ISSN: 2096-3246
Volume 56, Issue 03, Aug, 2024

A Reversible-Logic Based Architecture For Artificial Neural Network

[1] **Bhoopal Rao Gangadari ,** [2] **Badavathu Rajender**,

[1] Associate Professor , Department of ECE, Nalla Malla Reddy Engineering College, Hyderabad -500088

[2] M.Tech Student, Department of ECE, Nalla Malla Reddy Engineering College, Hyderabad-500088

ABSTRACT

Complex 2.5D/3D SOC design architectures with high-performance computing beyond sub-10 nm advanced node technology may be investigated and implemented. Thanks to node scalability, heterogeneous integration, and advanced design, we can think beyond the bounds of Moore's Law. limit the reach of the legislation and account for excessive power loss In the context of nanotechnology and low-power VLSI circuit designs, the science of quantum computation has seen extensive study of reversible logic functions. Digital circuits that are capable of reversible computation have significantly lower power consumption. In this work, we propose a ground-breaking new layout. Making use of logic gates that may be used in reverse, an ANN is developed. A After searching the relevant literature extensively, just a few pieces with similar content were located. To the best of our abilities, our proposed method is correct.

ISSN: 2096-3246
Volume 56, Issue 03, Aug, 2024

INTRODUCTION

As shown by Moore's law's exponential progression [1], node-technology scaling has already surpassed the sub-10 nm threshold. Each CPU (Central Processing Unit) can have several billion transistors integrated into its 3D SoC (System-On-Chip) architecture. recently, in the last few years. benefits and drawbacks of high performance Optimizations may be made to the circuit's complexity, cost, and computational power. However, there is now an issue with power consumption. significant barrier to high-performance computing, as described by ITRS (International Technology Research Symposium). Semiconductor Industry Technology Roadmap for 2008 weak force There are numerous points in the circuit design process when VLSI design might be useful. using a wide variety of standard methods. This essay explored the method via the lens of quantum computing and reversible logic [2].

Consider a standard AND gate as an example of an irreversible logic gate. Any bit in the input that is a logic '0' will produce an output bit that is also a logic '0'. only if both inputs have a logic 1. Recently, by Considering only the logic '0' state of the output might make it hard to ascertain which logic '0' input bit signal is needed. Counting the result as a logical "0" (0-0, 0-1, 1-0) is the state. Any of the possible two q states can hold q bits of signal information. According to the Landauer principle, for every piece of data that is lost, another KT ln2 needs to be created to account for it. Thermal energy is expressed as a power dissipation in (Joules), with K = 1.38 1023 (J/K), the Boltzmann constant.

Such power reduction is remarkable in modern node technology. C.H. Bennett calls the shift from conventional irreversible digital gates to "non-reversible gates" a "paradigm shift." Thanks to reversible logic gates, we are able to avoid this significant power consumption. a loss due to dissipation. A lossless reversible logic process is one that can translate every input signal to all possible output signals in the same way, in reverse order [2, 5]. This The study recommends a novel neural network hardware design. One of the many applications for a combination of a reversible logic circuit with a neural network is in the development of smart tools for classifying images [6].

LITERATURE SURVEY

Irreversibility And Heat Generation In The Computing Process

Abstract: It has been argued that computers always use parts that perform logical operations that don't have a unique inverse value. This logical irreversibility necessitates the production of some heat throughout each cycle of the machine, often on the order of kilocalories. Using dissipation, signals may be standardized and made to function independently of their particular logical history. To better understand the relationship between switching speed and energy dissipation and to quantify the impact of errors caused by thermalfluctuations, switching kinetics are studied in greater depth for two simple but typical models of bistable devices.

Logical Reversibility Of Computation:

Abstract: The typical general-purpose computer automaton (such a Turing machine) is potentially irreversible since its transition function does not have a single-valued inverse. It is shown here that such machines may be kept very straightforward while yet being able to do a wide variety of computations in a logically reversible fashion. This finding is of great scientific significance because it opens the door to the possibility of thermodynamically reversible computers capable of doing useful computations at practical speeds while requiring significantly less energy per logical step than kT. The logically reversible automaton conducts the same beginning computation as its counterpart irreversible automata, but without deleting any intermediate results. The second phase involves printingthe final product. The third and final stage made a U-turn

ISSN: 2096-3246
Volume 56, Issue 03, Aug, 2024

A Novel Design Gate Based Low-Cost Configurable Ro Puf Using Reversible Logic

Abstract:A A physical unclonable function (PUF) is a mapping function that cannot be replicated, and it can be either one-to-one or many-to-many. The input to a PUF function is called a Challenge, and the output is a response (or replies). The unique CRP behavior allows for the mathematical characterization of every PUF. Silicon PUFs are most useful for protecting intellectual property in hardware. On the other hand, we are entering the era of quantum computation, when reversible digital logics may gradually replace the current standard. During each digital logic operation, conventional circuits must discard bits of input data (fan-in) at the output end (fan-out), wasting significant amounts of power in the process.

