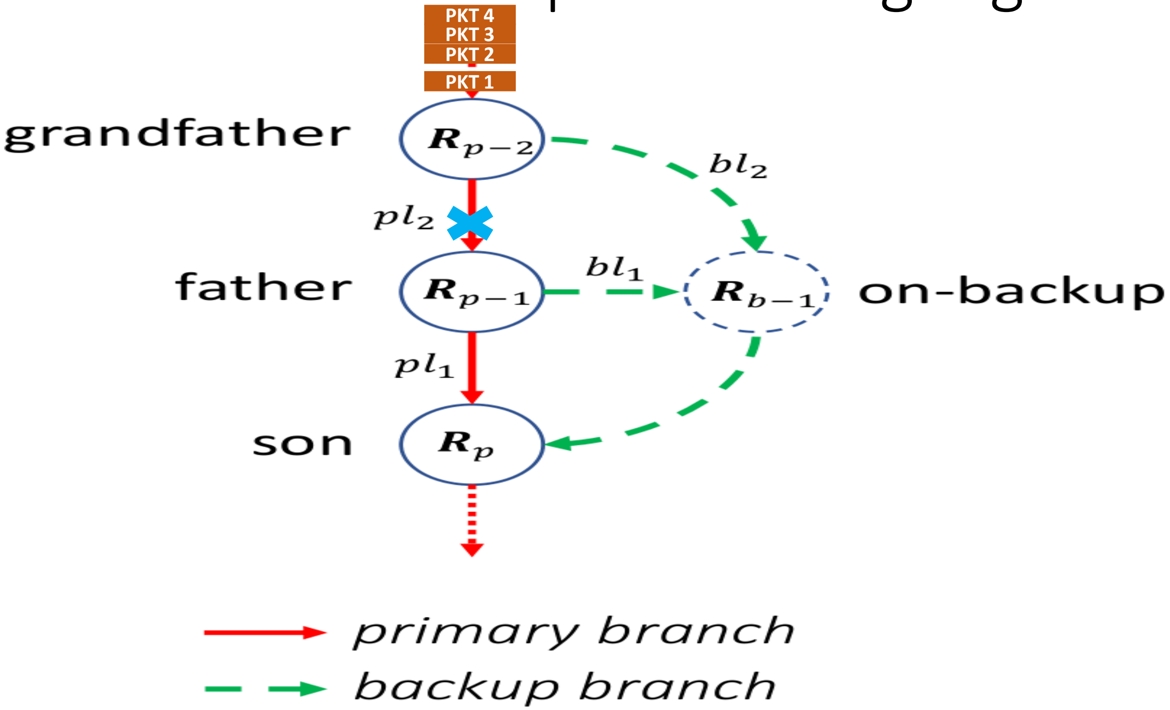

Fig. 7.12: Primary and backup branches.

- The FT-KMCR provides some backup branch(es) to bypass faulty links when there is a faulty primary branch.
- The backup branches are alternative routes to the primary ones.

