RESEARCH ARTICLE OPEN ACCESS

Feed Forward Pipelined Accumulate Unit for the Machine Learning Accelerator

K. Darani*, Dr. P. Sukumar**

*(Department of Electronics and Communication Engineering, Nandha Engineering College,Erode,India
Email: daranikettimuthu20@gmail.com)

** (Department of Electronics and Communication Engineering, Nandha Engineering College,Erode,India
Email: Sukumarwin@gmail.com)

----------**********----------

Abstract:

MAC calculations account for a huge piece of AI quickening agent activities. The pipelined structure is generally embraced to improve the execution by lessening the length of basic ways. An expansion in the quantity of flip-tumbles due to pipelining, be that as it may, for the most part brings about huge territory and power increment. An enormous number of flip-flops are regularly required to meet the feedforward-cutset rule. In view of the perception that this standard can be loose in AI applications, we propose a pipelining strategy that disposes of a portion of the flip-flounders specifically. The reenactment results demonstrate that the proposed MAC unit accomplished a 20% vitality sparing and a 20% zone decrease contrasted and the ordinary pipelined MAC.

Keywords: **Terms—Hardware accelerator, machine learning, multiply–accumulate (MAC) unit, Pipelining.**

----------**********----------

I. INTRODUCTION

As of late, the profound neural system (DNN) rose as an integral asset for different applications including picture arrangement and discourse acknowledgment. Since a gigantic measure of vector-grid augmentation calculations are required in a run of the mill DNN application, an assortment of committed equipment for AI have been proposed to quicken the calculations. In an AI quickening agent, countless duplicate amass (MAC) units are incorporated for parallel calculations, and timing-basic ways of the framework are regularly found in the unit.

A multiplier commonly comprises of a few computational parts including a halfway item age, a segment expansion, and a last expansion. A collector comprises of the convey engendering viper. Long basic ways through these stages lead to the presentation corruption of the general framework. To limit this issue, different techniques have been contemplated. The Wallace and Dadda multipliers are notable models for the accomplishment of a quick section expansion, and the convey lookahead (CLA) snake is regularly used to diminish the basic way in the aggregator or the last expansion phase of the multiplier.

It is outstanding that pipelining is one of the most prominent methodologies for expanding the activity clock recurrence. In spite of the fact that pipelining is an effective method to diminish the basic way delays, it brings about an expansion in the region and the power utilization because of the addition of many flip-flops. Specifically, the quantity of flip-flops will in general be huge on the grounds that the flip-flops must be embedded in the feed forward-cutset to guarantee useful equity when the pipelining. The issue exacerbates as the quantity of pipeline stages is expanded.

The fundamental thought of this paper is the capacity to loosen up the feed forward-cutset rule in the MAC structure for AI applications, in light of the fact that lone the last worth is utilized out of the huge number of increase collections. As it were, not quite the same as the use of the regular MAC unit, moderate amassing qualities are not utilized here, and thus, they don't should be right as long as the last worth is right. Under such a condition, the last worth can wind up right if every twofold contribution of the adders inside the MAC takes an interest in the figuring once and just once, independent of the cycle. Accordingly, it isn't important to define a precise pipeline limit.

In view of the recently clarified thought, this paper proposes a feedforward sans cutset (FCF) pipelined MAC design that is particular for a superior AI quickening agent. The proposed plan technique lessens the zone and the power utilization by diminishing the quantity of embedded flip-flops for the pipelining.

II.PRELIMINARY: FEEDFORWARD-CUTSET RULE FOR PIPELINING

It is outstanding that pipelining is one of the best approaches to diminish the basic way delay, in this way expanding the clock recurrence. This decrease is accomplished through the inclusion of flip-flops into the datapath. Fig. 1(a) demonstrates the square graph of a three-tap finiteimpulse reaction (FIR) channel [11] $y[n] = ax[n] + bx[n - 1] + cx[n - 2]$. (1) Fig. 1(b) and (c) demonstrates pipelining models with respect to the FIR channel. Notwithstanding decreasing basic way delays through pipelining, it is likewise critical to fulfill practical uniformity when pipelining. The time when the flip-flops are embedded to guarantee utilitarian fairness is called the feedforward-cutset. The meanings of cutset and feedforward-cutset are as per the following [11]: Cutset: A lot of the edges of a chart with the end goal that if these edges are expelled from the chart, and the diagram moves toward becoming disconnected. Feedforward-cutset: A cutset where the information move in the
forward heading on the majority of the cutset edges. Fig. 1(b) demonstrates a case of substantial pipelining. The two-organize pipelined FIR channel is developed by embeddings two flip-tumbles along feedforward-cutset. Interestingly, Fig. 1(c) appears a case of invalid pipelining. Useful correspondence isn't ensured for this situation in light of the fact that the flip-flops are not embedded effectively along the feedforward-cutset.

