

Triangular Systolic Array with Reduced Latency for QR-decomposition of Complex Matrices

Alexander Maltsev, Vladimir Pestretsov, Roman Maslennikov, Alexey Khoryaev
Broadband Wireless Division, Intel Corporation,
Nizhny Novgorod, Russia

Abstract — The novel CORDIC-based architecture of the Triangular Systolic Array for QRD of large size complex matrices is presented. The proposed architecture relies on QRD using a three angle complex rotation approach that provides *significant reduction of latency* (systolic operation time) and makes the QRD in such a way that the upper triangular matrix \mathbf{R} has *only real diagonal elements*.

I. INTRODUCTION

The upcoming 802.11n and 802.16e standards will include support of multiple antenna techniques such as MIMO and SDMA modes of transmission, which allow spatial multiplexing of data streams from one or multiple users. The performance and complexity of such systems will strictly depend on the number of antennas used. Thus the challenging task is arising to develop the most efficient architectures for realization of different signal processing algorithms in MIMO-OFDM systems with large number of antenna elements.

The MIMO detection schemes which will be used in 802.11n and 802.16e transceivers are based on Zero-Forcing (ZF) or Minimum Mean Square Error (MMSE) linear algorithms, because of their efficiency and relatively low computational complexity. As these algorithms are linear, their realization consists of calculating weight matrices for every subcarrier (or group of adjacent subcarriers) and multiplication of the OFDM subcarrier received signals (in frequency domain) from different antennas by a weighting matrix to get estimates of the transmitted data symbols. The weight matrices for ZF and MMSE algorithms are calculated as follows:

$$\mathbf{W}(i) = (\mathbf{H}(i)^H \mathbf{H}(i))^{-1} \mathbf{H}(i)^H \quad (1)$$

$$\mathbf{W}(i) = (\mathbf{H}(i)^H \mathbf{H}(i) + \rho(i)\mathbf{I})^{-1} \mathbf{H}(i)^H \quad (2)$$

where i is the subcarrier index, $\mathbf{W}(i)$ is the weight matrix for the i -th subcarrier, $\mathbf{H}(i)$ is the channel transfer matrix for the i -th subcarrier, \mathbf{I} is the identity matrix and $\rho(i)$ is the reciprocal of the SNR for the i -th subcarrier.

The hardware block implementing the ZF or MMSE algorithms will include a weight calculation unit, and a unit that combines data subcarriers from different antennas using

these weights (combiner unit). The implementation of the combiner unit is rather straightforward. Opposed to that, the implementation of the weight calculation unit is quite challenging. As it can be seen from (1) and (2) *the main computational problem* for the ZF and MMSE algorithms is *matrix inversion*, which should be done for every subcarrier (or group of several adjacent subcarriers). In 802.11n and 802.16e systems the channel is measured using training symbols and/or pilots, which are followed by data symbols with spatial multiplexing. So, weight matrix calculation should be done with a low latency for not impacting the overall RX latency significantly. To meet such stringent requirements, the usage of a dedicated hardware block for matrix inversion appears necessary.

The effective way for solving the hardware matrix inversion problem is the use of the QR Decomposition (QRD). The QRD for matrix \mathbf{V} determines matrices \mathbf{Q} and \mathbf{R} such that $\mathbf{V} = \mathbf{QR}$, where \mathbf{Q} is a unitary matrix and \mathbf{R} is an upper triangular matrix. If the QR decomposition is available, then the inverse matrix can be found as:

$$\mathbf{V}^{-1} = (\mathbf{QR})^{-1} = \mathbf{R}^{-1} \mathbf{Q}^{-1} = \mathbf{R}^{-1} \mathbf{Q}^H \quad (3)$$

The inverse matrix for unitary matrix \mathbf{Q} is simply found as its Hermitian transpose and the inversion of the triangular matrix \mathbf{R} is straightforward using the back substitution algorithm.

