2025 21st International Conference on Synthesis, Modeling, Analysis and Simulation Methods, and Applications to Circuits Design (SMACD) | 979-8-3315-2395-4/25/$31.00 ©2025 IEEE | DOI: 10.1109/SMACD65553.2025.11092008

A Practical PCB-based Framework for Spiking Neural Networks with a Half-Adder Example

Sevde Vuslat Çıkıkcı*, Eren Örek*, Ayhan Aysoy*, Ali Kağan Özgen†, Arda Yavuz†, Tuba Ayhan†
*Istanbul Technical University, Istanbul, Türkiye
{cikikcis24, orek18, aysoy24}@itu.edu.tr
†MEF University, Istanbul, Türkiye
{ozgena, yavuzar, ayhant}@mef.edu.tr

***Abstract*—This paper addresses the half-adder problem using Spiking Neural Networks (SNNs). In a previous study, the XOR operation was successfully realized on a breadboard and in this study it is integrated into the half-adder structure. The system uses input signals at frequencies of 50 Hz and 100 Hz and the neurons are generated by the Leaky Integrate and Fire (LIF) model. Unlike other neuron models, the LIF model is less complex. In addition, it was preferred because of its biological meaningfulness compared to the Integrate and Fire model. The network, consisting of 18 neurons in total, shows that basic arithmetic operations can be performed with SNN. Overall, this study demonstrates that basic logic operations can be implemented in neural networks, thus providing new perspectives for digital calculation. The successful solution of the Half Adder problem using SNNs not only proves the calculation capabilities of SNNs, but also opens new perspectives for the development of more complex logical circuits using these biologically inspired neural circuits.**

***Index Terms*—Spiking Neural Network (SNN), Leaky Integrate and Fire (LIF), Half-Adder, XOR, Circuit, Transistors, NMOS, PMOS**

I. Introduction

Today, Moore's Law has reached its limits because transistors cannot physically shrink any further [1]. Different architectures such as 3D transistors [2], NSFET [3], HPC [4] and SNN (Spiking Neural Networks) [1] are explored to improve processing performance.

In 3D transistors, lack of design for testability (DFT) and cooling issues pose significant challenges [3], [5]. NSFET's manufacturing difficulty and self-heating problem can negatively affect performance [6]. HPC systems face high energy consumption (30 MW) and frequent failures [4]. SNNs stand out in terms of energy efficiency with event-based computing. SNNs only operates when a threshold value is reached, preventing unnecessary energy consumption [7].

This study presents circuit modules that offer a practical framework for constructing SNNs, balancing simplicity, robustness, and scalability. The modules are implemented on PCBs using off-the-shelf components. Additionally, tunable components and an I/O structure ensure seamless communication between modules, enhancing the overall scalability of the framework. The accessible modular SNN framework was simulated on an XOR problem [8], using the topology from [9] as a guide. This work expands on the previous study by presenting the circuit implementation and introducing a new example: a half-adder system on SNN.

The paper first introduces the main circuit modules in Section II. Section III explains the half-adder topology and its experimental results are given in Section IV then the paper is concluded. For repeatability, the component values are given in the appendix.

II. Circuit Design

This system is designed to be implemented using six distinct circuits. This section covers the LIF, integrator, and monostable circuits, which are more crucial to the topology than the other circuits. In addition to these circuits, a summing amplifier performs the summing operation, an inverting operational amplifier implements the weights in the topology, and a band-pass and low-pass filter distinguishes the inputs in Hidden Layer 1.The low-pass filter is designed using a three-stage second-order low-pass Sallen & Key filter and a first-order Buffered RC.The Band-pass filter circuit consists of a buffer RC and a three-stage two-degree Sallen & Key high pass filter.

A. LIF Neuron Circuit

Figure 1a illustrates the electronic circuit of the neuron model used in the topology.The capacitor is a representation of the neuron membrane, and the R_1, R_2, and C components of the circuit control the neuron's leaky time moreover the refractory time is controlled by R_3 and C components. The Silicon Controlled Rectifier (SCR) component manages the switching function based on the time it remains in the on and off states [10].

Pairs of NMOS and PMOS transistors in the axon block losses in the network by strengthening the signal from the soma [10].