Embryonic Hardware Self-Repair Facilitated by Intelligent Fault Prediction

AbstractTo make bio-inspired hardware resistant against defects in their early stages, this study proposes novel approaches to self- healing, fault prediction, and fault-prediction assisted self-healing. The proposed self-healing method repairs a dysfunctional embryonic cell by repurposing components from other cells. Experiments have shown that self-healing is effective, but that hardware has a considerable recovery period after experiencing an unanticipated malfunction. To combat this lag, we provide novel deep-learning- based formulations for fault prediction. The suggested fault prediction methods and the proposed self-healing approach are then used to assess the efficiency and timeliness of hardware still in the development stage. The proposed fault prediction and self-healing methodshave been implemented using VHDL over FPGA. Very precise fault forecasts are suggested.

RELATED WORK

Overview Of Artificial Neural Network And Reversible Logic:

We've gone over the basics of the reversible logic paradigm and gone through the evaluation indices for a reversible logic circuit as trash outputs and quantum cost. hardware delay, input consistency, and complexity indication.

Reversible function:

It is possible to provide any reversible functionality by combining many reversible logic gates. In Fig. 1, a generalized block of reversible logic is represented by the N-dimensional input vector I = (I1, I2,, IN) and the N-dimensional output vector O = (F1, F2,

, FN). The

most basic realization of a reversible logic is a one-to-one mapping between any N-bit inputs and the corresponding N-bit outputs. Sincethere is no permanent loss of data bits during a reversible operation, Logic operations and reversal abilities, if possible, would be helpful.There is no loss of energy.

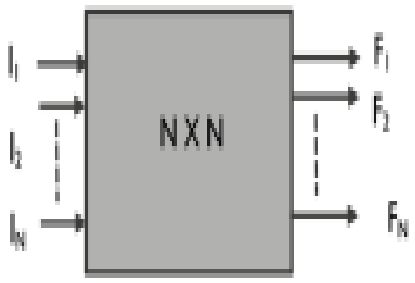


Fig. 1: Generalized (N × N) reversible logic functionality

ISSN: 2096-3246
Volume 56, Issue 03, Aug, 2024

Quantum cost:

The "quantum" cost of constructing a reversible circuit is equal to the sum of the numbers of its simplest reversible gates, either one (1) or two (2). Basic logic gates include SWAP (22), ControlledV (23), Controlled- V+ (14), NOT (11), and CNOT (23). So, for instance, if X can be utilized to construct reversible XOR functionality, we have an example of this. The functionality's quantum cost will be given by: Y Controlled-V/V+ gates and CNOT gates.

Constant Inputs:

Using these "trash" outputs, which are essentially additional bits, it is possible to achieve the "reversibility property" of any reversible circuit. However, these pieces of output are never used. importantly, cannot be undone for a given instance of reversible functioning. have to go. In Fig. 2, we have an illustration of the concept of "junk" outputs.

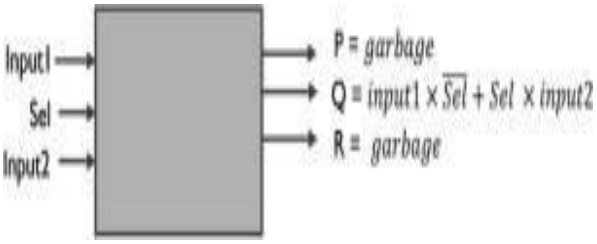


Fig. 2: Garbage output representation.

Delay:

The delay of a matching circuit may be calculated by counting the number of logic gates that can be utilized in both directions betweenthe circuit's input and output terminals.

Hardware Complexity:

The quantity of logical operations performed by a reversible circuit is one measure of its complexity. To find out the whole logical operation, utilize #NOT gates in conjunction with #AND and #XOR gates. Equation [7] may be rewritten in a more generic form as T

= + + +, where = #NOT gates, where = #(2 input)AND gates, and where = #(2 input)XOR gates. Any reversible logic must satisfy thefollowing requirements: features such as:

(a) Minimum reversible gate count

(b) And output trash count

(c) Minimal constant inputs

(d) A brief delay

Basic Reversible gates:

We have covered in depth the functioning of the four basic reversible gates known as Feynman gates, Fredkin gates, Toffoli gates, and Peres gates [2, 7, 8]. Table. This page details many of the available indexes. for these entryways. The Feynman gate seen in Fig. 3c is a(2 2) kind. P and Q are the output and input vectors, respectively. Output functionality is defined by P = A and Q = A B. The Feynmangate can also be used as CNOT when A = 1. B = 1 if Q = 0 and B = 0 otherwise. results produced By knowing the states of

ISSN: 2096-3246
Volume 56, Issue 03, Aug, 2024

the output P, we may accurately predict the values of B and Q, where A is the input that affects Q. In Fig. 3, we see an example of a (3 + 3) Toffoligate.

TABLE I: Basic reversible logic gate index

Reversible Gate	Quantum Cost [QC]	Logical calculation [T]
NOT Gate	0	1α
Controlled-V	1	–
Controlled-V+	1	–
CNOT gate	1	1δ
Feynman Gate	1	1δ
Fredkin Gate	5	$2\delta + 4\beta + 1\alpha$
Toffoli Gate	5	$1\delta + 1\beta$
Peres Gate	4	$2\delta + 1\beta$

Artificial Neural Network:

ANN attempts to mimic the computational and communicative abilities of the human brain [9]. Over the last decade [10], several hardware- and software-based algorithms have been proposed and developed for use in areas such as computer vision, identification of patterns, the processing of natural languages, adversarial learning of machines, and defect prediction. Ideas for machine learning, neural network modeling approaches, and the process of learning Algorithms are described in [11]. Analog, digital, and what are often called mix-signal circuits have all been used in various hardware ANN implementations [12, 13]. The fact that sigmoid brains use a digital method, with weights ranging from 0 to 1, presents the most significant challenge for numbers between 9 and 14. A prototype artificial neural processor using 2.5-CMOS technology was developed by Masa, Peter, and colleagues [14]. It can perform 61020 calculations per second on this CPU.

