Neuromorphic Computing

4. Emerging Memory Devices for Neuromorphic Systems (Part – I and II)

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 of Memory
2. Memory Technology
3. Memory Organization
4. Memory for Neuromorphic Systems
5. Dynamics of NVM Synapse
6. Conclusions

1. Introduction of Memory
Introduction

- Neuromorphic computing systems are generally built with thousands or even millions or neurons.
  - Neuromorphic systems’ parameters and temporal values are too large to be stored locally.
  - Storing and loading is necessary.
  - Accessing parameters and values requires a huge bandwidth.
- ➔ Designing memory for neuromorphic system is an extremely critical task:
  - Memory communication could be a bottle neck.
  - Power consumed for memory read/write instructions can be enourmous.

1. Introduction of Memory Hierachy

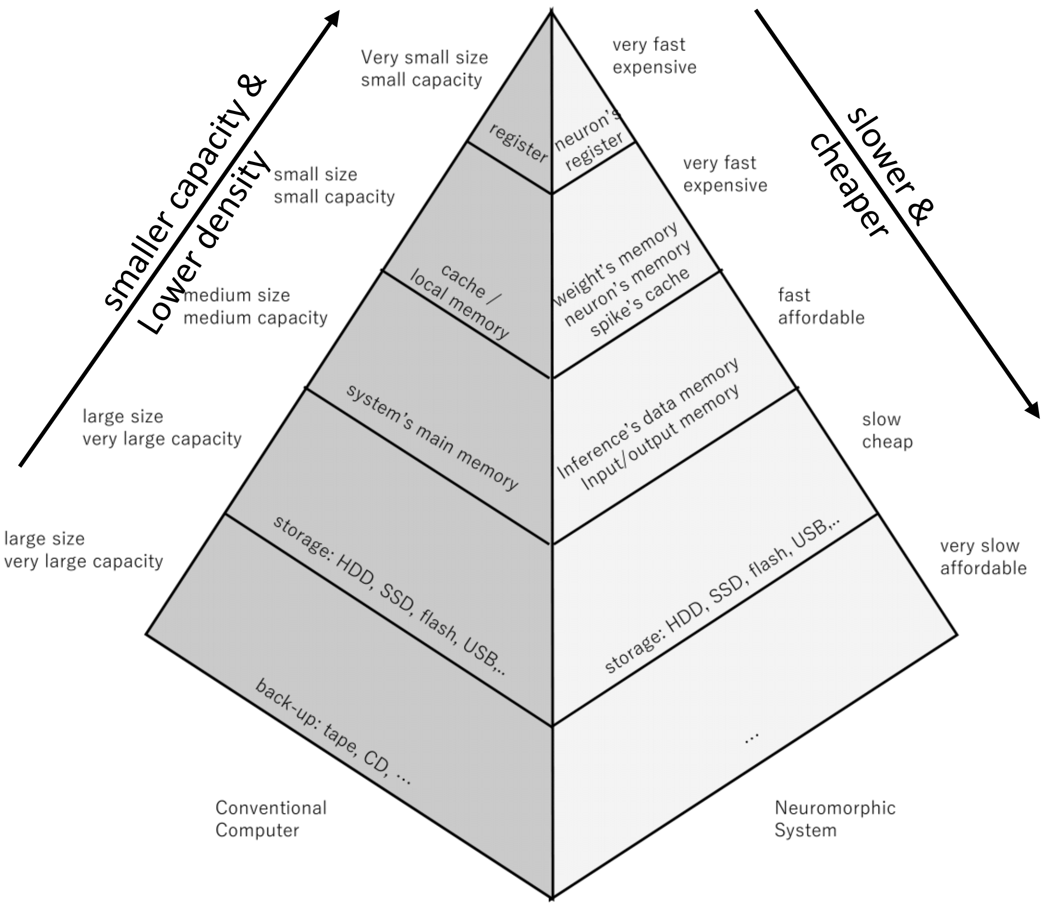


Fig. 4.1: Memory Hierarchy.

- Memory hierarchy for neuromorphic system is similar to the conventional computing systems.
  - Divide into multiple layers.
  - Smaller capacity ⇔ lower density ⇔faster ⇔ more expensive

1. Introduction of Memory
Neuron's structure

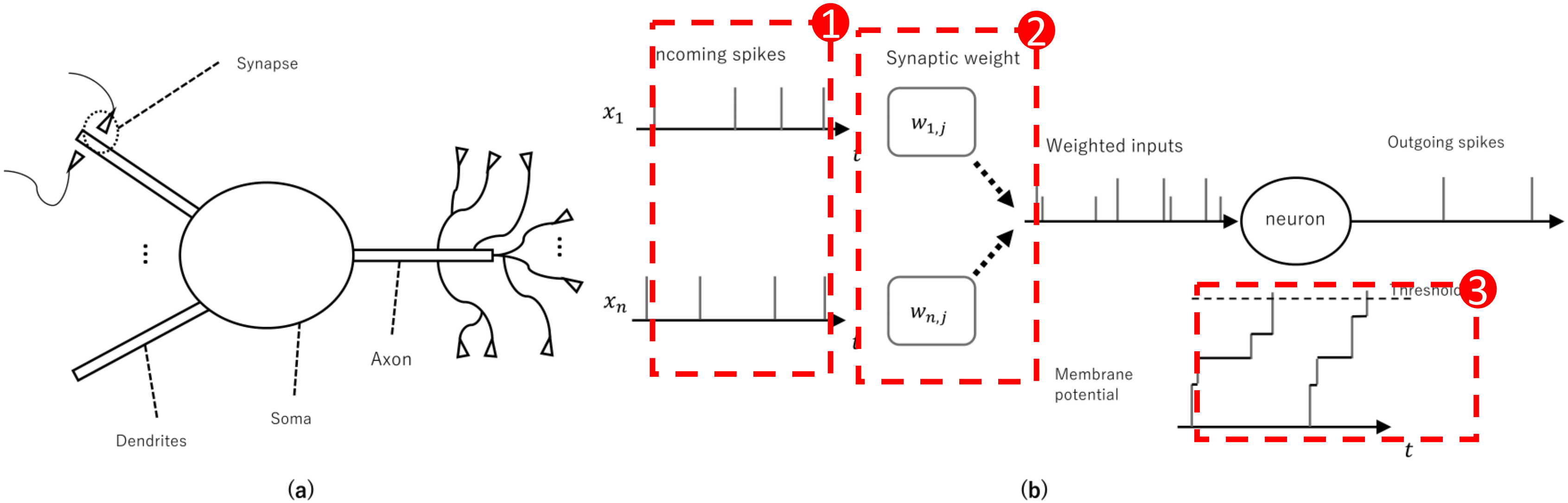


Fig. 4.2: (a) Biological neuron. (b) Spiking neuron.

In the spiking neuron models, there are three major parameters than need to be stored (memorized):

1. incoming spikes;
2. synaptic weights,
3. neuron's internal parameters (membrane potential, threshold, etc.).

Lecture Contents

1. Introduction of Memory
2. Memory Technology
3. Memory Organization
4. Memory for Neuromorphic Systems
5. Dynamics of NVM Synapse
6. Conclusions

2. Memory Technology
Introduction

- Memory, in general, consists of a set of memory cells:
  - Each memory cell exhibits in states or levels.
    - Typically binary value (0 or 1);
    - Can be in multiple levels.
  - Each memory cell can be read or written into states.
  - A typical memory cell has two control signals:
    - Select: to select the memory cell
    - Control: direction of the instruction
  - And two flows: Input and Output (can be one in Duplex mode)
  - Memory has mechanisms to access (read/write) the exact location of memory cells or sub-set of cells.

