2016 International Conference on Circuit, Power and Computing Technologies [ICCPCT]

Design of Register File using Reversible Logic

Chandana S
Department of Electronics and Communication Engineering
PES Institute of Technology
Bangalore, India
chandanas19071995@gmail.com

Navya C
Department of Electronics and Communication Engineering
PES Institute of Technology
Bangalore, India
cnavya03@gmail.com

Nagamani AN
Department of Electronics and Communication Engineering
PES Institute of Technology
Bangalore, India
nagamani@pes.edu

Vinod K Agrawal
Director of CORI
PES University
Bangalore, India
vk.agrawal@pes.edu

***Abstract*—Register file is the paramount aspect in computer memory unit. Eight bits (one memory unit) results in a single register and 32 of such register make up a register file. In this paper we have presented the design of a complete register file using reversible logic design. It consists of decoder, multiplexer, memory unit, read and write units. This has been verified using VHDL. In addition to that, we have implemented the register file in the design of Content Addressable Memory (CAM) as an application.**

I. Introduction

In current scenario, the reversible logic design attracting more interest due to its low power consumption. Reversible logic is very important in low-power circuit design. According to Landauer, the energy of each bit is equal to kT*ln2 (where 'k' is the Boltzman constant and 'T' is the temperature in Kelvin). [1]With the loss of each bit, kT*ln2 of energy is lost. In order to save this energy, Reversible logic gates are implemented. It has same number of input and output gates and all the inputs bits can be recovered, thus saving energy.There are various reversible gates available, but we have mainly used Feynman Gate, Fredkin gate, Toffoli gate . A reversible logic gate is an n-input n-output logic device with one-to-one mapping. To implement reversible computation, estimate its cost, and to judge its limits, it is formalized it in terms of gate level circuits. Reversible computing will also lead to improvement in power efficiency. Power efficiency will fundamentally affect the speed of circuits such as nano circuits and therefore the speed of most computing applications. Part I is introduction to the reversible logic design. Part II explains the preliminaries. In part III design of the register file is presented.

II. Preliminaries

There are many parameters for determining the complexity and performance of circuits. [2]

A. The number of Reversible gates (N)

This refers to the number of reversible gates used in circuit.

B. Number of constant inputs (CI)

This refers to the number of inputs that are to be maintained constant at either 0 or 1 in order to realize the given logical function.

C. The number of garbage outputs (GO)

This refers to the number of unused outputs present in a reversible circuit.

D. Quantum cost (QC)

This refers to the cost of the circuit in terms of the cost of a primitive gate. It is calculated knowing the number of primitive reversible logic gates (1*1 or 2*2) required to realize the given function.

E. Delay

The delay of a logic circuit is the maximum number of gates in a path from any input line to any output line. The definition is based on two assumptions:
(i) Each gate performs computation in one unit time and
(ii) all inputs to the circuit are available before the computation begins.
The delay of each 1x1 gate and 2x2 reversible gate is taken as unit delay 1. Any 3x3 reversible gate can be designed from 1x1 reversible gates and 2x2 reversible gates, such as CNOT gate, Controlled-V and Controlled-V+ gates (V is a square-root-of NOT gate and V+ is its Hermitian). Thus, the delay of a 3x3 reversible gate can be computed by calculating its logical depth when it is designed from smaller 1x1 and 2x2 reversible gates

F. BASIC REVERSIBLE LOGIC GATES USED IN THE DESIGN

The quatum diagram and the quantum cost of the gates which is prominently used in the paper is shown in Fig. 1

III. design of Register File

The core element in computer memory is Register File. As said earlier, each register file contains 32 registers and each register is made of n byte. Here, we have shown a register having 1 byte: 8 bits.
The general working is as follows:
The control signal is given to the first inputs of two Feynman gate and second input are assigned 1(constant). The outputs of Feynman gate 1 are READ 1 and WRITE . READ 1 acts as an enable/selection signal for first multiplexer .Similarly, the first output from Feynman gate 2 acts as select signal for second multiplexer, outputting READ DATA 2. WRITE signal

978-1-5090-1277-0/16/$31.00 ©2016 IEEE

is connected to the AND gates to enable Write operation on to the memory cell. The Multiplexer when enabled reads the written data from the selected register. The Register file block diagram is shown in the Fig. 2

A. Operation involved

The processor interacts with the memory unit through two different operations, namely: Read and Write. The explanation for each operation is given below.

