International Journal of Electrical and Computer Engineering (IJECE)
Vol. 11, No. 2, April 2021, pp. 1476~1484
ISSN: 2088-8708, DOI: 10.11591/ijece.v11i2.pp1476-1484

Design and implementation of dual-core MIPS processor for LU decomposition based on FPGA

Rusul Saad Khalil, Safaa S. Omran
Department of Computer Engineering Techniques, Electrical and Electronic Technical College,
Middle Technical University, Iraq

Article Info

Article history:

Received Mar 26, 2020
Revised Jun 29, 2020
Accepted Jul 11, 2020

Keywords:

Dual core
Field programmable gate array
LU decomposition
MIPS processor
Single core
VHDL

ABSTRACT

Many systems like the control systems and in communication systems, there is usually a demand for matrix inversion solution. This solution requires many operations, which makes it not possible or very hard to meet the needs for real-time constraints. Methods were exists to solve this kind of problems, one of these methods by using the LU decomposition of matrix which is a good alternative to matrix inversion. The LU matrices are two matrices, the L matrix, which is a lower triangular matrix, and the U matrix, which is an upper triangular matrix. In this paper, a design of dual-core processor is used as the hardware of the work and certain software was written to enable the two cores of the dual-core processor to work simultaneously in computing the value of the L matrix and U matrix. The result of this work are compared with other works that using single-core processor, and the results found that the time required in the cores of the dual-core is more less than using single-core. The designed dual-core processor is invoked using the VHDL language.

Corresponding Author:

Rusul Saad Khalil
Department of Computer Engineering Techniques
Electrical and Electronic Technical College
Middle Technical University
Al Doura district, Baghdad, Iraq
Email: asumustapha@gmail.com

1. INTRODUCTION

Many different systems require solving of matrix inversion, these systems like control or communication systems. The required time for solving the matrix inversion increases on the size of the matrix is become bigger. Hence, an alternative method were required in order to work in real-time, one of these methods is the LU decomposition [1].

In LU decomposition method the coefficient matrix [A] of the given system of equation $[A][X] = [B]$ is written as a product of a Lower triangular matrix (L) and an upper triangular matrix (U), such that $[A] = [L][U]$ where the elements of $L = (lij = 0\ for\ i < j)$ and the elements of $U = (uij = 0\ for\ i > j)$ that is, the matrices [L] and [U] look like [2, 3]. Following are set of equations for a 4x4 matrix.

$$[A] = [L][U] \tag{1}$$

$$\begin{bmatrix} A11 & A12 & A13 & A14 \\ A21 & A22 & A23 & A24 \\ A31 & A32 & A33 & A34 \\ A41 & A42 & A43 & A44 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ l21 & 1 & 0 & 0 \\ l31 & l32 & 1 & 0 \\ l41 & l42 & l43 & 1 \end{bmatrix} \begin{bmatrix} u11 & u12 & u13 & u14 \\ 0 & u22 & u23 & u24 \\ 0 & 0 & u33 & u34 \\ 0 & 0 & 0 & u44 \end{bmatrix} \tag{2}$$

where,

$$u11 = A11, u12 = A12, u13 = A13 \; and \; u14 = A14 \quad (3)$$

$$li1=\frac{Ai1}{u11} \text{ where } l21=\frac{A21}{u11}, \; l31=\frac{A31}{u11} \text{ and } l41=\frac{A41}{u11} \quad (4)$$

$$u22 = A22 - l21 \; x \; u12, u23 = A23 - l21 \; x \; u13, and \; u24 = A24 - l21xu14 \quad (5)$$

While

$$A32 = u12xl31 + u22xl32 \quad (6)$$

$$A42 = u12xlL41 + u22xlL42 \quad (7)$$

$$A33 = l31xu13 + l32xu23 + u33 \quad (8)$$

$$A34 = l31xu14 + l32xu24 + u34 \quad (9)$$

$$A43 = l41xu13 + l42xu23 + l43xu33 \quad (10)$$

$$A44 = l41xu14 + l42xu24 + l43xu34 + u44 \quad (11)$$

If one has a system of equations in the form of $[A][X] = [B]$, then the method of using the LU decomposition will make the solution easier by using the triangular matrices. After computing the LU matrices as shown in the next equations [4-7]:

$$[A][X] = [B] \leftrightarrow [L][U][X] = [B] \quad (12)$$

$$[U][X] = [Y] \quad (13)$$

$$[L][Y] = [B] \quad (14)$$

The objective of this paper is to program and build a 32-bit MIPS processor to perform the LU decomposition. Then designing and implementing a dual core MIPS processor, the results will be compared for the two designs system, each system been designed and implemented in VHDL [8-10].

