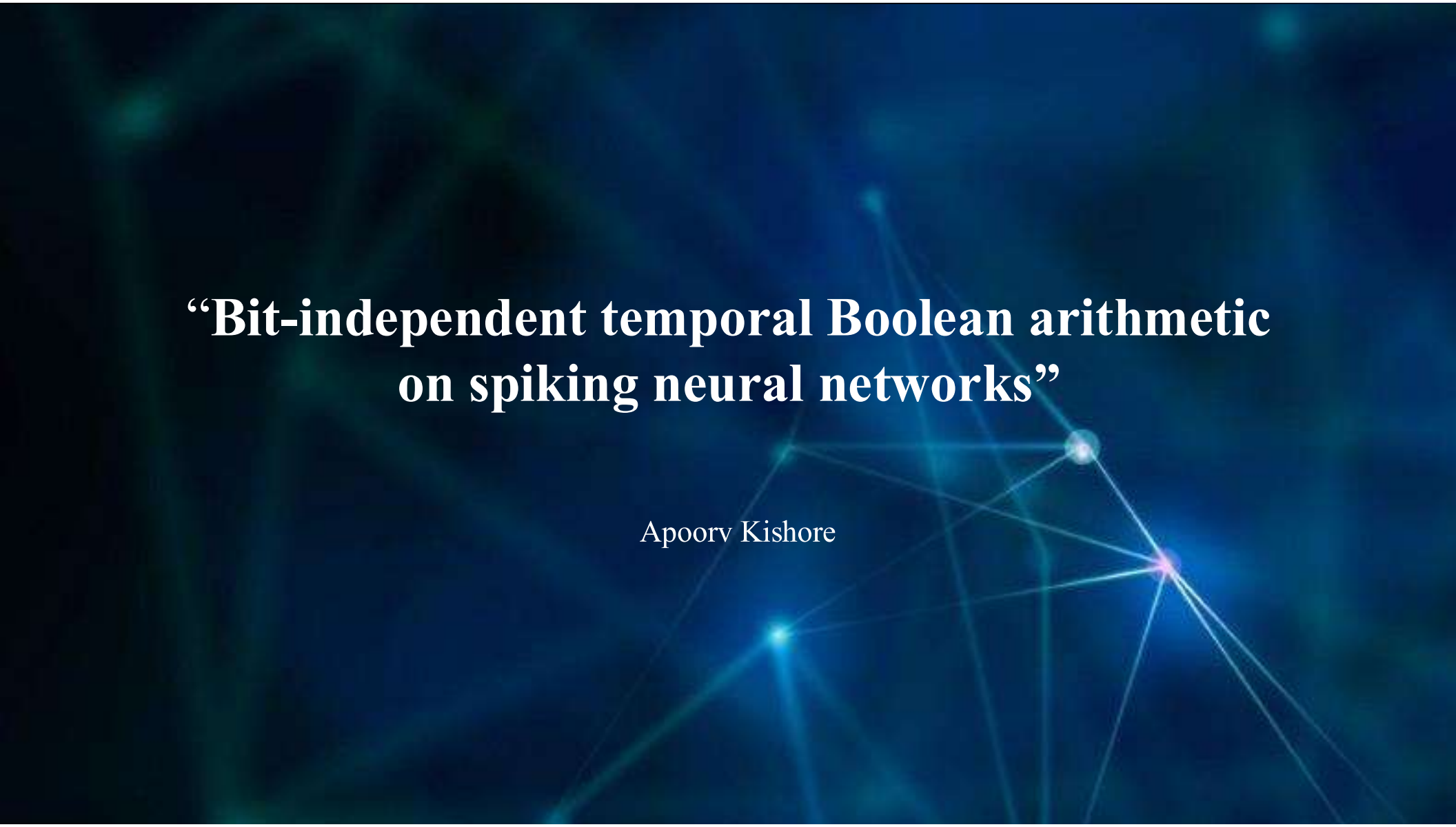
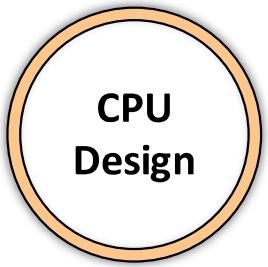


“Bit-independent temporal Boolean arithmetic on spiking neural networks”

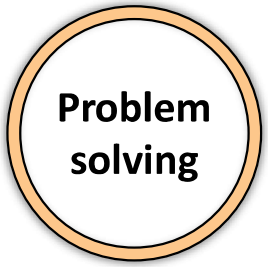
Apoorv Kishore

Motivation



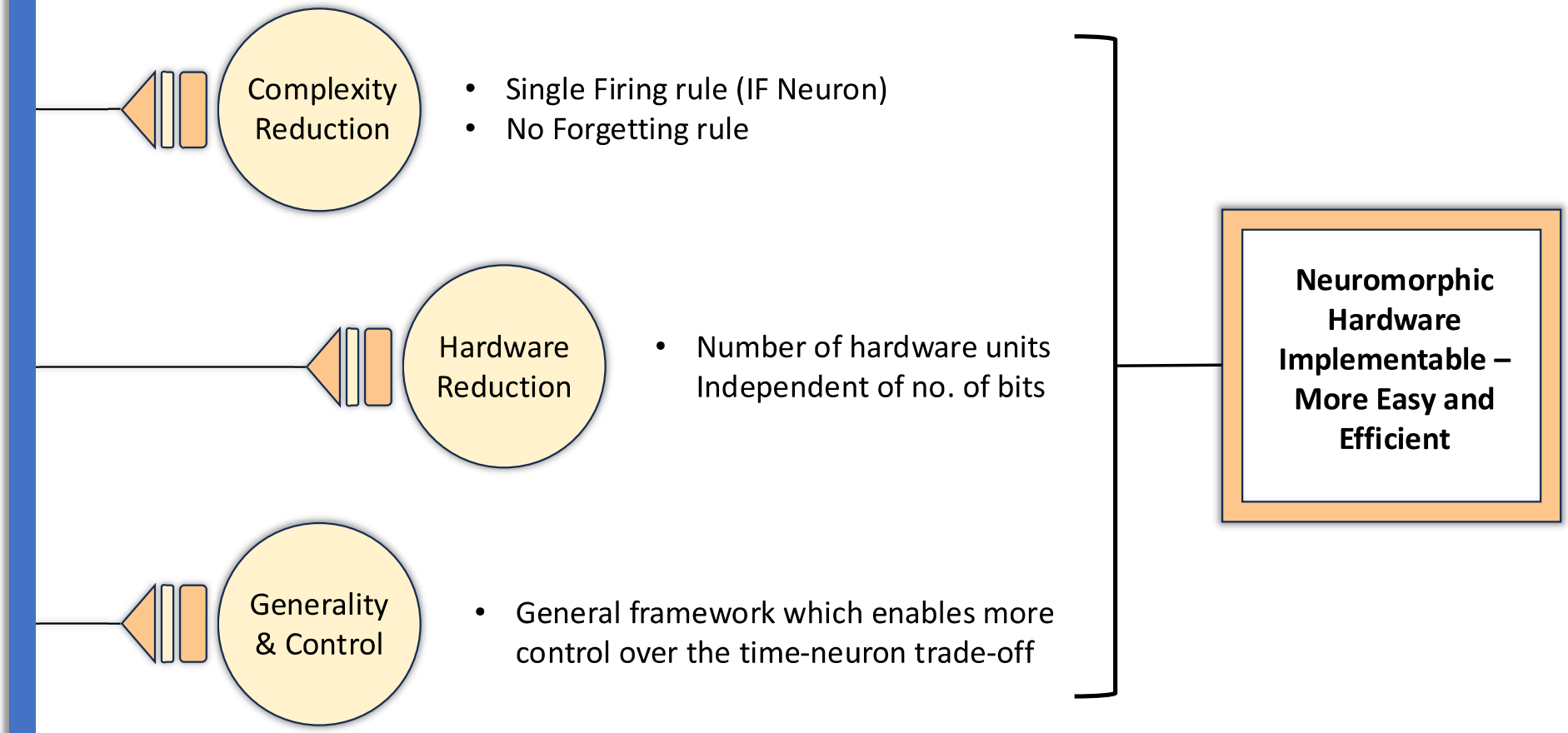
- Develop an alternative architecture for an ALU
- Can be further used to create a CPU entirely from Neural Processing Units

- Algorithmically, uses lesser number of computing units
- Moving towards a complexity-effective design of an ALU



- Seamless integration with DNNs : built upon the same computing unit
- A step towards a complete hardware solution without a digital processor
- Can be applied in problems dependent on quantitative operations – navigation, information fusion, risk assessment, etc