2. Memory Technology
Memory cell structure

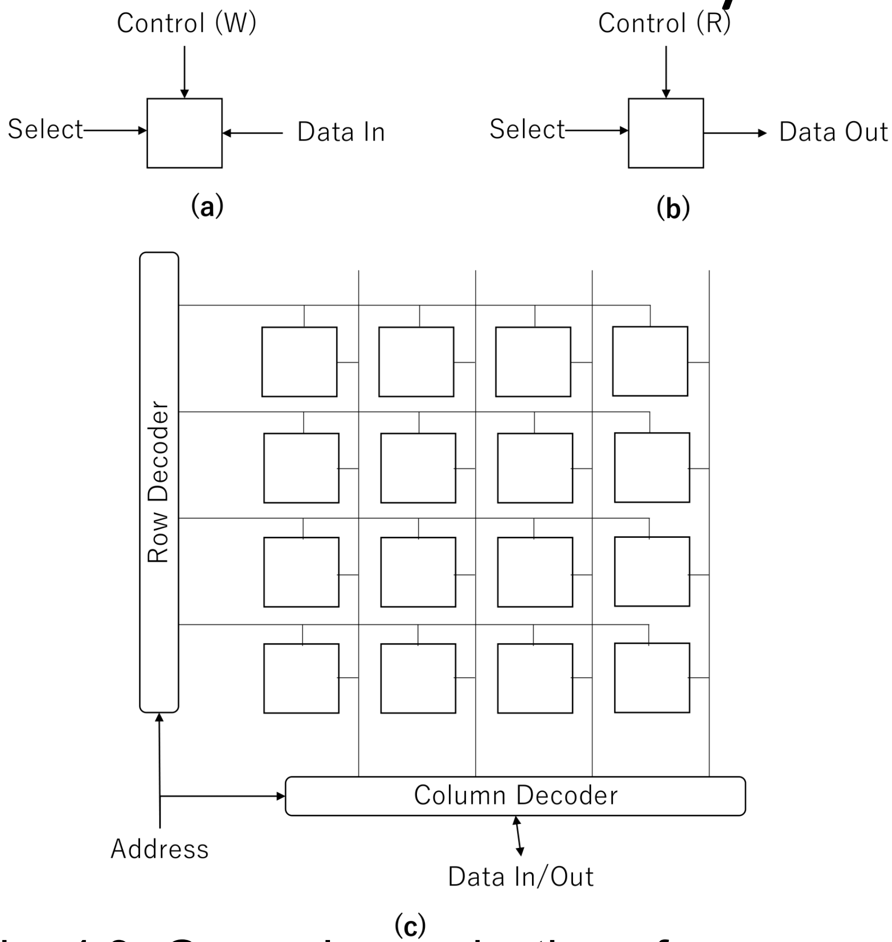


Fig. 4.3: General organization of a memory: (a) Memory cell write, (b) Memory cell read, (c) 2D array of memory cell.

- Memory cells are usually organized in a 2D array
- Accessing address is split into row and column addresses.
- Once row is selected, the whole content of the row will be read/written

2. Memory Technology
Overview of technologies

Technology	Cell size (F^2)	Write endurance	Speed (R/W)	Leakage power	Dynamic energy (R/W)	Retention period
Register	2200–3500	10^{16}	Extremely fast	Very high	Low	Voltage applied
SRAM	120–200	10^{16}	Very fast	High	Low	Voltage applied
eDRAM	60–100	10^{16}	Fast	Medium	Medium	30–100 μs
STT-RAM	6–50	4×10^{12}	Fast/slow	Low	Low/high	Years
RRAM	4–10	10^{11}	Fast/slow	Low	Low/high	Years
PCM	4–12	10^8–10^9	Slow/very slow	Low	Medium/high	Years
DWM	≥2	10^{16}	Fast/slow	Low	Low/high	Years
Flash (NAND)	1–4	10^4	Very slow	Very low	Low	Years

Table 4.1 The taxonomy of memory technologies with key design parameters

F: *feature size of the technology*

2. Memory Technology
SRAM cell

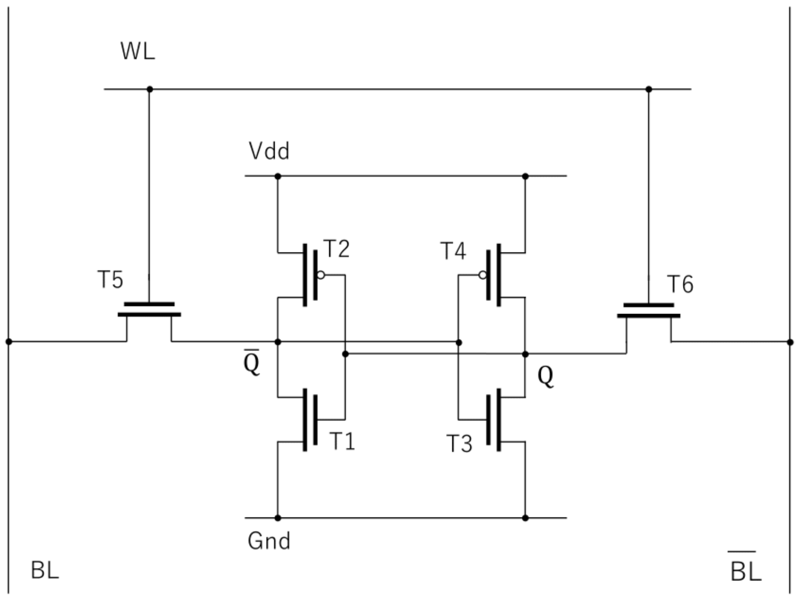


Fig. 4.4: A six transistors (6T) SRAM cell.

- Conventional Static Random Access Memory (SRAM) cell consists of 6 transisitors (6-T) which allow reading/writing and holding value as long as power is supplied.

2. Memory Technology
eDRAM cell

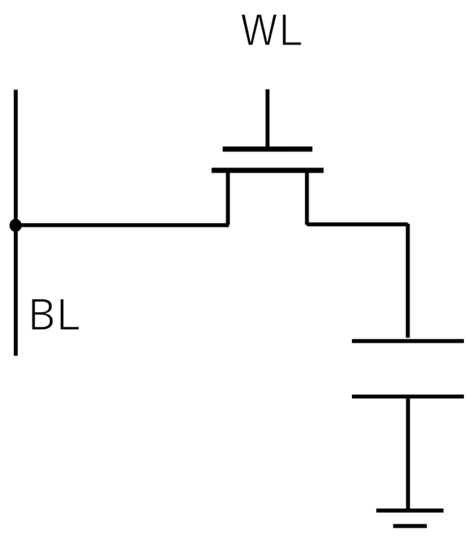


Fig. 4.5: eDRAM cell design: 1T1C.