2. MIPS PROCESSOR

It is a reduced instruction set computer (RISC) processor developed by MIPS technologies in the early 1980s which can fully implement instructions in single clock cycle. Therefore the slowest instructions can limit session time. In this paper a single core and dual core MIPS processors will be designed and implemented to perform mathematical requirements for the application of LU decomposition [8].

2.1. MIPS instruction set architecture (ISA)

32-bits MIPS Architecture been covered in this paper where transactions are either register or memory locations as shown in Table 1, Processor, to get to the word uses byte addressable [9, 11, 12].

2.2. Instruction formats

The MIPS has three different formats, which they are the R-type, I-type and J-type. Table 2 shows the different instructions formats for the MIPS processor [13-16].

Table 1. Processor registers

Name	Register number	Usage	Preserved on call?
$zero	0	The constant value 0	n.a.
$v0-$v1	2-3	Values for results and expression evaluation	no
$a0-$a3	4-7	Arguments	no
$t0-$t7	8-15	Temporaries	no
$s0-$s7	16-23	Saved	yes
$t8-$t9	24-25	More Temporaries	no
$gp	28	Global pointer	yes
$sp	29	Stack pointer	yes
$fp	30	Frame pointer	yes
$ra	31	Return address	yes

Table 2. Formats of processor instructions

Field size	6-Bits	5-Bits	5-Bits	5-Bits	5-Bits	6-Bits
Register	opration code	Source register	Target register	regisrer destination	Shift amount	function
Immediate	operation code	Source register	Target register		16-bits Imm	
Jump	opration code			26-bits address		

2.3. Single-core MIPS processor design

The MIPS processor is 32-bits processor which has 32 different registers each with size of 32-bits [17-23]. The main part in the MIPS processor is the control unit (CU). This unit consists of some registers and the arithmetic logic unit (ALU). Certain instructions where required for calculating the LU decomposition were designed and implemented [24-26]. Table 3 shows these different instructions. The design instructions set of the processor is suitable to perform LUD as shown in Table 4. Figure 1 shows the internal architecture of the control unit and Figure 2 shows the schematic design circuits that required in implementing the LU decomposition for single-core processor.

Table 3. Alu control for both processors

Code	Operation
000	Mul
001	Div
010	Add
011	not used
100	not used
101	not used
110	Sub

Table 4. Instruction set

Instructions	SW	LW	ADD	ADDi	SUB	MUL	DIV
Opcode	101011	100011	000000	001000	000000	000000	000000
Regwrite	0	1	1	1	1	1	1
Regdst	0	0	1	0	1	1	1
ALUSRC	1	1	0	1	0	0	0
ZERO	0	0	0	0	0	0	0
MEMWrite	1	0	0	0	0	0	0
MEMtoRegister	0	1	0	0	0	0	0
ALUopcode	00	00	10	00	10	10	10
Function	x	x	100000	x	100010	100100	100101
type	Immediate type	Immediate type	Register type	Immediate type	Register type	Register type	Register type
Description	Save word	Load word	Addition operation	Addition immediate operation	Subtraction operation	Multiplication operation	Division operation