Algorithmic Improvements



Algorithmic Improvements: Outcome

Adder

	Paper1	Paper2	Current
Time	N + 3	2N + 4	N+3
Neurons	3	N + 8	6
Types of Firing rules	2	3	1
Types of Forgetting rules	1	3	0

Subtractor

	Paper1	Paper2	Current
Time	N + 3	2N + 3	N+5
Neurons	10	N + 13	10
Types of Firing rules	4	7	1
Types of Forgetting rules	3	4	0

Multiplicator

	Paper1	Paper2	Current
Time	6+M+N	3N + 5	$\frac{(M-1)(N+3M-2)}{2}$
Neurons	K$\frac{N^2+15N+8}{2}$	3N + 8	K
Types of Firing rules	3 + N/2	5	1
Types of Forgetting rules	2 + N/2	4	0

Neuron Models

Used in the paper

a) A custom spiking rule: $\frac{a^s}{a^c} \rightarrow a, d$ → when neuron has 's' spikes in it, where $s \geq c$, 'c' spikes would be lost leaving 's-c' spikes stored in it and emit a single spike denoted by 'a' after a delay 'd'

b) A custom forgetting rule: $\frac{a^s}{a^k} \rightarrow \lambda$ → when neuron has 's' spikes in it, it will just forget the 'k' spikes stored in it, leaving 's-k' spikes stored and also emit nothing

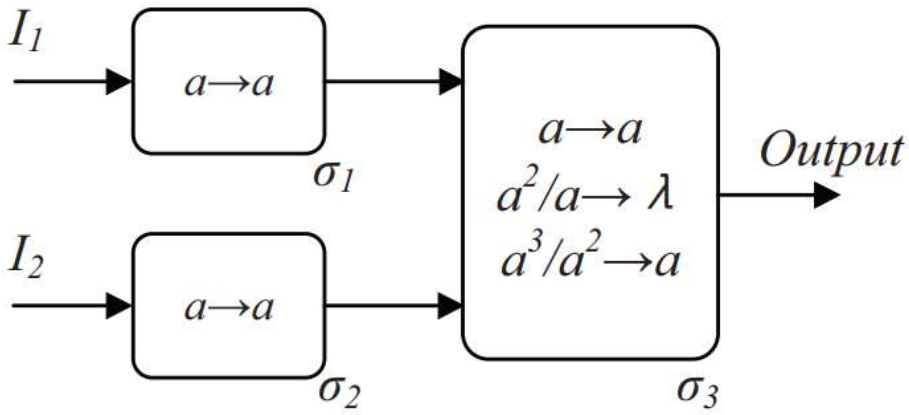
Used here

a) IF Neuron: $V(t) = V(t-1) + \sum_{j=1}^{n} w_j s_j^{in}$

$s^{out} = 1 \; if \; V(t) > V_{th}$

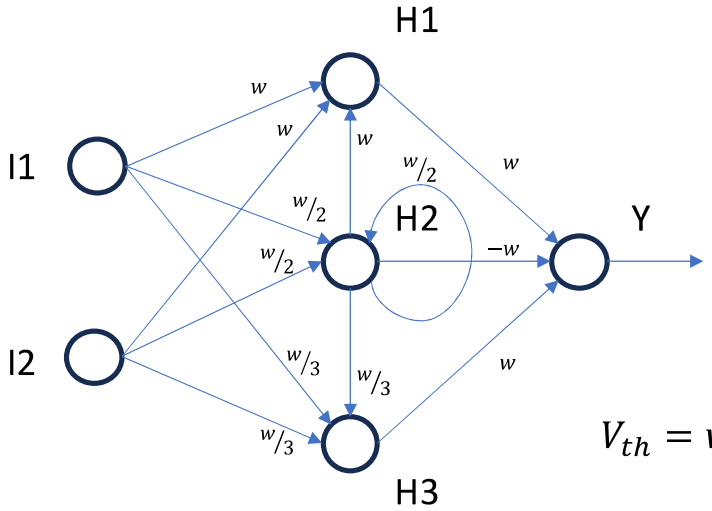
→ where w_j is the weight of the input synapse connected from j^{th} pre-neuron and s_j^{in} is the corresponding input spike. s^{out} is the output spike and V_{th} is the threshold voltage

Adder network: Prev Paper



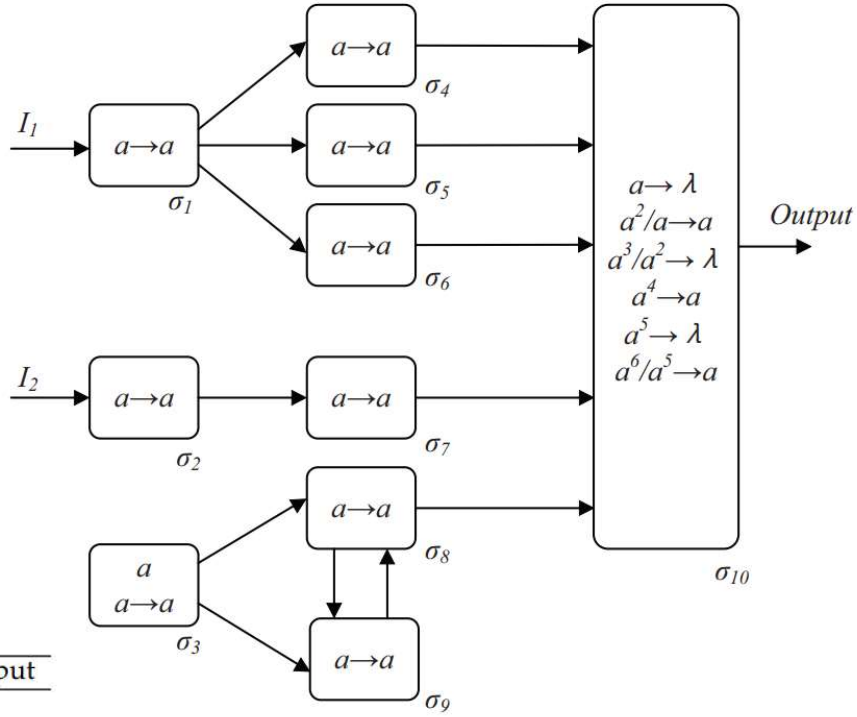
No. of steps	σ_1	σ_2	σ_3	output
0	0	0	0	-
1	**0**	**1**	0	-
2	**1**	**0**	1	-
3	**1**	**1**	1	**1**
4	**1**	0	2	**1**
5	0	0	2	**0**
6	0	0	1	**0**
7	0	0	0	**1**

Adder network: Proposed



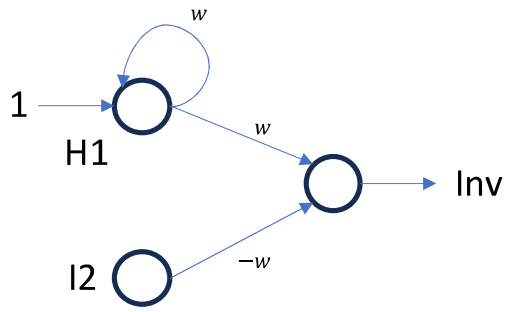
7	6	5	4	3	2	1	time
0	0	0	1	1	1	0	I1
0	0	0	0	1	0	1	I2
0	1	1	1	1	1	0	H1
0	0	1	1	0	0	0	H2
0	0	0	0	0	0	0	H3
1	0	0	1	1	0	0	Y

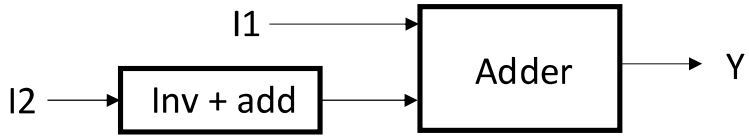
Subtractor network: Prev Paper



No. of steps	σ_1	σ_2	σ_3	σ_4	σ_5	σ_6	σ_7	σ_8	σ_9	σ_{10}	output
0	0	0	1	0	0	0	0	0	0	0	-
1	**0**	**1**	0	0	0	0	0	1	1	0	-
2	**1**	**0**	0	0	0	0	1	1	1	1	-
3	**1**	**1**	0	1	1	1	0	1	1	2	0
4	**1**	0	0	1	1	1	1	1	1	5	**1**
5	0	0	0	1	1	1	0	1	1	5	**0**
6	0	0	0	0	0	0	0	1	1	4	**0**
7	0	0	0	0	0	0	0	1	1	1	**1**

