Fast QR Decomposition Based on FPGA

Safaa S. Omran,
College of Electrical and Electronic Technique
Middle Technical University
Baghdad, Iraq
omran_safaa@ymail.com

Ahmed K. Abdul-abbas
College of Electrical and Electronic Technique
Middle Technical University
Baghdad, Iraq
ahmed89kareem@alkafeeluc.edu.iq

***Abstract*—The QR-decomposition (QRD) is an implementation necessary for many different detection algorithms such as MIMO (Multiple Input and Multiple Output) in wireless communication system. In this article, a QRD processor which decomposes the matrix into an orthogonal (Q matrix) and upper triangular matrix (R matrix) using Gram Schmidt algorithm is designed and implemented using a 32-bit High speed processor based on FPGA. This design requires 16 clock cycle to compute QR decomposition with 15.625 M QRDs per second throughput at 250 MHz operating frequency.**

Keywords—32-Bits processor; Gram-Schmidt; Verilog HDL, QR decomposition; CORDIC square root.

I. INTRODUCTION

With the increasing use of wireless communication systems the reliability requirements for high throughput are becoming more critical. In the current contract, multiple input and multiple output systems have generated enormous research interest as they offer high reliability and high throughput [1].

If A be an m × n matrix with full column ranks. The QR decomposition of A is A = QR, where Q is an m × m orthogonal matrix and R is an m × n upper triangular matrix [2] [3].

Usually, the Gram Schmidt algorithm, householder transformation, and givens rotations are known as the basic algorithms for QRD . The householder transformation and givens rotations illustrate the feature of numerical stability while the Gram Schmidt provides an occasion to perform successive orthogonalization. In this article the Gram Schmidt algorithm is used to decompose the matrix (A).

Many researchers were interested in QR decomposition based on field programmable gate array (FPGA). P. Luethi [4] proposed a VLSI architecture processor to detect multi-input multi-output (MIMO) up to 1.56 million complex valued of 4×4 dimensional matrices per second. Robert [1] proposed iterative decomposition architecture based on GS algorithm. Ji-Hwan Yoon [5] proposed an architecture to perform the QR decomposition using Givens Rotation method. While in [6] proposed another work to compute QRD, they developed the architecture for QRD based on CORDIC (Coordinate-Rotation Digital-Computer algorithm) and fixed point operations, the FPGA platforms was used in this work. And proposed in [7] a parallel architecture of an QRD systolic array based on the Givens rotation algorithm on FPGA, the direct mapping by 21-fixed point CORDIC was used in this architecture that can compute the QRD for a 4×4 real matrices.

The scope of this paper is on application of QRD based on Gram Schmidt method using fast processor to achieve high speed and low latency with high throughput. This fast processor is designed especially for this purpose and implemented using Verilog HDL language based on Virtex – 7 FPGA development board.

The arrangement of this article is as follows: Section two present an overview discussion on MIMO system model with QRD by using Gram Schmidt algorithm and a brief for mathematic procedure. Section three contains an explanation for the design of 32-bits High speed processor. Section four presents the Instruction Set and data presentation of this design. In section five, results, throughput and timing are discussed. Finally, the conclusion given in the last section.

II. OVERVIEW

In this section an explanation of MIMO systems model and QR decomposition using Gram Schmidt algorithm will be given.

A. MIMO system

In wireless communication systems, MIMO is a way to multiplying the space of a communication channel by using multiple receive and transmit antennas to utilize multipath propagation [8] as shown in Fig.1. MIMO becomes a main element of wireless communication standards including "IEEE 802.11n (Wi-Fi), IEEE 802.11ac (Wi-Fi), HSPA+ (3G), WiMAX (4G), and Long Term Evolution (LTE 4G)".