1. scalar inputs multiplied by a scalar weight

matrixThe sum of the scalar inputs after

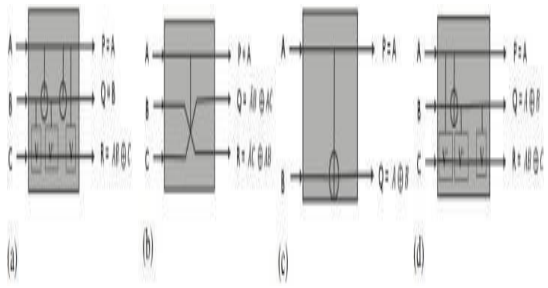


Fig. 3: Block diagram of (3 × 3) for (a) Toffoli gate (b) Fredkin gate (c) Feynman gate (d) Press gate.

ISSN: 2096-3246
Volume 56, Issue 03, Aug, 2024

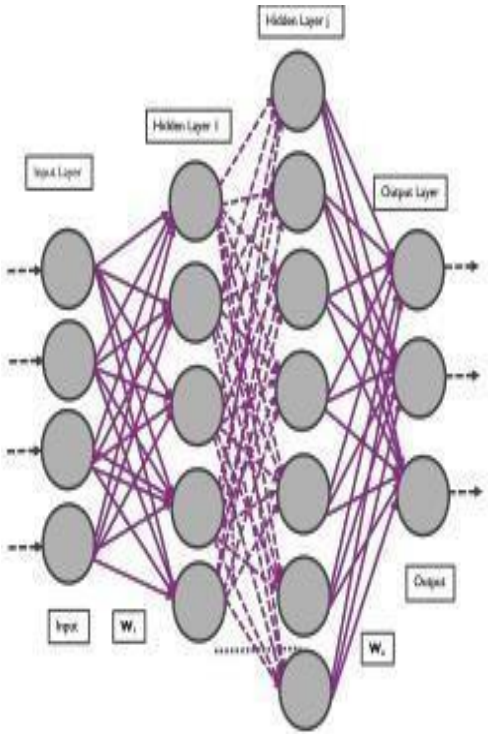


Fig. 4: Neural network architecture.

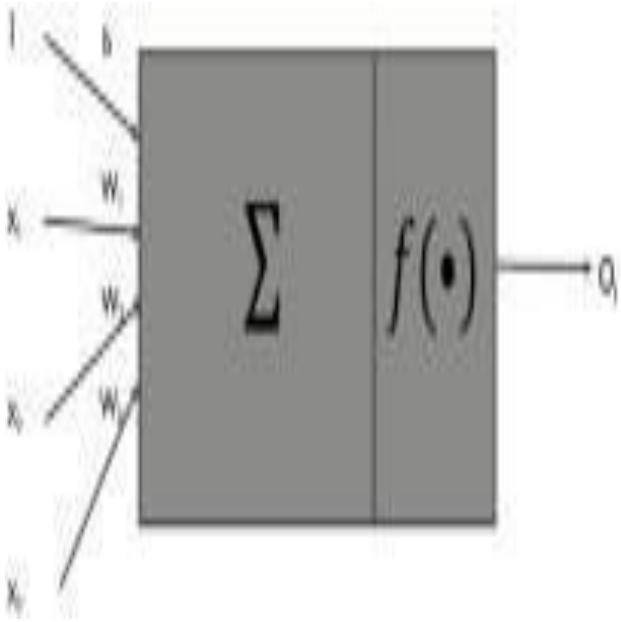


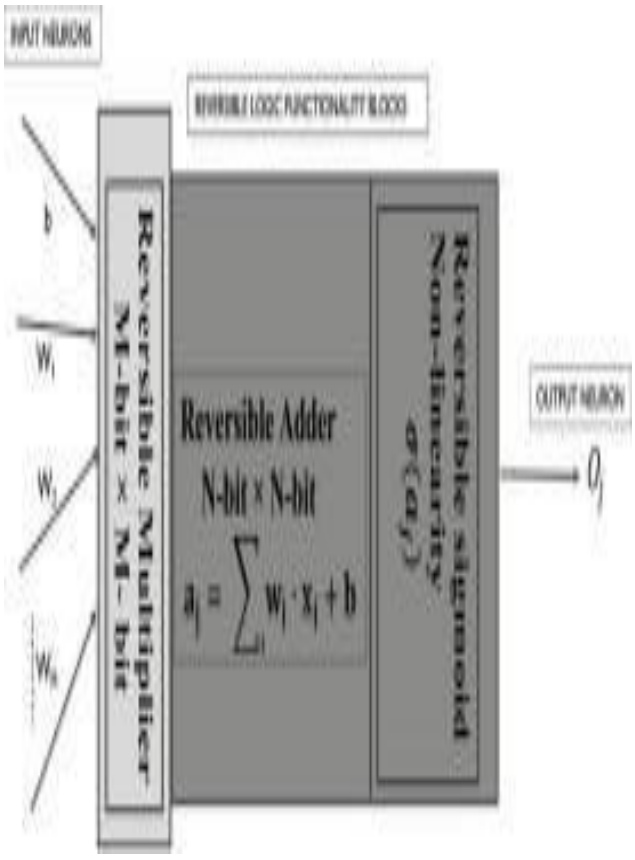
Fig. 5: Single artificial neuron: Symbolic hardware circuit realization.

