Hardware implementation of sorting algorithm using FPGA

*Gayathri K,HarshiniV S,**Dr Senthil Kumar K K

**Students. **Associate professor, Department of Electronics & Communication Engineering , Prince Shri Venkateshwara Padmavathy Engineering College, Chennai ,Tamilnadu,India*

ABSTRACT

Sorting is one of the most fundamental topics in computer science, It is the basic process in data analysis and in an enormous application.Software sorting doesn't provide efficiency so hardware sorting is used.In this proposed system the different hardware sorting is compared and the best sorting is justified.The register transfer level of each sorting is schematic by this paper.The model simulation also provided so, it is easy to compare the sorting.The bitonic merge sort and Bitonic odd even sort using parallelism concept are proposed in this paper.

Keyword:- *Bitonic merge,Bitonic odd even merge sort, Comparator, Bubble sort, Parallelism, Register transistor level,Comparison,Simulation.,4 bit,8 bit,16 bit,32 bit.*

1.Introduction

The process of sorting is one of the most fundamental problems in computer science, and sorting algorithms are vital for computer engineering [1]. A large number of applications employ sorting operations, including scientific computing [2], [3], image processing [4], multimedia [5], and network processing . For decades, numerous sorting algorithms have been studied and optimized. For a small input of size n, bubble sort and insertion sort are fast in-place sorting methods, but both algorithms require execution time in the worst case. The widely known quick sort algorithm applies the divide-and-conquer strategy, which has an average execution time of Q (n log2 n) asymptotically, but in the worst case, that is, the input elements are originally placed in a completely reverse order, it increases to Q(n2). Merge sort and heap sort are asymptotically optimal because the upper bound of running time O(n log2 n)matches the worst-case lower bound O(n log2 n) of comparison based sorting algorithm as shown in figure 1.

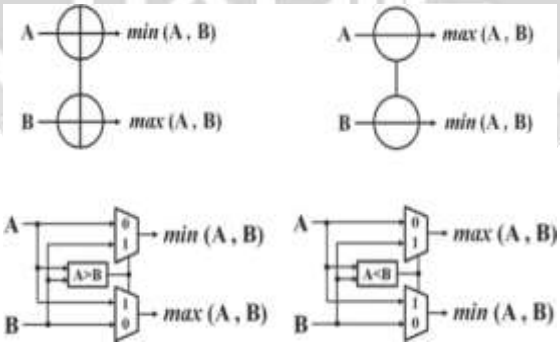

Figure. 1. The increasing (a) and decreasing (b) comparing block. (c) and(d) are the detail architectures

1.1 Existing system

In existing system they implemented some sorting techniques in software which is not feasible to achieve high speed and software sorting techniques are bubble sort,merge sort,quick sort. This can be enhanced by using hardware sorting using FPGA. This paper implements the hardware sorting in bubble sort, bitonic merge sort,bitonic odd even merge sort which enhanced using comparators and bitonic merge sort and bitonic odd even merge sort is invented by batcher.

2 Proposed scheme

This paper enhances the speed of various sorting techniques by using comparators and This compares the results for each sorting techniques with its delay and parallelism concept is used in this paper for various sorting techniques.

2.1 Bubble Sort and Its Parallel Design

For small inputs, bubble sort is a feasible solution and is also easy to implement. In each round, the largest (or smallest) sample is selected by a series of comparisons. The algorithm requires M comparison and switching events in the first round when the input size is M, M-1 events in the second round, M-2 events in the third round and so on until the complete sorted result is generated. The running time is O(M^2)where M is the input size. Parallization is a well-known solution to enhance the performance. Fig. 2 shows ahardware design for parallel bubble sort, where comparisonblocks in the same column are executed in parallel. Although the total number of comparing units is as same as that in the sequential version, a few operations could be executed simultaneously in a certain pipeline stage. The overall pipeline stage is defined as 2M-3, and the run time can bereduced to O(m)