B. Integrator Circuit

The input at 100 Hz, which represents logic 1, is converted to a $3mV$ output, and the input at 50 Hz, which represents logic 0, is converted to a 2.5 mV output using an integrator circuit to ascertain the values of the system weights.The output current of the integrator circuit shown in Figure 1b is given by

$$I_{Integrator} = I_{LIF}(exp^{-t/\tau_r} + exp^{-t/\tau_d})/\alpha_{scale} \quad (1)$$

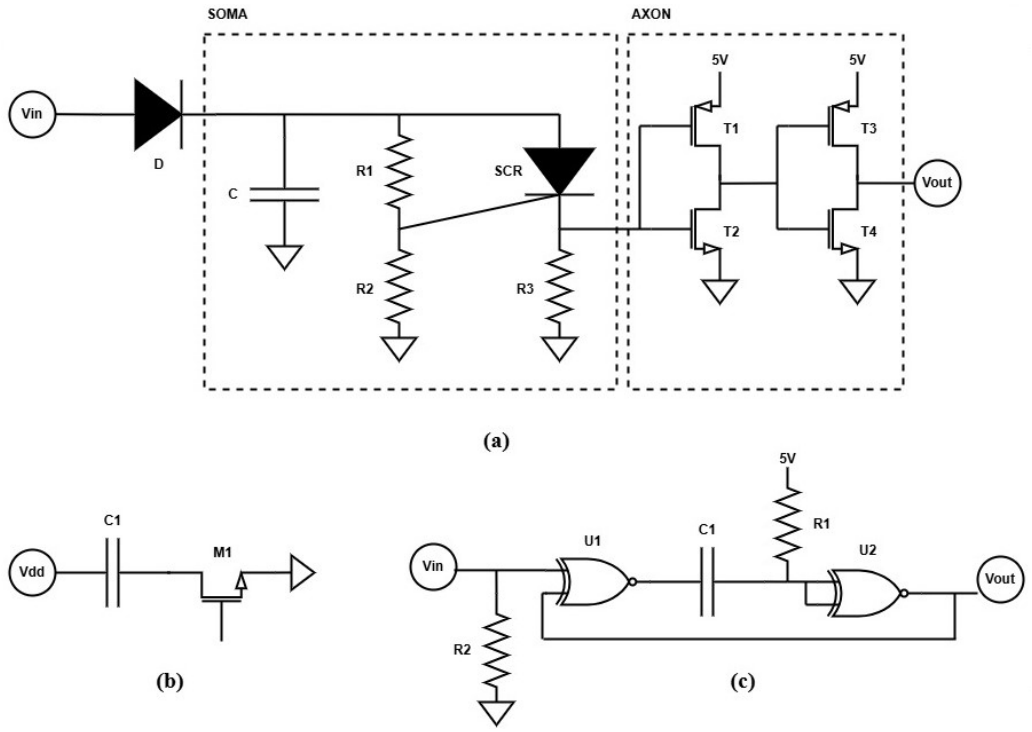


Fig. 1: Circuit schematics of topology (a) LIF Neuron (b) Integrator Circuit (c) Monostable Circuit

I_{LIF} controls the output current of the LIF analog circuit, and τ_d and τ_r control the rising and falling time constant of the integrator circuit current. The time constant of the circuit is dependent on the C_1 and gate voltage of the M_1. In addition,α_{scale} is a scaling factor of the peak value of current [11].

C. Monostable Circuit

As shown in Figure 1c, the monostable circuit remains at Logic 0 however when an external trigger signal exceeds the threshold value, the circuit temporarily switches its output to Logic 1 before returning to its default state.

The NOR gate plays a crucial role in maintaining the stability of the circuit, while the product of R_1 and C_1 determines the time constant, which defines the width of the output pulse.

In the topology, this circuit is used for two separate purposes. Filtering a signal with a small pulse width can be challenging. Therefore, the first step is to extend the pulse at the filter's input until the signal can be filtered. Regardless of the width of the monostable circuit's input pulse, the output signal's width is fixed at 3 ms. The input pulse widths applied to the neurons must be equal for the system to function effectively and accurately. Consequently, the second purpose is to transform the signal into spike form as it comes from the monostable circuit. For this purpose, a monostable circuit with an output width of 100 μs is utilized.