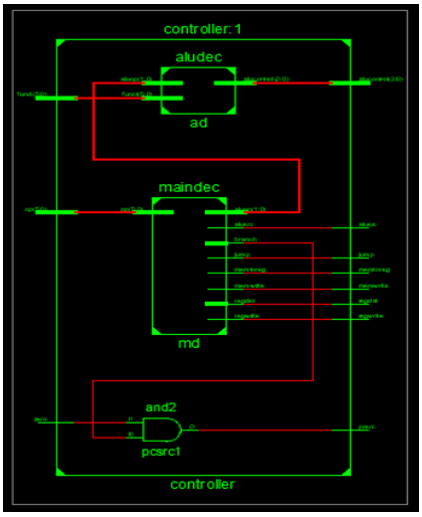

Figure 1. RTL for control unit internal architecture

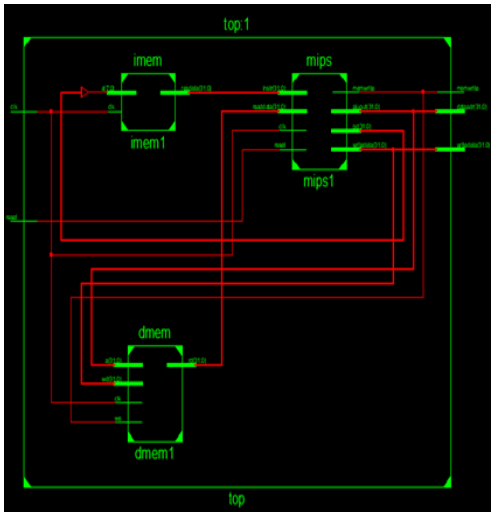

Figure 2. RTL for single core MIPS processor

2.4. Dual-core MIPS processor design

Dual-core consists of two cores and each one is responsible for specific function, both cores shared same data memory. Each core has their own instruction memory, register file and control unit, first core will be used to perform the lower (L) matrix while the second core will perform the upper (U) matrix depending on LU decomposition (factorization) [13, 27]. Figure 3 shows the designed Dual-core MIPS processor, the Lower core is used to compute the (L) matrix while the Upper core is used to compute the (U) matrix, So that, both cores were working simultaneously to compute LU matrices in less time than single-core, which gives a high level of parallelism.

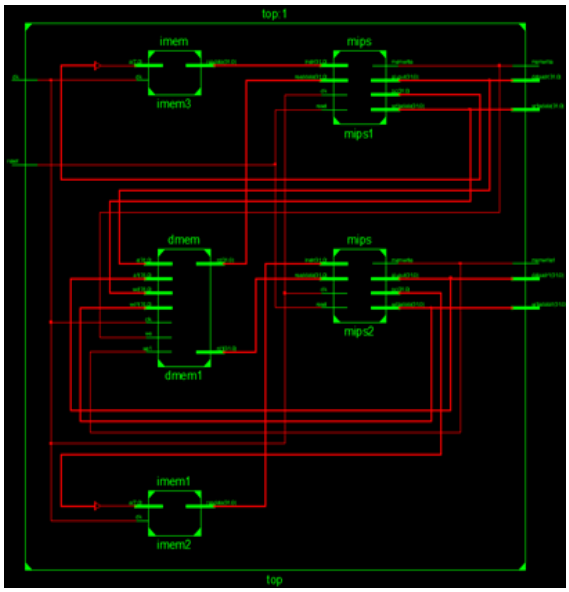

Figure 3. RTL for dual core MIPS processor

3. DATA REPRESENTATION

The fixed-point data representation is chosen in this paper, which is easier in the design consideration. Other method in data representation is floating which is excluded in this work because it requires a very large hardware component [28-30]. Figure 4 shows the format for 32-bits of data.

Sign	Integer value	decimal fraction
1 bit	11 bits	20 bits
31	30 20	19 0

Figure 4. Format for the used data

4. SIMULATION RESULT OF SINGLE-CORE

Single-core processor is implemented using FPGA development board Spartan-6 the simulation results which have been gotten from the Xilinx ISim simulator. Executing a set of instructions to compute LUD, both matrix and LUD is shown in (15) for a 4x4 matrix which also can lead into a 6x6 matrix, the time required to perform LU decomposition is 3070 ns (3.07 µs) at frequency 50 MHz. The results are found identical to the theoretical results when applied for the 4x4 matrix. Figure 5 and Figure 6 show the test-bench of waveform simulation for matrix A and it's LUD and Figure 7 shows the resources needed for the excuted design.

$$A = \begin{bmatrix} 2 & 3 & 1 & 5 \\ 6 & 13 & 5 & 19 \\ 2 & 19 & 10 & 23 \\ 4 & 10 & 11 & 31 \end{bmatrix} = L \begin{bmatrix} 1 & 0 & 0 & 0 \\ 3 & 1 & 0 & 0 \\ 1 & 4 & 1 & 0 \\ 2 & 1 & 7 & 1 \end{bmatrix} U \begin{bmatrix} 2 & 3 & 1 & 5 \\ 0 & 4 & 2 & 4 \\ 0 & 0 & 1 & 2 \\ 0 & 0 & 0 & 3 \end{bmatrix} \quad (15)$$