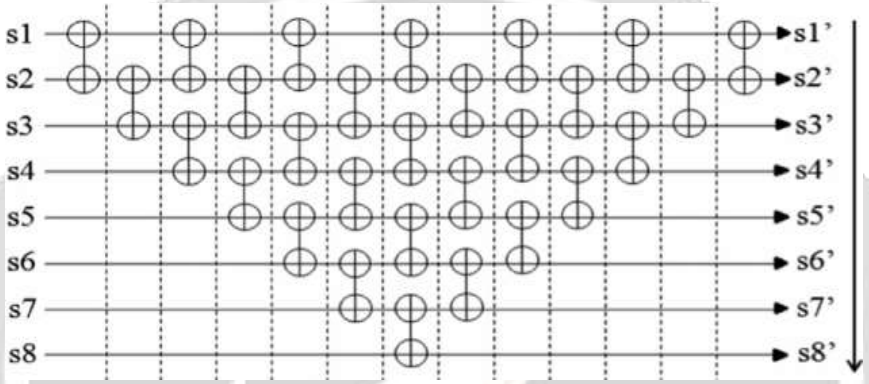

Figure. 2. The parallel architecture of an eight-input bubble sort**.**

2.2 Odd-Even Merge Sort

The odd-even merging unit proposed by Batcher merges two sorted sequences into a complete sorted result. A sorting network can be recursively constructed using the merging unit. An M-input odd-even mergingunit isdenoted by OE-M, where M should be the power of two.The sorted sequence could be generated through a seriesof parallel merging units from OE-2s, OE-4s, OE-8s ... toOE-M. The architecture is parallel and feasible for pipelinedesign.

To sort a data set with 2P samples, there are 2P-1OE-2s in the first stage, 2^P-2 OE-4sin the second stage, and soon, until there is one OE-2^P in the final stage. Further-more, an OE-2^P merging unit could be subdivided into P stages. The time complexity of an odd-even merging net-work with M inputs can be represented by O(log2^2 M)because there are 1 + 2 +…+ log2 M stages in total, and the area complexity is O(M log2^2 M)Fig. 3 illustrates an example of an eight-input odd-even merge sorting network composed of four parallel OE-2s, two parallel OE-4s, and one OE-8. The pipeline levels are 6 and there are 19 increasing comparison blocks.

.

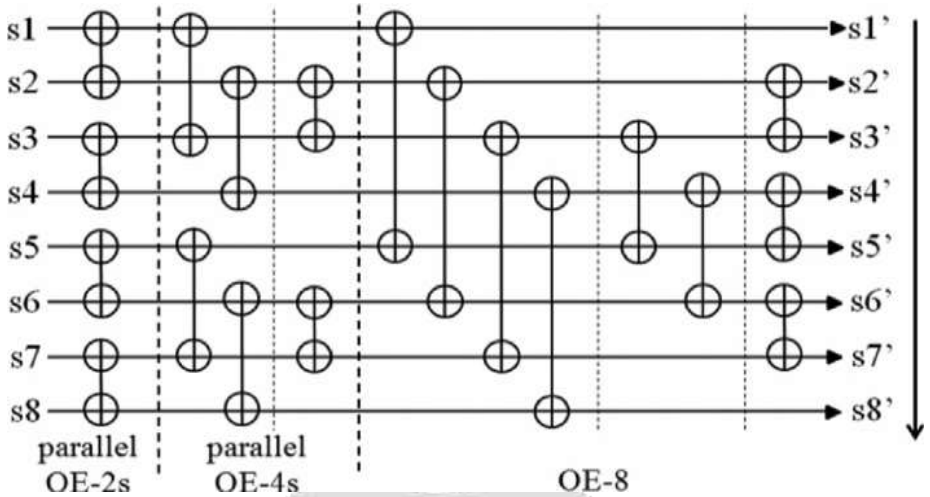

Figure 3. The architecture of an eight-input odd-even sorting network

2.3 Bitonic Merge Sort

Bitonic sort is another sorting network proposed by Batcher. A bitonic sequence is composed of one sequence inincreasing order and another one in decreasing order. Thebitonic merging unit merges the two sequences with equallength into a complete sorted result. Bitonic sort has beenused widely because of its regular structure, which makes itconsiderably simpler to implement than an odd-even sorting network. Similarly, the input size of bitonic mergingunit should be a power of two. The M-input merging unitreceives an ascending and a descending sequence, and bothof themcontain M/2 samples. It is called a BM-M, where Mcan be represented as 2^P. To construct a complete 26inputbitonic sorting network, a series of bitonic merging units areapplied recursively to generate the bitonic subsequence.