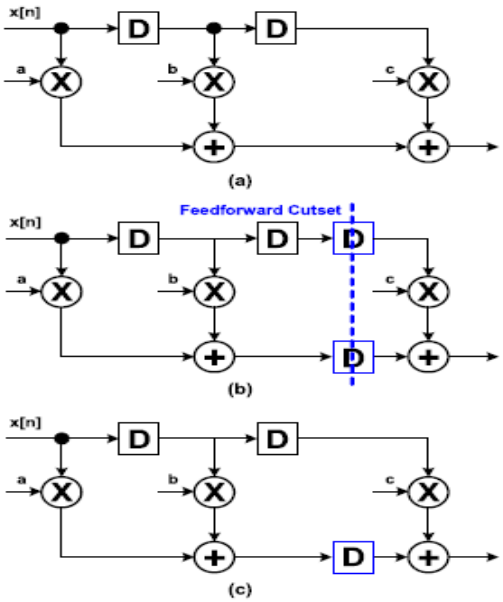

FIG.1 Block diagrams of the three-tap FIR filter [11]. (b) Valid pipelining. (c) Invalid pipelining. "D" indicates a flip-flop.

III. PROPOSED FCF PIPELINING

Fig. 2 shows instances of the two-arrange 32-piece pipelined collector (PA) that depends on the swell convey viper (RCA). A[31 : 0] speaks to information that move from the outside to the info cushion register. AReg [31 : 0] speaks to the information that are put away in the info cradle. S[31 : 0] speaks to the information that are put away in the yield support register because of the gathering. In the traditional PA structure [Fig. 2(a)], the flip-flops must be embedded along the feedforward-cutset to guarantee useful balance. Since the collector in Fig. 2(a) contains two pipeline arranges, the quantity of extra flip-flops for the pipelining is 33 (dark hued flip-flops). On the off chance that the gatherer is pipelined to the n-arrange, the quantity of embedded flip-flops ends up 33(n−1), which affirms that the quantity of flip-flops for the pipelining increments altogether as the quantity of pipeline stages is expanded. Fig. 2(b) demonstrates the proposed FCF-PA. For the FCF-PA, just one flip-flop is embedded for the two-stag pipelining. Accordingly, the quantity of extra flip-flops for the n-organize pipeline is n – 1 in particular. In the ordinary PA, the right collection estimations of the considerable number of contributions up to the comparing check cycle are created in each check cycle as appeared in the planning chart of Fig. 2(a). A two-cycle distinction exists between the information and the comparing yield because of the two-arrange pipeline. Then again, in the proposed design, just the last aggregation result is legitimate as appeared in the planning chart of Fig. 2(b).

A. Modified FCF-PA for Further Power Reductions

In spite of the fact that the proposed FCF-PA can decrease the zone and the power utilization by supplanting the CLA, there are sure information conditions in which the undesired information change in the yield support happens, in this manner diminishing the power productivity when 2's supplement numbers are utilized. Fig. 3 demonstrates a case of the undesired information progress. The information sources are 4-piece 2's supplement paired numbers. AReg[7 : 4] is the sign expansion of AReg[3], which is the sign piece of AReg[3 : 0]. In the regular pipelining [Fig. 3 (left)], the aggregation result (S) in cycle 3 and the information put away in the information cushion (AReg) in cycle 2 are included and put away in the yield cradle (S) in cycle 4. For this situation, the "1" in AReg[2] in cycle 2 and the "1" in S[2] in cycle 3 are included, consequently creating a convey. The convey is transmitted to the upper portion of the S, and thus, S[7:4] stays as "0000" in cycle 4.