A MIMO system is considered with M_t transmit and M_r receive antennas. The $M_t \times M_r$ refer to dimension of matrix H represents the MIMO channels, the M_t transmit wave vector is denoted by $s = [s_1, s_2, s_3, s_4, \dots\dots, sM_t]^T$, and the M_r vector n represents the additive zero mean complex Gaussian noise with difference σ_2^n per complex dimension. The energy of the transmitted symbol vector is normalized like equation (1), where I_{Mt} is the $M_t \times M_r$ dimension of identity matrix. The MR-dimensional receive vector $y = [y_1, y_2, y_3, y_4, \dots, yM_r]^T$, corresponds to equation (2). The signal to noise ratio per receives antenna is $\frac{I_{Mt}}{\sigma_2^n}$. MIMO detection is done using QR decomposition, where it begins by decomposing H into an Q matrix (orthogonal) and an R matrix (upper triangular) with real valued non-negative elements on the main diagonal.

$$E[ss^H] = I_{Mt} \quad (1)$$

$$y = Hs + n. \quad (2)$$

978-1-5386-6696-8/18/$31.00 ©2018 IEEE

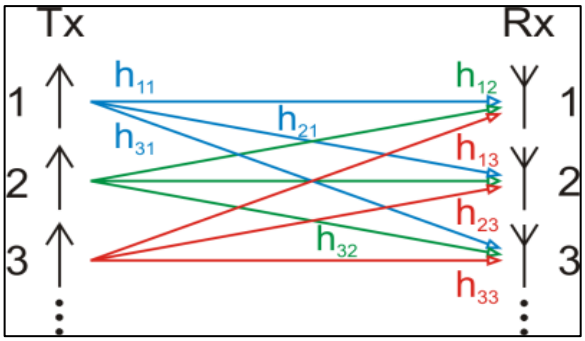

Fig. 1. Multipath propagation.

B. Gram-Schmidt algorithm

A matrix A can be decomposed into the product of an Q orthogonal matrix and an R upper triangular matrix as shown in equation (3). Fig.2 illustrates the GS algorithm [9].

$$A = QR, \text{ where } (QQ^T = I, Q^T = Q^{-1}). \quad (3)$$

A = QR, often used to solve **Ax = b**, where **x** is unknown
Find **d** = Q^T**b**, then **Rx = d** can be solved by back substitution
A = [a_1, a_2, a_3,…a_n] where a_i are column vectors

scaler $proj_v(a)$ = <v,a_1^T>/ <v*v^T> where <a,b^T> is dot product

$u_1 = e_1$ and $e_1 = u_1$ / norm (u_1)
$u_2 = a_2 - proj_{u1}(a_2)$ and $e_2 = u_2$ / norm (u_2)
$u_3 = a_3 - proj_{u1}(a_3) - proj_{u2}(a_3)$ and $e_3 = u_3$ / norm (u_3)

Q = [e_1, e_2, e_3,…e_n] where e_i are orthonormal column vectors

$$R = \begin{matrix} \langle e_1,a_1\rangle & \langle e_1,a_2\rangle & \langle e_1,a_3\rangle\ldots & \langle e_1,a_n\rangle \\ 0 & \langle e_2,a_2\rangle & \langle e_2,a_3\rangle\ldots & \langle e_2,a_n\rangle \\ 0 & 0 & \langle e_3,a_3\rangle\ldots & \langle e_3,a_n\rangle \\ 0 & 0 & 0\ldots & \langle e_4,a_n\rangle \\ \cdot & \cdot & \cdot \quad \cdot & \cdot \\ 0 & 0 & 0 \quad 0 & \langle e_n,a_n\rangle \end{matrix} \quad \text{(upper triangular)}$$

Fig. 2. GS algorithm.

III. Processor Design

In this work, new architecture has been designed to perform QR decomposition using Gram Schmidt algorithm to achieve high speed execution and to increase the throughput. A 32-bit architecture is used in this design for high accuracy and decreases the percent of error in arithmetic operations where the fixed point method is used as system of numbers. As shown in Fig.3, this design consists of Register set, instruction pointer, instruction memory unit, data memory unit, control unit, stage 1, stage 2, E-reg unit and R-unit. Following a discussion and details for each of units.

