PERFORMANCE ANALYSIS OF LOSS LESS COMPRESSION FOR DDR4 USING COMMAND TRACE APPLICATION

[1]K.Venkata Suma, [2] Dr. Mohammad Hayath Rajvee, [3]Jabeena Shaik
[1]PG Scholar,[2]Professor,[3]Associate Professor & HOD, Dept. of E.C.E
[1,2,3]Quba College of Engineering and Technology, Nellore, SPSR Nellore Dist., A.P

ABSTRACT

This paper proposes a highthroughput HW gas pedal for lossless compression of the command trace. The proposed HW is designed in a pipeline construction to handle Huffman tree age, encoding, and stream merge.To keep away from the HW cost increment attributable to highthroughput preparing, a Huffman tree is productively carried out by using static irregular access memory-based lines and bitmaps. What's more, factor length stream combine is performed for a minimal price by lessening the HW wire width utilizing the numerical properties of Huffman coding and handling the metadata and the Huffman codeword utilizing FIFO independently. Moreover, to further develop the compression productivity of the DDR4 memory command, the proposed design incorporates two preprocessing activities, the "don't mind bits supersede" and the "bits orchestrate," which use the working qualities of DDR4 memory.

I. INTRODUCTION

Performing various types of handling over an intricate type of a picture is a difficult errand with regards to hardware-based execution. In this angle, picture compression is one such cycle that fundamentally offers overhead in the hardware realization.All the issues begins with the memory arrangement of the hardware segment while handling huge number of picture pixels. It is accepted that the exhibition of the memory the board exclusively relies on the example of its getting to instrument. It is fundamental that analysis of the traces of the memory is needed to be investigated to further develop the memory the board of the hardware. The picture compression component intends to advance the size of picture for smooth transmission and storage.The presence of ancient rarities in the pictures can likewise eliminate through picture compression. Be that as it may, there is testing circumstance in accomplishing critical picture compression for the space and energy compelled gadgets. To play out a proficient picture compression, there are different conditions that are needed to be cooked up.The essential conditions to be fulfilled are that the use of lossless compression is significant for opposing loss of traced information in memory framework. The auxiliary condition to be fulfilled is that there ought to be higher level of throughput and sped up for higher data transfer capacity acceptability. With the expanding recurrence of activity and the higher information transmission capacity, there is a need of higher pace of throughput too.The fundamental explanation for this is that to offer better analysis of memory traces there is a need of creating speed increase arrangement of the compression as for hardware displaying and not the product based indicative approach.Another significant finding is that product based compression approach is in every case more slow in contrast with the hardware-based compression approach that downsize the compression execution. Hence, there is a huge tradeoff between hardware-based methodology and programming based methodology to accomplish the objective of compression productivity. The expanded utilization of Internet-of-Things (IoT) can prompt more use of asset limitation gadgets

associated each other through various organizations.

II.EXISTING SYSTEM

This load of past investigations in [21]–[24] use Huffman coding, however they are not an ideal Huffman coding. Nunez and Jones [21] and Fowers et al. [24] utilized a foreordained tree structure; they can't pack information while changing the tree structure as indicated by the information. If there should be an occurrence of Lin et al. [22], [23], they can't deal with compression and tree development simultaneously in light of the fact that they assemble an approximated tree structure on a disconnected process.Thus, the compression proportion is definitely forfeited. Moreover, the working recurrence of these past plans is moderately low and aside from [24], they can just handle the information with a little BW, delivering it hard to measure highthroughput information of 8 GB/s. This implies that these past plans are deficient as far as compression proficiency and HW intricacy/cost.Herein, to make up for this downside, the tradeoff between compression effectiveness and HW cost is painstakingly thought to be through the productive pipeline handling of blockbased compression and HW-improved design methods.

III.PROPOSED SYSTEM

Fig. 1 shows the general construction of the proposed HW design. It comprises of three sections: block Huffman, equal stream combine, and preprocessing. The block Huffman module comprises of a cradle, a Huffman forward portion, and a Huffman back part, as displayed in Fig. 2. In the Huffman forward portion, the recurrence tally module figures the image recurrence of the information and the arranging module orchestrates the images in the recurrence request by combine sort. In the Huffman back section, a Huffman tree is shaped utilizing the images and frequencies arranged in the Huffman forward portion. In this cycle, the module utilizes SRAM-based lines and bitmaps fittingly to frame the Huffman tree and theHuffmancodewordadequately.