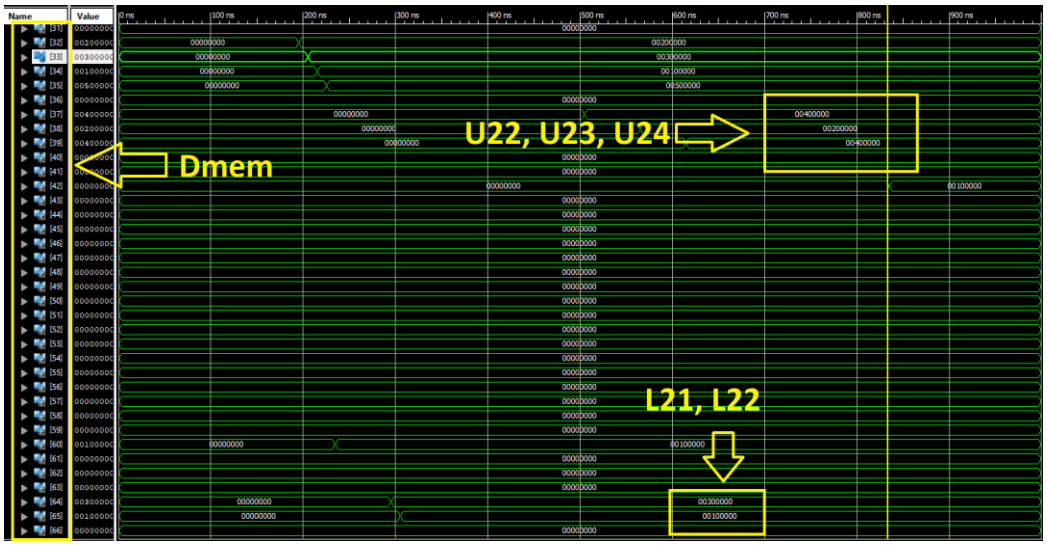

Figure 5. Single-core processor test bench of data memory

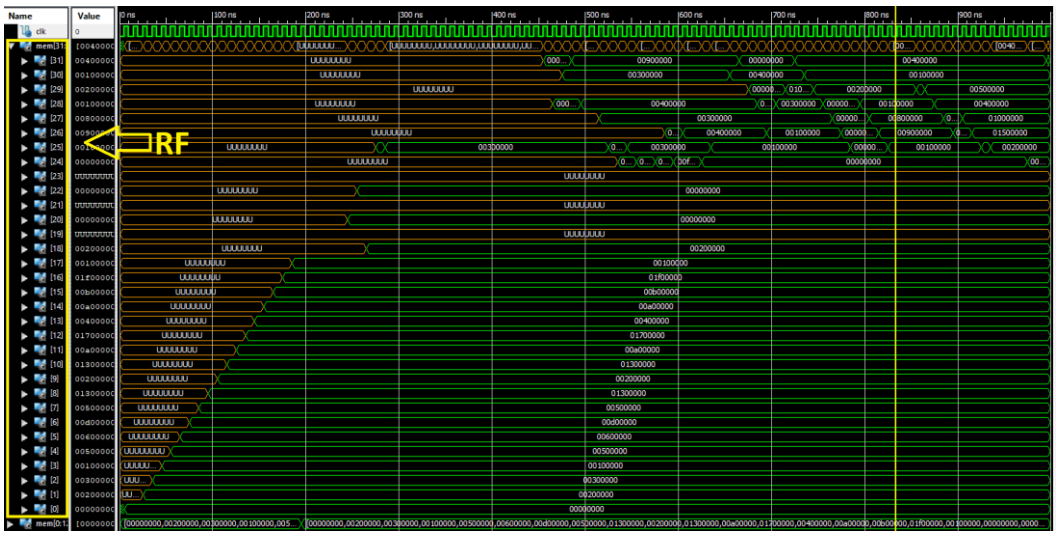

Figure 6. Single-core processor test bench of register file

Device Utilization Summary (estimated values)			[-]
Logic Utilization	**Used**	**Available**	**Utilization**
Number of Slice Registers	7	54576	0%
Number of Slice LUTs	5014	27288	18%
Number of fully used LUT-FF pairs	0	5021	0%
Number of bonded IOBs	67	296	22%
Number of BUFG/BUFGCTRLs	1	16	6%
Number of DSP48A1s	4	58	6%