1) Read operation: When the control signal is high , the READ 1/READ2 signal behaves as the select line for the Multiplexer. Once the multiplexer is enabled , the data which stored in the corresponding register is chosen and made available on READ DATA 1/ READ DATA 2 line. The Fig. 3 gives the diagrammatic approach to read operation.

2) Write operation: When control signal is low , the WRITE signal is enabled(READ 1 and READ 2 is disabled) . The WRITE enable the AND gate. One of the input to AND gate is the data from the decoder. Register number is fed to a 5:32 decoder . Each output of AND gate acts as code in register file and the 32 bit register data is directly fed into the register file. Fig. 4 shows the Write operation diagram.

B. Design of individual components used in the Block Diagram

1) Multiplexer (Enocders) using MFRG gate (32:1): Multiplexer, a combinational circuit, is used as data selector. It has many inputs and one output. Depending upon the select value, corresponding data line will be available at the ouput. We have proposed a 32:1 multiplexer using modified Fredkin gate which has a reduced quantum cost and power and play a vital role in read part of register file .we have 2 read output which requires two 32:1 multiplexer . For multiplexer block we modify the Frekdin gate. The input vector, Iv= (A ,B,C). [3]

Gate	Gate structure	Quantum cost
Feynman (FG)	A, B; P=A, Q=A ⊕ B; 2*2	1
Fredkin (FRG)	A, B, C; P=A, Q=A'B ⊕ AC, R=A'C ⊕ AB; 3*3	5
Toffoli (TG)	A, B, C; P=A, Q=B, R=AB ⊕ C; 3*3	5
Modified Fredkin Gate (MFRG)	A, B, C; MFRG Gate; P=A, Q=AB' ⊕ AC', R=A'C ⊕ AB; 3*3	4

Fig. 1. Preliminaries

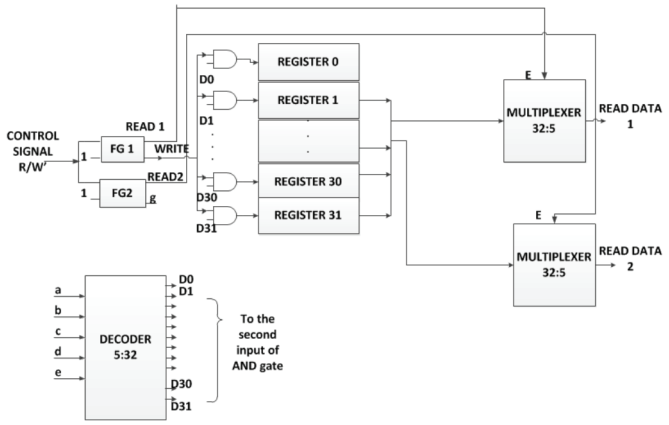

Fig. 2. Register File Block Diagram

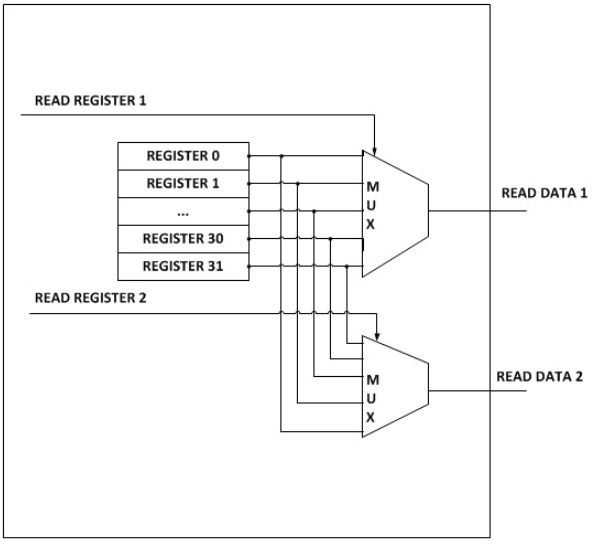

Fig. 3. Read operation [12]