ISSN: 2096-3246
Volume 56, Issue 03, Aug, 2024

PROPOSED SYSTEM

Figure 6 depicts the whole network, whereas Figure 7 shows the suggested node layout. The proposed architecture requires a reversible multiplier and then an A reverse adder. We have also developed a Sigmoid that can be inverted. using the same logic as its analogous digital form [23, To the best of our understanding, our An first attempt at implementing nonlinear sigmoid using just reversible logic gates. In Figs. 8 and 9, we can see the results. Logic-based adder [21] and multiplier [18] that can work in reverse. In Fig. 10, we see some of the data that supports the reversible sigmoid model: A reversible multiplier, consisting of 8 reversible adder circuits and 3 Peres gates, has been constructed using a reversible full-adder configuration with a combination of five Fredkin gates.

Fig. 6: The proposed reversible architecture of Neuron



.

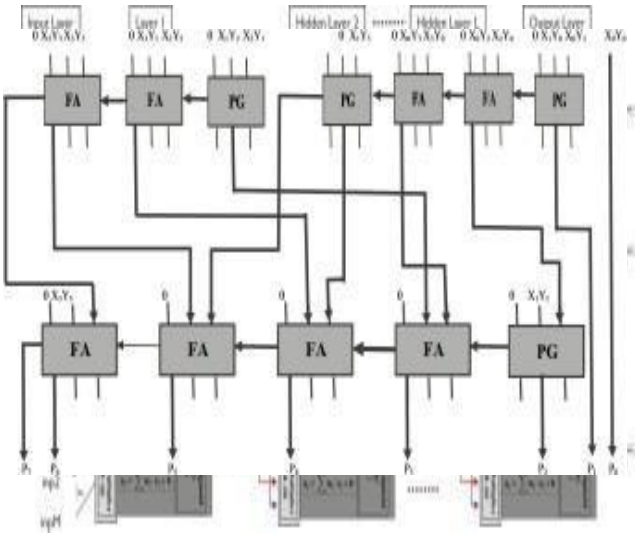


Fig. 7: The proposed architecture of neural network

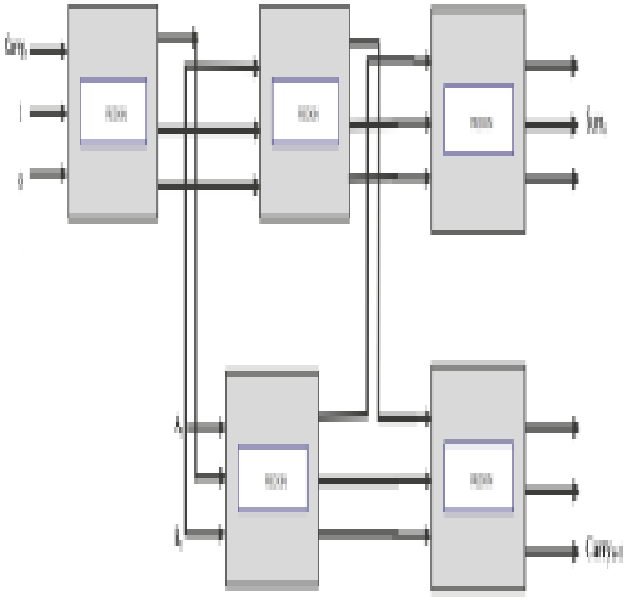


Fig. 8: Block diagram of reversible 1-bit Adder using Fredkin gate.

Fig. 9: Block diagram of reversible 4-bit multiplier using Peres gate and reversible FA.

The integer part must be complemented in accordance with the signbit. If the sign-bit is "1," just the integer part is complemented; otherwise, nothing changes. The next step is to add a "1" to the integer part. Locate the integer bit positions where there is a using the resultant's bit position. logic '1'. Bit positions n where the logic "1" occurs will undergo a left-shift operation of 2n times. If |Input| 3.875

= 10.1102, for instance. The sigmoid function of this input is close to 0.9375. Following the steps above, our final fixed-point

ISSN: 2096-3246
Volume 56, Issue 03, Aug, 2024

representation will be 0.11112 =. Our target sigmoid nonlinearity value is about similar to 0.937510, which is equal to (0.5 + 0.25 +

0.125 + 0.0625)10.