- Dynamic RAM (DRAM) is another technology.
- DRAM cell stores in its capacitor.
  - Leakage of capacitor can reduce the voltage ➔ refresh needed
  - Reading can lose capacitor voltage ➔ reading also mean writing again
- Most common DRAM cell is 1T1C (1 transistor 1 capacitor):
  - Higher density than SRAM

2. Memory Technology
STT-RAM cell

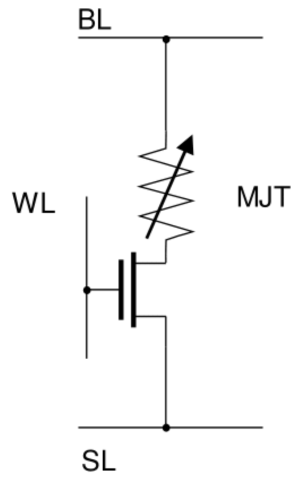
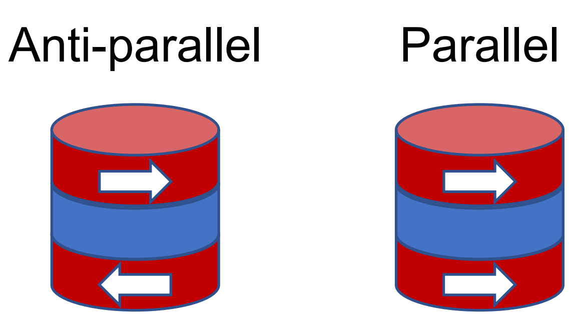
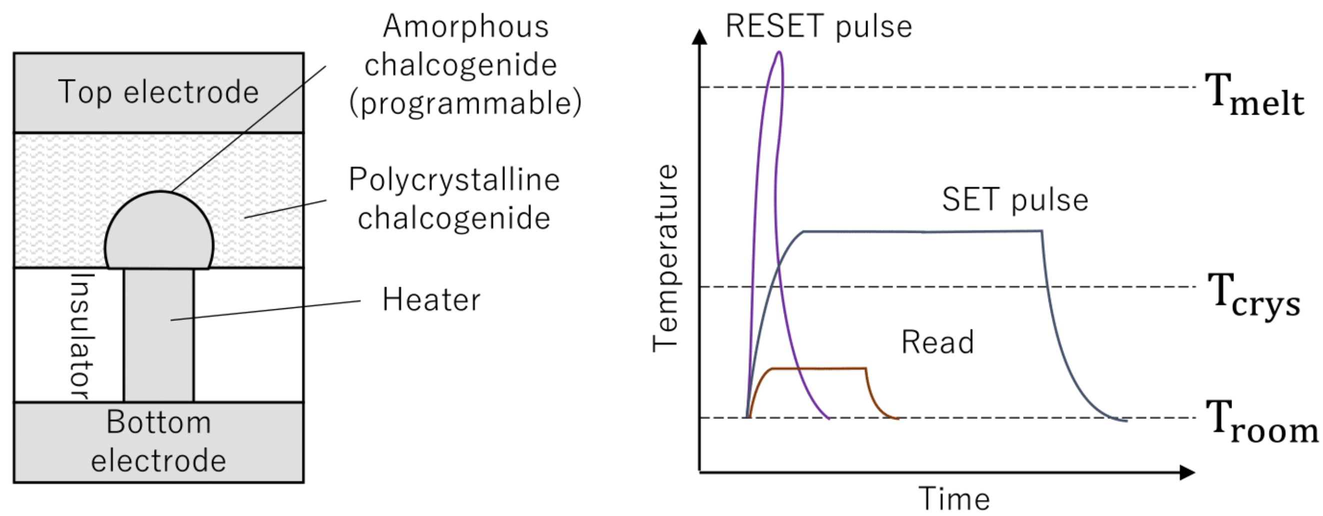


Fig. 4.6: A STT-RAM cell.



- A cell consists of a magnetic tunneling junction (MTJ)
- MJT consists of two ferromagnets (one is free, one is fixed) separated by a thin insulator
- MTJ is either:
    - low-resistive (parallel)
    - high-resistive (anti-parallel)
- STT-RAM is a non-volatile memory ➔ value will not lost after cutting power supply

2. Memory Technology
PCM Memory



(a) (b)

Fig. 4.9: Phase change memory: (a) A cross-section image of a mushroom-type PCM device. (b) The programming pulses and the resulting relative temperature for RESET, SET, and read operation in PCM.

2. Memory Technology
PCM Memory

- PCM is based on the property of certain materials, such as $Ge_2Sb_2Te_5$, which exhibit differences in resistivity in their two phases:
  - Crystallized: high resistance
  - Amorphous: low resistance
- In a PCM device, a small amount of one of the material is put between two metal terminals
- To program, SET or RESET pulses are put to PCM memory to increase/reduce the size the amorphous region

2. Memory Technology
RRAM cell

- Resistive Random Access Memory (RRAM) denote all memory technologies that rely on the resistance change to store the information.
- There are two structures:
  - *Conventional memory architecture*: Accessing like normal SRAM/DRAM using row/column decoder. RRAM cell stores binary bit (0/1).
  - *Resistive crossbar architecture:* Working with precise resistance value. One resistance = once synapse

2. Memory Technology
HfOx RRAM cell

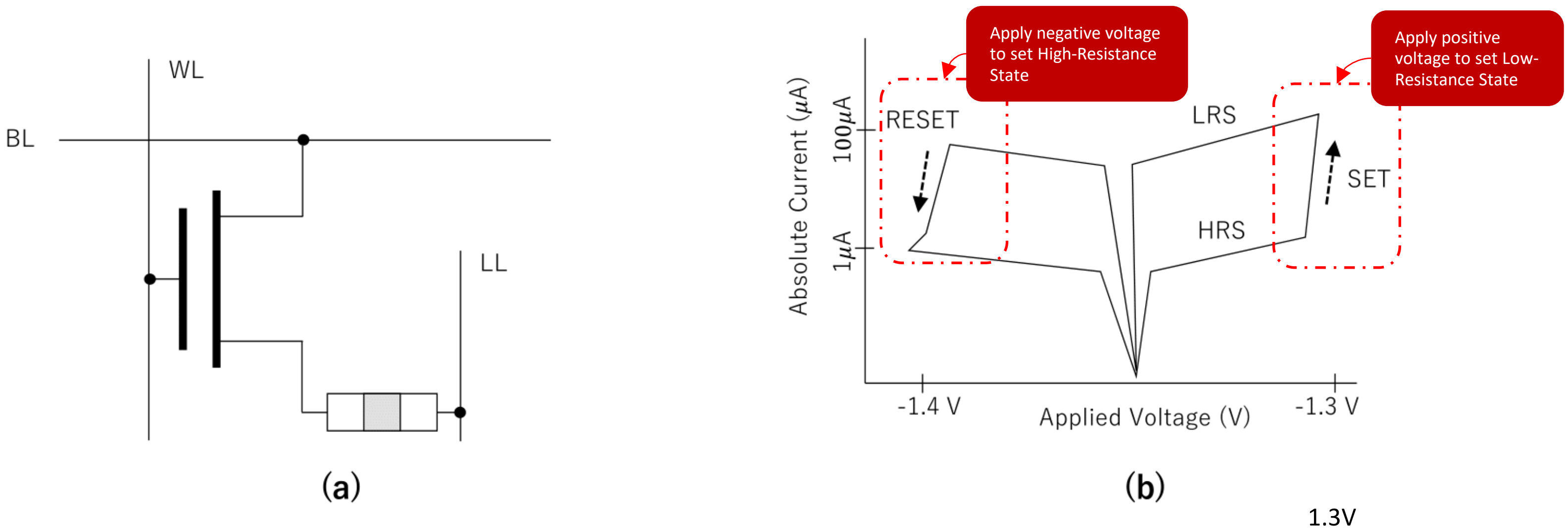


Fig. 4.7: RRAM cell: (a) Schematic. (b) I–V characteristics curve of a $HfOx$ RRAM cell [17]. Current is in absolute value. Readers may be more familiar with the I-V characteristics of memristor.