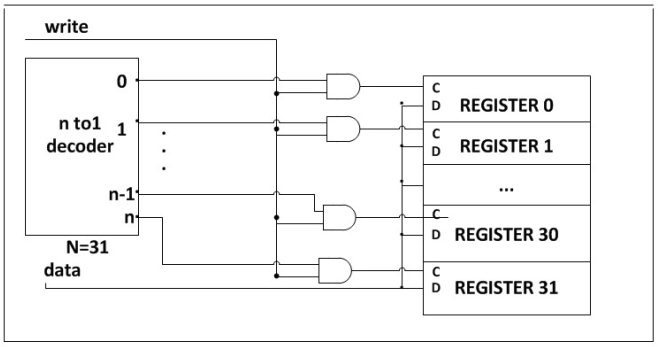

Fig. 4. Write operation [12]

This design consists of 5 stages. In first stage A will be select line and B and C are two data inputs. The first output of every gate which will be the select line to next MFRG of same stage. The second output is fed as second input for

the succeeding stage MFRG gates. The third output will be a garbage output. This repeats for every stage. In the fifth stage, second output will be the required multiplexed output.
The output of these two 32:1 multiplexer will be READ DATA 1 and READ DATA 2. The results are simulated in Xilinx by using VHDL language. The design is shown in the Fig.5

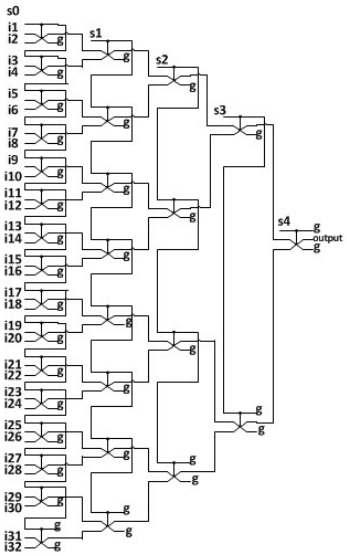

Fig. 5. Encoder

2) Decoder for a Register File: Decoder, also a combinational circuit, has n inputs and 2^n outputs. As, the name says, it decodes the coded inputs. [8]
The proposed design as shown in Fig. 6 is a 5 to 32 decoder, built using one Feynman gate and thirty Fredkin gates.
a and *0* are given input to the Feynman gate to produce a and a. At each stage , new inputs are introduced. Each stage has 2^n, n=0,1,2,3,4 number of gates. '*a*' and '$\bar{a}$' fed as the input to the next two Fredkin gate along with input '*b*' and ancilla *0*.
For the next stage, 4 input from these two Fredkin gates (second and third output of each Fredkin gate) are given to the second input gate of the Fredkin and the *0* is fed as input to the third input gate of the Fredkin. This is repeated for the next two stages as shown in the figure. It has 4 garbage values and 31 ancilla. The output at the end fourth stage are, a^- $\bar{b}\,\bar{c}\,\bar{d}\,\bar{e}$, $a\,\bar{b}\,\bar{c}\,\bar{d}e$, $a\,\bar{b}\,\bar{c}\,d\,\bar{e}$ ... $abc\bar{d}e$, $abcd\bar{e}$, abcde from 0 to 31.

3) AND gate using Toffoli gate: Toffoli gate is 3*3 reversible logic gate. The first input is write signal and decoder each output is given as second input for each gate. The third input is a constant 0. The third output gives the ANDed output of the first two inputs. This is fed as an input to each memory cell of a register file. The 1st and 2nd outputs are garbage outputs.

4) Memory Cell: The design of the memory cell is as shown in the Fig. 7 One Fredkin gate and one Feynman gate are used to design one D- latch [4] [6]. Output from the AND gate is connected to the first input of the Fredkin gate. Data to be stored is connected to the second input of this gate. First two output gate are garbage value. The third gate is given to the Feynman gate , to produce feedback Q. This feedback is connected to the third input of the Fredkin gate. There are 8 such D-latches [11] in series to form a register. There are 16 ancilla and 18 garbage values.

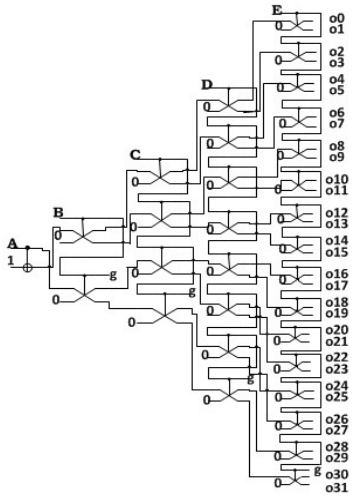

Fig. 6. Decoder [9]

TABLE I
PERFORMANCE PARAMETER OF ALL THE COMPONENTS

PROPERTY	Multiplexer	Decoder	AND Gate	8 bit memory cell.
Garbage values	36	4	2	18
Number of Gates	31	33	1	16
Number of ancilla	0	31	1	16
Quantum cost	124	121	5	55
Quantum delay	25Δ	24Δ	5Δ	15Δ
Quantum depth	5	5	1	3

The number of gates, garbage values , number of ancilla, quantum cost ,quantum delay and depth of each component designed above is tabulated in the Table I.