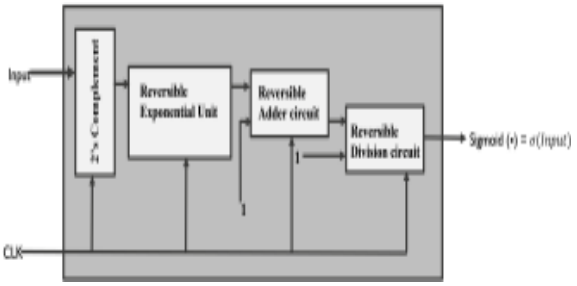
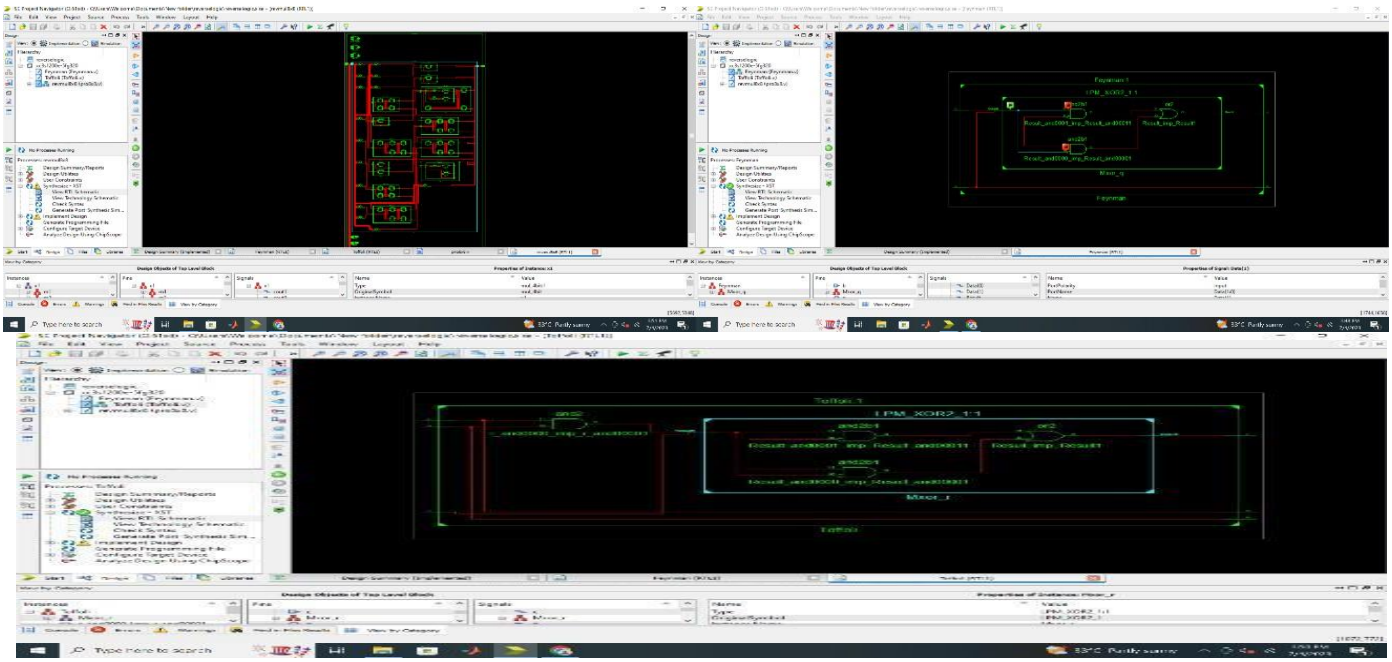
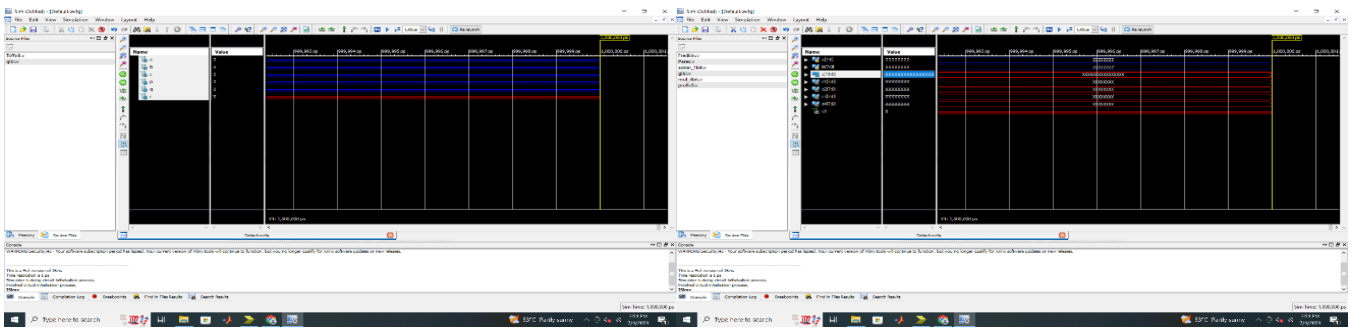


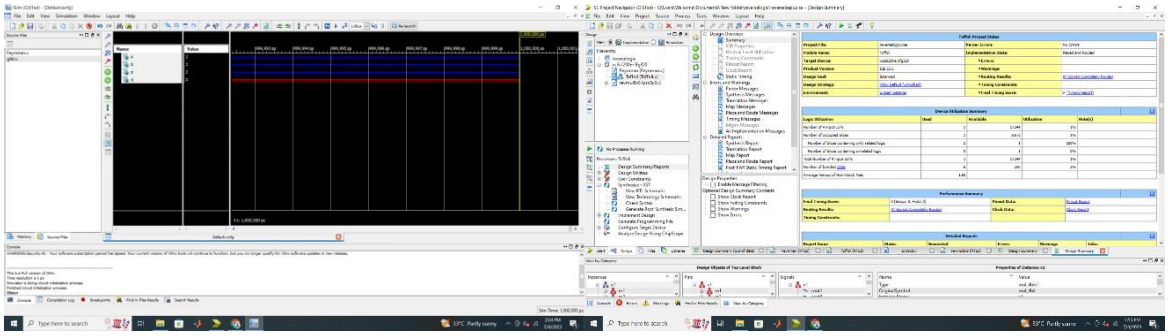
Fig. 10: Block diagram of reversible sigmoid function