III. Half Adder Topology

The XOR topology used in the previous study is adapted to build a new system, thus, the solution of half adder is provided. The aim is to solve the half adder problem with 18 LIF neurons using the SNN system shown in Figure 2. The layers in this network are defined as Input Layer, Hidden Layers and Output Layer. In the Input Layer, logic 0 and logic 1, the frequencies of the input values are defined as 50Hz and 100Hz respectively.In the half-adder design, the output of SUM is dependent on the result of XOR, and the output of CARRY is dependent upon the result of AND. Figure 2 shows the topology and circuits of the systems designed for the half adder. In contrast to the previous work, the number of neurons in Hidden Layer 2 is increased to 8 and the number of neurons in Output Layer is increased to 4.

Neurons 1.1 and 1.2 are connected to the input layer's first neuron, while neurons 1.3 and 1.4 are connected to the second neuron. Depending on the frequency of the incoming signal, these neurons in the first layer decide which neurons are activated. Neurons 1.1 and 1.3 only fire at 100 Hz due to the connections to them using a band-pass filter that only enables frequencies up to 100 Hz. Only 50Hz input can stimulate neurons 1.2 and 1.4 the reason the connections to these neurons are made using a low-pass filter.

The second layer receives the signals that are separated by frequencies. In hidden layer two, neurons 2.1 and 2.4 generate the Class B output of the SUM result, while neurons 2.2 and 2.3 create the Class A output. Analogously, 2.5 produces CARRY Class A output, whereas 2.6, 2.7, and 2.8 enable the

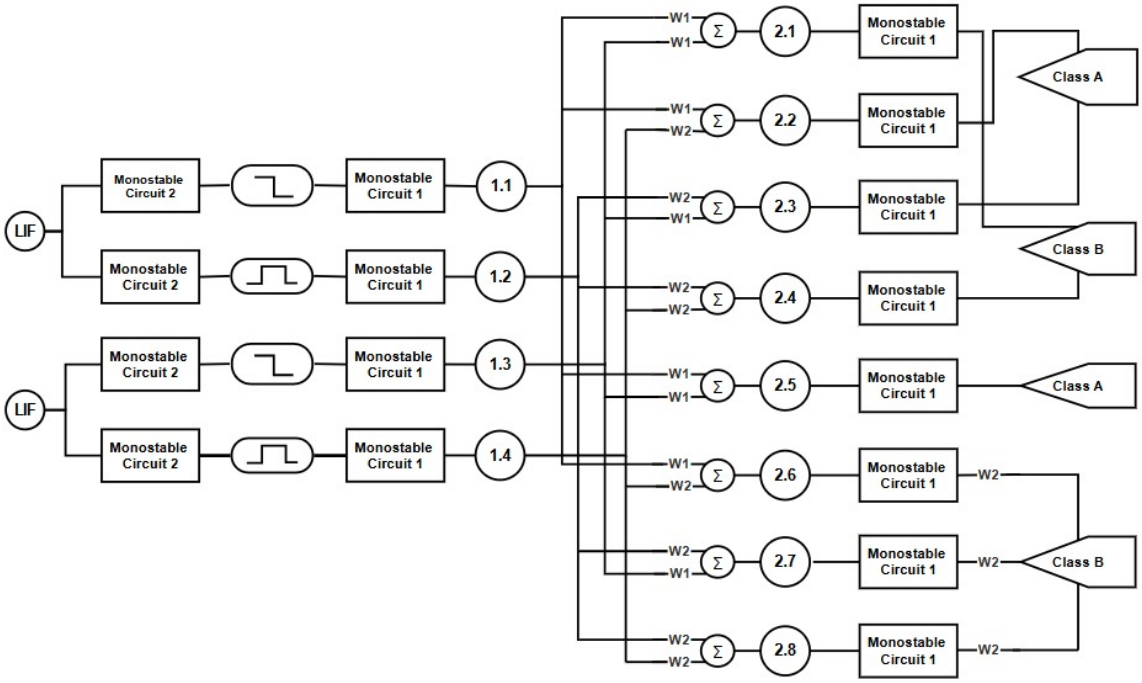


Fig. 2: Schematic Showing All Circuits of the System Used to Solve the Half-Adder Problem

firing of CARRY Class B. The corresponding Class A or B neurons of SUM and CARRY are fired after being transferred from Hidden layer 2 to the output layer.