IV. APPLICATION OF REGISTER FILE IN CONTENT ADDRESSABLE MEMORY

CAM -Content addressable memory is a special type of computer memory used in very high speed searching application .It is also known as associative memory or associative storage.

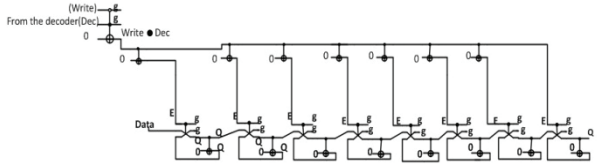

Fig. 7. 8 bit Register Block Diagram

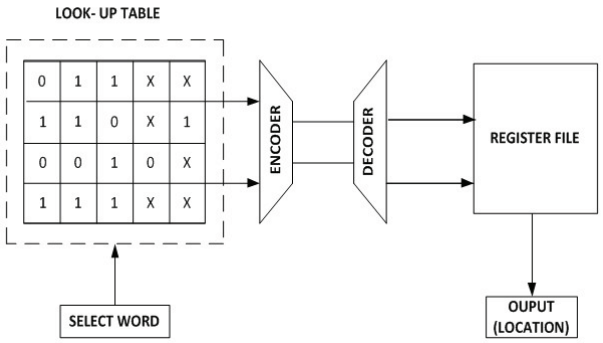

Fig. 8. CAM Block Diagram

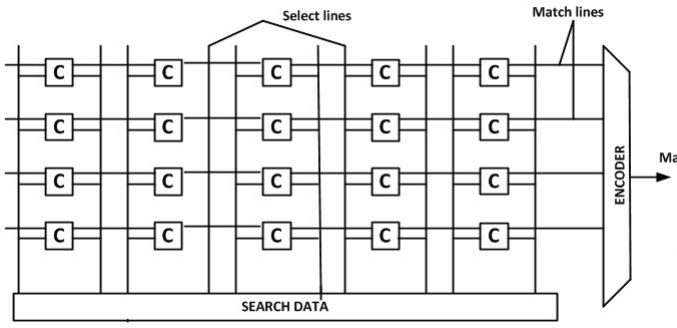

Fig. 9. 5x4 LUT

A. General working:

The design of the CAM is given in the Fig.8 It consist of a Look up table (LUT), encoder, decoder and Register file (memory) [5]. When the user provides a search word (input), its address is searched in the Look Up Table to check whether the search word is stored in it. If the search word is found, the information is sent to the encoder and eventually to the decoder. The decoder, decodes the information and returns the address of word stored in the memory. This decoder output is fed to the register file which gives the location of the search word (output).

The design of each component in detail is presented below: The Look up table consist of the conventional CAM cells which uses NOR gates. Here a 5x4 cell array is arranged(4 words with each containing 5 bit). The match line is selected, if a match is found and encoder gives the address location in the matched line.The lookup table is as shown in Fig 9

B. Design of Encoder and Decoder

The design of the encoder as shown in fig and decoder shown in figure.

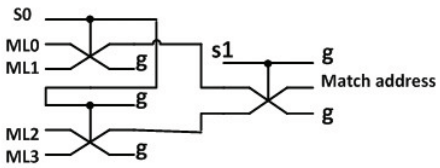

Fig. 10. Encoder used in CAM

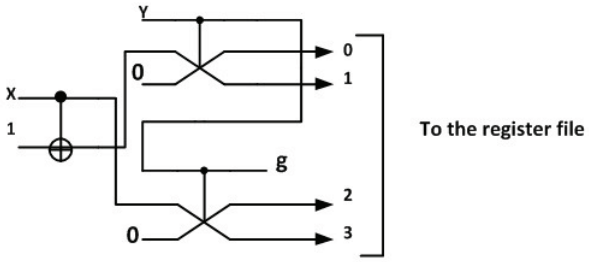

Fig. 11. Decoder used in CAM

ENCODER: The design is shown in Fig 10 uses 3 gates. Its quantum cost is 27. The delay of the circuit is 10Δ, whereas the depth is 2.It doesn't have any ancilla.