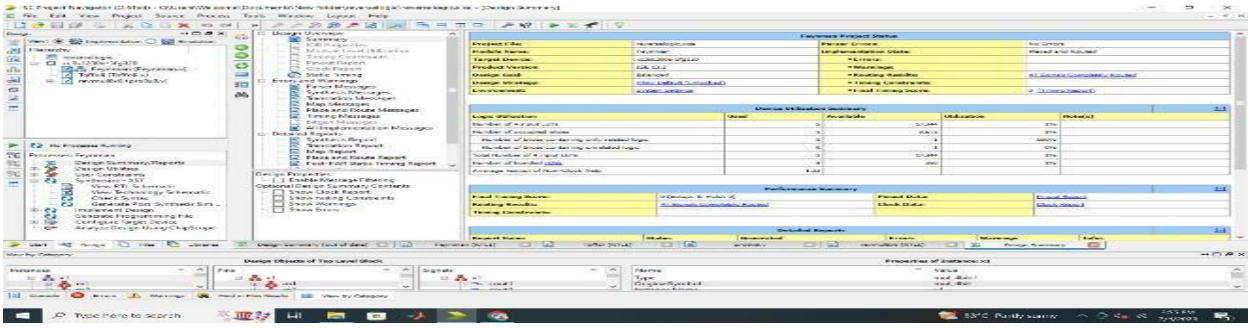
RESULT

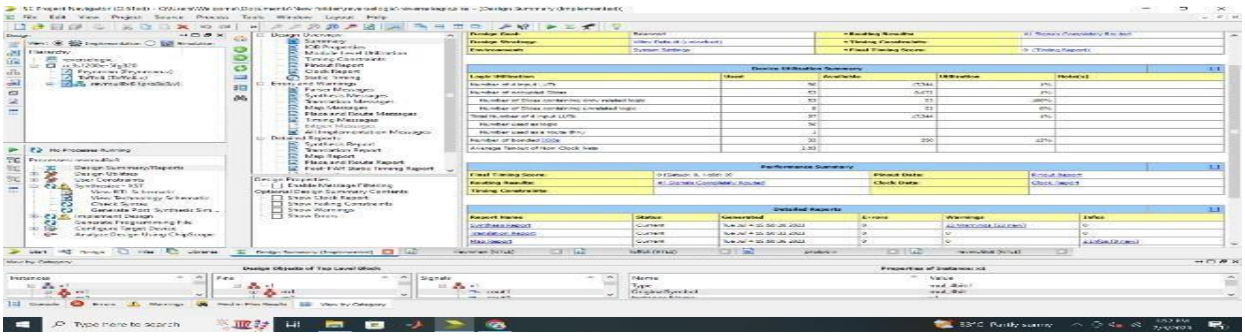


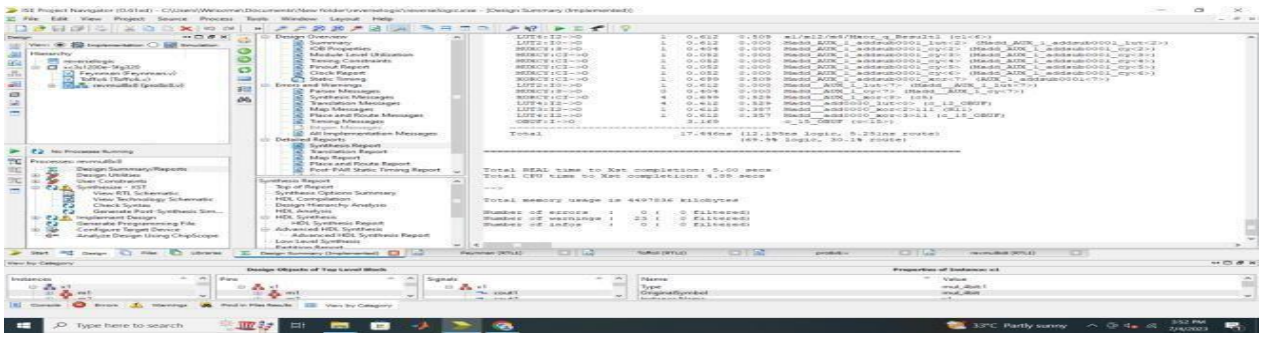


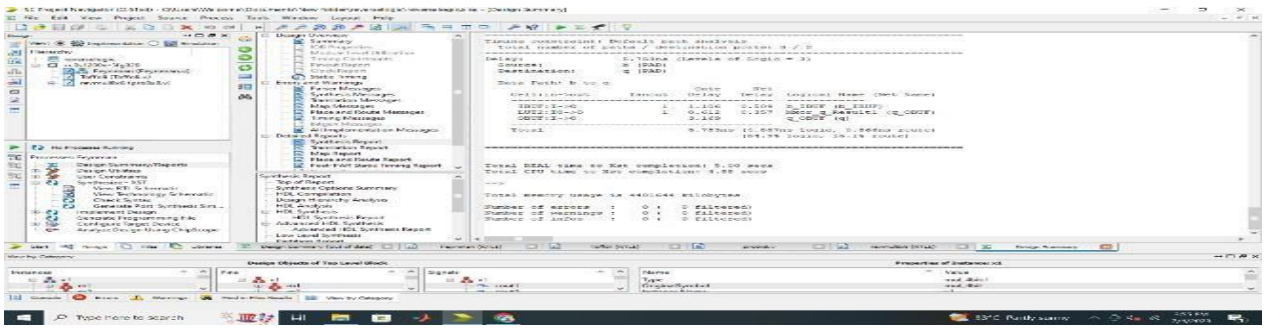
ISSN: 2096-3246
Volume 56, Issue 03, Aug, 2024



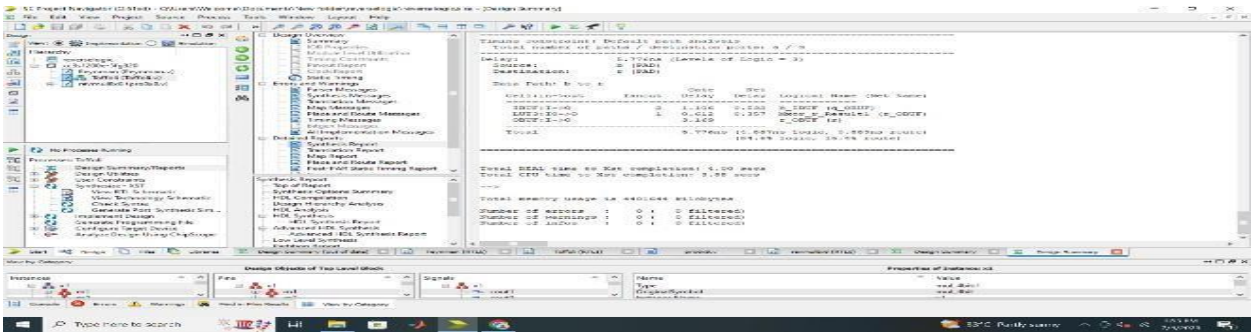








ISSN: 2096-3246
Volume 56, Issue 03, Aug, 2024



Implementation And Experimental Result:

The proposed approach is developed using VHDL on an Altera 10 GX FPGA. Table II shows the functional implementation of any artificial neuron in the ANN array, and Table III shows the synthesis result. There are 32 bits in total in this data. The proposed approach uses 0.78. W of power, while the conventional [9] method requires 0.94 W. We were able to cut down by roughly 16%. In general, less energy is lost as heat than in conventional processes. In To compare the energy lost by conventional logic design with reversible logic, we may use a metric called the Energy Saving Factor (ESF). They supply it because:

The efficiency index for this energy-saving measure is thus 1.2. Our approach in this paper relied only on a fully reversible logic-based hardware construction of a synthetic neuron. As a result, we believe there is further room for optimizing hardware outputs, complexity, and quantum cost in order to cut down on waste and improve design. Because of developments in the proposed circuit to

counteract the increase in area-overhead and delay, the situation may now be reversed. Therefore, we may claim that paradigms used to create high-performance computers that are impacted by the challenges humans have with power dissipation are restrictive. The energyefficiency (Flops/Watt) of a complex system might benefit from our proposed approach because of its greater scalability. The proposed method is useful for low power, high performance processing devices. We also performed a postlayout simulation to determine the effects on time, space, and energy consumption.

TABLE II: Summary of artificial neuron functionality

Neuron j		
Input from low-level neuron		Scalar (X_i)
Operation	- Weight multiplication - Sum	$a_j = \sum_i W_i.X_i + b$
	Nonlinear activation function	$O_j = f(a_j) = sigmoid(a_j)$
Output		Scalar (O_j)

ISSN: 2096-3246
Volume 56, Issue 03, Aug, 2024

TABLE II: Summary of artificial neuron functionality

Metrics	Using Reversible logic	Using Classic Logic Gate
Mechanism	ANN	ANN
Power	0.78 W	0.94 W
Slice LUTs	3854 of 53200	3789 out of 53200
Slice Registers	2342 of 105306	2315 of 105306
Frequency	120MHZ	120MHZ
Power	0.79 W	0.96 W
Delay	35 ms	26 ms

CONCLUSION

In this piece, we propose an ANN circuit that can function in reverse. The proposed layout was put up against several tried-and-true alternatives. The proposed method outperforms conventional solutions in terms of both power usage and heat dissipation. Power consumption has been cut by around 16 percent compared to the previous design. from the state of the art. VHDL on an Altera 10 GX FPGA is utilized to implement the proposed technique for the ANN array. In our next study, we will develop a sophisticated circuit fora convolutional neural network that makes use of both long-term and short-term memories and reversible logic.

REFERENCE

[1] G. E. Moore et al., "Cramming more components onto integrated circuits," 1965.