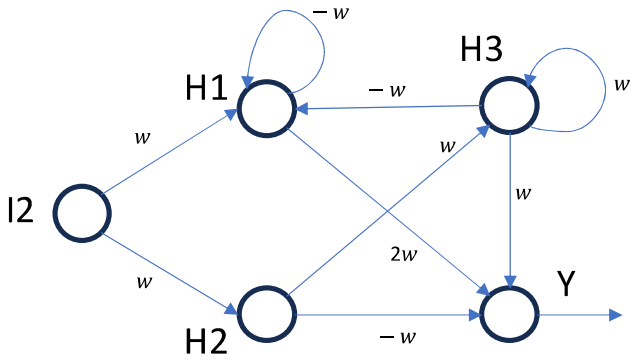
Subtractor network: Proposed

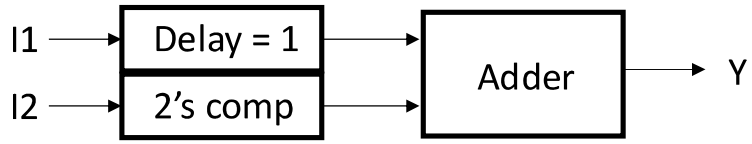




5	4	3	2	1	0	time
0	0	0	0	0	1	1
0	1	0	1	1	0	I2
1	1	1	1	1	0	H1
0	1	0	0	0	0	Y

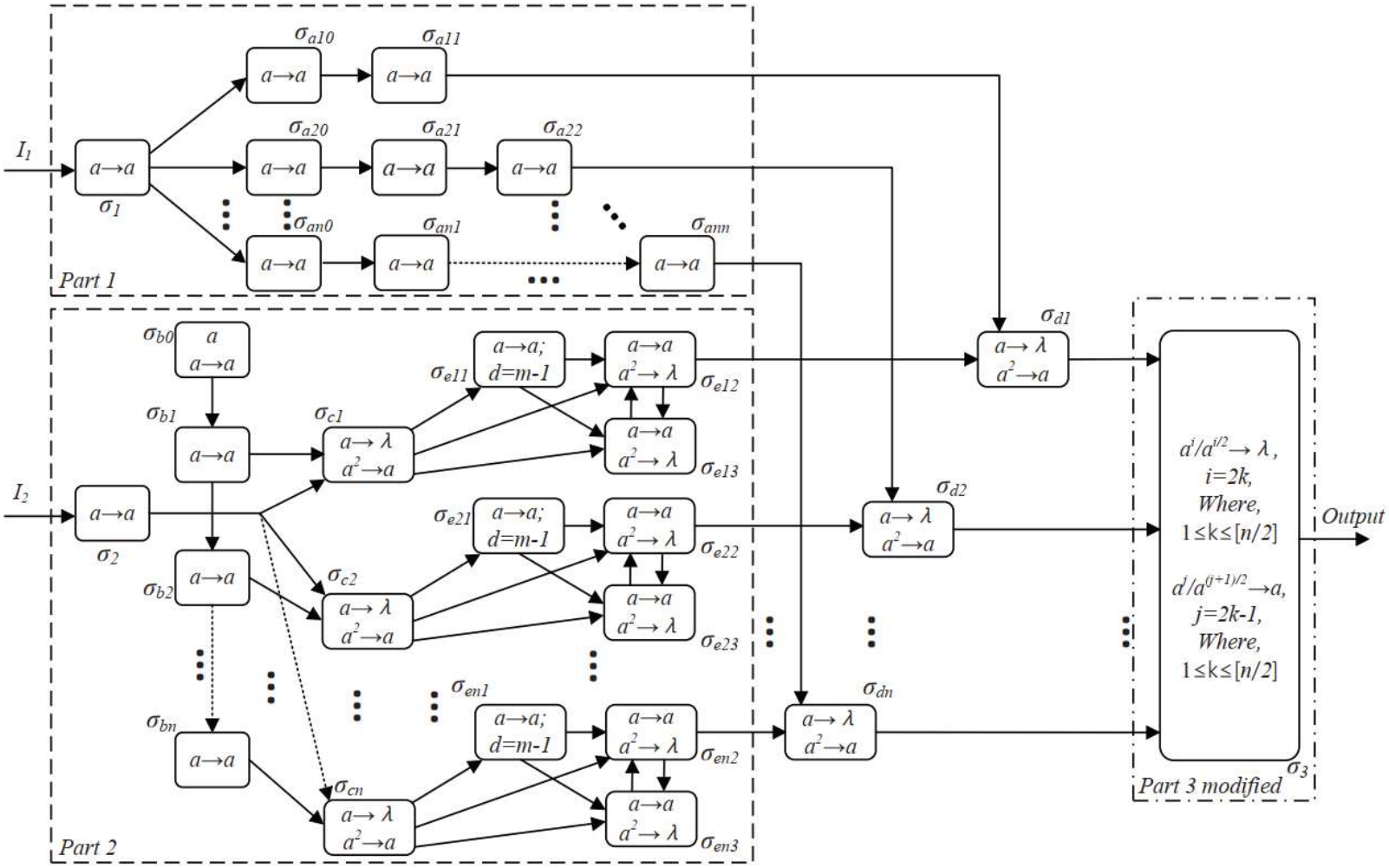
Subtractor network: Proposed



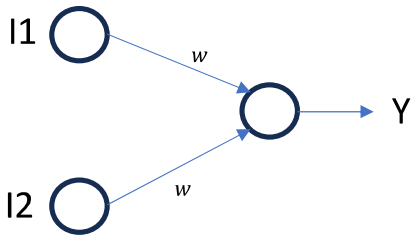


7	6	5	4	3	2	1	time
0	1	0	1	1	0	0	I2
0	0	0	1	0	0	0	H1
1	0	1	1	0	0	0	H2
1	1	1	0	0	0	0	H3
0	1	0	1	0	0	0	Y