DECODER: The circuit is as shown in Fig. 11 It has 3 gates, 1 output is a garbage value. Quantum cost, quantum delay and quantum depth is 22, 9Δ and 2 respectively.

V. Results and Discussion

Table II consolidates the comparative study of existing SRAM memory cell and our proposed memory cell using D-latch with respect to the logic design parameters. Table III gives the % improvement of the proposed design over Existing design.

VI. Conclusion

This paper proposed the complete design of Register File which uses D-Latch as the memory unit. This is advantageous because, it is asynchronous. Once the control signal is sent by the control unit of the processor, the data is written and read into/from the memory without requiring additional clock

TABLE II
Comparitive Overview

Property (1 memory cell)	Proposed	Exisiting I [1]	Existing II [6]
Number of Gates	4	6	3
Quantum Cost	12	19	21
Quantum Delay	15δ	17δ	19δ
Quantum Depth	3	4	5

TABLE III
% IMPROVEMENT

Property (1 memory cell)	% Improvement (compared wrt [1])	%,Improvement t(compared wrt [6])
Quantum Cost	36.8%	42.8%
Quantum Delay	11.7%	21%
Quantum Delay	25%	40%

timing constrain. This register file is used as an application in the design of the CAM.

REFERENCES

[1] A. Nagamani, V. K. Agrawal, R. M. Bhat, N. Shrilakshmi, and V. K. Sonnad, "Design and analysis of esop based online testable reversible sram array," in Advances in Electronics, Computers and Communications (ICAECC), 2014 International Conference on. IEEE, 2014, pp. 1–6.

[2] P. R. Yelekar and S. Sujata, "Introduction to reversible logic gates & its application," in 2nd National Conference on Information and Communication Technology, 2011, pp. 5–9.

[3] A. Malhotra, C. Singh, and A. Singh, "Efficient design of reversible multiplexers with low quantum cost and power consumption," International Journal of Emerging Technology and Advanced Engineering, vol. 4, no. 7, 2014.

[4] H. Thapliyal and N. Ranganathan, "Design of reversible latches optimized for quantum cost, delay and garbage outputs," in VLSI Design, 2010. VLSID'10. 23rd International Conference on. IEEE, 2010, pp. 235–240.

[5] K. Pagiamtzis and A. Sheikholeslami, "Content-addressable memory (cam) circuits and architectures: A tutorial and survey," Solid-State Circuits, IEEE Journal of, vol. 41, no. 3, pp. 712–727, 2006.

[6] M. Morrison, M. Lewandowski, R. Meana, and N. Ranganathan, "Design of static and dynamic ram arrays using a novel reversible logic gate and decoder," in Nanotechnology (IEEE-NANO), 2011 11th IEEE Conference on. IEEE, 2011, pp. 417–420.

[7] M. Mamun, S. Al, I. Mandal, and M. Hasanuzzaman, "Efficient design of reversible sequential circuit," arXiv preprint arXiv:1407.7101, 2014.

[8] A. Majumder, P. L. Singh, N. Mishra, A. J. Mondal, and B. Chowdhury, "A novel delay & quantum cost efficient reversible realization of 2 i× j random access memory," in VLSI Systems, Architecture, Technology and Applications (VLSI-SATA), 2015 International Conference on. IEEE, 2015, pp. 1–6.

[9] A. Majumder, P. Singh, B. Chowdhury, and R. Rai, "Synthesis and realization of n-bit reversible register file used in bus organization of processor architecture," Procedia Computer Science, vol. 57, pp. 305–312, 2015.

[10] M. Morrison, "Theory, synthesis, and application of adiabatic and reversible logic circuits for security applications," in VLSI (ISVLSI), 2014 IEEE Computer Society Annual Symposium on. IEEE, 2014, pp. 252–255.

[11] H. Thapliyal, N. Ranganathan, and S. Kotiyal, "Design of testable reversible sequential circuits," Very Large Scale Integration (VLSI) Systems, IEEE Transactions on, vol. 21, no. 7, pp. 1201–1209, 2013.

[12] Gojko Babić, Introduction to Computer Architecture, Register File Design and Memory Design.(2016). Retrieved from http://http://web.cse.ohiostate.edu/~babic/Cse3421.E.MemoryDesign.02-04-2016.pdf