Register Set, is an internal memory, which consists of a set of storage locations, where each location consists of 32 bits (32 flip flops), the design consists of 32 registers. Instruction Pointer unit (IP), is a 32-bit register used to store 32-bits address of memory location from which an instruction to be fetched. In single cycle processor the memory divided into two parts instruction and data, the first part to store the instructions and the second part to store the data, this arrangement enables programmer to read instruction and data at the same time or in the same cycle, this type of design is known as Harvard memory architecture [10]. In this processor the memory is designed as four way interleaving Harvard memory. Control unit is responsible for controlling the data flow and an interface between the units. The proposed control unit is the largest unit. The control unit is designed to produce six control signals (regwrite, qen, lq1, lq2, lq3 and eout) details of these signals are shown in Table I.

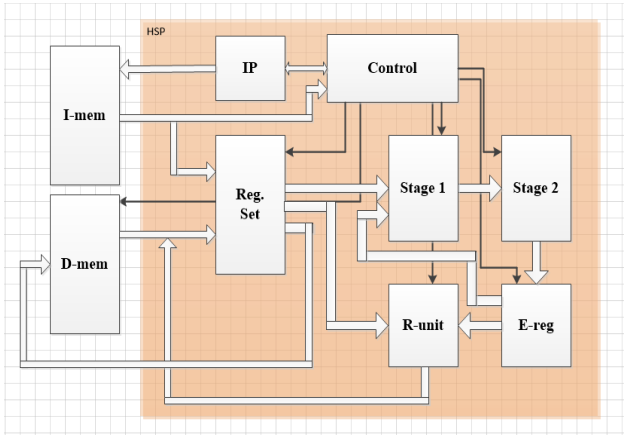

Fig. 3. Block diagram of the designed processor.

TABLE I. CONTROL SIGNALS.

sort	Signal name	description	bits
1	regwrite	Writing into register (active high).	1
2	qen	Write into E unit	1
3	lq1	Enable latch one	1
4	lq2	Enable latch two	1
5	lq3	Enable latch three	1
6	eout	Enable output port	1

The core of this design is stage 1 unit and stage 2 unit. Where, these two units are used to compute Q and R matrices using Gram Schmidt algorithm, u1 is computed by using stage 1, while, e1 is computed by using stage 2. Following details for each one.

As shown in Fig.4, stage 1 unit consists of three parts to compute:

$$u_n = a_n - proj_{u_{n-1}}(a_n) \quad (4)$$

Where

$$proj_{u_{n-1}}(a_n) = (a_n * e_{n-1})\, e_{n-1} \quad (5)$$

This processor is designed for matrix A (4 x 4) elements, so, equation (4) can be rewritten as in equation (6) to become suitable for each column.

$$u_n = a_n - (a_n * e_1)\, e_1 - (a_n * e_2)\, e_2 - (a_n * e_3)\, e_3 \quad (6)$$

Where the initial values for each e_n equal to zero. The stage 1 unit is designed depending on equation (6). As shown in Fig.4 each part refers to "proj", so, the parts of stage 1 unit consist of eight multipliers and three adders connected as parallel, then, the subtraction operation is performed between a_n and them (part 1, part 2 and part 3). From this unit u vector (4 x 1) will be produced and then sends it to the next stage unit under control signals, Fig.5 shows the block diagram of part section.

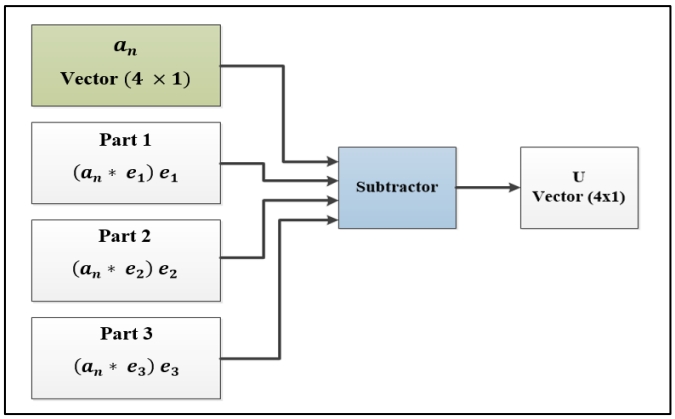

Fig. 4. Block diagram of Stage 1.