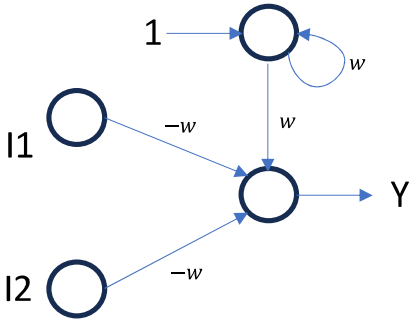
Multiplicator network: Prev Paper



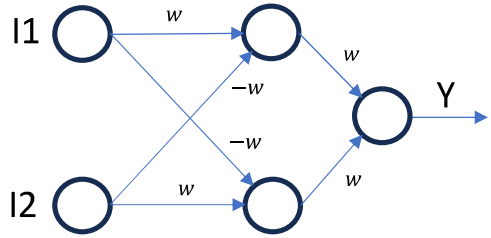
Basic Logic networks



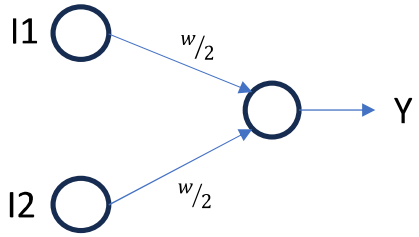
OR



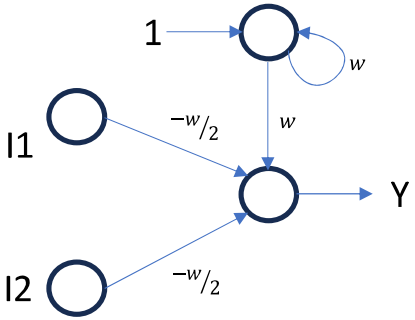
NOR



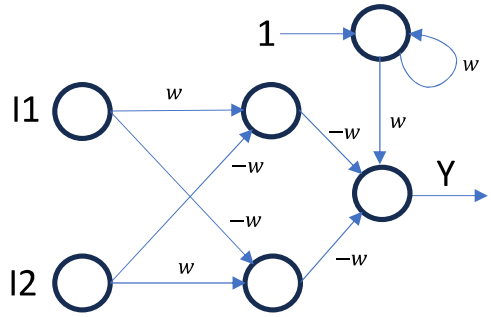
XOR



AND

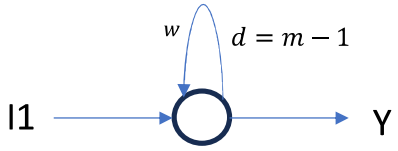


NAND

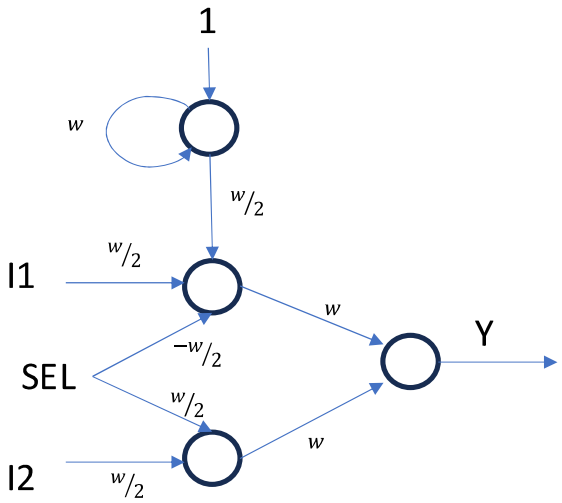


XNOR

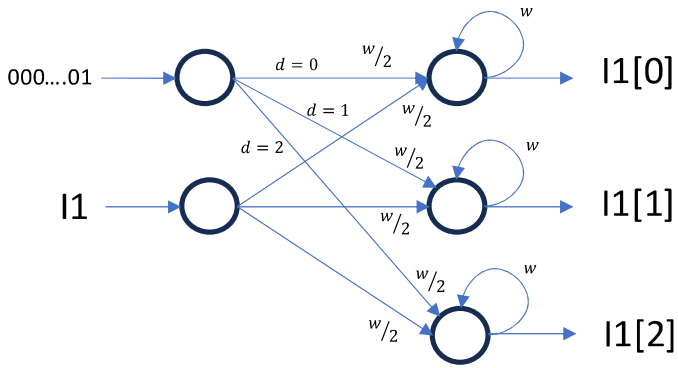
Basic Logic networks



REPEATER

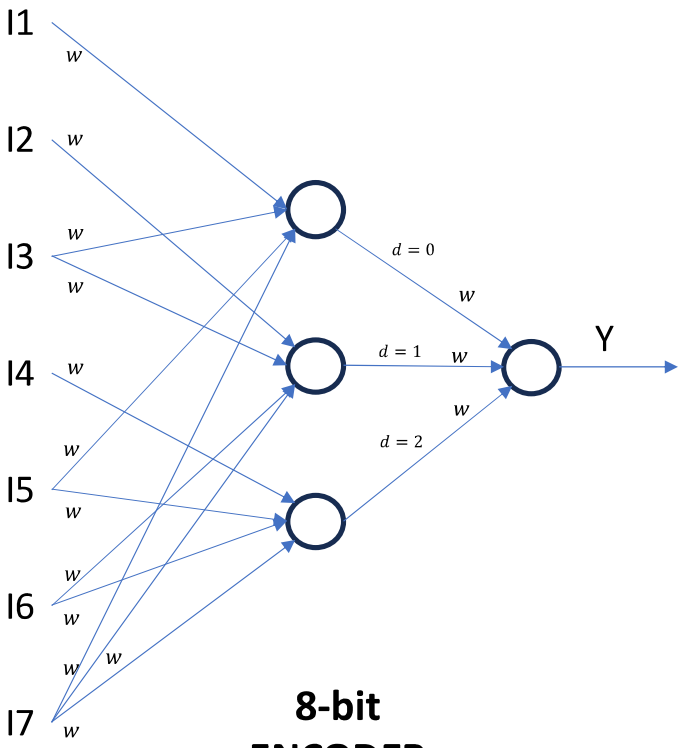


MUX

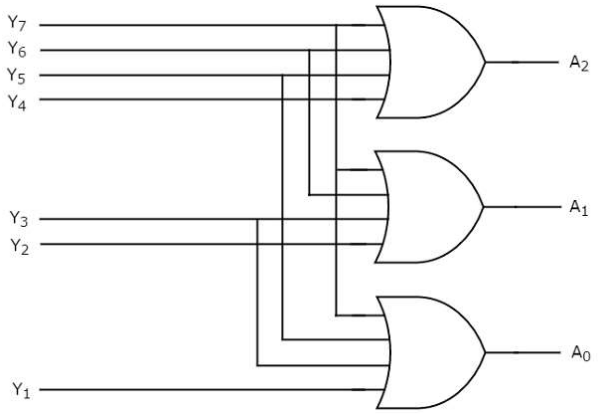


DESERIALIZER

Basic Logic networks

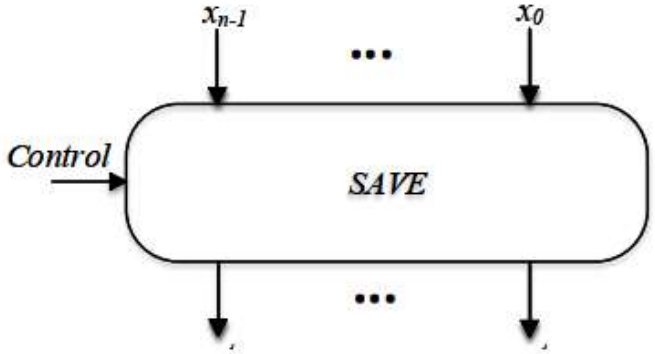


