# A Low Complexity Complex QR Factorization Design for Signal Detection in MIMO OFDM Systems

Yin-Tsung Hwang and Wei-Da Chen
*Department of Electrical Engineering,*
*National Chung Hsing University, Taichung, Taiwan*
*hwangyt@dragon.nchu.edu.tw*

*Abstract*— **Complex QR factorization is a fundamental operation used in various MIMO signal detection algorithms. In this paper, we revise the Givens Rotation based factorization algorithm and develop an efficient scheme working in the real number domain. The complex matrix is first extended into a block-wise symmetric real number counterpart. The proposed scheme can reduce the computing complexity to almost one half by exploiting the symmetric property. Computing complexity analysis also shows the superiority of our scheme over various factorization schemes. Finally, subject to the EWC 802.11n recommendation, a novel systolic array design featuring fully parallel and deeply pipelined processing was presented. CORDIC algorithm is employed to implement the required rotation operations with low circuit complexity. Synthesis results in TSMC 0.18μm process indicate the proposed design, with a gate count of merely 17.06K and a maximum clock rate of 202 MHz, can admit a new 2×2 complex matrix for factorization in every 8 clock cycles.**

## I. INTRODUCTION

MIMO OFDM based communications such as WiFi and WiMax have become very popular these years due to its merits of spectral efficiency and resilience to frequency selective fading channels. An important property of MIMO processing is its intensive computing, which grows quadratically with the number of antennas used. In particular, signal detection is one of the most demanding modules in MIMO OFDM systems. The signal vectors must be resolved tone by tone in the frequency domain subject to the pace of sampling rate. Each detection basically needs to handle a matrix operation of size $N_r \times N_t$, where $N_r$ and $N_t$ stand for the numbers of receiving and transmitting antennas, respectively. Various MIMO signal detection schemes such as zero-forcing, V-blast, QR-blast, ML, sphere decoder have been proposed. Among them, zero-forcing, QR-blast and sphere decoder all employ QR factorization (QRF) operations in their schemes. The biggest advantage of QR factorization is that it is free of noise enhancement problem due to the property of unitary operations. This helps minimize the chance of erroneous detection arising from the noise factor. QR factorization itself, however, bears an order $O(N^3)$ computing complexity, where $N$ is the matrix size. The computation is even more complicated when the matrix entries are in complex forms, i.e. complex QR factorization. Due to the massive computations of complex QR factorization, hardwired approach is the only viable solution to meet the real time processing constraint. Algorithm development, nonetheless, is crucial to the computing complexity and may facilitate a simpler hardware design. Previous algorithms can be classified into three categories. Algorithms in category 1 basically resort to a sequence of Givens rotations. In [1], the complex matrix elements are first converted from Cartesian forms into polar forms. The rotation matrix is next calculated and applied for nullification. This approach requires frequent conversions between polar and Cartesian forms and will not only introduce excessive computation overheads but also hinder computing parallelism. The algorithm is modified in [2] where the rotation matrix contains three angles instead of two and the diagonals of the matrix can be kept real after rotations. It claims a shorter computing latency compared to [1] and only real divisions are required in the following backward substitution process. Algorithms in category 2 basically extend a complex matrix (size $N \times N$) into a real counterpart (size $2N \times 2N$) first and apply modified Gram-Schmidt scheme to do the factorization. Since all matrix elements are now real, so are the multiplication, division and square root operations; this facilitates a simpler hardware implementation. In [3], the data is further converted into log domain to simplify the multiplication and division operations. Considerable memory storage, however, is needed to hold the look-up table for conversion. The third category of algorithms utilize Householder [4,5] schemes to perform the factorization. The triangularization process in the Householder scheme is performed column by column. Despite its efficiency in the number of iterations, the algorithm is generally too complex to be realized in hardware efficiently. As for the hardware designs, systolic array is the most common architecture adopted in reported literatures. Depending on the computing parallelism supported, the designs can be a 2-D tri-array [2], a 1-D linear array [6] or simply a complicated function unit in charge of all iterations one by one [3]. In this paper, we will present a novel low complexity QR factorization module design. We start with the development of an efficient factorization algorithm based on Givens Rotation scheme and

978-1-4244-1684-4/08/$25.00 ©2008 IEEE

working in the real number domain. The nullification ordering is carefully scheduled to keep the matrix symmetric so that redundant computations can be totally eliminated. A highly parallel and deeply pipelined systolic array design is next developed. It requires as few as 8 clock cycles to admit a new 2-by-2 factorization, the fewest known in the literature. The circuit complexity, however, is kept low by the employment of a new QRF scheme and CORDIC modules.

