Complexity Analysis of MMSE Detector Architectures for MIMO OFDM Systems

Markus Myllylä, Juha-Matti Hintikka, Joseph R. Cavallaro and Markku Juntti

University of Oulu, Centre for Wireless Communications

P.O. Box 4500, FIN-90014 University of Oulu, Finland

{markus.myllyla, juhintik, cavallar, markku.juntti}@ee.oulu.fi

Matti Limingoja, Aaron Byman

Elektrobit Ltd.

Tutkijantie 8, FIN-90570 Oulu, Finland

{matti.limingoja, aaron.byman}@elektrobit.com

Abstract— In this paper, a field programmable gate array (FPGA) implementation of a linear minimum mean square error (LMMSE) detector is considered for MIMO-OFDM systems. Two square root free algorithms based on QR decomposition (QRD) are introduced for the implementation of LMMSE detector. Both algorithms are based on QRD via Givens rotations, namely coordinate rotation digital computer (CORDIC) and squared Givens rotation (SGR) algorithms. Linear and triangular shaped array architectures are considered to exploit the parallelism in the computations. An FPGA hardware implementation is presented and computational complexity of each implementation is evaluated and compared.

I. INTRODUCTION

The ever increasing data rates in wireless communication systems require the use of large bandwidths. Orthogonal frequency division multiplexing (OFDM) [1] has become a widely used technique to significantly reduce receiver complexity in broadband wireless systems. Multiple-input multiple-output (MIMO) channels offer improved capacity and significant potential for improved reliability compared to single antenna channels [2]. In the case of rich scattering environment layered space-time (LST) architectures [3], [4] combined with channel coding represent pragmatic yet powerful methods to increase the user data rate in systems with multi-element antenna arrays (MEAs). MIMO techniques in combination with OFDM technique (MIMO-OFDM) have been identified as a promising approach for high spectral efficiency wideband systems [5], [6].

The OFDM technique drastically simplifies receiver design by decoupling the intersymbol interference, i.e., a frequency selective, MIMO channel into a set of parallel flat fading MIMO channels [6]. However, the reception of the MIMO-OFDM signal has to be performed separately for each subcarrier. The optimal joint detection and decoding for LST architectures would require the use of a maximum likelihood (ML) algorithm. However, the computational complexity of optimal ML decoding is beyond the limit of most systems, and, thus, such an approach is not feasible. A suboptimal approach is to use separate suboptimal solution steps for detection and decoding, such as zero forcing (ZF) and minimum mean square error (MMSE) criterion based methods [3]. In this paper, a linear MMSE (LMMSE) based detector is considered for MIMO-OFDM systems.

Several approaches exist to solve the matrix inversion required by the LMMSE detector [7], [8]. Often these methods include operations such as square root and division which are very complex in implementation and should, if possible, be avoided. In this paper, two square root free methods are introduced for the implementation of a LMMSE detector. Both algorithms are based on QR decomposition (QRD) via Givens rotations, namely the coordinate rotation digital computation (CORDIC) [9] algorithm and the squared Givens rotation (SGR) [10] algorithm.

Architectural design of matrix operations in the literature is often based on systolic array structures with communicating processing elements (PEs) [11], [12]. In this paper, detector architectures are presented and compared for 2×2 and 4×4 antenna systems. A fast and parallel architecture is considered for lower dimensional systems, and a less complex architecture with easy scalability and time sharing PEs is considered for larger systems. An FPGA hardware implementation is presented and the computational complexity of each implementation is evaluated and compared.

The paper is organized as follows. The system model is presented in Section II. The LMMSE detector and the proposed algorithms are introduced in Section III. The architectural design is presented in Section IV. The hardware implementation in FPGA is presented in Section V. Conclusions are presented in Section VI.

II. SYSTEM MODEL

An orthogonal frequency division multiplexing (OFDM) based multiple antenna system with N transmit antennas and M receive antennas is considered. A block diagram of the system is shown in Figure 1. The received signal can be expressed in terms of code symbol interval as

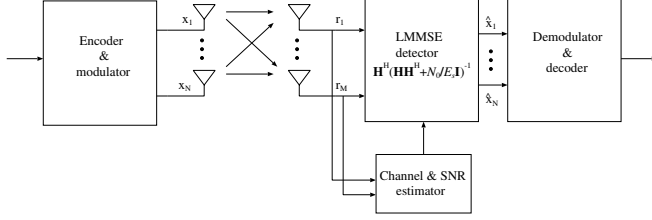$$\mathbf{r}_p = \mathbf{H}_p \mathbf{x}_p + \boldsymbol{\eta}_p, \quad p = 1, 2, \dots, P, \tag{1}$$

Fig. 1. Model of a MIMO system with N transmit and M receive antennas.

where P is the number of sub carriers and the received signal vector, the transmit symbol vector and the noise vector are defined in the frequency domain, respectively, as $\mathbf{r}_p = [r_{p,1}, r_{p,2}, \ldots, r_{p,M}]^T$, $\mathbf{x}_p = [x_{p,1}, x_{p,2}, \ldots, x_{p,N}]^T$, $\boldsymbol{\eta}_p = [\eta_{p,1}, \eta_{p,2}, \ldots, \eta_{p,M}]^T$. The elements of $\boldsymbol{\eta}_p$ are independent and complex Gaussian with equal power real and imaginary parts, i.e., $\boldsymbol{\eta}_p \sim \mathcal{CN}(\mathbf{0}, N_0\mathbf{I}_M)$ and represent the frequency domain thermal noise at the receiver. The channel matrix $\mathbf{H}_p \in \mathbb{C}^{M \times N}$ contains complex Gaussian fading coefficients with unit variance.