8-bit ENCODER

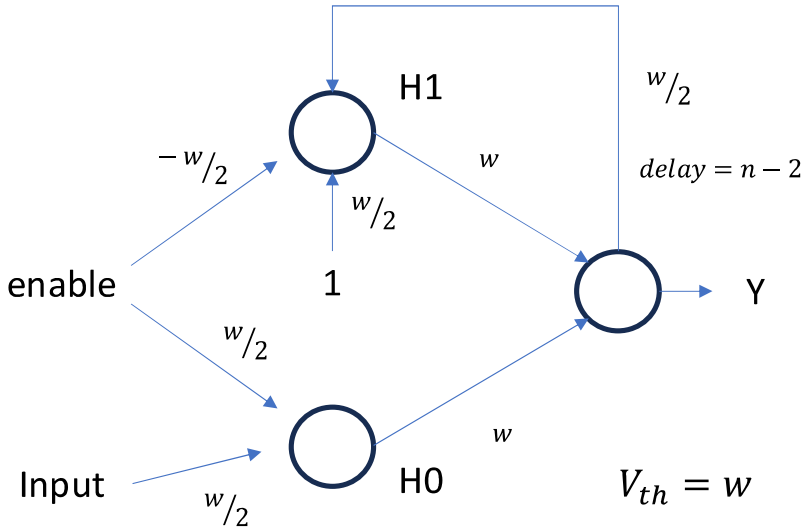


Digital Equivalent

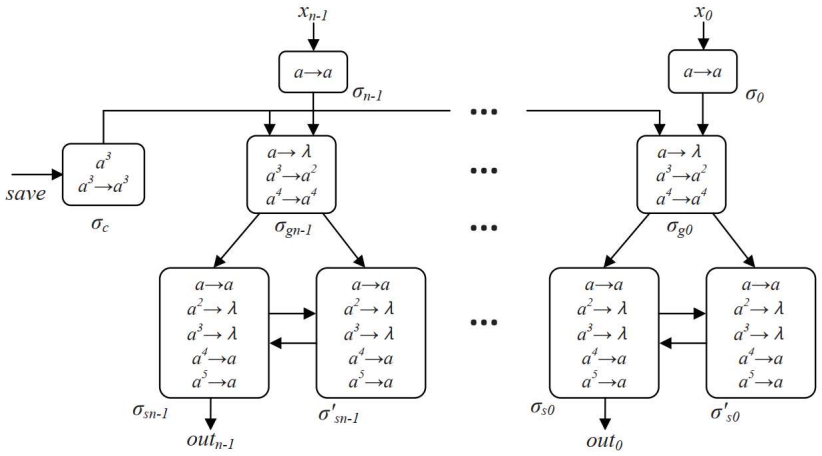
Storage unit

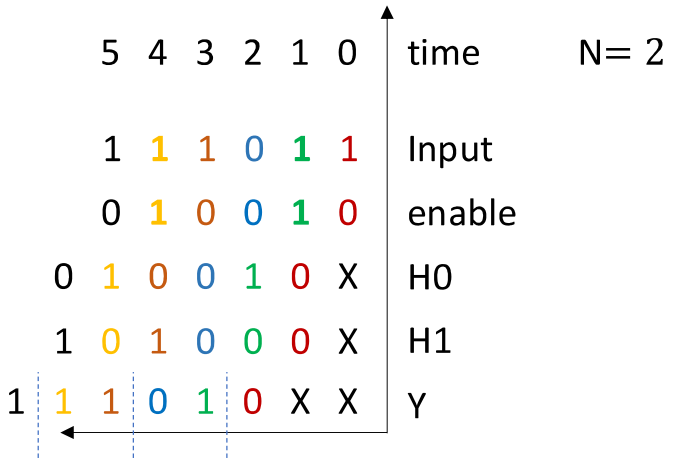


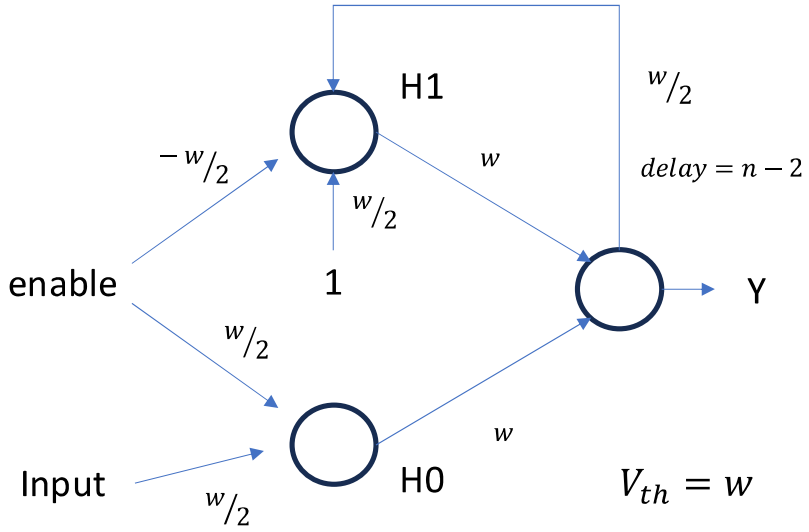
Paper



Currently Proposed





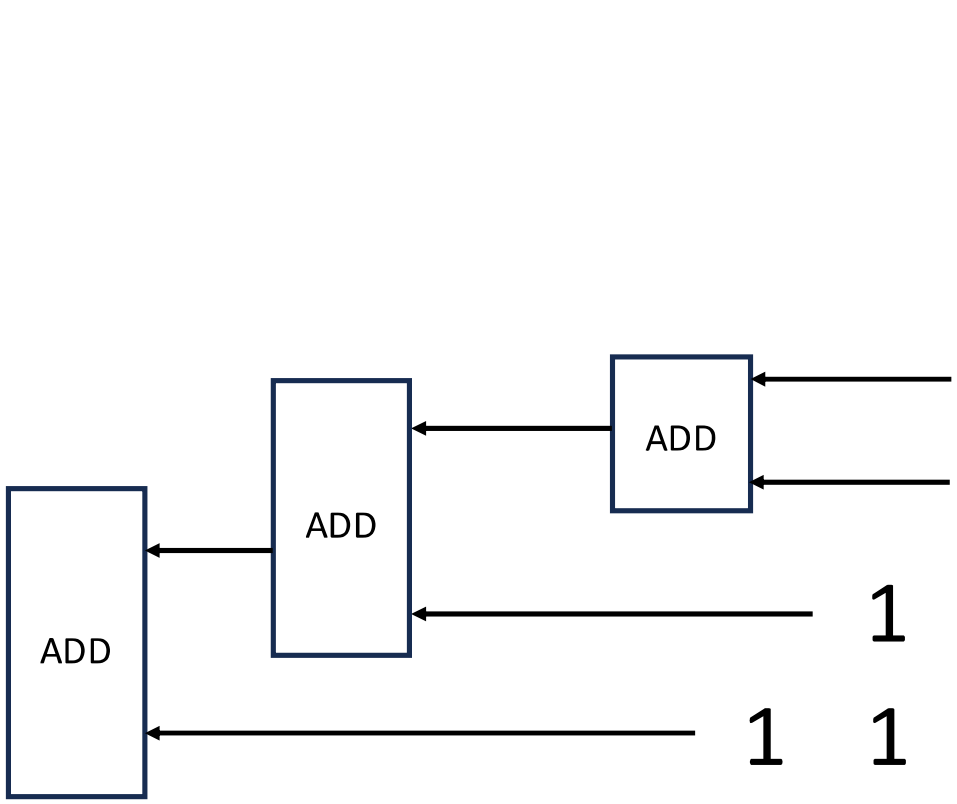


N-bit storage Unit

N= 2

	5	4	3	2	1	0	
							time
	1	1	1	0	1	1	Input
	0	1	0	0	1	0	enable
0	1	0	0	1	0	X	H0
1	0	1	0	0	0	X	H1
1	1	1	0	1	0	X	X	Y

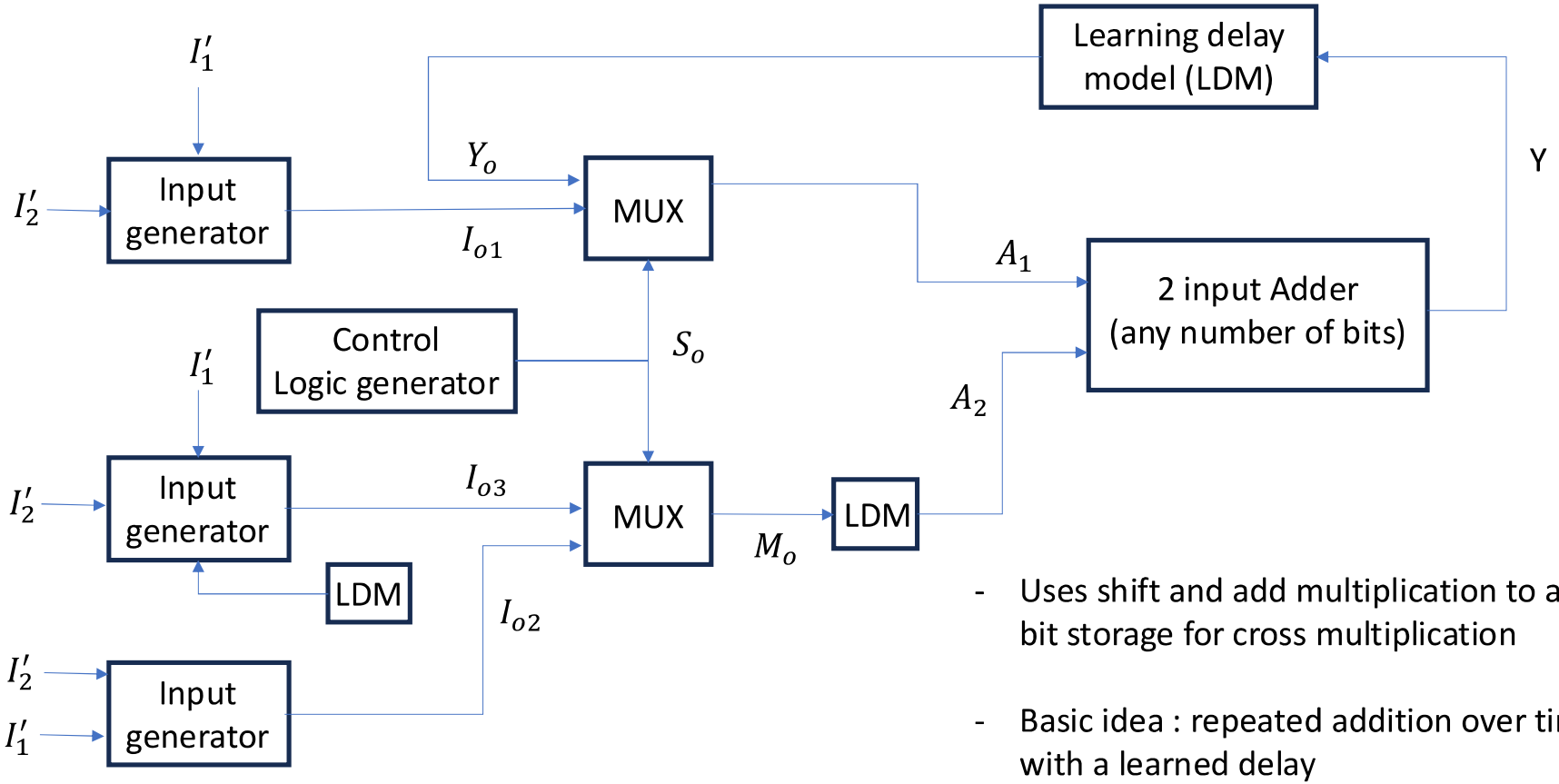
Multiplicator network: Proposed



		1	1	0	1	
X	1	1	1	1		
		1	1	0	1	First Pass
1	1	0	1	X		
1	1	0	1	X	X	Iterative
1	1	0	1	X	X	X