- HfOx RRAM can be written by applying voltage (positive and negative)
- Within the writing voltage values, RRAM cells work like a resistance in two modes:
  - High Resistance: Current is around 1 μA
  - Low Resistance : Current is around 100 μA

2. Memory Technology
Resistive Crossbar

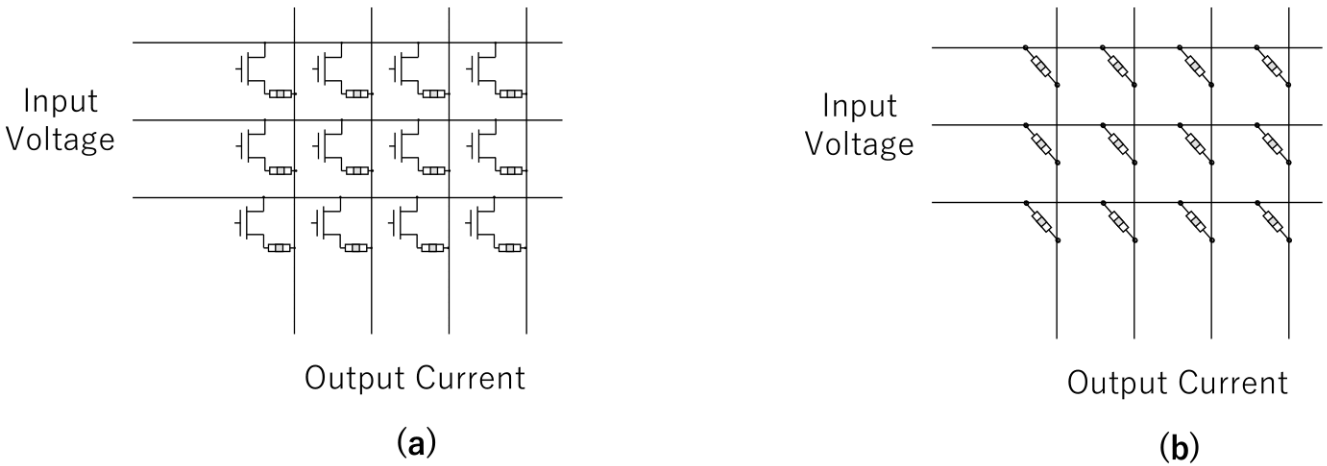


Fig. 4.8: Resistive crossbar design: (a) 1T1R. (b) 1 0T1R

- There are two design of resistive crossbar:
  - With transistor (1T1R): Reading is done via row selection (input current) and column selection (transistor enabling).
  - Without transistor (0T1R): Reading is done via row selection (input currents) and measuring output current.

Lecture Contents

1. Introduction of Memory
2. Memory Technology
3. Memory Organization
4. Memory for Neuromorphic Systems
5. Dynamics of NVM Synapse
6. Conclusions

3. Memory Organization
Introduction

- A semiconductor memory consists of a 2D array of M×N cells (M rows and N columns)
  - If the number of the columns (N) is the accessing bit-width (word's width), no column decoder is needed.
  - If the number of the columns (N) is a multiples of bit-width, column decoder is need.
  - If the number of the columns (N) is a divisors of the bit-width, reading process must take several cycles

3. Memory Organization

Memory Block-diagram

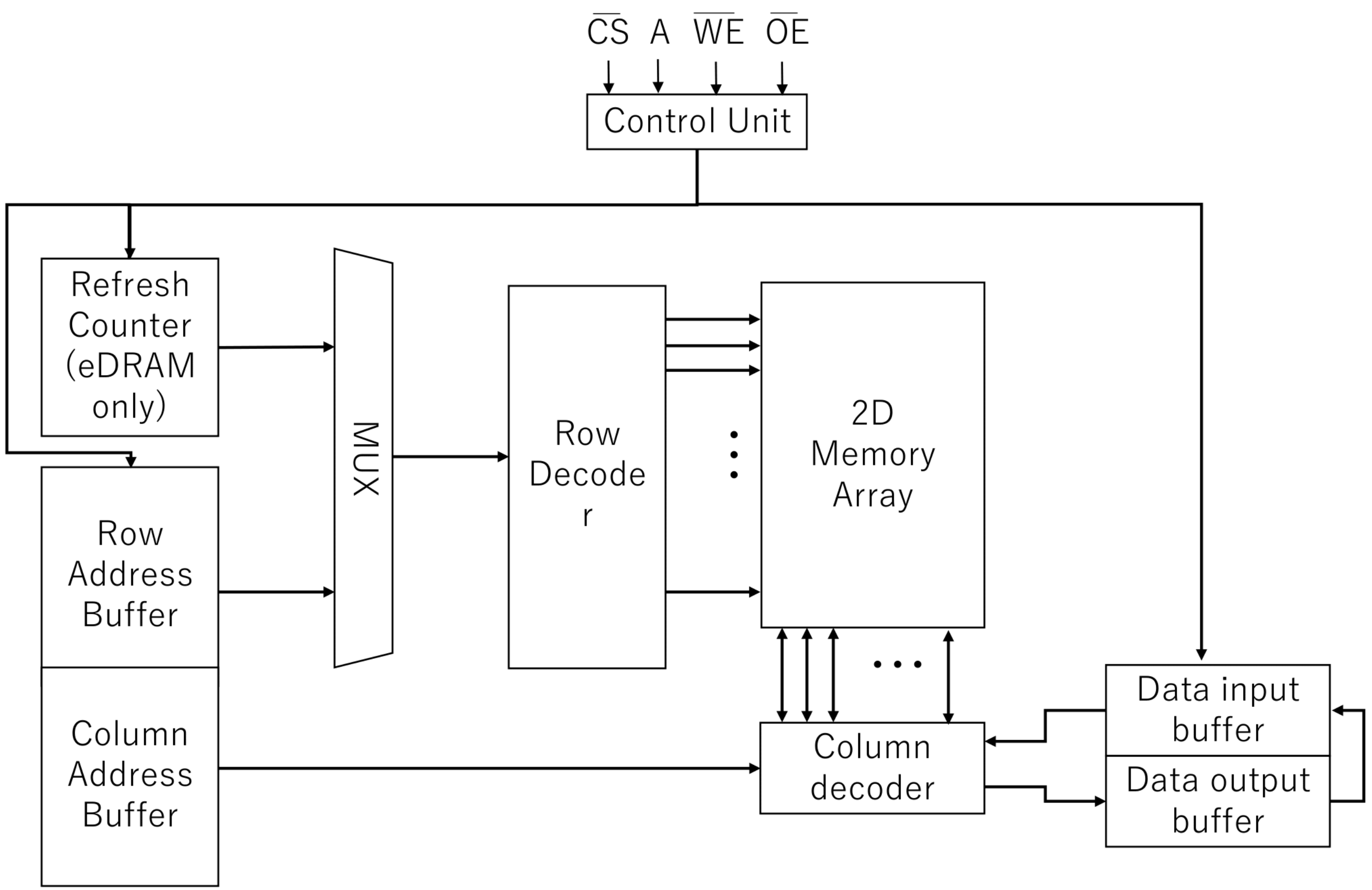


Fig. 4.10: Organization of a semiconductor memory

3. Memory Organization
Writing process

- Writing process is usually done:
  - Enabling chip select (CS) signal
  - Enabling the write enable (WE) signal
  - Putting the corresponding address (A)
  - Putting the data into the data line
- Depending on the technology, writing process can take one cycle (SRAM, DRAM) or multiple cycle (STT, PCM,…)