7. Reconfigurable Neuromorphic Computing System:

Fault-Tolerant Spike Routing Algorithm: Fault Management Algorithm Flow Chart

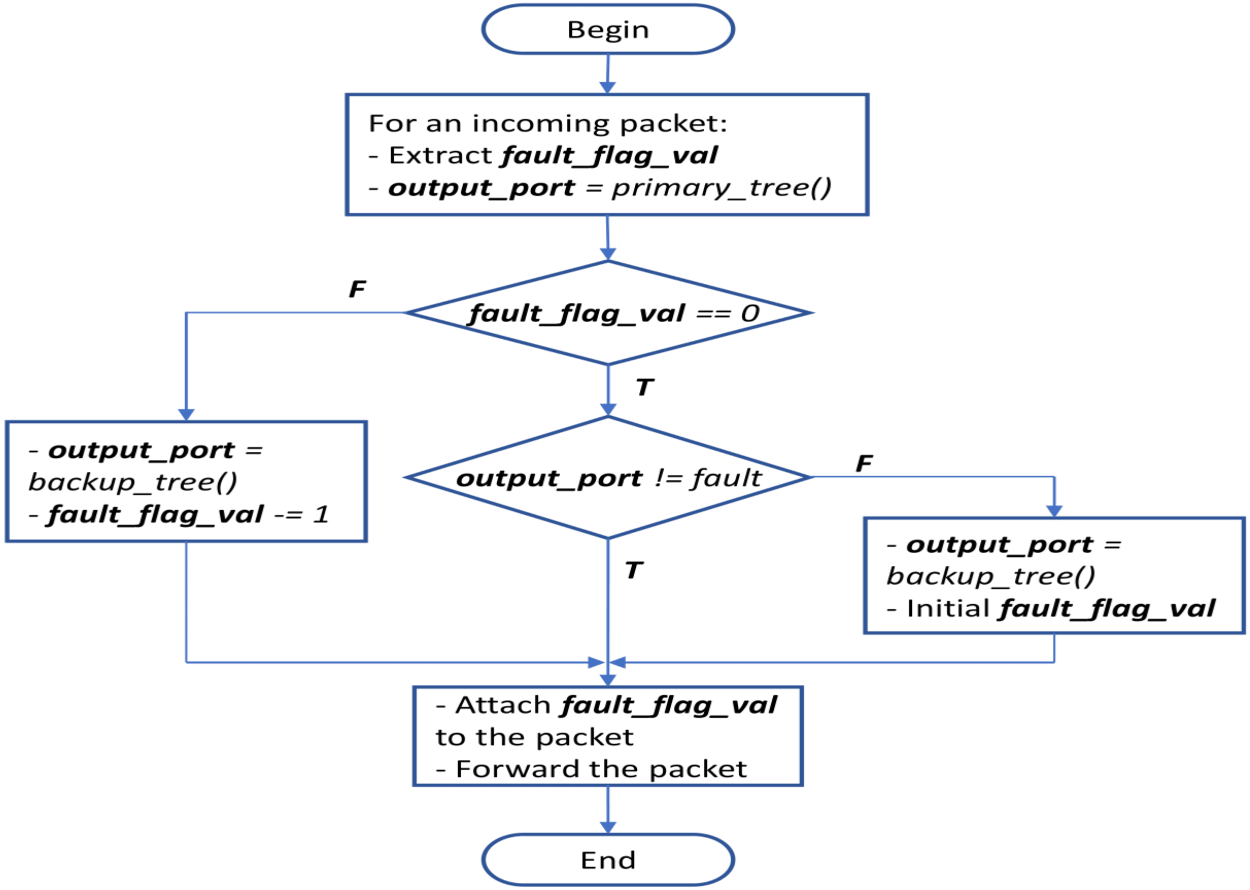

Fig. 7.13: Fault-management algorithm applied for ”son”, on-backup, ”father” and ”grandfather” routers.