Figure 7. The FPGA resources of single processor

5. SIMULATION RESULT OF DUAL-CORE

The proposed design of dual core processor has been coded by using VHDL, XILINX Spartan 6 with sets of instructions that compute LU decomposition, a testbench was created to implement same 4x4

matrix as shown in Figure 8, Figure 9 and Figure 10 with resource required as shown in Figure 11. The time required to perform L decomposition in dual core processor is 850 ns (0.85 µs) at frequency 50 MHz with number of instruction 41. As shown in Table 5, and the time required to perform U decomposition is 1170 ns (1.17 µs) for the same frequency with 57 instructions that has been used as shown in Figure 12.

Table 5. Single core and dual core comparisons

Processor	Time (ns)	Instructions used	Clock period (ns)
Single Core	3070	142	20
Dual Core (First core)	850	41	20
Dual Core (Second core)	1170	57	20

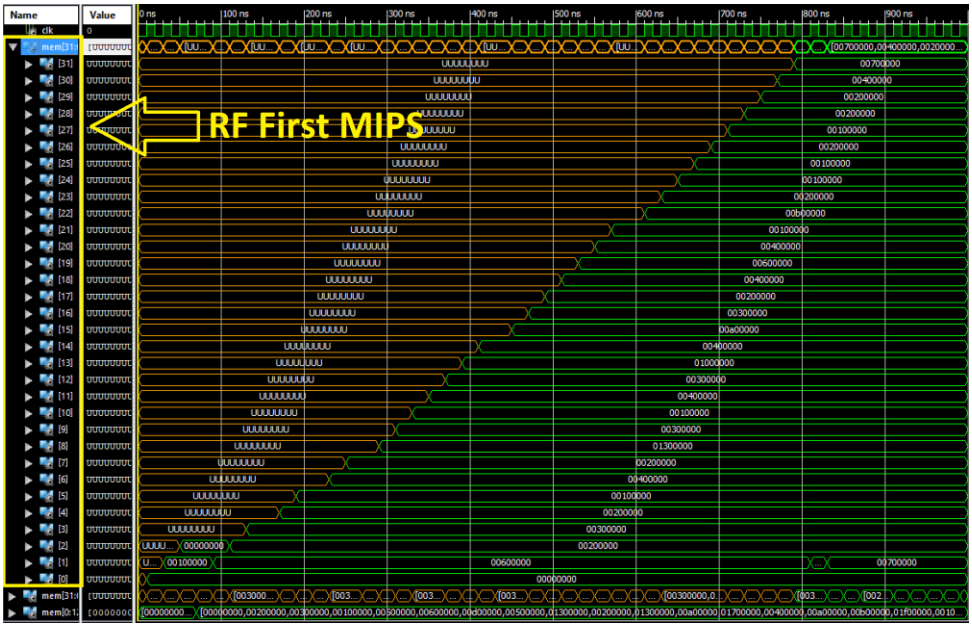

Figure 8. Dual core processor test bench of register file 1

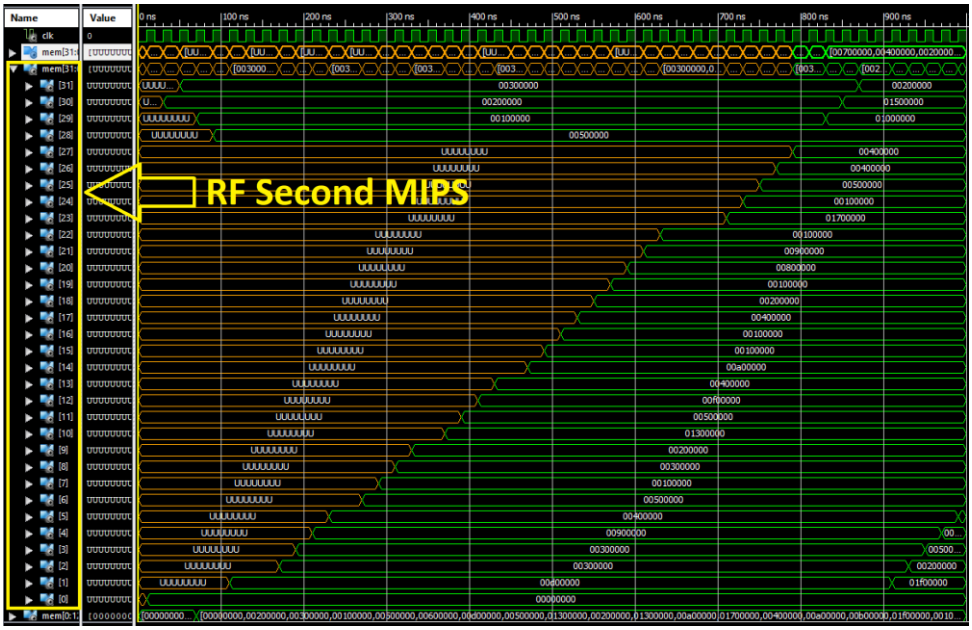

Figure 9. Dual core processor test bench of register file 2

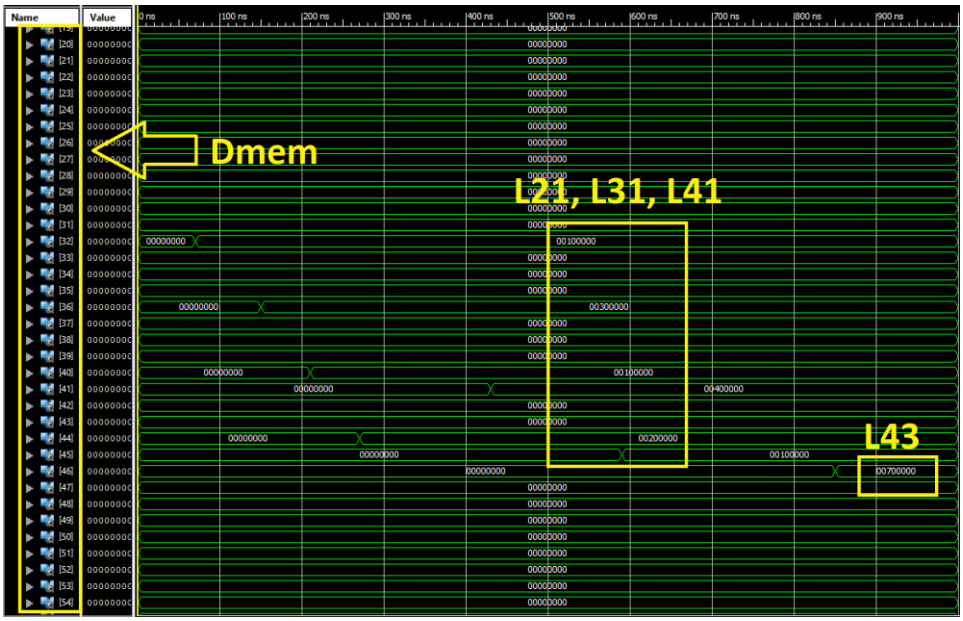

Figure 10. Dual core processor test bench of data memory

Device Utilization Summary (estimated values)			[-]
Logic Utilization	**Used**	**Available**	**Utilization**
Number of Slice Registers	4104	54576	7%
Number of Slice LUTs	17323	27288	63%
Number of fully used LUT-FF pairs	4096	17331	23%
Number of bonded IOBs	132	296	44%
Number of BUFG/BUFGCTRLs	1	16	6%
Number of DSP48A1s	8	58	13%