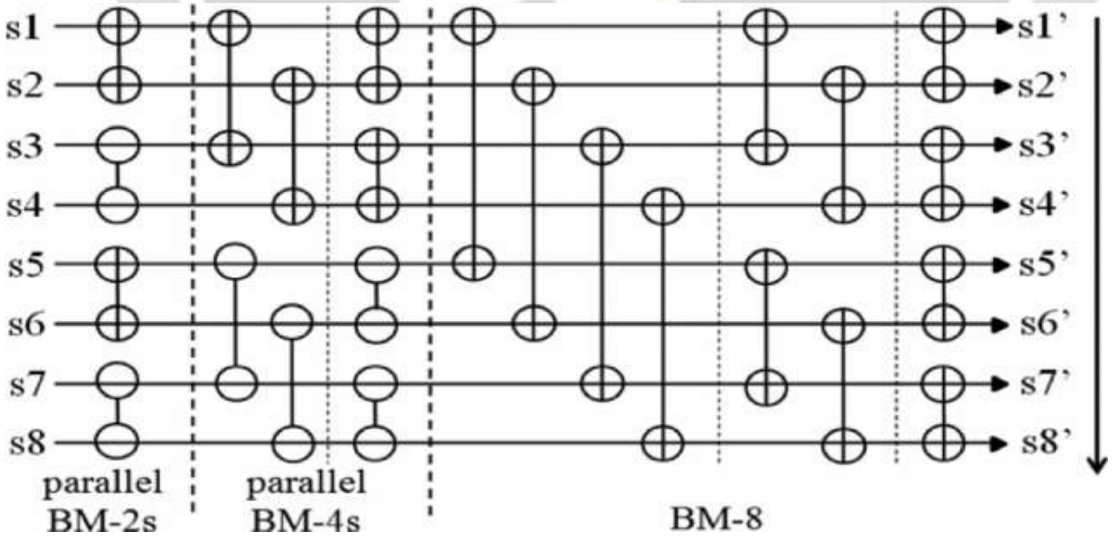

Figure. 4. The architecture of an eight-input bitonic merge sorting network.

The 2^P-input bitonic sorting network consists 2^P-1 parallel.BM-2s in the first level, 2^P-2 parallel BM-4s in the second level .and one BM-2^P in the final level. The time complexity and area cost are the same as those of the odd-even sorting network. Fig. 4 illustrates an example of an eight-input bitonic sorting network composed of four parallel BM-2s, two parallel BM-4s, and one BM-8. The pipeline levels are 6 and there are 24 comparison blocks. A few of the comparing blocks produce increasing sequences, whereas others produce decreasing results, which is the most notable difference between the odd-even and the bitonic sorting

Table.1. Comparison between the existing and proposed method for input 4-bit width

Sorting technique	IOs	Slices	4 input LUT	Delay(ns)
Bitonic odd even merge sort	64	124	218	34.87
Bitonic merge sort	64	160	263	37.26
Bubble sort	64	176	309	73.82

Table. 2. Comparison between the existing and proposed method for input 8-bit width

Sorting technique	IOs	Slices	4input LUT	Delay(ns)
Bitonic merge	128	302	549	34.92
Bitonic odd even merge sort	128	251	456	33.87
Bubble sort	128	370	672	65.91

Table 3 Comparison between the existing and proposed method for input 16-bit width

Sorting technique	IOs	Slices	4input LUT	Delay(ns)
Bitonic merge	256	620	1126	37.07
Bitonic odd even merge sort	256	502	912	36.97
Bubble sort	256	739	1344	72.91

Table.4. Comparison between the existing and proposed method for input 32-bit width

Sorting technique	IOs	Slices	4input LUT	Delay(ns)
Bitonic merge	512	1238	2250	42.29
Bitonic odd even merge sort	512	1003	1824	42.76
Bubble sort	512	768	1400	74.12

3. RESULT AND ANALYSIS

In result and analysis this compares delay for various sortiing it is simulated it is represents in figure 5. This compares the results for slices for different input bitwidth like 4 bit,8bit ,16 bit,32 bit for various sorting techniques.