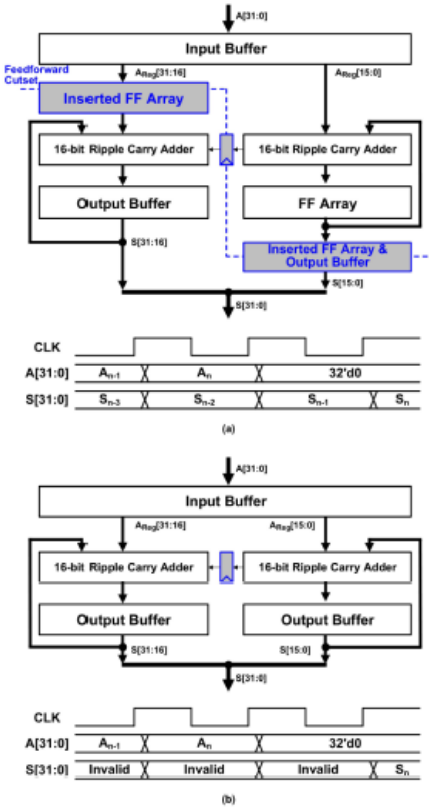

Fig. 2. Schematics and timing diagrams of two-stage 32-bit accumulators.(a) Conventional PA. (b) Proposed FCF-PA.

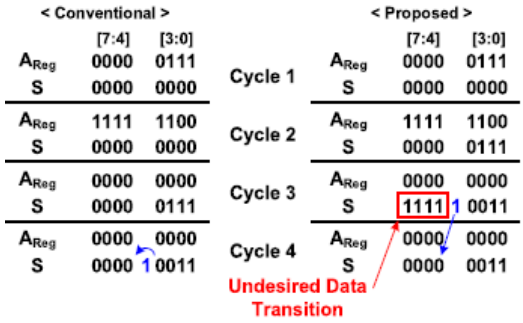

Fig. 3. Example of an undesired data transition in the two-stage 8-bit Pas with 4-bit 2's complement input numbers.

IV. WORKING

The segment expansion in the MAC activity is for the figuring of twofold numbers in every expansion stage utilizing the half-adders or potentially full adders and after that for the death of the outcomes to the following expansion arrange. Since MAC calculations depend on such increments, the proposed pipelining strategy can likewise be applied to the AI explicit MAC structure. In this area, the proposed pipelining strategy is applied to the MAC engineering by utilizing the one of a kind normal for Dadda multiplier. The Dadda multiplier plays out the section expansion along these lines to the Wallace multiplier which is broadly utilized, and it has less zone and shorter basic way delay than the Wallace multiplier Fig. 4 demonstrates the pipelined section expansion structures in the Dadda multiplier. The Dadda multiplier plays out the section expansion to diminish the tallness of each stage. On the off chance that a specific section as of now fulfills the objective tallness for the following segment expansion organize, at that point no activity is performed during the stage Using this property, the proposed pipelining technique can be applied to the MAC structure also. Fig. 4 (a) is a case of pipelining where the traditional technique is utilized. The majority of the edges in the feedforward-cutset are liable to pipelining. Then again, in the proposed FCF pipelining case [Fig. 4 (b)], if a hub in the section expansion stage doesn't have to take part in the stature decrease, it tends to be barred from the pipelining [the bunch in the spotted box of Fig. 4(b)]. At the end of the day, in the traditional pipelining technique, every one of the edges in the feedforward-cutset must be pipelined to guarantee practical uniformity paying little mind to a planning slack of each edge [Fig. 4(a)].

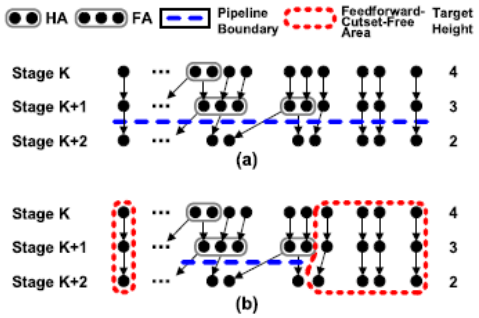

Fig. 4. Pipelined column addition structure with the Dadda multiplier. (a) Conventional pipelining. (b) Proposed FCF pipelining. HA: half-adder. FA: full adder

Therefore, fewer flip-flops are required contrasted and the traditional pipelining case. Then again, in the Wallace multiplier, however many halfway items as would be prudent are engaged with the computation for every segment expansion organize. Since the fractional items need more planning leeway to be avoided from pipelining, the adequacy of the proposed FCF pipelining strategy is littler in the Wallace multiplier case than in the Dadda multiplier case. Fig. 5 demonstrates the square outlines of pipelined MAC models. The proposed MAC design [Fig. 5(b)] joins the FCF (MAC with the proposed FCF pipelining) for the segment expansion and the MFCF-PA for the collection. Rather than pipelining the majority of the last hubs in the segment expansion organize as in Fig. 5(a), the proposed FCF-MAC engineering is utilized to specifically pick the hubs for the pipelining. For the proposed design, the combined duplicate aggregation style is received. The last viper is put in the last phase of the MAC activity. When all is said in done, the last viper is structured utilizing the CLA to accomplish a short basic way delay. Interestingly, the proposed plan utilizes the MFCF-PA style in the amassing stage in light of the more prominent power and the zone proficiency of the MFCF-PA