## II. PROPOSED QR FACTORIZATION SCHEME

A frequency domain MIMO channel model can be expressed as,

$$\mathbf{y}_{N_r\times 1} = \mathbf{H}_{N_r\times N_t}\mathbf{x}_{N_t\times 1} + \mathbf{n}_{N_r\times 1}, \quad (1)$$

where **y**, **H**, **x** and **n** denote the received signal, the channel matrix, the transmitted signal and the noise, respectively. The signal detection problem can be formulated as a least square (LS) problem. The basic of QRF is to decompose a matrix into the product of a unitary matrix **Q** and an upper triangular matrix **R**. After factorization, the LS problem can then be solved by backward substitutions as shown in Eq. (2).

$$\mathbf{y} = (\mathbf{QR})\cdot\mathbf{x} + \mathbf{n} \;\Rightarrow\; \hat{\mathbf{x}} = \mathbf{R}^{-1}\cdot\left(\mathbf{Q}^{\mathbf{H}}\mathbf{y}\right) = \mathbf{R}^{-1}\cdot\hat{\mathbf{y}} \quad (2)$$

Since all aforementioned operations are in complex number forms, the channel model in Eq. (1) can be converted into a real number counterpart as indicated in Eq. (3).

$$\begin{bmatrix}\mathrm{Re}(\mathbf{y})\\ \mathrm{Im}(\mathbf{y})\end{bmatrix} = \begin{bmatrix}\mathrm{Re}(\mathbf{H}) & -\mathrm{Im}(\mathbf{H})\\ \mathrm{Im}(\mathbf{H}) & \mathrm{Re}(\mathbf{H})\end{bmatrix}\cdot\begin{bmatrix}\mathrm{Re}(\mathbf{x})\\ \mathrm{Im}(\mathbf{x})\end{bmatrix} + \begin{bmatrix}\mathrm{Re}(\mathbf{n})\\ \mathrm{Im}(\mathbf{n})\end{bmatrix}, \quad (3)$$

where Re(·) and Im(·) denote the real and imaginary parts. Therefore, we may operate QR factorizations on an extended real channel matrix (size $2N_r\times 2N_t$) instead of a complex one. To simplify the illustration of the proposed QRF scheme, we will assume $N_r = N_t = 2$. As shown in Eq. (4), an extended real channel matrix $\tilde{\mathbf{H}}$ augmented with **y** is first derived.

$$\tilde{\mathbf{H}} = \begin{bmatrix} h_{r11} & h_{r12} & -h_{i11} & -h_{i12} & y_{r1}\\ h_{r21} & h_{r22} & -h_{i21} & -h_{i22} & y_{r2}\\ h_{i11} & h_{i12} & h_{r11} & h_{r12} & y_{i1}\\ h_{i21} & h_{i22} & h_{r21} & h_{r22} & y_{i2}\end{bmatrix}, \quad (4)$$

where subscript *r* and *i* each indicates the real and imaginary part of a complex component. The augmentation of the received signal **y** is for the use in backward substitutions to solve the estimate $\hat{\mathbf{x}}$. Conventional Givens Rotation scheme [7] is to triangularize the matrix $\tilde{\mathbf{H}}$ through a sequence of rotations. This, however, fails to explore the symmetric property (except for the minus sign in Im(**H**)) of the matrix and leads to excessive computations. In our scheme, we try to keep the block-wise symmetric property of matrix $\tilde{\mathbf{H}}$ intact during the triangularization process. In addition, 4 triangular sub-matrices are sought instead of one triangular conglomerate. This can be illustrated by Figure 1. It can be seen that these 4 triangular sub-matrices actually correspond to the extended real number version of a complex upper triangular matrix **R** after QR factorization. Another advantage of the proposed scheme is that the imaginary parts of the diagonal elements are all zeros. This means only real division is needed during the backward substitutions.