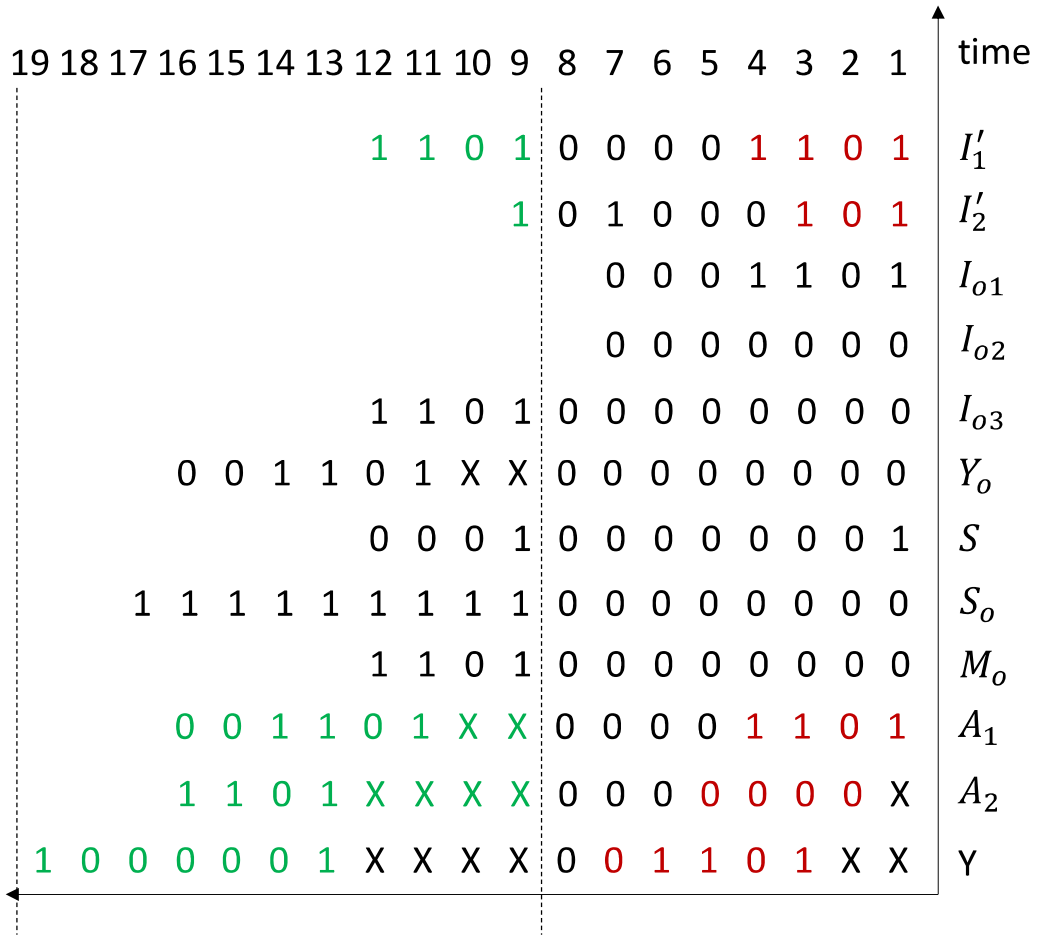
Multiplicator network: Proposed

Rough Top level architecture of the proposed multiplier network

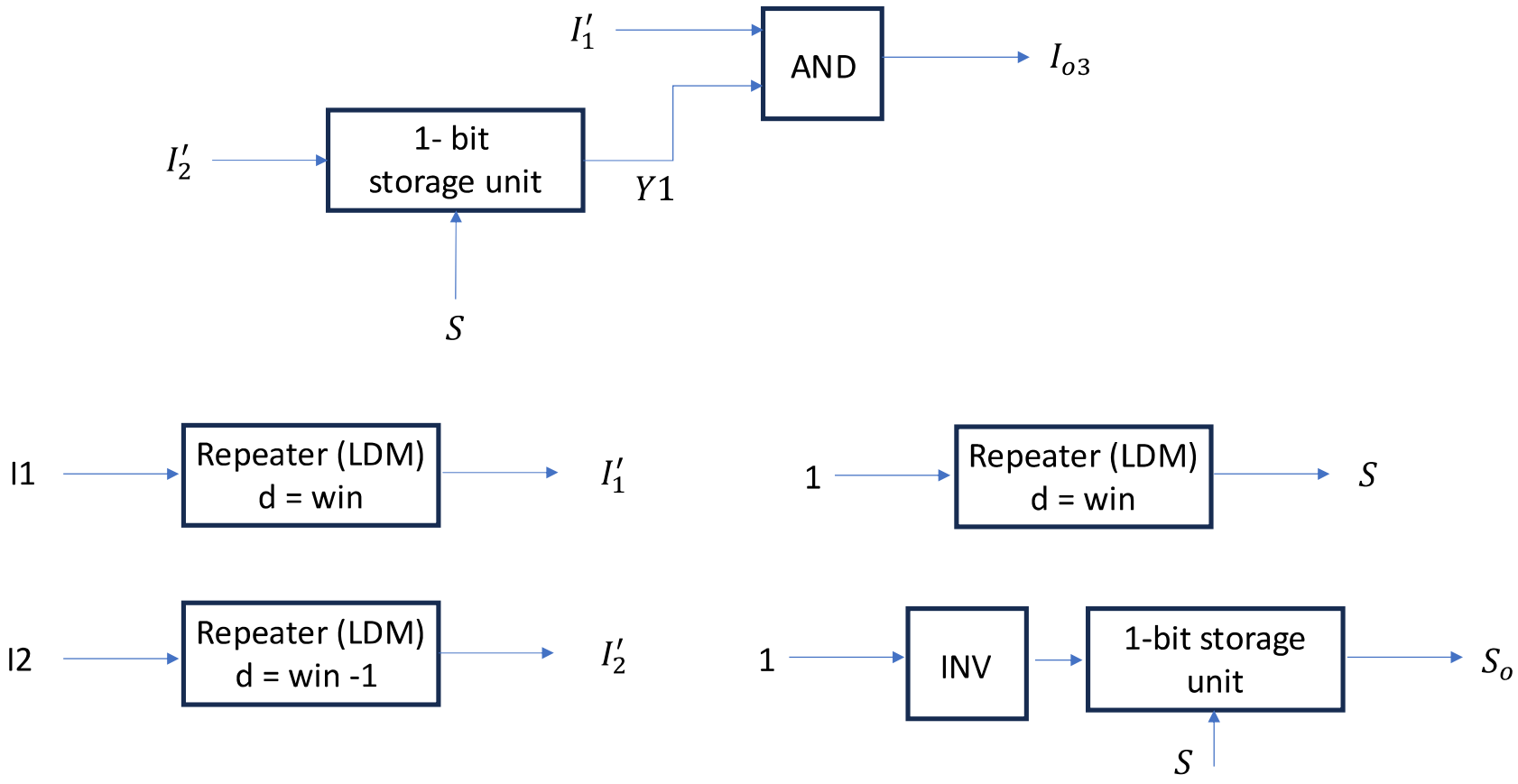


- Uses shift and add multiplication to avoid bit storage for cross multiplication
- Basic idea : repeated addition over time with a learned delay