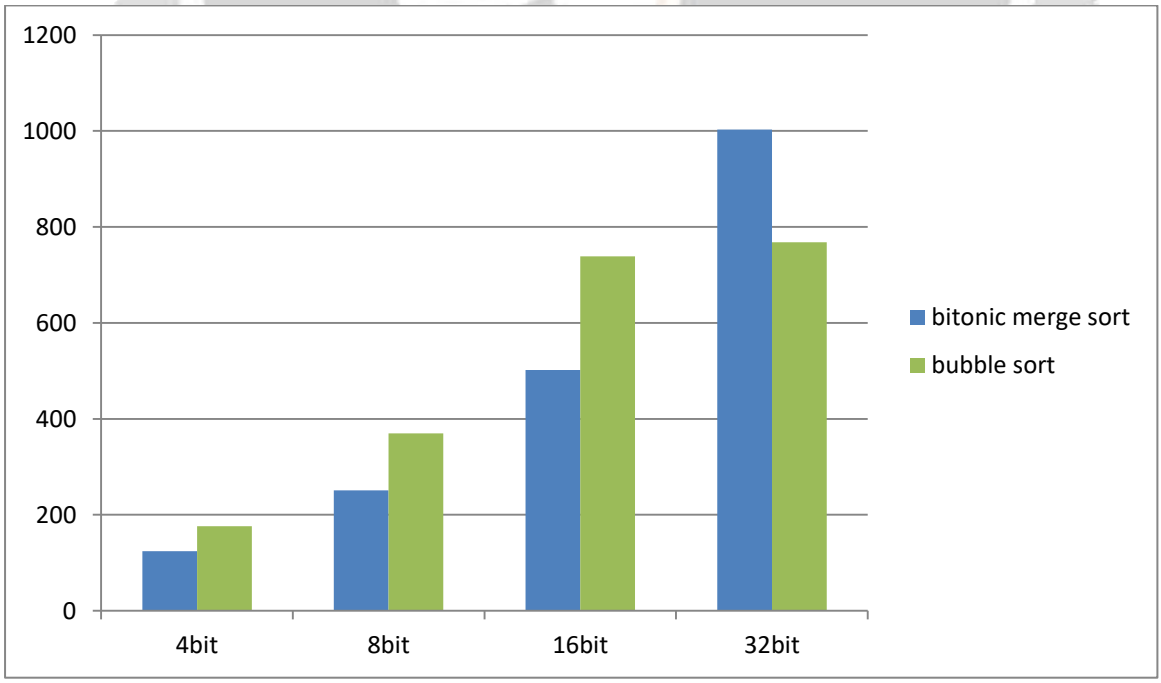

Figure 5:Slices

3.1 Simulation Output

Simulation output is obtained by running Verilog code in modelsim software.Figure 9 shows the simulation results of bubble sort implementation.The Figure 10,11 shows the RTL diagrams obtained from Xilinx.