A.PROPOSED ARCHITECTURE

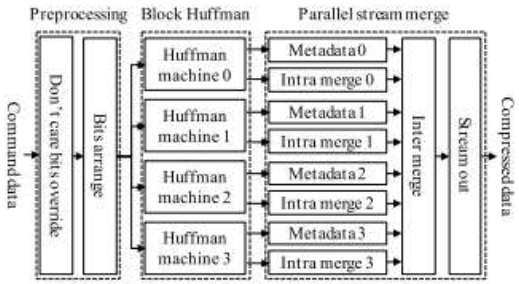

Fig 1: Overall structure of the proposed hardware design.

B.BLOCK DAIGRAM DESCRIPTION

An equal stream blend module packs the yield surges of four equal compression machines into one stream. It comprises of the metadata, intramerge, intermerge, and stream out module. In the metadata module, image and recurrence data are stuffed into the single stream since they are expected to develop the tree at the deciphering cycle. The intramerge module combines the eight variable length streams from the single compression machine into the single stream utilizing shifters as well as tasks. The intermerge module packs the metadata and intramerge streams into the single stream by shifters as well as tasks. The stream out module cuts the last compacted stream from the intermerge module to a specific size and sends it to the last yield. This stream combine module is fundamental for variable length coding yet requires colossal HW cost for high-throughput information preparing. To address this disadvantage, thus, the equal stream blend module proficiently pipelines these consolidation measures and advances the HW wire width utilizing the numerical properties of Huffman coding, to empower high-throughput information handling with an extremely low HW cost. Thusly, the proposed HW gas pedal performs lossless information compression with a high throughput of 8 GB/s and accomplishes incredible HW cost per throughput execution. Besides, it can protect the compression proportion of ideal

Huffman coding since it creates the new Huffman tree consistently as per the information. This minimal expense HW design of lossless compression, which keeps up with high compression productivity and high throughput.