Figure 11. Summarize the FPGA resource of dual processor

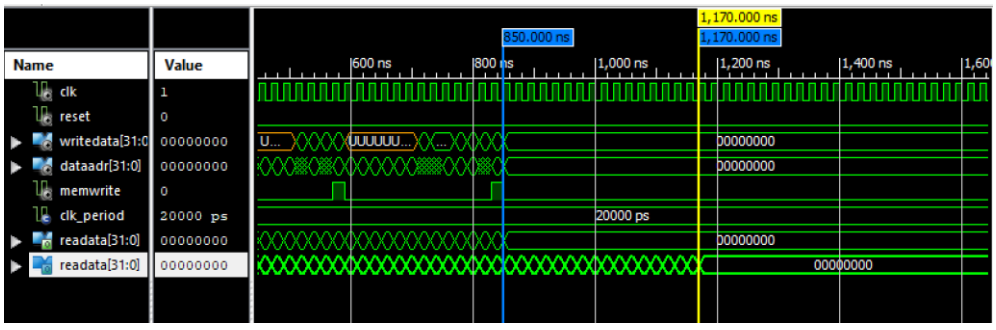

Figure 12. Simulation of LU decomposition using dual processor

6. CONCLUSION

A single core and dual core were designed to perform LU 4x4 matrix calculation for the purpose of teaching studies of the MIPS architecture course for master student. Designing and implementing single core and dual core processors with the required instructions for each processer sufficient to implement the LU decomposition using decomposition process. The time of single core processer to perform the LU 4x4 matrices was 3.07 µs at frequency 50 MHz while designing dual core processor where the first core of the processor used to compute the L matrix and the second core of the processor used to compute U matrix. This design can achieve high performance with timing of 1.17 µs. The most consuming processor is the Dual core processor. However, it gives higher performance.