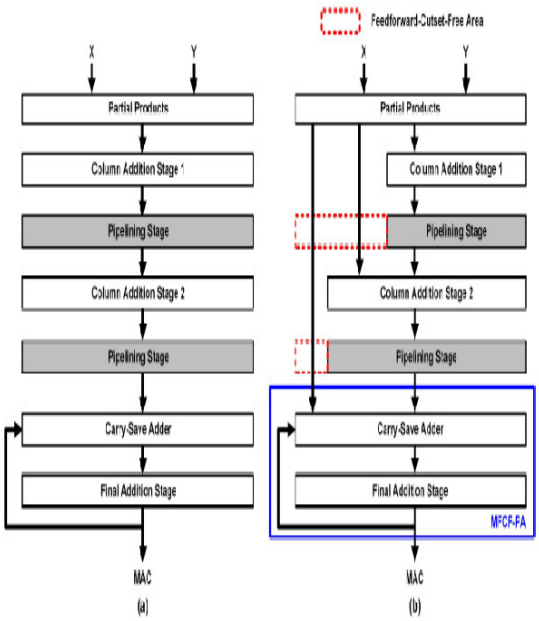

Fig. 5. Block diagrams of pipelined MAC architectures. (a) State-of-the-art merged MAC [12] with pipelining. (b) Proposed FCF-MAC with MFCF-PA.Dotted box: FCF area where the flip-flops are removed from the conventional pipelined MAC.

V. RESULT

We assess the proposed FCF pipelining strategy in this segment. To start with, for use of the collector just case, twofold weight-systems are considered. , we decide the quantity of bits in the collector to be [16 (InputFeature) +11 (Accumulation) =] 27-piece. For the MAC case, the 16-piece 2's supplement number is utilized for both the information highlight and weight. All things considered, the quantity of bits in the last snake is resolved to be [16 × 2 (Multiplication) +11 (Accumulation) =] 43-piece. The plan is combined with the entryway level cells in a 65-nm CMOS innovation utilizing Synopsys Design Compiler. For some specially crafts, "set_dont_touch" direction of Design Compiler is utilized after the immediate launch of the cells in the standard library,

instead of the union of cells utilizing the register-move level depiction.

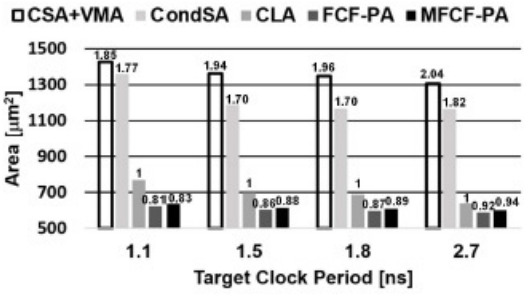

Fig. 6. Comparison of area among CSA + VMA,CondSA CLA-based accumulators, proposed FCF-PA, and MFCF-PA.

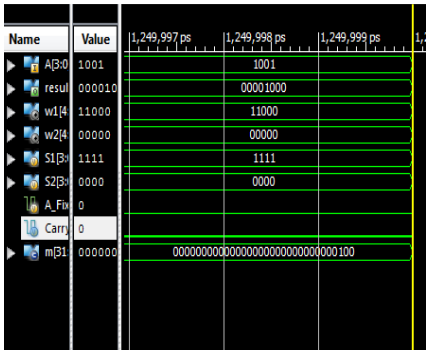

Fig. 7. Simulation Result

For the assessment of the power utilization, we run the time sensitive investigation with a worth charge dump record in the PrimeTime PX. Both the real info highlights (ImageNet informational collection) and arbitrary vectors produced by the pseudorandom number generator (PRNG) are nourished as info information. We structure the CLA by just portraying it as "A + B" in Verilog Hardware Description Language and by integrating/streamlining utilizing Design Compiler. All recreation results for the zone and the control utilization incorporate info and yield cushions. The numbers above bars in the figures show standardized region/capacity to the CLA-based aggregator (gatherer just case) or "Macintosh + CLA" design (MAC case).