[2] R. Garipelly, P. M. Kiran, and A. S. Kumar, "A review on reversible logic gates and their implementation," International Journal ofEmerging Technology and Advanced Engineering, vol. 3, no. 3, pp. 417–423, 2013.

[3] R. Landauer, "Irreversibility and heat generation in the computing process," IBM journal of research and development, vol. 5, no.

3, pp. 183–191, 1961.

[4] C. H. Bennett, "Logical reversibility of computation," IBM journal of Research and Development, vol. 17, no. 6, pp. 525–532, 1973.

[5] B. Dey, K. Khalil, A. Kumar, and M. Bayoumi, "A novel design gate based low-cost configurable ro puf using reversible logic," in2019 IEEE 62nd International Midwest Symposium on Circuits and Systems (MWSCAS). IEEE, 2019, pp. 211–214.

[6] K. Khalil, O. Eldash, A. Kumar, and M. Bayoumi, "Intelligent fault-prediction assisted self-healing for embryonic hardware," IEEETransactions on Biomedical Circuits and Systems, 2020.

[7] M. Surekha, "Efficient approaches for designing quantum costs of various reversible gates," International Journal of Engineering,vol. 9, no. 1, pp. 57–78, 2017.

[8] P. R. Yelekar, S. S. Chiwande et al., "Introduction to reversible logic gates & its application," in 2nd National Conference onInformation and Communication Technology, 2011, pp. 5–9.

[9] M. Forssell, "Hardware implementation of artificial neural networks," 2014.

[10] K. Khalil, O. Eldash, A. Kumar, and M. Bayoumi, "Machine learning-based approach for hardware faults prediction," IEEETransactions on Circuits and Systems I: Regular Papers, 2020.

[11] I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning. The MIT Press, 2016.

ISSN: 2096-3246
Volume 56, Issue 03, Aug, 2024

[12] K. Khalil, O. Eldash, A. Kumar, and M. Bayoumi, “An efficient approach for neural network architecture,” in 2018 25th IEEEInternational Conference on Electronics, Circuits and Systems (ICECS). IEEE, 2018, pp. 745–748.

[13] K. Khalil, O. Eldash, B. Dey, A. Kumar, and M. Bayoumi, “A novel reconfigurable hardware architecture of neural network,” in2019 IEEE 62nd International Midwest Symposium on Circuits and Systems (MWSCAS). IEEE, 2019, pp. 618–621.

[14] H. Faiedh, Z. Gafsi, K. Torki, and K. Besbes, “Digital hardware implementation of a neural network used for classification,” inProceedings. The 16th International Conference on Microelectronics, 2004. ICM 2004., Dec 2004, pp. 551–554.

[15] M. Kawaguchi, N. Ishii, and M. Umeno, “Analog neural circuit and hardware design of deep learning model,” Procedia ComputerScience, vol. 60, pp. 976–985, 2015.

[16] H. Jang, O. Simeone, B. Gardner, and A. Gruning, “An introduction to spiking neural ¨ networks: Probabilistic models, learningrules, and applications,” 2018.

[17] I. Sourikopoulos, S. Hedayat, C. Loyez, F. Danneville, V. Hoel, E. Mercier, and A. Cappy, “A 4-fj/spike artificial neuron in 65 nmcmos technology,” Frontiers in Neuroscience, vol. 11, p. 123, 2017.

[18] N. P. Thanh, Y.-S. Kung, S.-C. Chen, and H.-H. Chou, “Digital hardware implementation of a radial basis function neural network,”Comput. Electr. Eng., vol. 53, no. C, pp. 106–121, Jul. 2016. [Online]. Available: https://doi.org/10.1016/j.compeleceng.2015.11.017

[19] A. N. Al-Rabadi, “Reversible logic neural networks,” in 2004 IEEE International Joint Conference on Neural Networks (IEEE Cat.No.04CH37541), vol. 4, July 2004, pp. 2677–2682 vol.4.

[20] K. Khalil, O. Eldash, A. Kumar, and M. Bayoumi, “N 2 OC: Neural-network-on-chip architecture,” in 2019 32nd IEEEInternational System-on-Chip Conference (SOCC). IEEE, 2019, pp. 272–277.

[21] G. Wang, “A unified unsigned/signed binary multiplier,” in Conference Record of the Thirty-Eighth Asilomar Conference onSignals, Systems and Computers, 2004., vol. 1, Nov 2004, pp. 513–516 Vol.1.

[22] V. Thenmozhi and R. Muthaiah, “Optimized low power full adder design,” in 2017 International Conference on Networks Advancesin Computational Technologies (NetACT), July 2017, pp. 86–89.

[23] S. Gomar, M. Mirhassani, and M. Ahmadi, “Precise digital implementations of hyperbolic tanh and sigmoid function,” in 201650th Asilomar Conference on Signals, Systems and Computers, Nov 2016, pp. 1586–1589.

[24] M. Al-Nsour and H. S. Abdel-Aty-Zohdy, “Implementation of programmable digital sigmoid function circuit for neuro- computing,” in 1998 Midwest Symposium on Circuits and Systems (Cat. No. 98CB36268). IEEE, 1998, pp. 571–574.

[25] N. M. Nayeem, M. A. Hossain, L. Jamal, and H. M. H. Babu, “Efficient design of shift registers using reversible logic,” in 2009International Conference on Signal Processing Systems, May 2009, pp. 474–478.