7. Reconfigurable Neuromorphic Computing System:

Mapping

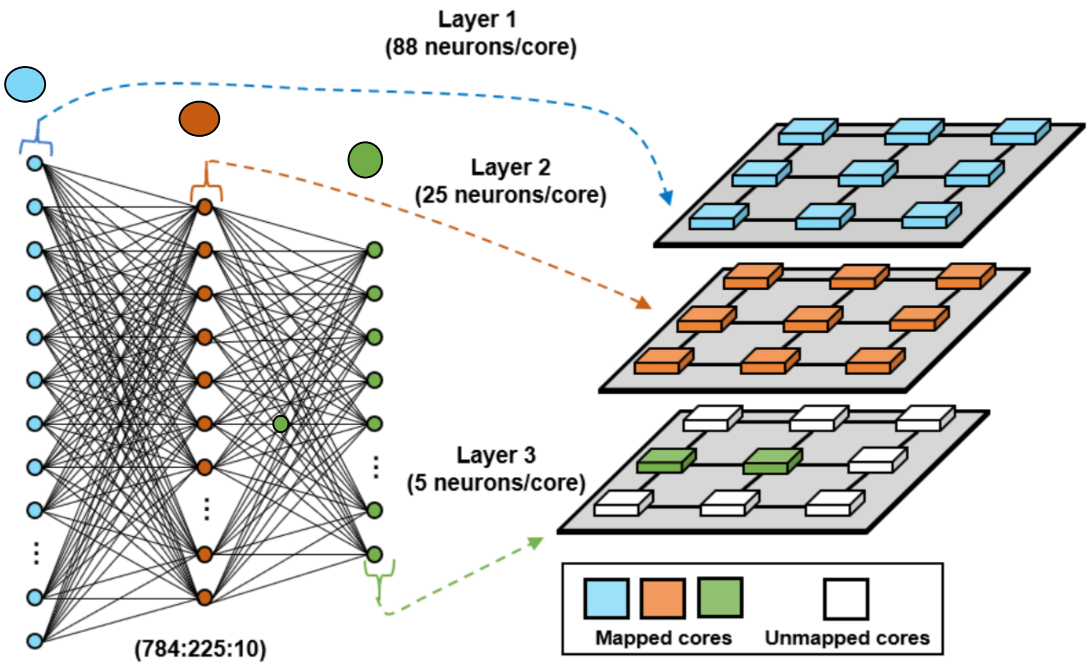

Fig. 7.14: 784:225:10 SNN mapping on a 3 × 3 × 3 NASH configuration for MNIST classification application

- The aim of a mapping is to establish measurable links between the parameters of the SNN application and the NASH.
- The NASH mapping approach is layer-based where each network layer is mapped to a corresponding NASH layer.