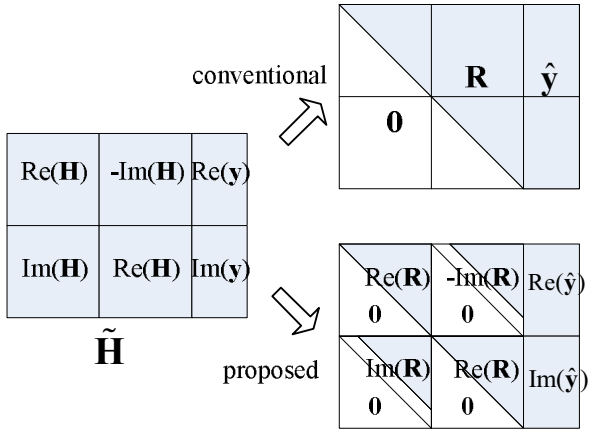


Figure 1. Conventional vs. proposed triangularization schemes for QRF

The scheme begins with nullifying the first column of Im(**R**). Givens rotations are applied to the corresponding rows in Re(**R**) and Im(**R**). Due to the block-wise symmetric property in $\tilde{\mathbf{H}}$, the first column in both Im(**R**) and - Im(**R**) can be nullified simultaneously as shown in Eq. (5) & (6),

$$\begin{bmatrix}\cos\theta_1 & 0 & \sin\theta_1 & 0\\ 0 & 1 & 0 & 0\\ -\sin\theta_1 & 0 & \cos\theta_1 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}\begin{bmatrix} h_{r11} & h_{r12} & -h_{i11} & -h_{i12} & y_{r1}\\ h_{r21} & h_{r22} & -h_{i21} & -h_{i22} & y_{r2}\\ h_{i11} & h_{i12} & h_{r11} & h_{r12} & y_{i1}\\ h_{i21} & h_{i22} & h_{r21} & h_{r22} & y_{i2}\end{bmatrix} = \begin{bmatrix} h^{(1)}_{r11} & h^{(1)}_{r12} & 0 & -h^{(1)}_{i12} & y^{(1)}_{r1}\\ h_{r21} & h_{r22} & -h_{i21} & -h_{i22} & y_{r2}\\ 0 & h^{(1)}_{i12} & h^{(1)}_{r11} & h^{(1)}_{r12} & y^{(1)}_{i1}\\ h_{i21} & h_{i22} & h_{r21} & h_{r22} & y_{i2}\end{bmatrix} \quad (5)$$

$$\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & \cos\theta_2 & 0 & \sin\theta_2\\ 0 & 0 & 1 & 0\\ 0 & -\sin\theta_2 & 0 & \cos\theta_2\end{bmatrix}\begin{bmatrix} h^{(1)}_{r11} & h^{(1)}_{r12} & 0 & -h^{(1)}_{i12} & y^{(1)}_{r1}\\ h_{r21} & h_{r22} & -h_{i21} & -h_{i22} & y_{r2}\\ 0 & h^{(1)}_{i12} & h^{(1)}_{r11} & h^{(1)}_{r12} & y^{(1)}_{i1}\\ h_{i21} & h_{i22} & h_{r21} & h_{r22} & y_{i2}\end{bmatrix} = \begin{bmatrix} h^{(1)}_{r11} & h^{(1)}_{r12} & 0 & -h^{(1)}_{i12} & y^{(1)}_{r1}\\ h^{(1)}_{r21} & h^{(1)}_{r22} & 0 & -h^{(1)}_{i22} & y^{(1)}_{r2}\\ 0 & h^{(1)}_{i12} & h^{(1)}_{r11} & h^{(1)}_{r12} & y^{(1)}_{i1}\\ 0 & h^{(1)}_{i22} & h^{(1)}_{r21} & h^{(1)}_{r22} & y^{(1)}_{i2}\end{bmatrix} \quad (6)$$

where $\theta_1 = \tan^{-1}\left(h_{i11}/h_{r11}\right)$, $\theta_2 = \tan^{-1}\left(h_{i21}/h_{r21}\right)$. After the nullification of the first column in Im(**H**), we work on the nullification of the first column in Re(**H**). All elements below the diagonal will be nullified and this is accomplished by applying Givens rotations to the successive rows in Re(**H**) in a bottom up sequence. Since there are two Re(**H**) sub-matrices, the same nullification process must be performed twice. Eq. (7) & (8) are used to illustrate the process.