3. Memory Organization
Writing waveform

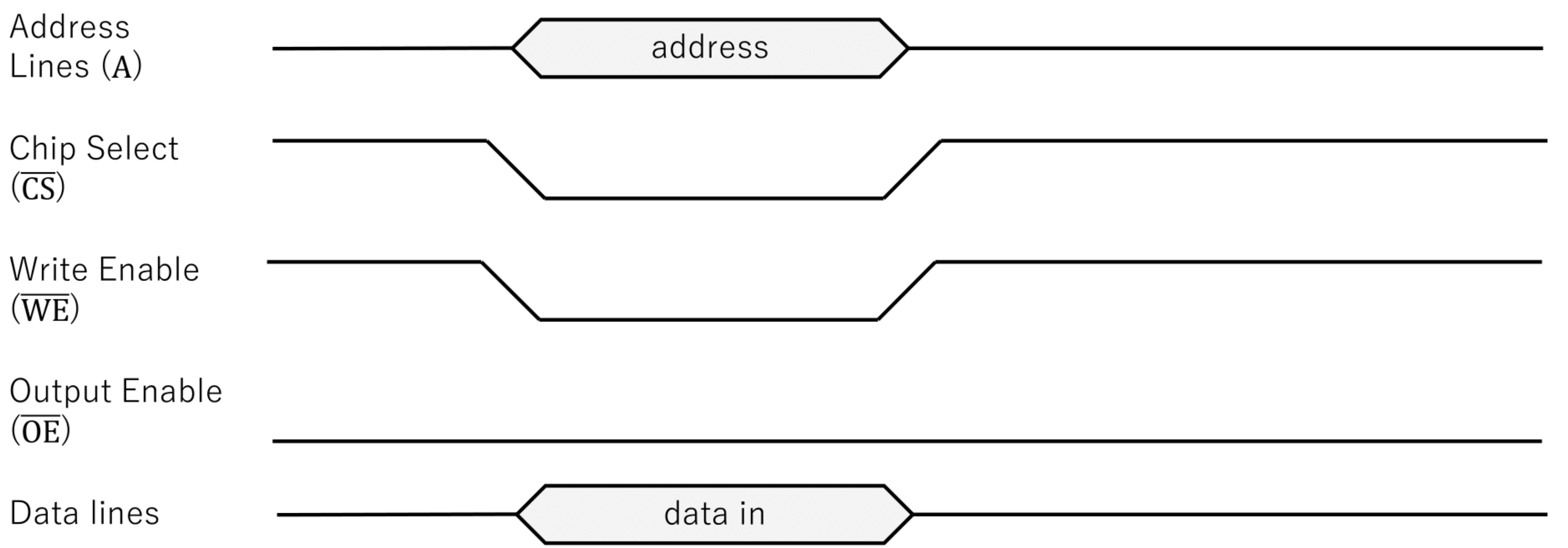


Fig. 4.11: Simplified memory waveform: writing data.

3. Memory Organization
Reading process

- Similar to writing, reading process is usually done:
  - Enabling chip select (CS) signal
  - Enabling the read enable (RE) signal
  - Putting the corresponding address (A)
  - Reading the data from the data line after a certain inveral
- Depending on the technology, reading process can take one cycle (SRAM, DRAM) or multiple cycle (PCM,NAND,…)

3. Memory Organization
Reading waveform

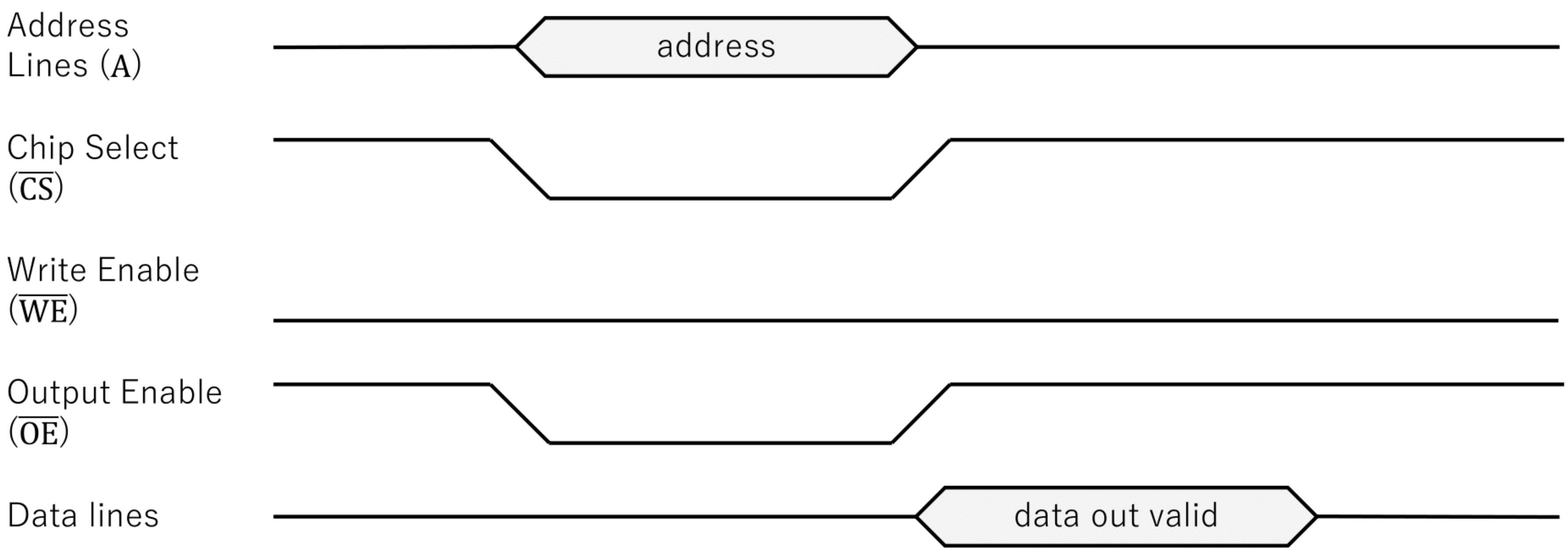


Fig. 4.12: Simplified memory waveform: reading data.

Lecture Contents

1. Introduction of Memory
2. Memory Technology
3. Memory Organization
4. Memory for Neuromorphic Systems
5. Dynamics of NVM Synapse
6. Conclusions

4. Memory for Neuromorphic Systems
Overview

- Neuromorphic systems typically need to store three major types of data: spikes, neuron states, and weights
- Spike are usually stored in registers or SRAM for low latency reading processes.
- Memory design for spike:
  - FIFO: first in first out
  - Sorting/scheduling structure: enabling finding the proper spikes for processing
- Neuron's state can be stored internally for parallel neuron design or externally for serial neuron design:
  - Serial neuron parameter must be load/stored up on request.

4. Memory for Neuromorphic Systems
Neuron's architecture

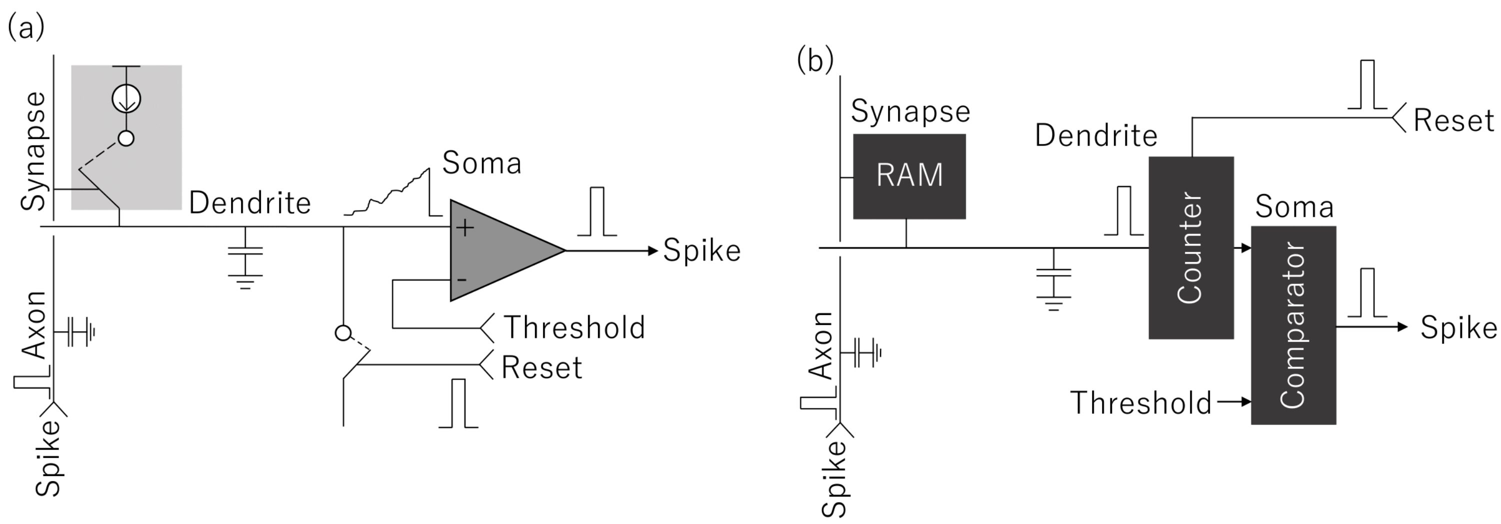


Fig. 4.13: Analog and digital silicon neurons. (a) Analog implementation. (b) Digital implementation.