The QRD is advantageous for hardware realization of matrix inversion as it can be efficiently implemented by a systolic array with processing elements based on the CORDIC algorithm, and the total QR decomposition can be performed with no multiplications.

Thus, the QRD technique looks like *the most promising approach* for implementation in MIMO and SDMA systems *with large number of TX and RX antenna elements*.

Recently an efficient parallel triangular systolic processor array realization of the QR Decomposition-based Recursive Least Square (QRD-RLS) algorithm using Givens rotations was introduced in [1].

In the present work for the QRD realization we suggest to use a new CORDIC based architecture of the Triangular Systolic Array (TSA) exploiting Three Angle Complex Rotation (TACR) approach instead of a Complex Givens Rotation (CGR). Compared to the CGR approach, the proposed TSA/TACR architecture provides *significant reduction of latency* (systolic operation time) by using the same hardware resources of CORDIC modules. As a second advantage of the TSA/TACR approach, it makes the QRD in such a way that the upper triangular matrix \mathbf{R} has *only real diagonal elements* instead of complex ones always produced by CGR. This fact essentially simplifies the following inversion of matrix \mathbf{R} when using back substitution, which requires many divisions on the diagonal elements of \mathbf{R} .

II. QRD BACKGROUND

At the beginning let briefly review the Complex Givens Rotation and introduce the Three Angle Complex Rotation approaches for the QR decomposition of a complex matrix.

The Givens Rotation technique is used to selectively introduce a zero into a matrix. This approach is widely applied for the QRD to reduce the input matrix to a triangular form by applying successive rotations to matrix elements below the main diagonal [2]. The idea of complex Givens rotation can be easily illustrated on the example of 2x2 complex matrix $\mathbf{V}_{2 \times 2}$ defined by:

$$\begin{bmatrix} Ae^{j\theta_a} & Ce^{j\theta_c} \\ Be^{j\theta_b} & De^{j\theta_d} \end{bmatrix} \quad (4)$$

where $j = \sqrt{-1}$; A, B, C, D represent the magnitudes and $\theta_a, \theta_b, \theta_c, \theta_d$ stand for the angles of the matrix entries.

The CGR is described by two rotation angles θ_1, θ_2 through the following matrix transformation:

$$\begin{bmatrix} \cos \theta_1 & \sin \theta_1 e^{j\theta_2} \\ -\sin \theta_1 e^{-j\theta_2} & \cos \theta_1 \end{bmatrix} \cdot \begin{bmatrix} Ae^{j\theta_a} & Ce^{j\theta_c} \\ Be^{j\theta_b} & De^{j\theta_d} \end{bmatrix} = \begin{bmatrix} Xe^{i\theta_x} & Ye^{i\theta_y} \\ 0 & Ze^{i\theta_z} \end{bmatrix} \quad (5)$$

where angles θ_1, θ_2 are chosen to set to zero the matrix element below the main diagonal, and defined by:

$$\begin{aligned} \theta_1 &= \tan^{-1}(B/A) \\ \theta_2 &= \theta_a - \theta_b \end{aligned} \quad (6)$$

It is easy to verify that using (6) leads to an upper triangular matrix with complex diagonal elements (5).

The alternative approach for the QRD may be realized through a unitary matrix transformation described by the TACR technique. This transformation for the QRD of $\mathbf{V}_{2 \times 2}$ in (4) may be presented by:

$$\mathbf{Z} = \begin{bmatrix} \cos \theta_1 e^{j\theta_2} & \sin \theta_1 e^{j\theta_3} \\ -\sin \theta_1 e^{j\theta_2} & \cos \theta_1 e^{j\theta_3} \end{bmatrix} \quad (7)$$

To obtain an upper triangular matrix the presented unitary transformation requires three angles $\theta_1, \theta_2, \theta_3$ determined by:

$$\begin{aligned} \theta_1 &= \tan^{-1}(B/A) \\ \theta_2 &= -\theta_a \\ \theta_3 &= -\theta_b \end{aligned} \quad (8)$$