$$\begin{bmatrix}\cos\theta_3 & \sin\theta_3 & 0 & 0\\ -\sin\theta_3 & \cos\theta_3 & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}\begin{bmatrix} h^{(1)}_{r11} & h^{(1)}_{r12} & 0 & -h^{(1)}_{i12} & y^{(1)}_{r1}\\ h^{(1)}_{r21} & h^{(1)}_{r22} & 0 & -h^{(1)}_{i22} & y^{(1)}_{r2}\\ 0 & h^{(1)}_{i12} & h^{(1)}_{r11} & h^{(1)}_{r12} & y^{(1)}_{i1}\\ 0 & h^{(1)}_{i22} & h^{(1)}_{r21} & h^{(1)}_{r22} & y^{(1)}_{i2}\end{bmatrix} = \begin{bmatrix} h^{(2)}_{r11} & h^{(2)}_{r12} & 0 & -h^{(2)}_{i12} & y^{(2)}_{r1}\\ 0 & h^{(2)}_{r22} & 0 & -h^{(2)}_{i22} & y^{(2)}_{r2}\\ 0 & h^{(1)}_{i12} & h^{(1)}_{r11} & h^{(1)}_{r12} & y^{(1)}_{i1}\\ 0 & h^{(1)}_{i22} & h^{(1)}_{r21} & h^{(1)}_{r22} & y^{(1)}_{i2}\end{bmatrix} \quad (7)$$

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \cos\theta_4 & \sin\theta_4 \\ 0 & 0 & -\sin\theta_4 & \cos\theta_4 \end{bmatrix} \begin{bmatrix} h_{r11}^{(2)} & h_{r12}^{(2)} & 0 & -h_{i12}^{(2)} & y_{r1}^{(2)} \\ 0 & h_{r22}^{(2)} & 0 & -h_{i22}^{(2)} & y_{r2}^{(2)} \\ 0 & h_{i12}^{(1)} & h_{i11}^{(1)} & h_{r12}^{(1)} & y_{i1}^{(1)} \\ 0 & h_{i22}^{(1)} & h_{r21}^{(1)} & h_{r22}^{(1)} & y_{i2}^{(1)} \end{bmatrix} = \begin{bmatrix} h_{r11}^{(2)} & h_{r12}^{(2)} & 0 & -h_{i12}^{(2)} & y_{r1}^{(2)} \\ 0 & h_{r22}^{(2)} & 0 & -h_{i22}^{(2)} & y_{r2}^{(2)} \\ 0 & h_{i12}^{(2)} & h_{r11}^{(2)} & h_{r12}^{(2)} & y_{i1}^{(2)} \\ 0 & h_{i22}^{(2)} & 0 & h_{r22}^{(2)} & y_{i2}^{(2)} \end{bmatrix} \quad (8)$$

where $\theta_3 = \theta_4 = \tan^{-1}\left(h_{r21}^{(1)} / h_{r11}^{(1)}\right)$. After working on the 1st columns of all sub-matrices, we now proceed to work on the 2nd columns. The same procedures apply except that the sub-matrices are reduced to size ($N$-1)×($N$-1). We may repeat the same nullification process until the triangular forms as indicated in Figure 1 are achieved. For the 2×2 example in our illustration, one more nullification suffices to accomplish the entire triangularization process. This is given by Eq. (9).