Neuron's architecture can be digital or analog based. For storing analog neuron's parameter, sampling and storing digitally is needed.

4. Memory for Neuromorphic Systems
Serial neuron

- In serial neuron design, one physical neuron is used for multiple neurons' computations.
  - It starts by loading the parameters of the computing neurons from the memory.
  - It then compute the neuron
  - At the end of the time-step, parameter are stored back to the memory.
  - After finishing the current computing neuron, the next neuron is computed.
- The major benefit of serial neuron design is low hardware cost; however, it requires multiple reading/writing processes for the computing.

4. Memory for Neuromorphic Systems
Serial neuron

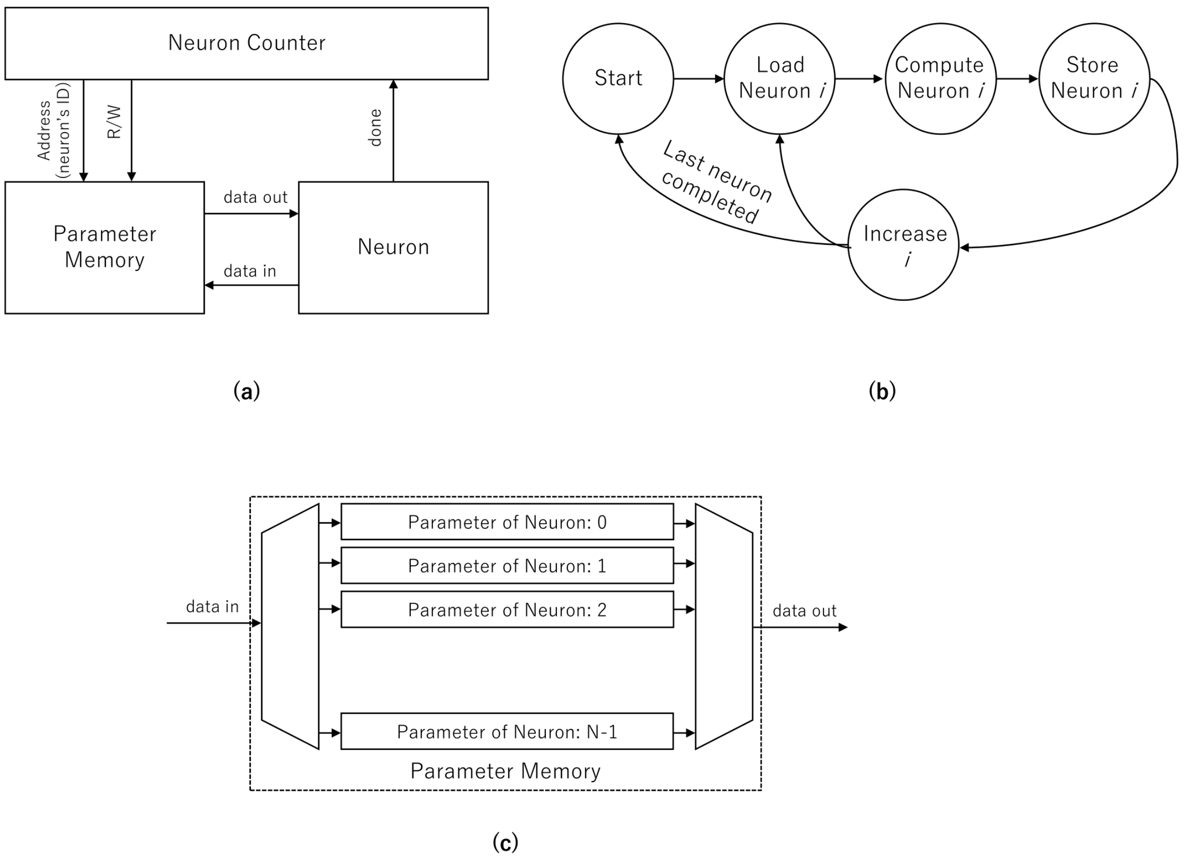
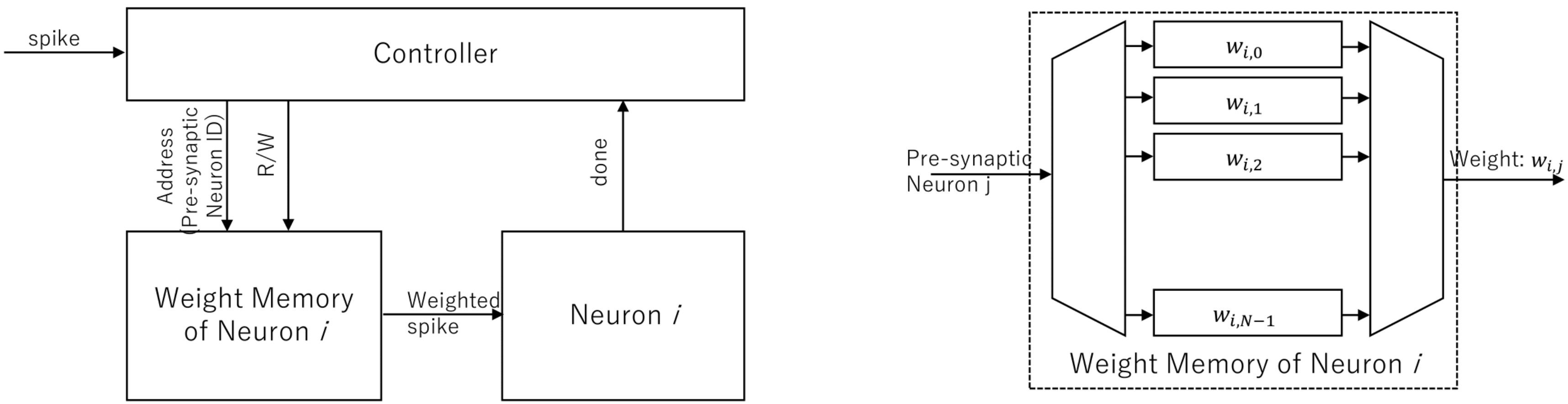


Fig. 4.14: The serial neuron model. (a) The model architecture (b) The finite state machine. (c) The parameter structure.

4. Memory for Neuromorphic Systems
Serial neuron

- In serial neuron design, one physical neuron is used for multiple neurons’ computations.
  - It starts by loading the parameters of the computing neurons from the memory.
  - It then compute the neuron
  - At the end of the time-step, parameter are stored back to the memory.
  - After finishing the current computing neuron, the next neuron is computed.

4. Memory for Neuromorphic Systems
Parallel neuron



(a) (b)

Fig. 4.15: The parallel neuron weight model. (a) The model architecture (b) The weight structure.

4. Memory for Neuromorphic Systems
Parallel neuron

- In parallel neuron design, one physical neuron is used for one neuron' computations.
  - Parameter is loaded at the initialization stage.
  - No loading and storing needed during the inference
- The major benefit of parallel neuron design is non existent loading/storing time.
- However, the hardware cost for parallel neuron is problematic.

4. Memory for Neuromorphic Systems
Weight memory

- Weights (or synapses) are usually stored in memory nearby the neuron.
- Neuron (physical) has its own dedicated memory due to bottle neck issue of shared memory.
- One word can store one weight or several weight (merged).