The suggested TACR technique results in a new upper triangular matrix:

$$\begin{bmatrix} \cos \theta_1 e^{j\theta_2} & \sin \theta_1 e^{j\theta_3} \\ -\sin \theta_1 e^{j\theta_2} & \cos \theta_1 e^{j\theta_3} \end{bmatrix} \cdot \begin{bmatrix} Ae^{j\theta_a} & Ce^{j\theta_c} \\ Be^{j\theta_b} & De^{j\theta_d} \end{bmatrix} = \begin{bmatrix} X & Ye^{i\theta_y} \\ 0 & Ze^{i\theta_z} \end{bmatrix} \quad (9)$$

Note that the TACR matrix transformation introduces the real element X on the matrix diagonal. Application of the TACR approach for an $N \times N$ square matrix will lead to the appearance of real elements on the matrix diagonal except at the lowest one. For further inversion of the obtained upper triangular matrix it is advantageous to eliminate the complex lowest diagonal element in order to avoid complex division. To produce a matrix with only real diagonal elements, the additional simple unitary transformation is required. For the considered example of the 2x2 matrix in (4), the additional matrix transformation has the form:

$$\begin{bmatrix} 1 & 0 \\ 0 & e^{-j\theta_z} \end{bmatrix} \cdot \begin{bmatrix} X & Ye^{i\theta_y} \\ 0 & Ze^{i\theta_z} \end{bmatrix} = \begin{bmatrix} X & Ye^{i\theta_y} \\ 0 & Z \end{bmatrix} \quad (10)$$

Note that this additional step at the last phase of the QRD does not complicate its hardware realization and can be easily embedded into the CORDIC-based TSA/TACR architecture.

It is shown in the next section that utilizing the TACR approach for the QRD results in a CORDIC-based systolic array architecture achieving latency reduction compared to the TSA architecture based on CGR.

III. DESCRIPTION OF TSA/TACR ARCHITECTURE

The structure of the suggested TSA/TACR architecture for the QRD is shown in Fig. 1. It consists of three main processing cells with different functionality: Delay Units (DU), Processing Elements (PE) and one Rotational Unit (RU).

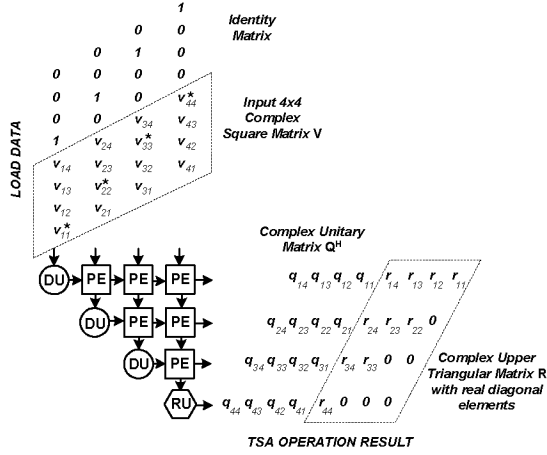


Figure 1. Block diagram of Triangular Systolic Array

Fig. 1 also illustrates the input/output signal organization of TSA/TACR on the example of a 4x4 matrix \mathbf{V} to be decomposed. According to TSA/TACR, the input matrix \mathbf{V} is loaded in the temporally (skewed) triangular shape and followed by the identity matrix to produce unitary matrix \mathbf{Q}^H at the TSA/TACR output. Note that the diagonal elements of input matrix \mathbf{V} are marked by token '*' which represents the control signal propagated together with the data element through the TSA/TACR and controls the mode (functionality) of PE and RU cells.