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta_5 & 0 & \sin\theta_5 \\ 0 & 0 & 1 & 0 \\ 0 & -\sin\theta_5 & 0 & \cos\theta_5 \end{bmatrix} \begin{bmatrix} h_{r11}^{(2)} & h_{r12}^{(2)} & 0 & -h_{i12}^{(2)} & y_{r1}^{(2)} \\ 0 & h_{r22}^{(2)} & 0 & -h_{i22}^{(2)} & y_{r2}^{(2)} \\ 0 & h_{i12}^{(2)} & h_{i11}^{(2)} & h_{r12}^{(2)} & y_{i1}^{(2)} \\ 0 & h_{i22}^{(2)} & 0 & h_{r22}^{(2)} & y_{i2}^{(2)} \end{bmatrix} = \begin{bmatrix} h_{r11}^{(2)} & h_{r12}^{(2)} & 0 & -h_{i12}^{(2)} & y_{r1}^{(2)} \\ 0 & h_{r22}^{(3)} & 0 & 0 & y_{r2}^{(3)} \\ 0 & h_{i12}^{(3)} & h_{r11}^{(2)} & h_{r12}^{(2)} & y_{i1}^{(2)} \\ 0 & 0 & 0 & h_{r22}^{(3)} & y_{i2}^{(3)} \end{bmatrix} \quad (9)$$

with $\theta_5 = \tan^{-1}\left(h_{r22}^{(2)} / h_{i22}^{(2)}\right)$. Finally the LS estimate $\hat{\mathbf{x}}$ can be obtained by backward substitutions as indicated in Eq. (10).

$$\begin{aligned} &[\hat{x}_{r1} \quad \hat{x}_{r2} \quad \hat{x}_{i1} \quad \hat{x}_{i2}]^t \\ &= \left[ \frac{y_{r1}^{(2)} + h_{i12}^{(2)}\hat{x}_{i2} - h_{r12}^{2}\hat{x}_{r2}}{h_{r11}^{(2)}} \quad \frac{y_{r2}^{(3)}}{h_{r22}^{(3)}} \quad \frac{y_{i1}^{(2)} - h_{i12}^{(2)}\hat{x}_{r2} - h_{r12}^{(2)}\hat{x}_{i2}}{h_{r11}^{(2)}} \quad \frac{y_{i2}^{(3)}}{h_{r22}^{(3)}} \right]^t \end{aligned} \quad (10)$$

It should be noted that these four sub-matrices are always kept symmetric during the nullification process. So are they in the final triangularized $\tilde{\mathbf{H}}$ matrix. This means only the two sub-matrices in the first part (i.e. Re(**H**) and Im(**H**)) need to be calculated in practice. The computations of the other two sub-matrices are simply mirrors of the first part and can be totally eliminated. This saves the computation complexity by almost one half. The simplified scheme now starts with a reduced real matrix $\tilde{\mathbf{H}}'$.

$$\tilde{\mathbf{H}}' = \begin{bmatrix} h_{r11} & h_{r12} & y_{r1} \\ h_{r21} & h_{r22} & y_{r2} \\ h_{i11} & h_{i12} & y_{i1} \\ h_{i21} & h_{i22} & y_{i2} \end{bmatrix} \quad (11)$$

The proposed QR factorization scheme is then formulated as finding a sequence of Givens rotations $\mathbf{Q}_k^H$ to convert $\tilde{\mathbf{H}}'$ into a triangular form as shown in Eq. (12).

$$\left(\prod_k \mathbf{Q}_k^H\right) \cdot \tilde{\mathbf{H}}' = \begin{bmatrix} \mathrm{Re}(\mathbf{R}) & \mathrm{Re}(\hat{\mathbf{y}}) \\ \mathrm{Im}(\mathbf{R}) & \mathrm{Im}(\hat{\mathbf{y}}) \end{bmatrix} = \begin{bmatrix} h_{r11}^{(2)} & h_{r12}^{(2)} & y_{r1}^{(2)} \\ 0 & h_{r22}^{(3)} & y_{r2}^{(3)} \\ 0 & h_{i12}^{(2)} & y_{i1}^{(2)} \\ 0 & 0 & y_{i2}^{(3)} \end{bmatrix} \quad (12)$$