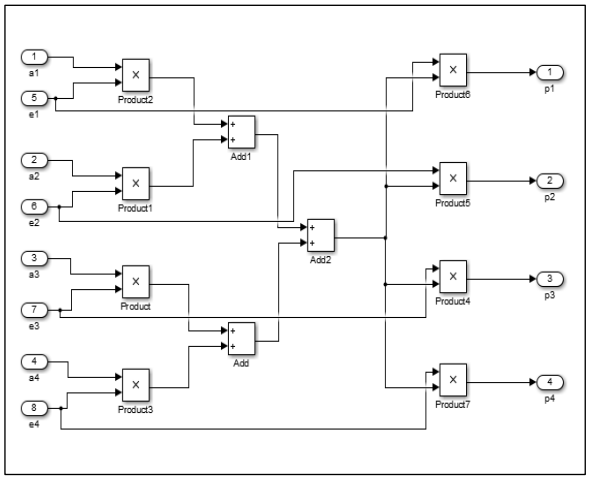

Fig. 5. Block diagram of part section.

Stage 2 unit is responsible for computing:

$$e_n = \frac{u_n}{norm\,(u_n)} \tag{7}$$

$$\text{Where } norm\,(u_n) = \sqrt{\sum u^2} \tag{8}$$

Fig.6 shows the internal architecture for the stage 2 unit. As shown in above equations (7) and (8) the square root operation and division operation are needed to perform this equation. **C**oordinate **R**otation **Di**gital **C**omputer algorithm (CORDIC) is used to calculate a square root operation. This algorithm depends on shift-add operations. To increase the speed of execution, multiplier is used. Fig.7 shows an example for this method to compute a square root for number (x) which consists of 16-bits. 16 iterations are needed to compute the square root for number consists of 32-bits. So, to decrease the latency this stage is designed to work in parallel. Fig.8 shows the block diagram for this design.

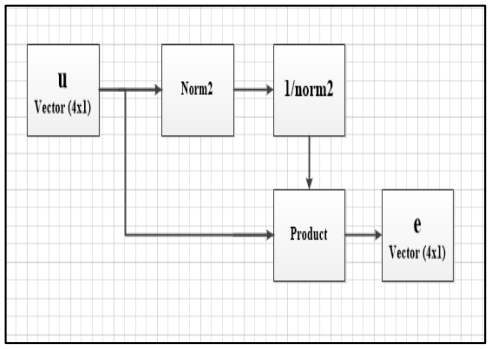

Fig. 6. Block diagram of Stage 2.

L	2^L	y	x= 12056	
		0	initial value	
7	128	0	128 x 128 > 12056	do nothing
6	64	64	64 x 64 < 12056	add 64 to $y_{initial}$ --> 64
5	32	96	$(64 + 32)^2$ < 12056	add 32 to last y --> 96
4	16	96	$(96 + 16)^2$ > 12056	do nothing
3	8	104	$(96 + 8)^2$ < 12056	add 8 to last y --> 104
2	4	108	$(104 + 4)^2$ < 12056	add 4 to last y --> 108
1	2	108	$(108 + 2)^2$ > 12056	do nothing
0	1	109	$(108 + 1)^2$ < 12056	add 1 to last y --> 109
-1	0.5	a.s.o.	and so on	and so on

Fig.7. Example about CORDIC algorithm.

Where: X equal to 12056 and the square root is 109.
L equal to N-bits / 2.
Y is square root value.

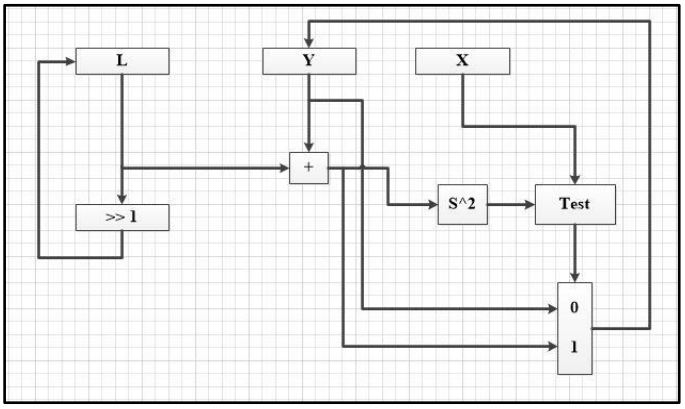

Fig. 8. Block diagram for CORDIC.