4. Memory for Neuromorphic Systems
Weight operation

- Once a spike is received, the corresponding weight address is decoded.
- With non-merged weight, each address is for one weight; therefore, the reading process is used to compute the weighted spike
- With merged weight, each address is for multiple weights, therefore, after reading, a column decoding is need to split the weighted spike.

4. Memory for Neuromorphic Systems

Serial neuron: Weight operation

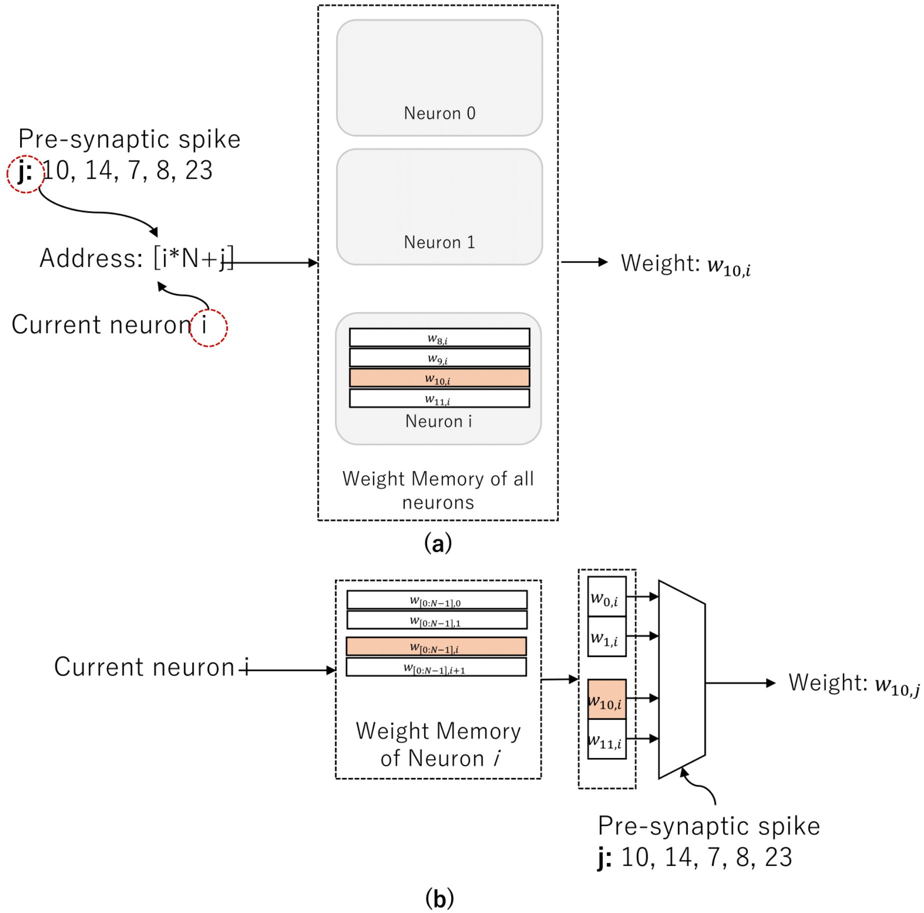


Fig. 4.18: The serial neuron weight memory operation: (a) normal weight, (b) merged weight.

4. Memory for Neuromorphic Systems
Parallel neuron: Merged weight

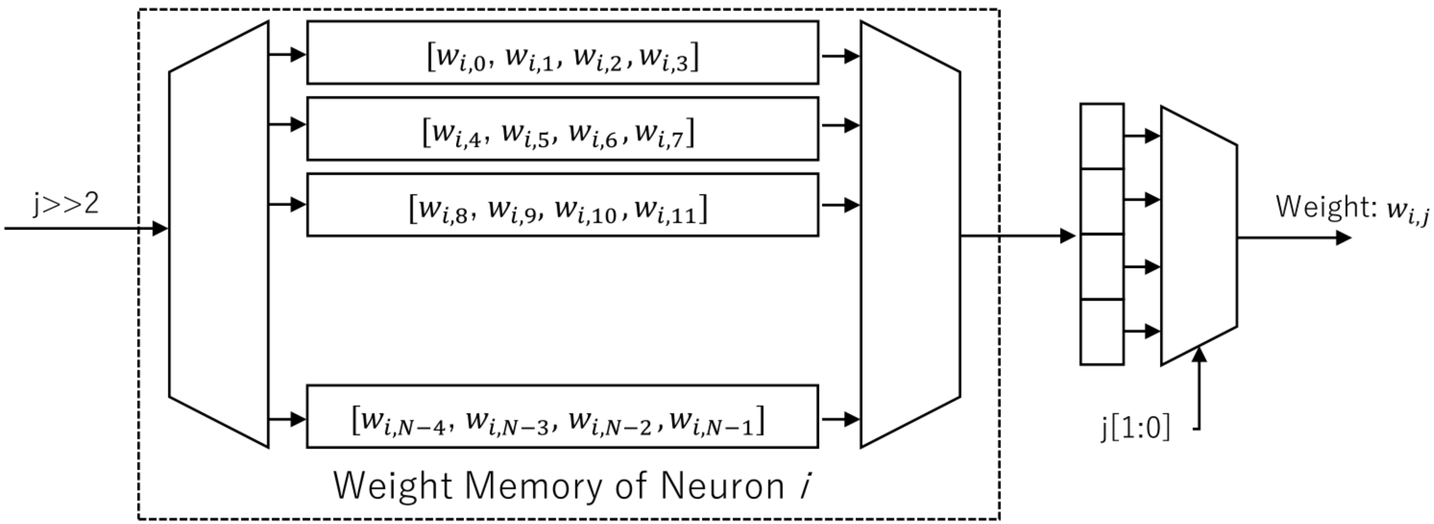


Fig. 4.16: The parallel neuron weight memory with merged four weights in a memory row.

- Instead of storing a single weight, several adjacent weights are stored in the same address
- It can increase the density; however, power consumption may not be efficient