## III. Algorithm Performance Evaluation

We first conduct computing complexity analyses on various QRF algorithms. These include the proposed scheme and both complex and real versions (by matrix extension as suggested in Eq. (3)) of Householder (HD), Gram-Schmidt (GS) and Givens Rotation (GR). The complexity is measured in the units of real multiply, add, division and square root operations. Since a CORDIC scheme will be used to implement the trigonometric operations, the computing complexity of CORDIC is used in analysis. In our design, it corresponds to 8 (the iteration number) add operations. The results for a 2×2 matrix are summarized in Table I. All GR based schemes use trigonometric operations and only add operations are needed in CORDIC implementation. Our scheme enjoys the least computing complexity among all due to the removal of redundant operations. Both Gram-Schmidt and Householder schemes, either real or complex version, are much more complicated than the Givens Rotation schemes and require 4 types of function unit.

TABLE I
Computing Complexity Analyses of Various QRF Schemes

| scheme | MPY | ADD | SQRT | DIV |
|---|---|---|---|---|
| proposed | - | 140 | - | - |
| complex HD | 64 | 42 | 2 | 2 |
| complex GS | 32 | 21 | 1 | 2 |
| complex GR | - | 200 | - | - |
| real HD | 135 | 68 | 8 | 4 |
| real GS | 52 | 45 | 4 | 4 |
| real GR | - | 368 | - | - |

Since these schemes lead to different factorization results, we will next analyze the impact of factorization schemes on the overall MIMO signal detection performance. A simple 2×2 MIMO OFDM system presented in [8] is used as the test platform. It basically follows the 802.11n environment setting and a frequency domain windowing function [8] is used in channel estimation. For simplicity, BPSK modulation is assumed and QR-Blast detection algorithm is adopted. The BER curves of signal detection using different factorization schemes are given in Figure 2. All curves are virtually overlapped and exhibit almost identical BER performance. It can be concluded that different factorization schemes, in spite of their computing complexities, do not affect the BER performance as long as the same signal detection algorithm is applied.

## IV. Hardware Design & Implementation

Subject on the QRF scheme presented in section II, a novel systolic array is developed for 2×2 MIMO systems. This design supports two levels (i.e. inter- and intra-iteration) of computing concurrency and can provide much more computing power than the linear array or single processor can. The design specs also follow the EWC 802.11n recommendation, which requires solving 114 channel estimations in an 8μs long preamble period for later signal

detection use. In other words, one complex QRF must be performed in every 70ns. For a target working frequency of 120MHz, only 8 clock cycles are allowed for each complex QRF computation. Therefore, a 2-D systolic array is needed to provide the necessary computing power. Figure 3 shows the block diagram of the proposed systolic array design, where "GG" stands for the Givens Generation (angle calculation) and "GR" stands for the Givens Rotation. The computing modules are divided into 5 sections, corresponding to the concurrent calculations of Eq. (7)~(11). Since Eq. (9) and (10) apply the same rotation angle, only one GG module is needed for these two sections. To reduce the circuit complexity, a CORDIC design is employed to implement both GG and GR modules. CORDIC is an efficient algorithm to implement trigonometric functions via iterative operations of shift and add. No look-up table is needed in this case, either. It simplifies the circuit complexity significantly at the cost of lengthened computing latency. Since the nullification process must be performed in order, increased cycle counts in either GG or GR module will have adverse effects on the overall throughput rate of QRF. To overcome this problem, the computing parallelism at the inter-iteration level is explored across neighboring GG/GR modules in our design. In other words, the GR module can begin its computation only one iteration behind the GG module. The proposed design is thus characterized as a highly parallel and deeply pipelined QRF systolic processor. 8 times of iteration are adopted in our design and simulations indicate further iterations do not help in precision but will degrade the speed. Synthesis results using TSMC 0.18μm process indicate that the proposed design consists of only 17.06K logic gates and can operate at a maximum clock rate of 202 MHz (better than the specified 120MHz for real time operations). In our design, the initiation interval between two successive complex 2×2 QRF computations is only 8 clock-cycle long, which is much smaller than those presented in other designs such as [2,3,6].