Where:

- *Test has a value of "1" if [S^2 < X], and "0" otherwise.*
- *All values are unsigned.*
- *Initial value of L = ($\frac{N}{2} - 1$), where N is a length of X.*
- *Initial value of Y = 0.*

The DSP divider is used to perform a division operation. This technique is used to achieve low latency. This type of divider supports only in top series of FPGA boards as Virtex-7 and Spartan-6 [11].

E-reg unit is a set of registers (16 registers) each one consists of 32-bits used to store the orthogonal matrix Q (Q = e). This unit directly connected with stage 2 unit and R-unit to reduce the time for data transfer between them. While R-unit is responsible on calculating upper triangular matrix R, it consists of four multipliers and three adders to compute one element of R matrix each time. This unit designed to work as parallel with the other units to reduce time of execution and increasing the throughput.

IV. Instruction Set and Data Presentation

A 32-bit architecture is used to design the proposed processor. So, 32-bits format instruction is used to control the work of processor. Fig.9 explains the format instruction; it is clear from this figure that the format of the instruction consists of six parts. Opcode part consists of 6-bits, used to select the type of instruction format, Source register (Rs) and Destination register (Rd) are of 5-bits each which are used for storage of data, Rq, Ra and Rp are of 5-bits used to select the columns of matrices that required in process.

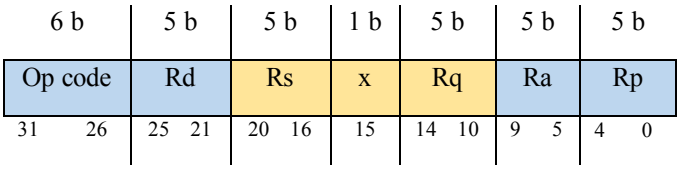

Fig. 9. Instruction format.

Finally, there are two different types of representation to the real numbers: fixed point and floating point numeration systems. The fixed-point arithmetic is an extension of the integer representation which allows to represent relatively reduced range of numbers with a constant absolute precision as well as a low complexity in implementation. So, it is used in this design with a word length equal to 32 bits divided to 20 bits as fraction point and 11 bits as integer part and last bit as sign number. Fig.10 shows the fixed point format that used in this design.

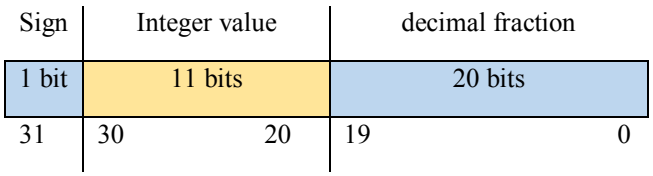

Fig.10. 32 bits fixed point format.

V. Results of Implementation

The proposed design of High speed 32-bits processor has been written using Verilog HDL (Hardware Description Language) and implemented using Virtex- 7 FPGA board. 32-bits fixedpoint number representation has been used. A testbench has been created for testing the results of QRD with Matlab program, the result of simulated behavior model using ISE Design Suite 14.7 program shows that the total latency is 16 clock cycles. Matrix A has been given from Matlab program as test matrix for this design:

$$A = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 \end{bmatrix}$$

The results as computed using Matlab program using floatingpoint representation are:

$$Q = \begin{bmatrix} 0.5774 & 0.2582 & 0.1690 & -0.7559 \\ 0.5774 & 0.2582 & -0.6761 & 0.3780 \\ 0.5774 & -0.5164 & 0.5071 & 0.3780 \\ 0 & 0.7746 & 0.5071 & 0.3780 \end{bmatrix}$$

$$R = \begin{bmatrix} 1.7321 & 1.1547 & 1.1547 & 1.1547 \\ 0 & 1.2910 & 0.5164 & 0.5164 \\ 0 & 0 & 1.1832 & 0.3381 \\ 0 & 0 & 0 & 1.1339 \end{bmatrix}$$