IV. Experimental Results

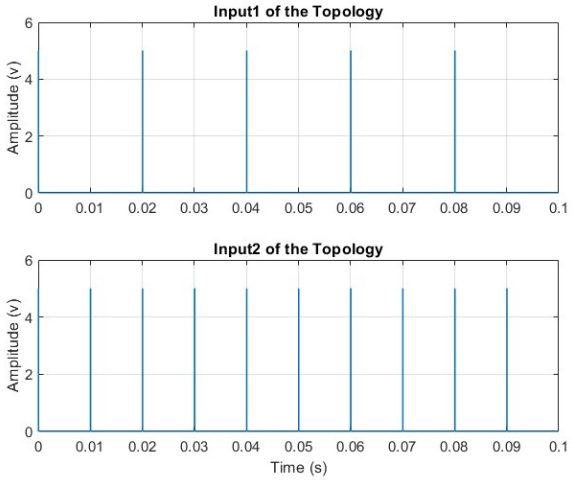


Fig. 3: Inputs applied to the input layer of the topology, representing logic 0 and logic 1 with distinct frequency values

As seen in figure 3, 50 Hz, logic 0, is applied to the first input of the system, and 100 Hz, logic 1, is applied to the second input. The purpose of this case study is to demonstrate how the system behaved under various input conditions. In addition, the input amplitudes are $5V$, and the spike widths are 100 μs.

The outputs in Figure 4 are the results of applying the inputs to the system depicted in Figure 1. The results obtained show that the SUM result of the problem is logic 1, which means that the CLASS A neuron fires and the CLASS B neuron does not fire. Moreover, the CARRY result is logic 0, which means that the CLASS B neuron fires and the CLASS A neuron does not fire.

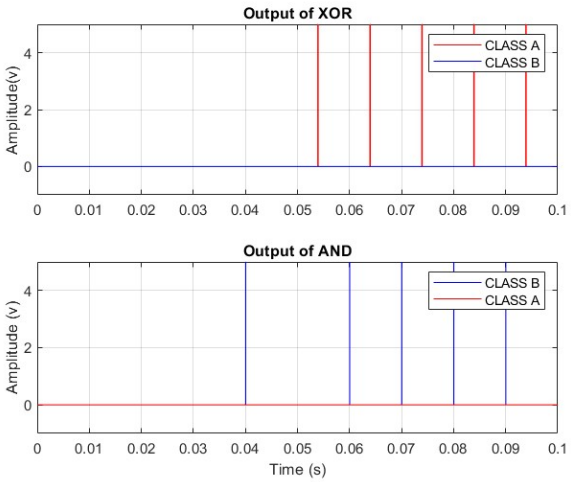
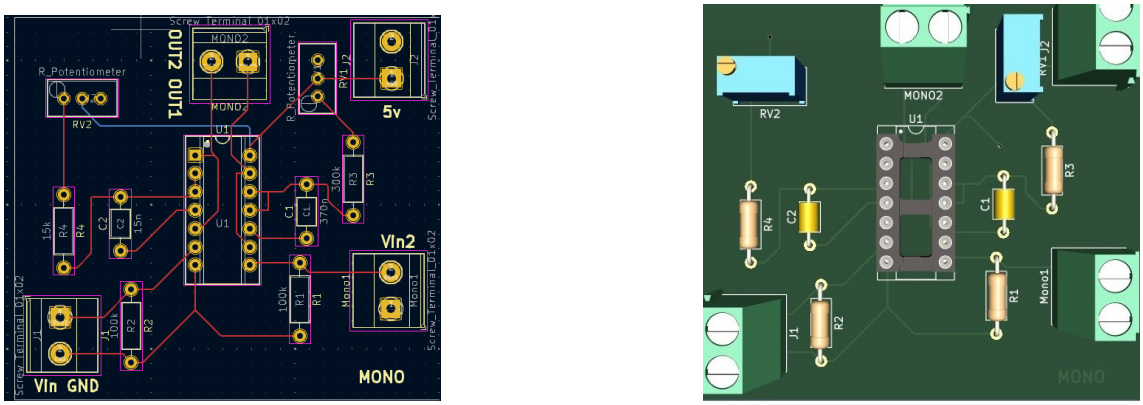


Fig. 4: The result of Topology output showing SUM and CARRY

The PCB design of the Monostable Circuit, one of the circuits run in the simulation, is shown schematically and in 3D in Figure 5. The PCBs of the designed circuits were tested in the laboratory environment and PCB data were taken.