VI. CONCLUSION

We presented the FCF pipelining strategy in this paper. In the proposed plan, the quantity of flip-slumps in a pipeline can be decreased by unwinding the feedforward-cutset limitation, because of the extraordinary normal for the AI calculation. We applied the FCF pipelining technique to the aggregator (FCF-PA) structure, and afterward enhanced the power dispersal of FCF-PA by diminishing the opportunity of undesired information advances (MFCF-PA). The proposed plan was moreover extended, and applied to the MAC unit (FCF-MAC). For the assessment, the customary and proposed MAC designs were combined in a 65-nm CMOS innovation. The proposed collector demonstrated the decrease of region and the power utilization by 17% and 19%, individually, contrasted and the collector with the traditional CLA snake based structure. On account of the MAC design, the proposed plan diminished both the region and power by 20%. We accept that the proposed thought to use the one of a kind normal for machine learning calculation for progressively effective MAC configuration can be received in numerous neural system equipment quickening agent structures later on.

REFERENCES

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Image Net classification with deep convolution neural networks," in Proc. Adv. Neural Inf. Process. Syst., 2012, pp. 1097–1105.

[2] K. Simonyan and A. Zisserman. (2014). "Very deep convolution networks for large-scale image recognition." [Online]. Available: https://arxiv.org/abs/1409.1556

[3] C. Szegedy et al., "Going deeper with convolutions," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2015, pp. 1–9.

[4] A. Graves, A.-R. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP), May 2013, pp. 6645–6649.

[5] J. K. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, "Attention-based models for speech

recognition," in Proc. Adv. Neural Inf. Process. Syst., 2015, pp. 577–585.

[6] Y. H. Chen, T. Krishna, J. S. Emer, and V. Sze, "Eyeriss: An energy-efficient reconfigurable accelerator for deep convolution neural networks," IEEE J. Solid-State Circuits, vol. 52, no. 1, pp. 127–138, Jan. 2017.

[7] B. Moons, R. Uytterhoeven, W. Dehaene, and M. Verhelst, "Envision: A 0.26-to-10tops/w sub word-parallel dynamic-voltage-accuracy frequency-scalable convolution neural network processor in 28nm FDSOI," in Proc. IEEE Int. Solid-State Circuits Conf. (ISSCC), Feb. 2017, pp. 246–247

[8] C. S. Wallace, "A suggestion for a fast multiplier," IEEE Trans. Electron. Comput., vol. EC-13, no. 1, pp. 14–17, Feb. 1964.

[9] L. Dadda, "Some schemes for parallel multipliers," Alta Frequenza, vol. 34, no. 5, pp. 349–356, Mar. 1965.

[10] P. F. Stelling and V. G. Oklobdzija, "Implementing multiply-accumulate operation in multiplication time," in Proc. 13th IEEE Symp. Comput. Arithmetic, Jul. 1997, pp. 99–106.

[11] K. K. Parhi, VLSI Digital Signal Processing Systems: Design and Implementation. New Delhi, India: Wiley, 1999.

[12] T. T. Hoang, M. Sjalander, and P. Larsson-Edefors, "A high-speed, energy-efficient two-cycle multiply-accumulate (MAC) architecture and its application to a double-throughput MAC unit," IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 57, no. 12, pp. 3073–3081, Dec. 2010.

[13] W. J. Townsend, E. E. Swartzlander, and J. A. Abraham, "A comparison of Dadda and Wallace multiplier delays," Proc. SPIE, Adv. Signal Process. Algorithms, Archit., Implement. XIII, vol. 5205, pp. 552–560, Dec. 2003, doi: 10.1117/12.507012.

[14] M. Courbariaux, Y. Bengio, and J.-P. David, "Binary Connect: Training deep neural networks with binary weights during propagations," in Proc. Adv. Neural Inf. Process. Syst., 2015, pp. 3123–3131.

[15] M. Rastegari, V. Ordonez, J. Redmon, and A. Farhadi, "XNOR-Net: Image Net classification using binary convolution neural networks," in Proc. Eur. Conf. Comput. Vis. Springer, 2016, pp. 525–542.

[16] M. Gao, J. Pu, X. Yang, M. Horowitz, and C. Kozyrakis, "Tetris: Scalable and efficient neural network acceleration with 3d memory," in Proc. 22nd Int. Conf. Archit. Support Program. Lang. Oper. Syst., 2017, pp. 751–764.

[17] A. Parashar et al., "SCNN: An accelerator for compressed-sparse convolution neural networks," in Proc. 44th Annu. Int. Symp. Comput. Archit., Jun. 2017