4. Memory for Neuromorphic Systems

Parallel neuron: Weight operation

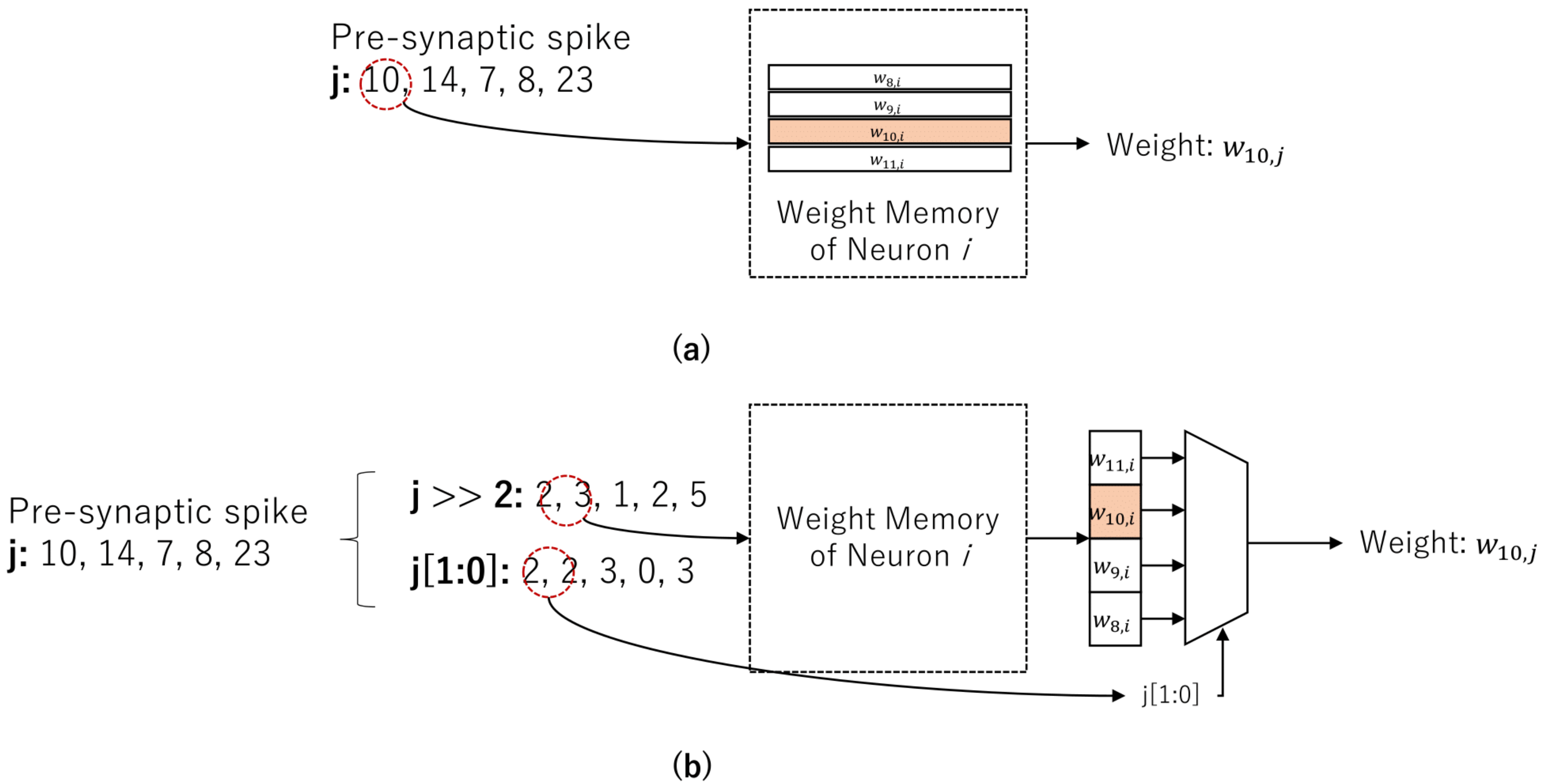


Fig. 4.17: The parallel neuron weight memory operation: (a) separated weight, (b) merged weight.

4. Memory for Neuromorphic Systems

Crossbar

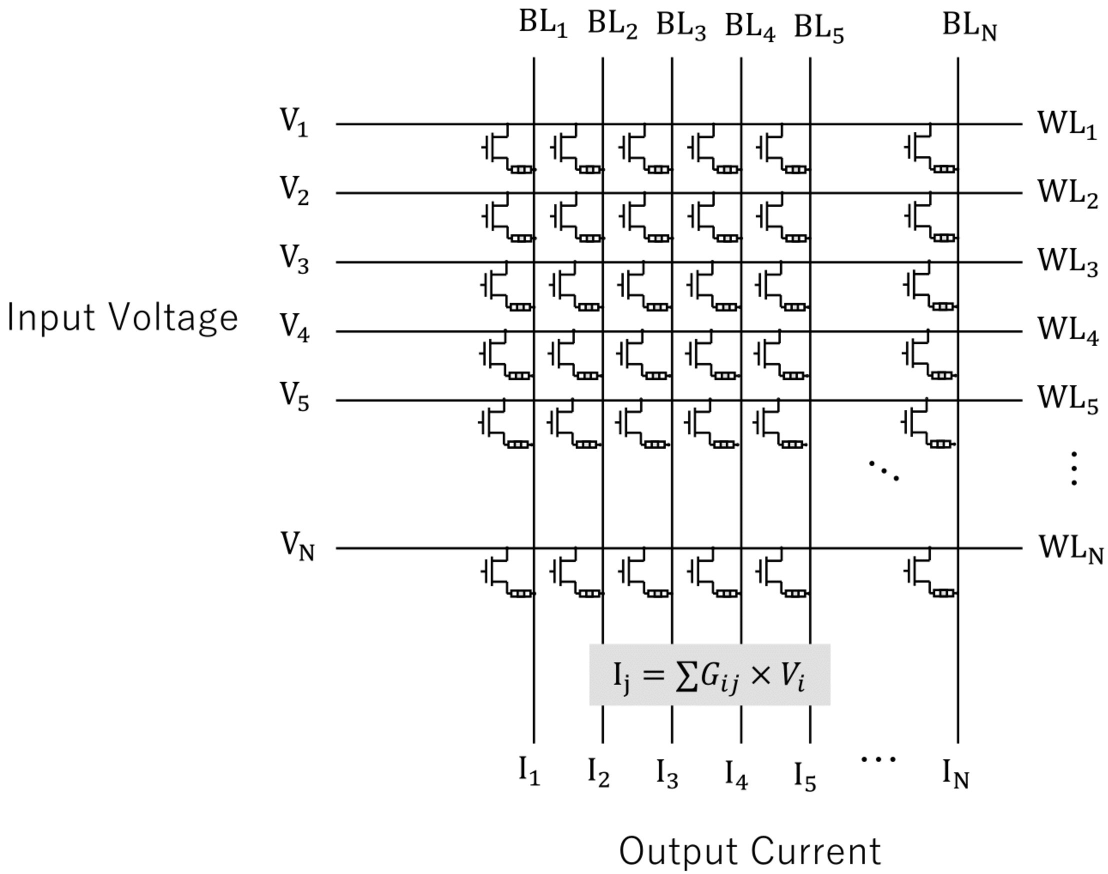


Fig. 4.19: Schematic for multiple layer neural network using NVM: Crossbar for two connected layers.

4. Memory for Neuromorphic Systems
Crossbard & In-memory computing

- The output current for neuron j (I_j) is calculated as the summary of the current provided by all presynaptic neuron voltage (I_{ij}) (the Kirchhoff's law):

$$I_j = \sum I_{ij}$$

- where I_{ij} is dependent on the applied voltage and the conductance of the NVM cell (as the Ohm's Law):

$$I_{ij} = V_i \times G_{ij}$$

Hence:

$$I_j = \sum V_i \times G_{ij}$$

This act like the multiplication and accumulation process

Lecture Contents

1. Introduction of Memory
2. Memory Technology
3. Memory Organization
4. Memory for Neuromorphic Systems
5. Dynamics of NVM Synapse
6. Conclusions

5. Dynamics of NVM Synapse
Overview

- Unlike conventional memories, non volatile memory (NVM) has a drifting phenomenon:
  - Writing process may enable not "accurate" resistance, especially with analog in-memory computing
    - Material is not homogeneous, therefore, the resistance value are different between memory after the writing process (assuming with the same writing time)
  - The resistance value will be "drifted" over time.
    - Low resistance becomes higher resistance
    - High resistance becomes lower resistance

5. Dynamics of NVM Synapse
Learning related

- With *ex-situ* learning process, weight are not adjusted after training.
  - It has little effect for binary NVM as low flipped bit rate has small impact on overall performance.
  - For in memory computing based, adjustment is needed to alleviate the affect
- With *in-situ* learning, the drifting process must be taken into account:
  - The new adjust weight value may not be as desired

Lecture Contents

1. Introduction of Memory
2. Memory Technology
3. Memory Organization
4. Memory for Neuromorphic Systems
5. Dynamics of NVM Synapse
6. Conclusions

5. Conclusion

- In this chapter, we have reviewed several memory technologies:
  - Conventional memory: SRAM, DRAM
  - Non-volatile memory: STT-RAM, PCM, RRAM
- Memory structure is also analyzed:
  - Serial vs parallel neuron design
  - Merged vs non-merged design
- The other issues such as in-memory computing and the drifting process of NVM are also reviewed