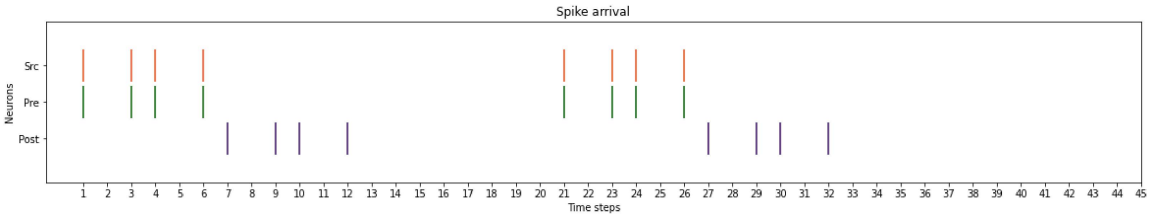
Multiplicator circuit: Proposed

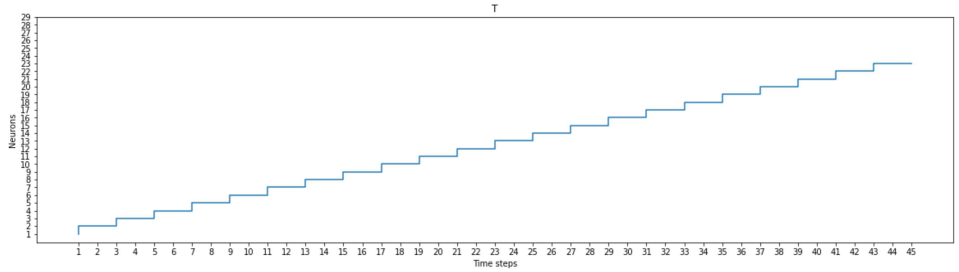


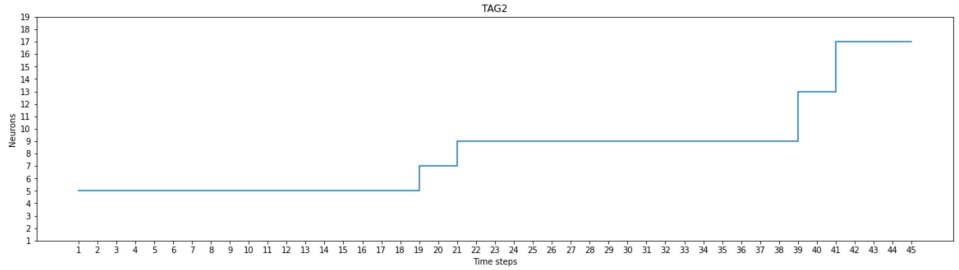
Iterative Input Generator



Learning Delay Model (LDM) - LAVA

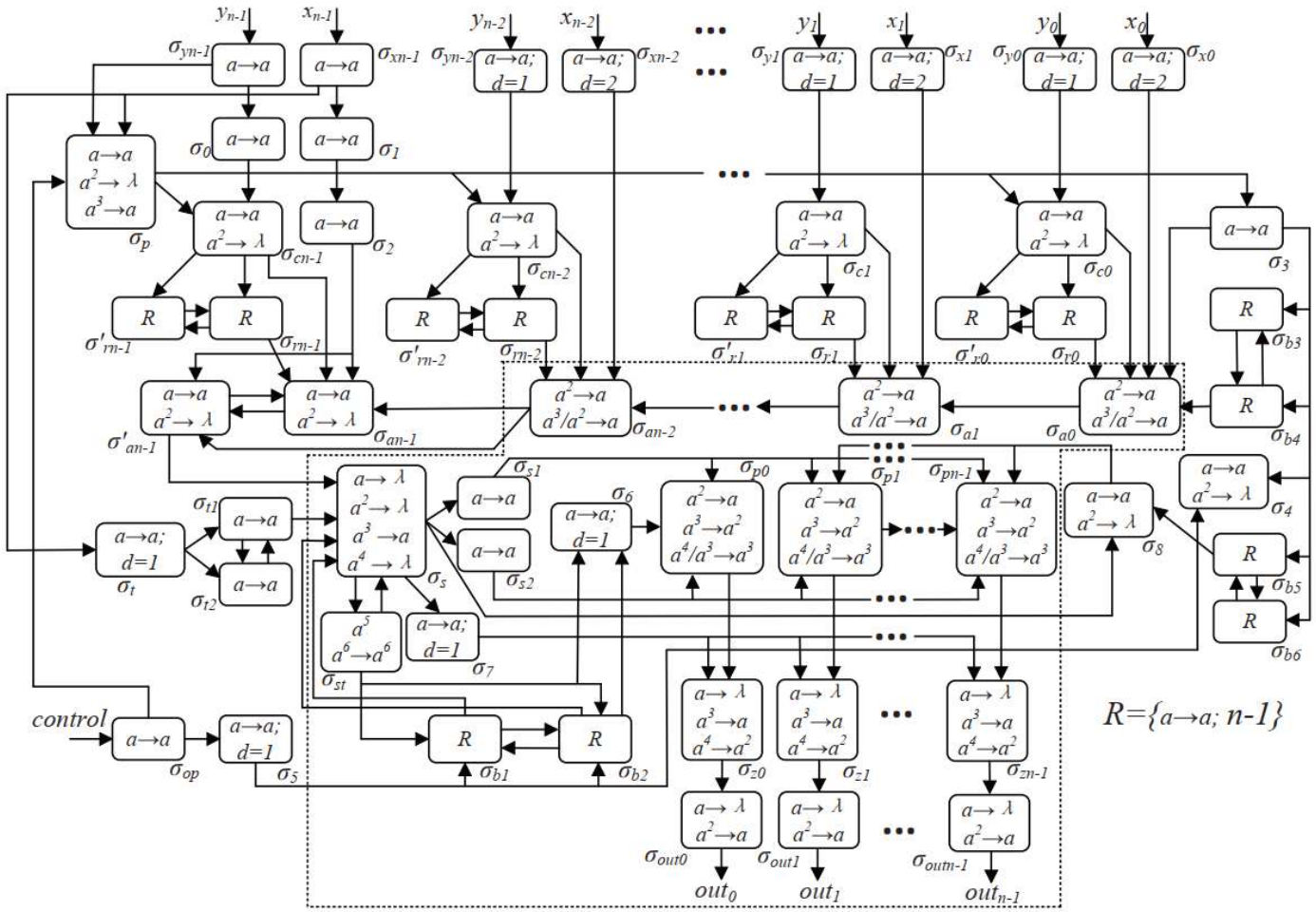






- Learning Delay Model is a synaptic learning algorithm proposed in this research project
- The synapse modifies its synaptic delay over time based on the actual time passed since it started receiving any input
- This enables the repeated multiplier to perform looped addition using the same network over time

Divider circuit



Application

- The previous paper uses this arithmetic calculator to perform information fusion based on Dempster-Shafer evidence theory (DST)

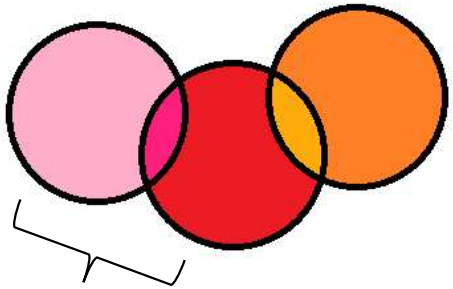
- The evidence theory says that the Basic Probability Assignment (BPA) of an event/evidence can be found using the BPAs of multiple internal sub-events/evidences

- Consider For example, that there is a probabilistic reliability factor assigned to each sensor in an AI automotive chip. Then the overall BPA can be calculated to give a more realistic picture of the environment. Using the above arithmetic calculator such things can be internally calculated in the Neural Processing Unit (NPU) and incorporated with the DNN/robot/control system.

Application : Example

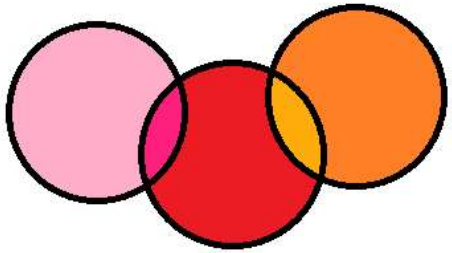
HSV Camera 1 – Lighting dark

M1(A)



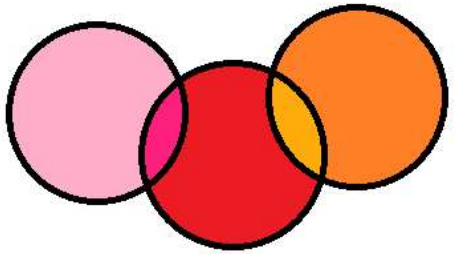
Hypothesis A – detected colour is either pink or red

HSV Camera – Inner Filters



M2(A)

HSV Camera 2 – Lighting Bright



M3(A)

- Use DS evidence theory to compute the BPA of the combined system

- A Possible output could be – “the combined system is saying colour could be Pink or Red with total reliability of 79%”

Action Items

- Motivation and Plausibility
  - Motivation [CPU Design, Complexity & H/W reduction, complete Neuromorphic solution, etc]
  - Plausibility ? [Beyond CMOS]

- Algorithm
  - Multiplicator [bit-independent H/W, Basic IF neurons]
  - Basic Logic units [NAND, NOR, XOR, MUX, De-serialiser, Encoder, repeater, etc]
  - Storage Unit [only 3 Neurons for 'n' bit storage]
  - Division Unit ?
  - Binary to frequency encoded spike train converter ?

- Implementation
  - Proof of Concept in Loihi
  - Full multiplicator Implementation ?

- Application
  - Basic Probability Assignment (reliability factor) calculation [Showed by the previous paper]
  - Shortest-path Navigation or another multiplicator application ?