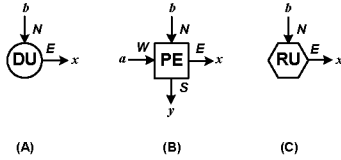


Figure 2. Triangular Systolic Array Units:
(A) – Delay Unit, (B) – Processing Element, (C) – Rotational Unit

A DU, see Fig. 2A, reads complex samples from the north input N , delays them (for period equal to PE operation time), and then passes to the east output E . The same functionality is valid for the control signal $*$.

A PE, see Fig. 2B, is the main signal processing cell of TSA/TACR. Its structure is based on the CORDIC processor that may be designed with one or more hardwired CORDIC modules. The number of used CORDIC blocks and their architecture may be chosen according to the TSA design constraint that is a tradeoff between TSA area and computation latency.

Each PE may operate in two different modes: Vectoring and Rotation. The PE mode is controlled by flag $*$. If the data sample carries flag $*$ and enters PE from the west port, then PE operates in Vectoring mode. In all other cases the Rotation mode is active.

Vectoring Mode (CORDIC modules of PE are configured to calculate angle and amplitude of input complex sample [3]): In Vectoring mode the PE takes complex samples a and b from west and north input ports W and N , respectively (see Fig. 2); then, it computes three angles $\theta_a = \arg(a)$, $\theta_b = \arg(b)$, $\theta_1 = \arctg(|b|/|a|)$, and stores results into three internal angle registers. In Vectoring mode the PE delivers magnitude $\sqrt{|a|^2 + |b|^2}$ at the east port E and a zero sample at south output S . Note that Vectoring mode requires usage of three CORDICs operations (see Fig. 3). In Vectoring mode the input control signal $*$ (which is present in the west port W) should be passed to the output port E (from West to East).

Rotation Mode (CORDIC modules of PE are configured to perform vector rotation in polar coordinates [3]): In Rotation mode the samples a and b (taken from west and north input ports respectively) are transformed to complex samples x and y according to:

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \cos(\theta_1) & \sin(\theta_1) \\ -\sin(\theta_1) & \cos(\theta_1) \end{bmatrix} \begin{bmatrix} \exp(-i\theta_a) & 0 \\ 0 & \exp(-i\theta_b) \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} \quad (11)$$

Here, θ_a , θ_b , and θ_1 are the angles computed and stored during the Vectoring mode of operation. The transformed complex samples x and y are sent to PE output ports E and S , respectively. To perform the transformation described by (11), 4 CORDIC operations are required (see Fig. 3). If the control signal $*$ is present at the north input port of PE, then it is passed to the south output (from North to South).

The RU, see Fig. 2C, is used to eliminate the complex lowest diagonal element of the output upper triangular matrix \mathbf{R} by means of additional rotation. It is based on one CORDIC module and also may operate in Vectoring or Rotation mode. If control signal $*$ appears at the north input, then RU operates in Vectoring mode.

In Vectoring mode, the RU takes complex sample b from north port N and applies CORDIC to compute angle $\theta_b = \arg(b)$ and magnitude $|b|$. As a result of the RU operation, the angle $\theta_b = \arg(b)$ is stored to an internal angle register $\theta_{reg} = \arg(b)$ and the magnitude $|b|$ is sent to the east output port E .

In Rotation mode, complex sample b from input N is rotated by angle θ_{reg} stored in the internal angle register. As a result, the complex sample $x = \exp(-i\theta_{reg}) \cdot b$ appears at the output port E .

IV. PE TIME COMPLEXITY - COMPARISON OF CGR AND TACR TSA ARCHITECTURES

It is clear that the TSA operation time required for the QRD is directly determined by the latency of PE cells. To illustrate the advantages of the proposed TSA/TACR architecture, the latencies of PE cells based on CGR and TACR approaches are compared. Note that in both approaches the PE may be realized by using one or more hardwired CORDIC modules. Fig. 3 shows the data flows of CGR and TACR when PE cells are realized with *two hardwired CORDIC modules* for both Vectoring and Rotation modes of operation. As it can be seen from Fig. 3, in both modes the suggested TACR PE internal architecture outperforms the CGR based PE architecture in terms of required CORDIC operations, and provides *full parallel usage* of all available hardware CORDIC resources (there are no idle CORDIC modules in any step of signal processing).