REFERENCES

[1] G. H. Golub and C. F. V. Loan, "Matrix Commutations," *4th ed., Johns Hopkins University Press*, 2013.
[2] A. Yang, C. Liu, J. Chang, X. Guo, "Research on Parallel LU Decomposition Method and It's Application in Circle Transportation," *Journal of software*, vol. 5, no. 11, pp. 1250-1255, 2010.
[3] T. Tiruneh, Tesfamariam Y. Debessai, G. C. Bwembya, S. J. Nkambule, "The LA=U decomposition method for solving systems of linear equations," *Journal of Applied Mathematics and Physics*, vol. 7, no. 9, pp. 2031-2051, 2019.
[4] X. Wang, and S. G. Ziavras, "Parallel LU Factorization of sparse matrices on FBGA based configurable computing engines," *Concurrency and Computation Practice and Experience*, vol. 16, no. 4, pp. 319-343, 2004.
[5] Y. wang, H. tao, S. xiao, H. Dai, "An implementation architecture design of LU decomposition in resource-limited system," *2015 IEEE International Symposium on Systems Engineering (ISSE)*, Rome, 2015, pp. 261-265.
[6] Y. Shao, L. Jiang, Q. Zhao, Y. Wang, "High Performance and Parallel Modle forLU decomposition on FPGAs," *2009 Fourth International Conference on Frontier of Computer Science and Technology*, Shanghai, 2009, pp. 75-79.
[7] A. A. Hussain, N. Tayem, M. O. Butt, A. Soliman, A. Alhamed, S. Alshebeili, "FPGA Hardware Implementation of DOA Estimation Algorithm Employing LU decomposition," *IEEE Access*, vol. 6, pp. 17666-17680, 2018.
[8] M. Mounika and A. Shankar, "Design & implementation of 32-bit Risc (MIPS) processor," *International Journal of Engineering Trends and Technology (IJETT)*, vol. 4, no. 10, pp. 4466-4474, 2013.
[9] V. Robio and J. Cook, "A FPGA Implementation of A MIPS RISC Processor for Computer Architecture Education," New Mexico State University, MSc. Thesis, 2004.
[10] V. R. Wadhankar and V. Tehre, "A FPGA Implementation of a RISC Processor for Computer Architecture," *National Conference on Innovative Paradigms in Engineering & Technology*, Nagpur, India, 2012, pp. 24-28.
[11] M. N. Topiwala, N. Saraswathi, "Implemantation of a 32-bit MIPS based RISC processor using cadence," in *2014 IEEE International Conference on Advanced Communications, Control and Computing Technologies*, Ramanathapuram, 2014, pp. 979-983.
[12] R. S. Balpande, R. S. Keote, "Design of FPGA based Instruction Fetch & Decode Module of 32-bit RISC (MIPS) Processor," *2011 International Conference on Communication Systems and Network Technologies*, Katra, Jammu, 2011, pp. 409-413.
[13] J. L. Hennessy, J. Norman, S. Przybylski, C. Rowen, T. Gross, F. Baskett, J. Gill, "Mips: A microprocessor architecture," *IEEE Press, ACM SIGMICRO Newsletter*, vol. 13, pp. 17-22, 1982.
[14] V. N. Sireesha, D. Santosh, "FPGA Implementation of A MIPS RISC Processor," I*nternational journal Computer Technology & Applications*, vol. 3, no. 3, pp. 1251-1253, 2012.
[15] M. B. Ibne Reaz, M. S. Islam, M. S. Sulaiman, "A Single clock cycle MIPS RISC processor design using VHDL," *ICONIP '02. Proceedings of the 9th International Conference on Neural Information Processing. Computational Intelligence for the E-Age (IEEE Cat. No.02EX575)*, Penang, Malaysia, 2002, pp. 199-203.
[16] H. S. Mehta, "Design of MIPS processor," California State University, MSc. Thesis, 2012
[17] S. S. Omran, A. J. Ibada, "FPGA Implementation of MIPS RISC Processor for Educational Purposes," *Journal of Babylon University and Applied Sciences*, vol. 24, no. 7, pp. 1745-1761, 2016.