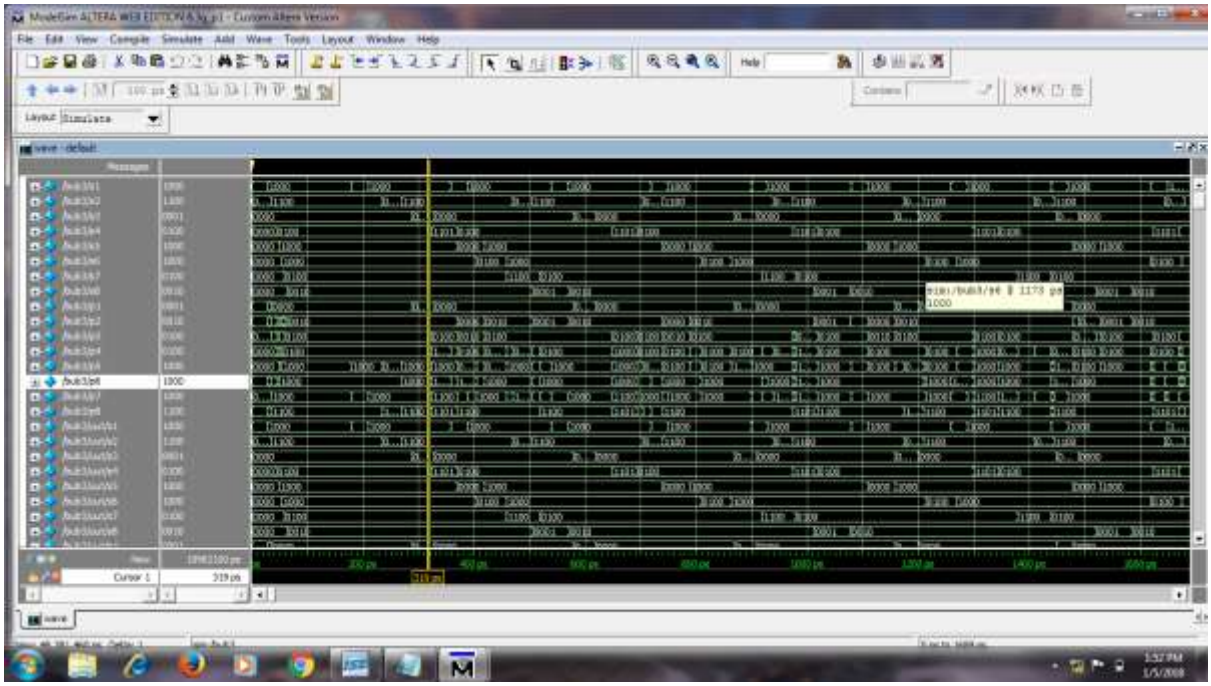

Figure 9:simulation output for bubble sort

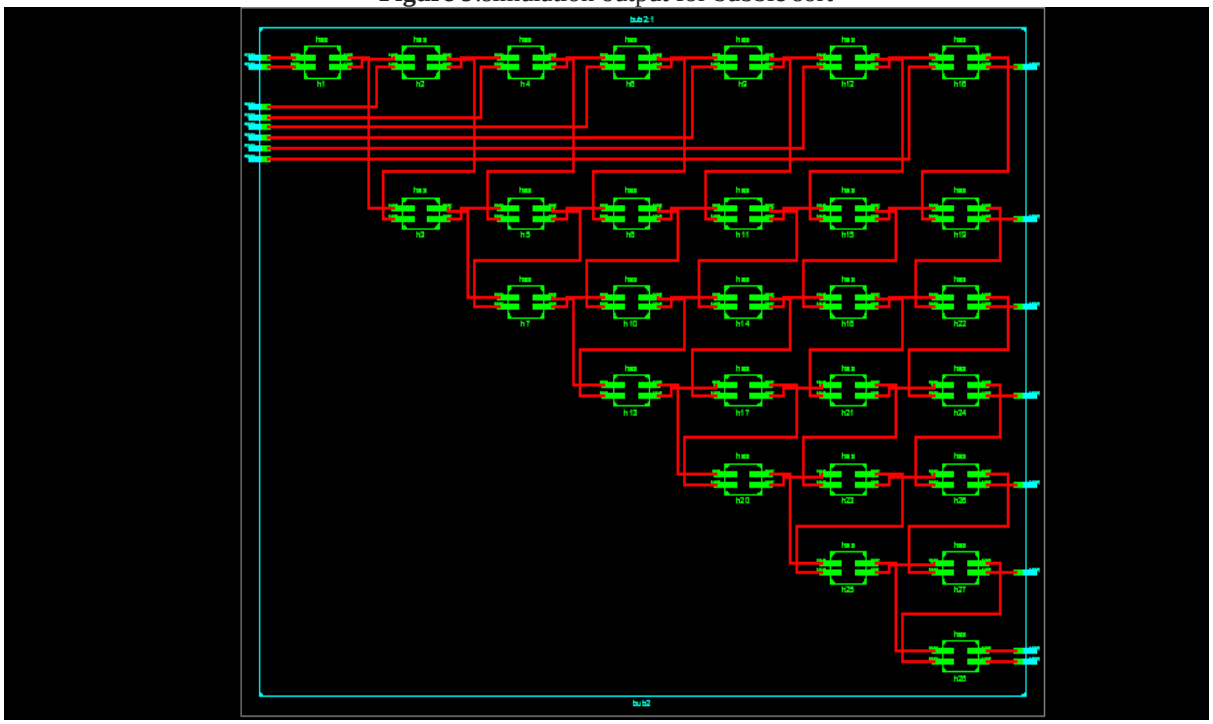

Figure 10: RTL Schematic for bubble sort

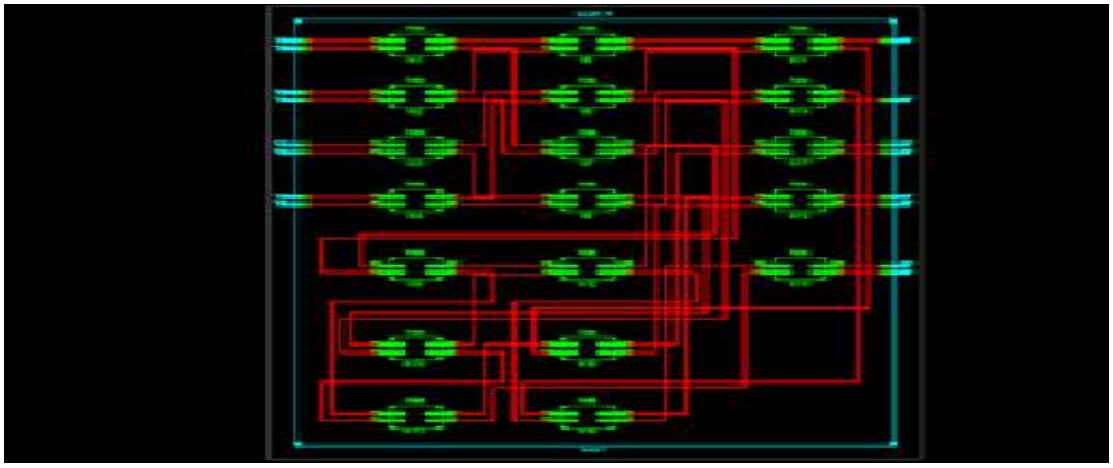

Figure 11: RTL Schematic for bitonic odd even merge sort

4.Conclusion

From the comparison of hardware sorting that is bubble sort, Bitonic odd-even sorting and Bitonic merge sorting using parallelism concept.It is concluded that the bitonic odd-even is fast with less delay but the structure is not fixed.Bitonic merge is more or less equal delay as bitonic odd even, but it provides fixed structure.Therefore tradeoff plays a specific role because bitonic odd even uses less space than bitonic merge.Depending on the application the sorting must be chosen.

5. REFERENCE

[1] K. E. Batcher, "Sorting networks and their applications," in Proc.AFIPS Proc. Spring Joint Computer Conf., 1968, pp. 307–314.

[2] L. Njejimana, et al., "Design of a real-time FPGA-based dataacquisition architecture for the LabPET II: An APD-based scannerdedicated to small animal PET imaging," IEEE Trans. Nucl. Sci.,vol. 60, no. 5, pp. 3633–3638, Oct. 2013.

[3] A. Colavita, E. Mumolo, and G. Capello, "A novel sorting algorithmand its application to a gamma-ray telescope asynchronousdata acquisition system," Nuclear Instruments Methods Phys. Res.Section A: Accelerators, Spectrometers, Detectors Associated Equipment,vol. 394, no. 3, pp. 374–380, 1997.

[4] A. Gabiger-Rose, M. Kube, R. Weigel, and R. Rose, "An FPGAbasedfully synchronized design of a bilateral filter for real-timeimage denoising," IEEE Trans. Ind. Electron., vol. 61, no. 8,pp. 4093–4104, Aug. 2014.

[5] G. Dimitrakopoulos, C. Mavrokefalidis, K. Galanopoulos, andD. Niolos, "Sorter based permutation units for media enhancedprocessors," IEEE Trans. Very Large Scale Integr. Syst., vol. 15, no.[6], pp. 711–715, Jun. 2007.