7. Reconfigurable Neuromorphic Computing System:

Complexity Analysis: Area

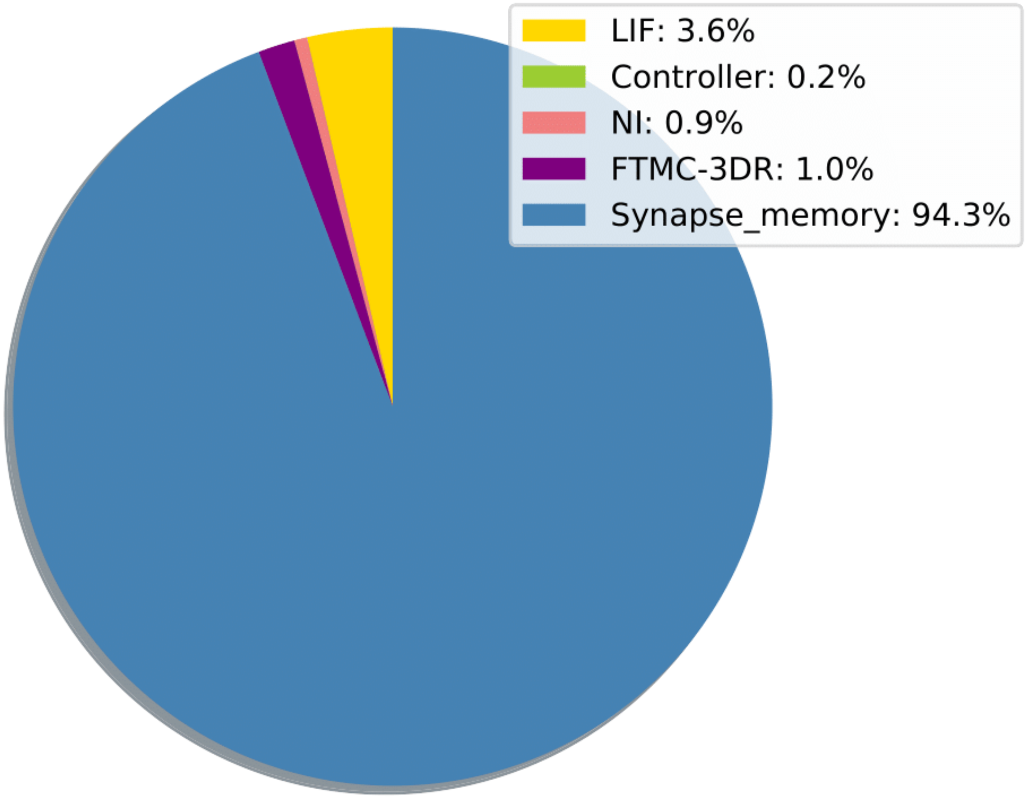

Fig. 7.15: Area analysis of NASH node

7. Reconfigurable Neuromorphic Computing System:

Complexity Analysis: Accuracy Evaluation

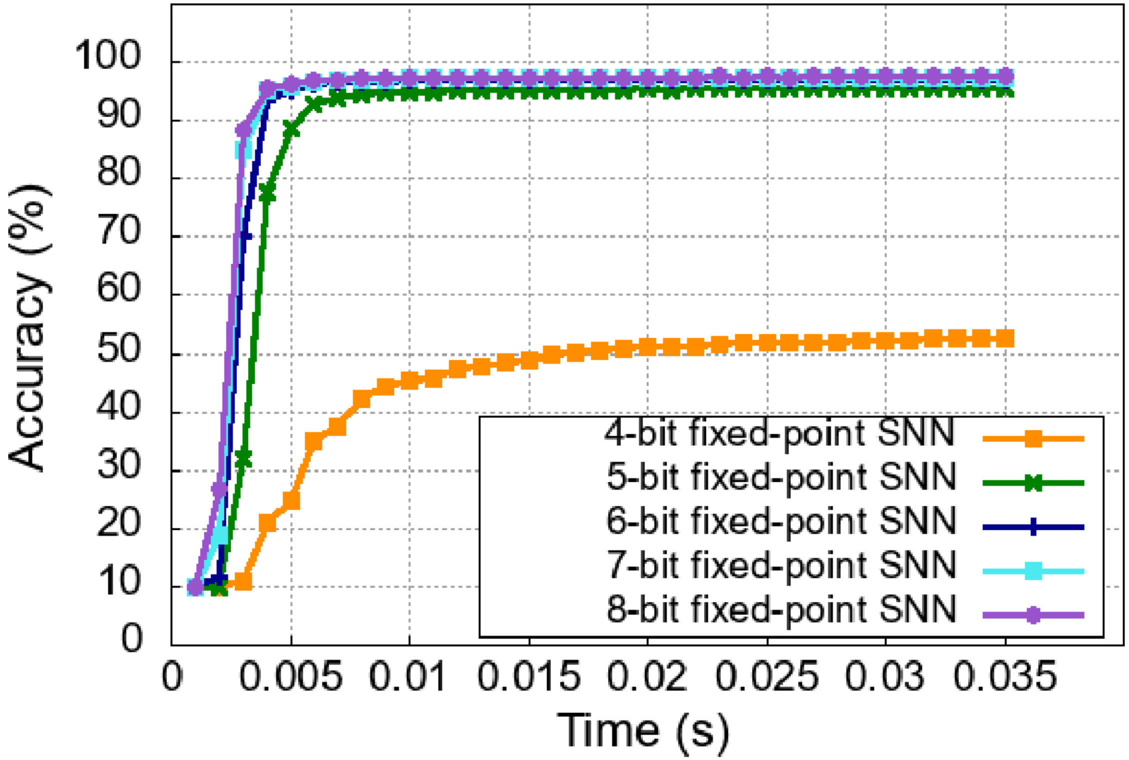

Fig. 7.16: Accuracy evaluation over various synapse precision

7. Reconfigurable Neuromorphic Computing System:

Complexity Analysis: Synapse Precision

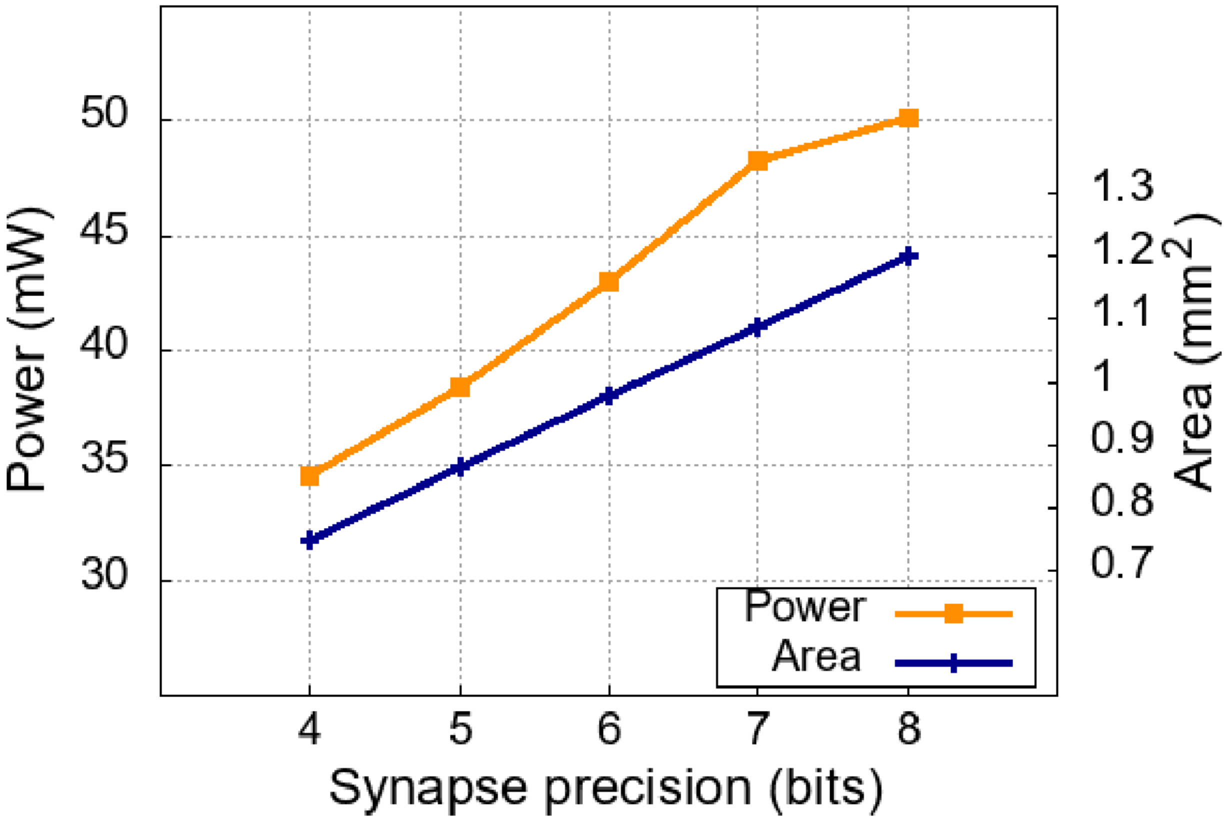

Fig. 7.17: Area and power evaluation over various synapse precision

7. Reconfigurable Neuromorphic Computing System:

NASH Layout Design and Floor Plan

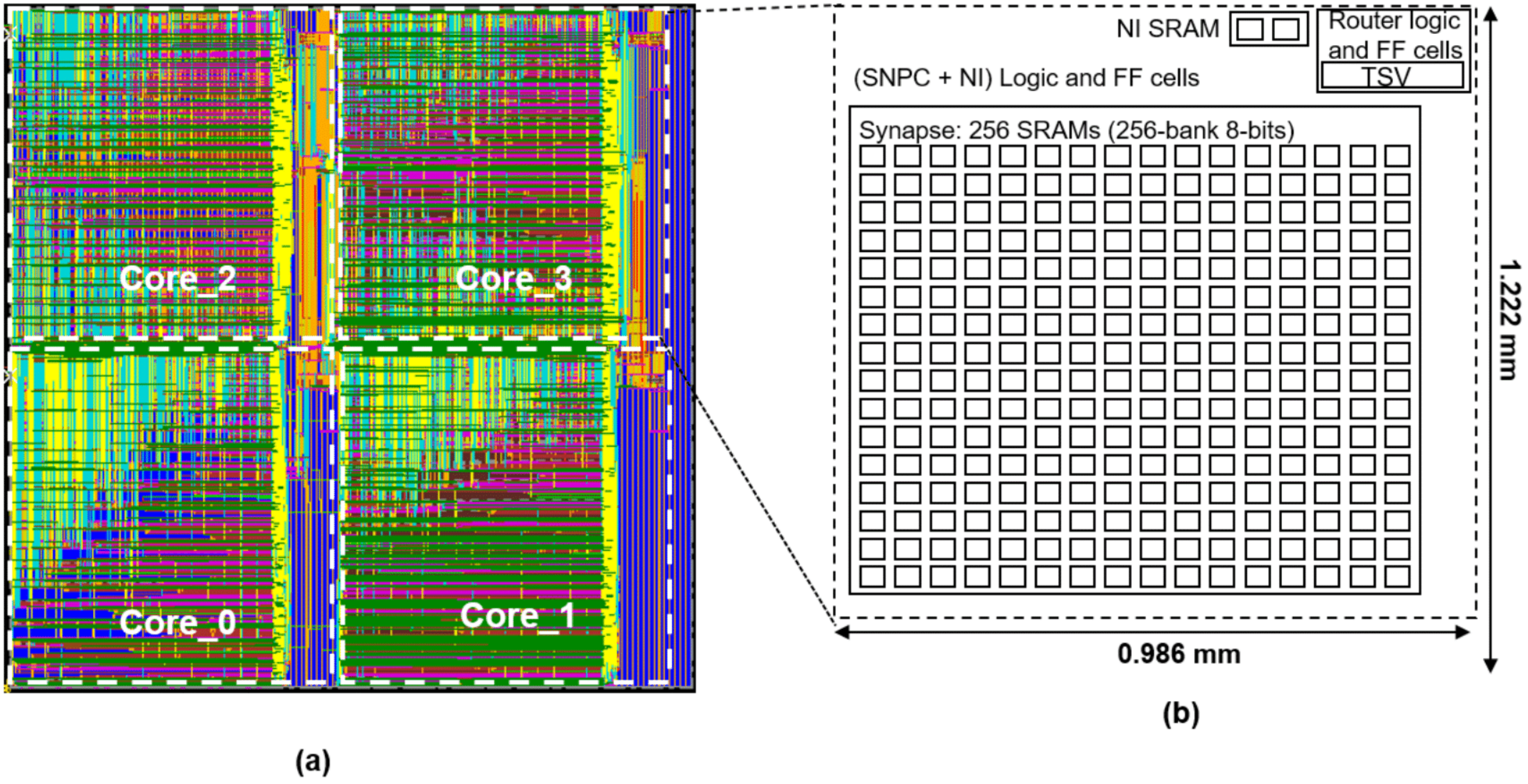
Core_2
Core_3
Core_0
Core_1
(a)
NI SRAM
Router logic and FF cells