[18] S.S. Omran and L. F. Jumma, "Design of SHA-1 and SHA-2 MIPS processor using FPGA," *2017 annual conference on new trends in information and communication technolongy applications (NTICT)*, Baghdad, Iraq, 2017, pp. 268-273.
[19] B. C. Alecsa and A. D. Ioan, "FPGA Implementation of a Matrix Structure for Integer Division," *2010 3rd International Symposium on Electrical and Electronics Engineering (ISEEE)*, Galati, 2010, pp. 238 - 243.
[20] S. Aslan, E. Oruklu, J. Saniie, "Architecture design Tool for Low Area Band Matrix LU Factorization," *2011 IEEE International Conference on Electro/Information Technology*, Mankato, MN, 2011, pp. 1-6.
[21] R. Srinidhi, "MIPS Processor Implementation," California State University Northridge, MSc. Thesis, 2012.
[22] M. B. Ibne Reaz, "Single Core Hardware Modeling of 32-bit MIPS RISC Processor with A Single Clock," *Research Journal of Applied Sciences, Engineering and Technology*, vol. 4, no. 7, pp. 825-832, 2012.
[23] K. Bhattacharyya, R. Biswas, A. S. Dhar, S. Banerjee, "Architectural design and FPGA implementation of radix-4 CORDIC processor," *Microprocessors and Microsystems*, vol. 34, no. 2-4, pp. 96-101, 2010.
[24] J. L. Hennessy, D. A. Patterson, "Computer Organization and Design: The Hardware/Software Interface," *Morgan Kaufmann, 4th ed.*, Waltham, 2012.
[25] M. N. Thakare, S. P. Ritpurkar, "Design and simulation of 32-Bit RISC architecture based on MIPS using VHDL," *2015 International Conference on Advanced Computing and Communication Systems*, Coimbatore, 2015, pp. 1-6.
[26] R. Anjana and G. Krunal, "VHDL Implementation of a MIPS RISC Processor," *International Journal of Advanced Research in Computer Science and software Engineering*, vol. 2, pp. 83-88, 2012.
[27] M. Herlihy and N. Shavit, "The Art of Multiprocessor Programming," *1st ed., Morgan Kaufmann*, 2008.
[28] C. K. Singh, S. H. Prasad, P. T. Balsara, "A fixed-point implementation for QR decomposition," *2006 IEEE Dallas/CAS Workshop on Design, Applications, Integration and Software*, Richardson, TX, 2006, pp. 75-78.
[29] M. Eljammaly, Y. Hanafy, A. Wahdan, A. Bayoumi, "Hardware Implementation of LU decomposition Using dataflow architecture on FPGA," 20*13 5th International Conference on Computer Science and Information Technology*, Amman, 2013, pp. 298-302.
[30] S. Gao, D. Al-Khalili, J. M. Pierre Langlois, N. Chabini, "Decimal Floating-Point Multiplier with Binary-Decimal Compression Based Fixed-Point Multiplier," *2017 IEEE 30th Canadian Conference on Electrical and Computer Engineering (CCECE)*, Windsor, ON, 2017, pp. 1-6.

BIOGRAPHIES OF AUTHORS

Rusul Saad Khalil was born in Baghdad, Iraq in 1992. She graduated from AlMamon University college in 2014, and now studying master at Electrical Engineering College/Middle Technical University, Baghdad, Iraq, her main interest in Computer Architecture Design, Computer engineering, embedded system and design.

Safaa S. Omran was born in Baghdad, Iraq in 1956. He graduated from University of Baghdad in 1978, and then he got the MSc from the same University in 1984. He is now a professor working at the Electrical Engineering College/Middle Technical University, Baghdad, Iraq. His main interest working researches are in the field of microprocessor design for embedded systems, Image processing and cryptography system design.