While the results as computed by ISE Design Suite 14.7 program using 32-bits fixed-point representation are:

$$Q = \begin{bmatrix} 0.5774 & 0.2582 & 0.1690 & -0.7560 \\ 0.5774 & 0.2582 & -0.6762 & 0.3779 \\ 0.5774 & -0.5166 & 0.5071 & 0.3780 \\ 0 & 0.7747 & 0.5071 & 0.3780 \end{bmatrix}$$

$$R = \begin{bmatrix} 1.7322 & 1.1548 & 1.1548 & 1.1548 \\ 0 & 1.2911 & 0.5163 & 0.5163 \\ 0 & 0 & 1.1832 & 0.3380 \\ 0 & 0 & 0 & 1.1339 \end{bmatrix}$$

Fig.11 shows a screen shot of the resources utilization for this QRD processer designed using Xilinx Virtex-7, while Fig.12 shows the screen shot of simulation for Q and R matrices by ISE Design Suite 14.7 program. The time required to find the QR decomposition of matrix A [4 × 4], is 64 ns (16 clock cycles) which is clear in Fig.11. This time for QR process is (0.064 µ sec), which means the throughput of this design is 15.625 M QRDs per second

Table II shows a comparison between the Fast QRD processor with other works. This comparison is based on data word length, type of algorithm (Gram Schmidt (GS), Modified Gram Schmidt (MGS) and Givens Rotation (GR)), latency and throughput. It is clear from this table that this design has the lower latency from other works and has a higher throughput. These results achieved because the high parallelism of the designed Fast QRD processor.

VI. Conclusion

In this article, a high-speed architecture of processor is presented to find QR decomposition based on Gram Schmidt algorithm. The clock cycles required for finding the QR decomposition was 16 clock cycles with 15.625 M QRDs per second throughput at 250 MHz. The proposed design was implemented using virtex-7 FPGA board.

TABLE II: GENERAL COMPARISION.

Items	**Works**			
	[12]	[5]	[13]	**This design**
Data word	12 bits	-	14 bits	32 bits
Algorithm	GR	GS	MGS	GS
Frequency	128 MHz	179 MHz	-	250 MHz
Latency (clock cycles)	137	40	35	16
Throughput (QRD/s)	0.934 M	4.48 M	11.42 M	15.625 M
FPGA board or CMOS	Virtex-6	0.13 µm	0.18 µm	Virtex-7

Device Utilization Summary (estimated values)			[-]
Logic Utilization	**Used**	**Available**	**Utilization**
Number of Slice Registers	558	607200	0%
Number of Slice LUTs	4299	303600	1%
Number of fully used LUT-FF pairs	551	4306	12%
Number of bonded IOBs	178	700	25%
Number of BUFG/BUFGCTRLs	1	32	3%
Number of DSP48E1s	192	2800	6%

Fig. 11. Resource utilization.