(SNPC + NI) Logic and FF cells
TSV
Synapse: 256 SRAMs (256-bank 8-bits)
1.222 mm
0.986 mm
(b)

Fig. 7.18: (a) Layout of a 2 × 2 NASH layer. (b) A NASH node comprising of 256 neuron logic and 65k synapses in 256 SRAMs (256-bank 8-bits each), network interface logic and memory, and an FTMC-3DR logic and TSVs

Chapter Summary

- This chapter presented the architecture, hardware design, and complexity analysis of a reconfigurable neuromorphic system NASH.
- The focus is on:
  - The SNPC, learning, interconnect, spike routing, and mapping.
- The system leverages the high scalability and parallelism, low communication cost, and high throughput available in 3D-NoC-based systems.

Conclusions

- Neuromorphic Computing is the use of hardware (VLSI) to simulate the biological architecture of the human nervous system (brain, complex network of nerves, etc.),

- Spiking Neural Network:
  - More analogous to the brain, communicating via spikes in a sparse event-driven manner.
  - Exploits spike sparsity to achieve low power.

- Synaptic dynamics is the time-dependent changes in synaptic currents that change the strength of coupling between neurons.

- There are various training/learning algorithms for SNNs:
  - Unsupervised Spike-timing-dependent plasticity (STDP)
  - ANN to SNN conversion

- Synthesizing a Neuromorphic System:
  - Define Problem→ Partition AI Tasks → Understand Constraints → Develop AI HW/SW Model → Embed into Device → Solve the Problem

Exercises