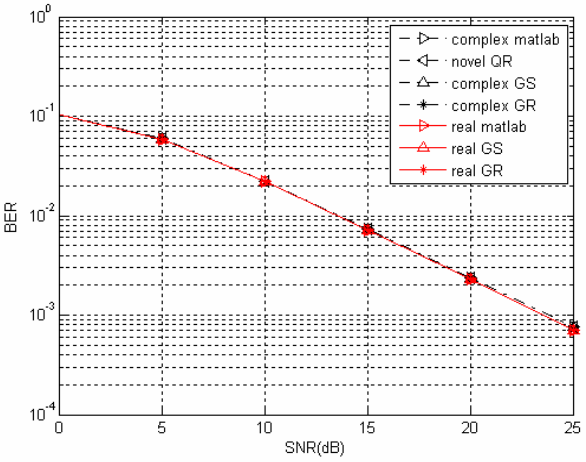


Figure 2. BER performances by different QRF schemes

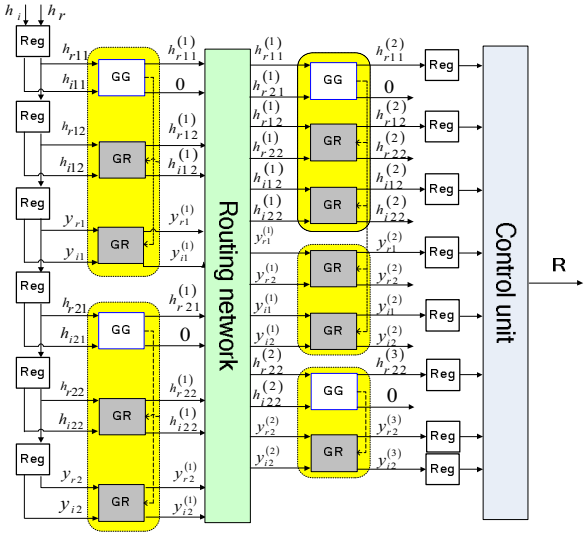


Figure 3. Block diagram of the proposed QRF systolic array design

## V. CONCLUSION

In conclusion, in this paper, we first derive an efficient complex QRF scheme capable of working in real number formats with low computing complexity. The claim is verified by the computing analyses on various QRF schemes, either in real or complex formats. To meet the demanding speed requirement of 802.11n specs, we next present a novel systolic array design using CORDIC modules. Only add operations are required in our implementation and the synthesis results prove the merits of our design in both circuit complexity and speed.

## VI. REFERENCE


[1] T.F. Coleman and C.F.Van Loan., "Handbook for Matrix Computations." SIAM, Philadelphia, PA, 1988.

[2] A. Maltsev, V. Pestretsov, R. Maslennikov, A. Khoryaev, "Triangular systolic array with reduced latency for QR-decomposition of complex matrices," *IEEE ISCAS,* vol. 21-24, May 2006.

[3] C.K. Singh, S.H. Prasad, P.T. Balsara, " VLSI Architecture for Matrix Inversion using Modified Gram-Schmidt based QR Decomposition," *International Conference on VLSI Design,* Jan. 2007.

[4] C.F.T. Tang, K.J.R. Liu, S.A. Tretter, " On systolic arrays for recursive complex Householder transformations with applications to array processing," *ICASSP*, vol.14-17, April 1991.

[5] K.-L. Chung, W.-M. Yan, " The complex Householder transform," *IEEE Transactions on S ignal Processing,* Vol.45, Issue 9, Sept. 1997

[6] F. Sobhanmanesh, S. Nooshabadi, " Parametric minimum hardware QR-factoriser architecture for V-BLAST detection," *IEEE Proceedings-Circuits, Devices and Systems,* Vol.153, Issue 5, October 2006.

[7] A. El-Amawy, K.R. Dharmarajan, "Parallel VLSI algorithm for stable inversion of dense matrices," *IEEE Proceddings on Computers and Digital Techniques,* vol. 136, no.6, pp.575-580, Nov.1989.

[8] Y.-T. Hwang, S.-C. Peng, J.-K. Lain, "A Low Complexity Channel Estimator for MIMO-OFDM Systems," *International Conference on Communications, Circuits and Systems Proceedings,* Vol.2, June 2006