Plausibility

Beyond CMOS

- Tunnel Field-Effect Transistor (TFET)
- Optical Transistor
- Tunnel Magnetoresistance (TMR)
- Quantum Computing (qubit)
- Spin-tronics (Quantum dot, Single Electron Transistor (SET))
- Vertical cavity surface emission Laser - neuron (VCSEL - neuron)

Plausibility

Single Electron Transistor (SET)

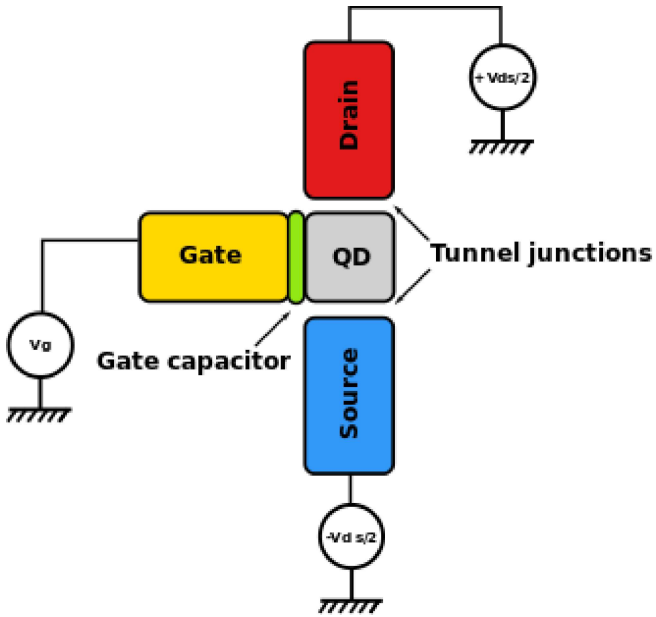
- Works on the principle of Quantum dot and Coulomb blockade

- **Quantum dot (QD):**
  - A minute piece of a semiconductor
  - Quantum mechanical effects in play
  - electrons tightly confined – like particle in a box

- When Photons excite the electrons to conduction band in a QD, it excites as per quantum energy levels

- When it de-excites from conduction to valence band, photon is released based on the quantum energy level difference

- SET has → a capacitor with a perfect insulator in between, two tunnel junctions and a QD



Single Electron Transistor

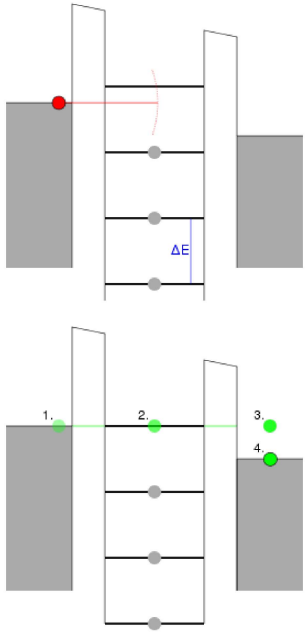
Plausibility

Single Electron Transistor (SET)

- **Coulomb blockade:** Conductance of very small devices decreases on very small bias voltages, so much so that is vanishes under a particular voltage

- Coulomb blockade occur in SETs as follows →
  - When $V_{bias} < {}^{e}/_{C}$ where C is self-capacitance of the QD
  - lower quantum levels occupied by electrons
  - Due to quantum principles, no. of electrons in a level is fixed
  - Source electron can't tunnel into the QD

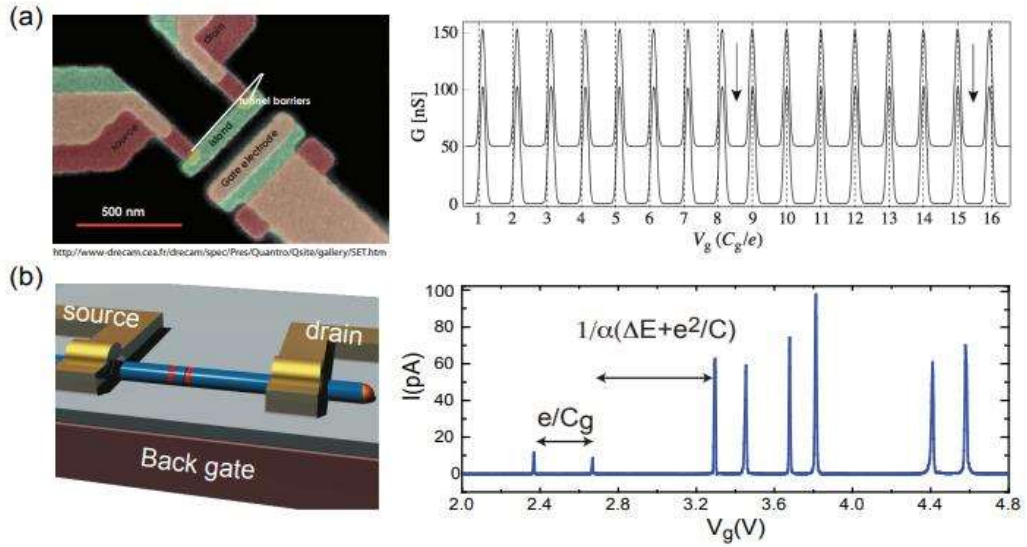
- Once V_{bias} is increased, the energy levels are lowered and electron can tunnel onto the QD and then onto the drain



Quantum Dot in a SET

Plausibility

Single Electron Transistor (SET)



Conductance/current waveform for a SET and nanowire QD

a) SET → energy levels are equally spaced hence peaks at equal intervals

b) Nanowire QD → the energy levels are dependent on external factors like a magnetic field and are irregularly placed

Plausibility

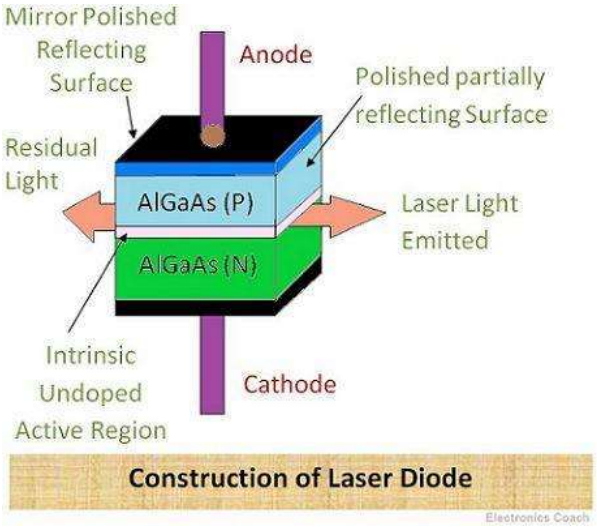
VCSEL Neuron – an optical device

- Vertical cavity surface emission laser

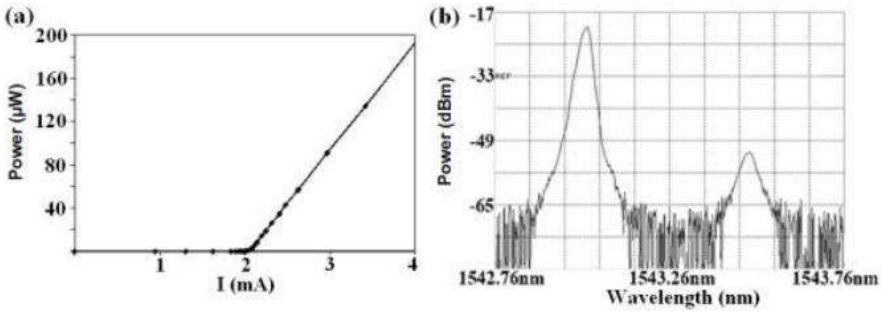
- Based on the concept of Laser Diode, injection locking and polarisation switching

- **Laser Diode:** releases a beam of light non-linearly based on bias voltage

- **Injection Locking:** The frequency of the output beam of light can be locked based on another reference (or seed) beam of a particular stable frequency



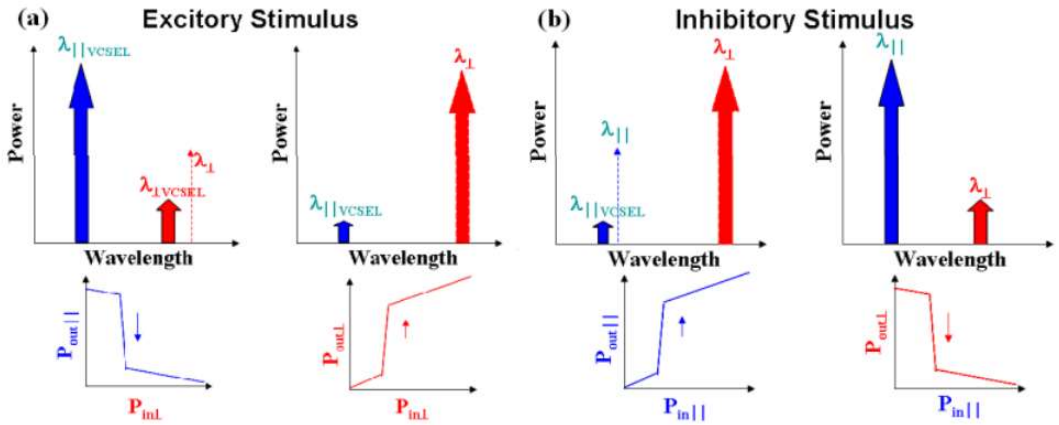
Laser Diode



VCSEL power spectrum and profile

Plausibility

VCSEL Neuron – an optical device



VCSEL neuron activity

- **Polarisation Switching (PS):** switching of the laser wavelength to the subsidiary wavelength non-linearly depending on the power of the injection beam

- This non-linear dynamics is used as the neuronal dynamics