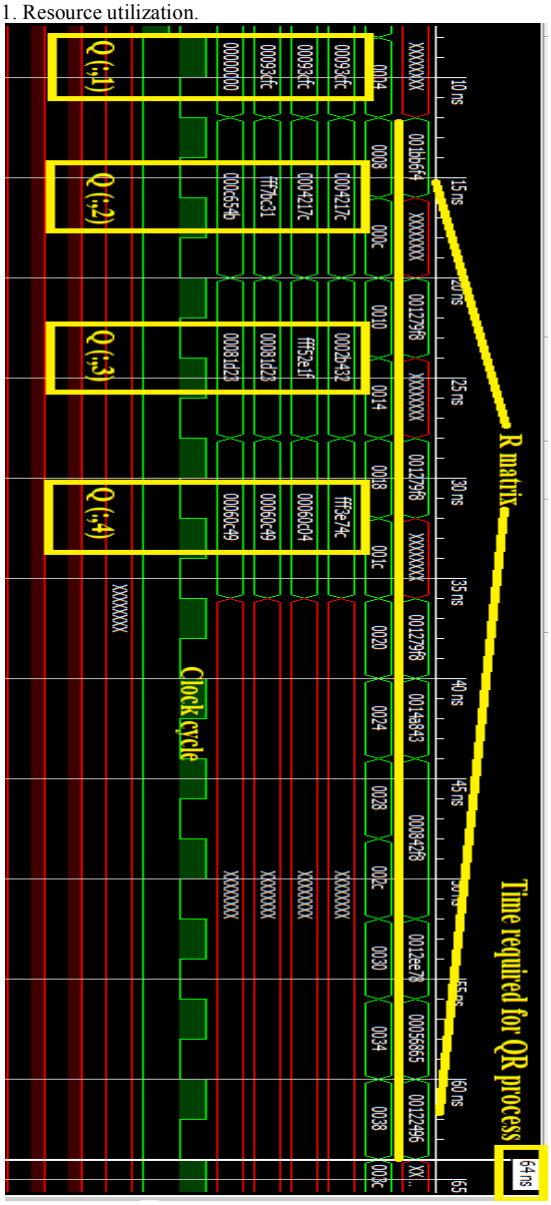

Fig.12. Simulation of Q and R matrices.

REFERENCES

[1] Robert Chen-Hao Chang, "Iterative QR Decomposition Architecture Using the Modified Gram–Schmidt Algorithm for MIMO Systems," *IEEE Transactions On Circuits And Systems,* vol. 57, no. 5, pp. 1095-1102, 2010.

[2] G. H. Golub and C. F. V. Loan, Matrix Comutations, Baltimore, Maryland: John Hopkins Univ. Press, 2013.

[3] S. S. Omran and A. K. Abdul-abbas, "Design of 32-bits RISC processor for hardware efficient QR decomposition," in *2018 International Conference on Advance of Sustainable Engineering and its Application (ICASEA)*, Wasit - Kut, Iraq, Iraq, 2018.

[4] P. Luethi, "Gram-Schmidt-based QR Decomposition for MIMO Detection: VLSI Implementation and Comparison," in *IEEE Asia Pacific Conference on Circuits and Systems* , Macao, China , 2008.

[5] Dongyeob Shin, Ji-Hwan Yoon, "Gram-schmidt tailed high-throughput QR decomposition architecture for MIMO detector," in *International SoC Design Conference (ISOCC)*, Jeju, South Korea , 2014.

[6] S. Aslan, S. Niu and J. Saniie, "FPGA implementation of fast QR decomposition based on givens rotation," in *IEEE International Conference on Midwest Symposium on Circuits and Systems*, USA, 2012.

[7] D. C. a. M. SIMA, "Fixed-Point CORDIC-Based QR Decomposition by Givens Rotations on FPGA," in *Reconfigurable Computing and FPGAs*, Canada, 2011.

[8] H. Lipfert, "Part I, Technical Basis," in *MIMO OFDM Space Time Coding – Spatial Multiplexing, Increasing Performance and Spectral Efficiency in Wireless Systems*, Institut für Rundfunktechnik., 2007, p. 22.

[9] M. Parker, V. Mauer and D. Pritsker, "QR Decomposition using FPGAs," in *EEE National Aerospace and Electronics Conference and Ohio Innovation Summit*, Dayton, OH, USA, 2016.

[10] S. S. Omran and H. S. Mahmood, "Pipelined MIPS processor with cache controller using VHDL implementation for educational purposes," in *IEEE International Conference on Electrical Communication, Computer, Power, and Control Engineering*, Mosul, Iraq, 2014.

[11] xilinx, "www.xilinx.com," Xilinx Inc., 2017. [Online]. Available: https://www.xilinx.com/products/intellectual-property/divider.html#overview.

[12] W. Zhao, J. Lin and S.-C. Chan, "Throughput/Area Efficient FPGA Implementation of QR Decomposition for MIMO Systems," in *2016 IEEE International Conference on Digital Signal Processing (DSP)*, Beijing, China, 2016.

[13] R. C. H. Chang and C. H. Lin and K. H. Lin and C. L. Huang and F. C. Chen, "Iterative QR decomposition architecture using the modified gram–schmidt algorithm for MIMO systems," *IEEE Transactions on Circuits and Systems I: Regular Papers,* vol. 57, no. 5, pp. 1095-1102, 2010.