IV.RESULTS

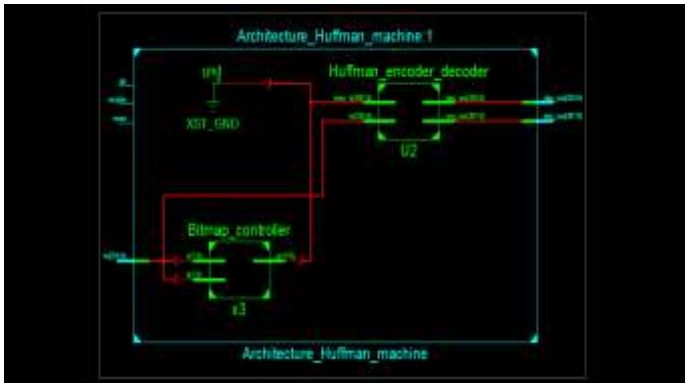

Fig 2:RTL Schematic

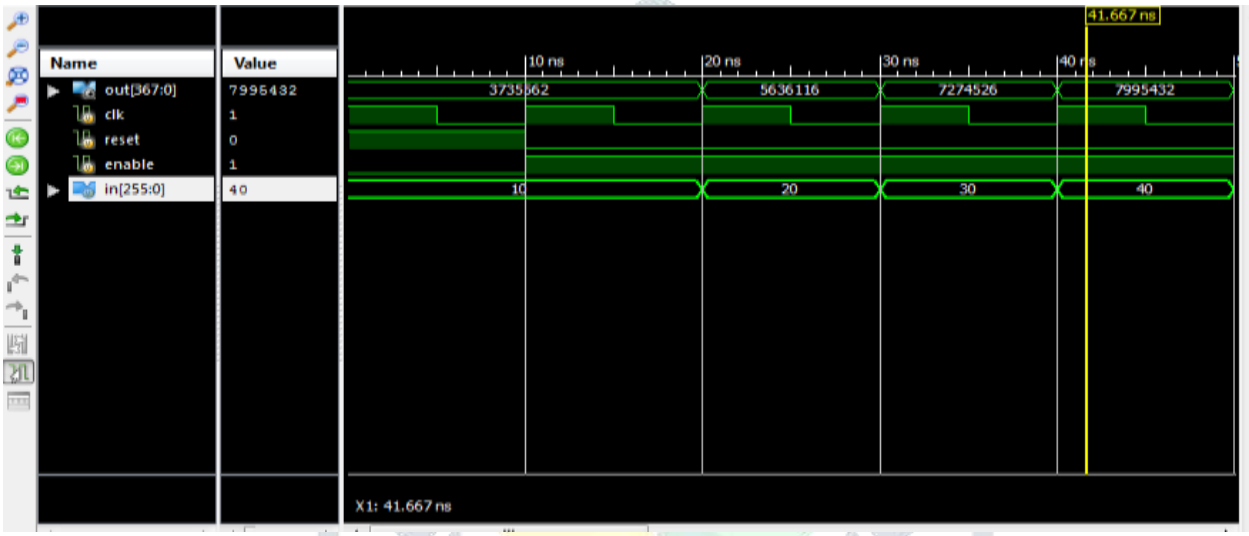

Fig 3:Simulation Results

V.CONCLUSION

As memory command trace analysis gets huge in further developing memory execution, memory command trace compression is getting progressively significant. Specifically, to help the fast expansion in the measure of information, the interest for a high-throughput HW design of lossless compression is expanding extensively. The two principle commitments of this paper are as per the following. Initial, a minimal expense HW design of lossless compression, which keeps up with the high compression proficiency and high throughput, is proposed. Trial results show that the proposed HW design has a vastly improved all out HW asset per throughput than the past examinations. Second, to additional improve the compression proportion for the memory command trace information, two preprocessing techniques, the "don't mind bits supersede" and the "bits orchestrate," which use the working attributes of the DDR4 memory, are proposed.

REFERENCES

[1] H. Choi, J. Lee, and W. Sung, “Memory access pattern-aware DRAM performance model for multi-core systems,” in Proc. IEEE Int. Symp. Perform. Anal. Syst. Softw., Apr. 2011, pp. 66–75.

[2] S. Rixner, W. J. Dally, U. J. Kapasi, P. R. Mattson, and J. D. Owens, “Memory access scheduling,” in Proc. 27th Annu. Int. Symp. Comput. Archit., May 2000, pp. 128–138.

[3] Y. Huang et al., “HMTT: A hybrid hardware/software tracing system for bridging the DRAM access trace’s semantic gap,” ACM Trans.

Archit. Code Optim., vol. 11, no. 1, Feb. 2014, Art. no. 7.

[4] D.-I. Jeon, M.-K. Lee, J.-C. Kim, and K.-S. Chung, "Runtime memory controller profiling with performance analysis for DRAM memory controllers," J. Circuits, Syst. Comput., vol. 27, no. 8, p. 1850126, Nov. 2017.

[5] C. F. Kao, S. M. Huang, and I. J. Huang, "A hardware approach to realtime program trace compression for embedded processors," IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 54, no. 3, pp. 530–543, Mar. 2007.

[6] M. Burtscher, I. Ganusov, S. J. Jackson, J. Ke, P. Ratanaworabhan, and N. B. Sam, "The VPC trace-compression algorithms," IEEE Trans. Comput., vol. 54, no. 11, pp. 1329–1344, Nov. 2005.

[7] K.-U. Irrgang and T. B. Preußer, "An LZ77-style bit-level compression for trace data compaction," in Proc. 25th Int. Conf. Field Program. Logic Appl., Sep. 2015, pp. 1–4.

[8] V. Uzelac and A. Milenkovic, "A real-time program trace compressor utilizing double move-to-front method," in Proc. 46th ACM/IEEE Design Automat. Conf., Nov. 2009, pp. 738–743.

[9] A. Milenkovic, V. Uzelac, M. Milenkovic, and M. Burtscher, "Caches and predictors for real-time, unobtrusive, and cost-effective program tracing in embedded systems," IEEE Trans. Comput., vol. 60, no. 7, pp. 992–1005, Jul. 2011.

[10] High Bandwidth Memory DRAM, JEDEC Standard JESD235A, 2015