Table 1 summarizes the latency estimates for PE based on CGR and TACR approaches. Two possible PE embodiments with one and two hardwired CORDIC modules are considered. The PE latency is determined by taking into account two main factors:

N - number of clock cycles required by CORDIC, and N_{SFC} - number of clock cycles to make CORDIC Scale Factor Correction (SFC).

Note that for latency calculation we have used the fact that the SFC may be implemented by using the same CORDIC hardware resources, see, e.g., [3].

TABLE I. PE LATENCY FOR CGR AND TACR QRD

QRD approach	Latency of PE based on two hardwired CORDIC modules	
	Vectoring mode	Rotation mode
CGR	$3N+2N_{SFC}$	$3N+2N_{SFC}$
TACR	$2N+N_{SFC}$	$2N+N_{SFC}$
Latency improvement	$N+N_{SFC}$	$N+N_{SFC}$

It can be seen from Table 1 that the suggested TACR approach *reduces the TSA operation time up to 40 percent*, if it is assumed that N_{SFC} is equal to N . However, the best known value for N_{SFC} is $N/4$, see, e.g., [3], in this case the reduction is equal to 35 percent.

V. CONCLUSION

In this work the VLSI hardware-oriented TSA/TACR architecture for the QR decomposition of large size complex matrices is presented. It is demonstrated that based on the CORDIC algorithm a significant latency reduction of QRD may be achieved by applying the TACR approach instead of the widely used Complex Givens Rotation technique.

REFERENCES

- [1] Tim Zhong Mingqian, A. S. Madhukumary and Francois Chin "QRD-RLS adaptive equalizer and its CORDIC-based implementation for CDMA systems" // *International Journal on Wireless & Optical Communications* Vol. 1, No. 1, P. 25-39, 2003.
- [2] A.El-Amawy, K.R. Dharmarajan "Parallel VLSI algorithm for stable inversion of dense matrices" // *IEEE Proceedings*, Vol. 136, No. 6, P. 575-580, 1989.
- [3] J.H. Cavallaro, F.T. Luk "CORDIC arithmetic for SVD processor" // *Journal of Parallel and Distributed Computing*, Vol. 5, 271-290, 1988.

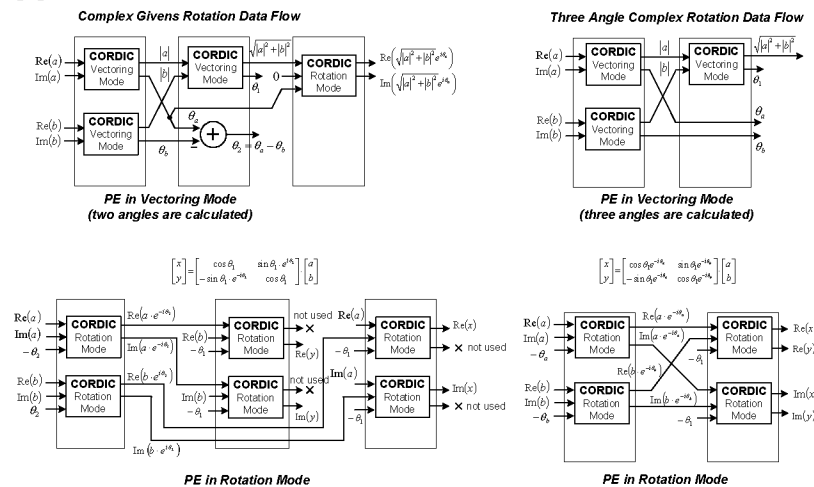


Figure 3. Comparison of flows required by PE based on CGR and TACR