(a) PCB schematic

(b) 3D view

Fig. 5: Monostable circuit designed for the Topology

V. Conclusions

This work demonstrates the implementation of a modular, cost-effective SNN framework using PCB-based circuit modules. The concept was previously simulated on an XOR problem. In this study, the circuit modules of an SNN (including LIF neurons and monostable circuits for firing, as well as parametrized connection elements filters, integrators, and amplifiers) were implemented. The system was then rearranged and tested as a half-adder. The results show that the proposed framework can support various systems by either adjusting the number of circuits or rearranging those used in the earlier study, as demonstrated through the implementation of another problem.

For future work, the software side of the framework will be developed to model and train SNNs targeting the presented hardware modules. This will make implementing and testing more complex problems on SNNs easier, potentially accelerating research in the field.

Acknowledgment

This project is supported by The Scientific and Technological Research Council of Türkiye (TÜBİTAK).

References

[1] S. Kumar, "Fundamental limits to moore's law," Nov 2015.

[2] S. Valasa, S. Tayal, L. R. Thoutam, J. Ajayan, and S. Bhattacharya, "A critical review on performance, reliability, and fabrication challenges in nanosheet fet for future analog/digital ic applications," *Micro and Nanostructures*, vol. 170, p. 207374, Oct 2022.

[3] H.-H. S. Lee and K. Chakrabarty, "Test challenges for 3d integrated circuits," *IEEE Design Test of Computers*, vol. 26, no. 5, pp. 26–35, 2009.

[4] P. O. A. Navaux, A. F. Lorenzon, and M. d. S. Serpa, "Challenges in high-performance computing," *Journal of the Brazilian Computer Society*, vol. 29, p. 51–62, Aug 2023.

[5] V. Kumar and A. Naeemi, "An overview of 3d integrated circuits," in *2017 IEEE MTT-S International Conference on Numerical Electromagnetic and Multiphysics Modeling and Optimization for RF, Microwave, and Terahertz Applications (NEMO)*, pp. 311–313, 2017.

[6] S. Rathore, R. Kumar Jaisawal, P. Kondekar, and N. Bagga, "Trap and self-heating effect based reliability analysis to reveal early aging effect in nanosheet fet," *Solid-State Electronics*, vol. 200, p. 108546, Feb 2023.

[7] M. Pfeiffer and T. Pfeil, "Deep learning with spiking neurons: Opportunities and challenges," *Frontiers in Neuroscience*, vol. 12, Oct 2018.

[8] S. V. Çıkıkcı, E. Örek, A. K. Özgen, A. Yavuz, and T. Ayhan, "Solving xor in spike neural network (snn) with component-off-the-shelf," in *2024 15th National Conference on Electrical and Electronics Engineering (ELECO)*, pp. 1–5, 2024.

[9] J. J. Wade, L. J. McDaid, J. A. Santos, and H. M. Sayers, "A biologically inspired training algorithm for spiking neural networks," *Irish Systems and Signals Conference (ISSC)*, Sep 2007.

[10] M. J. Rozenberg, O. Schneegans, and P. Stoliar, "An ultra-compact leaky-integrate-and-fire model for building spiking neural networks," *Scientific Reports*, vol. 9, Jul 2019.

[11] A. Gautam and T. Kohno, "An adaptive stdp learning rule for neuromorphic systems," *Frontiers in Neuroscience*, vol. 15, Sep 2021.

Appendix

In the circuit in 1a, D: 1N4148, SCR: CR02, C: 15nF, R1: $68k\Omega$, R2: $390k\Omega$, R3: $1.5k\Omega$, and CD4007 for CMOS are used. In 1b, C1: 47μF and M1: BS170 NMOS are used. To increase pulse width in 1c, R1: $300k\Omega$, R2: $100k\Omega$, C1: 370nF, and 74HCT02 NOR gates are used. For equalizing neuron input signals, R1: $15k\Omega$, R2: $100k\Omega$, C1: 15nF, and 74HCT02 NOR gates are used in the monostable circuit. The inverting operational amplifier circuit sets weight values: for w1, R1: $1k\Omega$, R2: $209k\Omega$; for w2, R1: $1k\Omega$, R2: $170k\Omega$. The summing circuit uses $1k\Omega$ resistors. LMC662 is used as the OPAMP model in all circuits.