The LMMSE based detector [3] minimizes the MSE between the transmitted signal vector $\mathbf{x}_p$ and the soft output vector of the LMMSE front end $\hat{\mathbf{x}}_p = \mathbf{W}_p^H\mathbf{r}_p$. The design criterion is

$$\mathcal{D}^2_{\text{LMMSE}} = \min_{\mathbf{W}} \mathrm{E}\{\| \mathbf{x}_p - \mathbf{W}^H\mathbf{r}_p \|^2_F\}, \qquad (2)$$

where $\mathbf{W}_p$ is the coefficient matrix, $\mathbf{r}_p$ is the received signal vector, and $\|\mathbf{A}\|^2_F = \mathrm{tr}\left(\mathbf{A}\mathbf{A}^H\right)$ denotes a squared Frobenius norm of the matrix $\mathbf{A}$. By using the well known Wiener solution [13], the LMMSE detector for MIMO-OFDM can be then reduced to

$$\mathbf{W}_p = \left(\mathbf{H}_p\mathbf{R}_{xx}\mathbf{H}_p^H + \mathbf{R}_{\eta\eta}\right)^{-1}\mathbf{H}_p\mathbf{R}_{xx}. \qquad (3)$$

Because the LMMSE detector has no prior knowledge of the channel code structure, we assume $\mathbf{R}_{xx} = E_s\mathbf{I}_N$. The thermal noise between receive antennae and subcarriers is also considered to be uncorrelated, i.e., $\mathbf{R}_{\eta\eta} = N_0\mathbf{I}_M$. Then the solution of (3) becomes

$$\mathbf{W}_p = \left(\mathbf{H}_p\mathbf{H}_p^H + \frac{N_0}{E_s}\mathbf{I}_M\right)^{-1}\mathbf{H}_p. \qquad (4)$$

III. LMMSE DETECTOR

The calculation of the LMMSE solution in (4) requires a matrix inversion operation which is computationally a very complex task. The solution for the LMMSE front-end coefficients $\mathbf{W}_p$ can be seen as a common problem of solving a linear system

$$\mathbf{A}\mathbf{X} = \mathbf{B} \qquad (5)$$

where the matrix to be inverted, the desired LMMSE coefficients and the right hand side of the equation are defined, respectively, as $\mathbf{A} = \mathbf{H}_p\mathbf{H}_p^H + \frac{N_0}{E_s}\mathbf{I}_M \in \mathbb{C}^{M \times M}$, $\mathbf{X} = \mathbf{W}_p \in \mathbb{C}^{M \times N}$, $\mathbf{B} = \mathbf{H}_p \in \mathbb{C}^{M \times N}$.

In this paper, two square root free methods based on QRD via Givens rotations are considered for the calculation of LMMSE detector coefficients. The CORDIC algorithm is an iterative algorithm introduced by Volder [9]. For an overview, see [14]. The SGR [10] algorithm is developed based on the work by Gentleman and Hammarling [10, references in]. Some related work with SGR algorithm can be found, e.g., from [15], [16], [17].

A. QRD with CORDIC Algorithm

In QRD a symmetric positive definite matrix $\mathbf{A}$ from (5) can be factored as follows

$$\mathbf{A} = \mathbf{Q}\mathbf{R} \qquad (6)$$

where $\mathbf{Q} \in \mathbb{C}^{M \times M}$ is unitary matrix, i.e., $\mathbf{Q}^H\mathbf{Q} = \mathbf{Q}\mathbf{Q}^H = \mathbf{I}$ and $\mathbf{R} \in \mathbb{C}^{M \times M}$ is upper triangular matrix. The CORDIC method provides pipelined implementations of the Givens rotations for QRD using shifts and addition/subtractions without the need to compute trigonometric functions or square roots [9], [14]. Then (5) can be written as

$$\mathbf{Q}\mathbf{R}\mathbf{X} = \mathbf{B} \qquad (7)$$
$$\mathbf{R}\mathbf{X} = \mathbf{Q}^H\mathbf{B} \qquad (8)$$

The matrix $\mathbf{X}$ can be solved from upper triangular system using back substitution algorithm [7].

The two dimensional rotation step in Givens rotations annihilates one element at a time from the given appropriate pairs of rows. The rotation step is repeated several times for the matrix $\mathbf{A}$ in (6) in order to construct $\mathbf{R}$ and $\mathbf{Q}$. In one rotation step the kth element of the row $\mathbf{a} = [0, \ldots, 0, a_k, \ldots, a_M]$ is to be annihilated by the rotation. Another row $\mathbf{r} = [0, \ldots, 0, r_k, \ldots, r_M]$ is applied in order to obtain QRD. For real valued $\mathbf{a}$ and $\mathbf{r}$ the rotation is

$$\begin{bmatrix} \bar{\mathbf{r}} \\ \bar{\mathbf{a}} \end{bmatrix} = \begin{bmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} \mathbf{r} \\ \mathbf{a} \end{bmatrix}$$
$$= \cos(\theta) \begin{bmatrix} 1 & \tan(\theta) \\ -\tan(\theta) & 1 \end{bmatrix} \begin{bmatrix} \mathbf{r} \\ \mathbf{a} \end{bmatrix}, \qquad (9)$$

where θ is chosen so that $\bar{\mathbf{a}}_k = 0$. If the angle of θ is such that $\tan(\theta)$ is a power of 2, the multiplication can be done using only bit-shift operations. A general angle can be constructed as a series of such angles with the tangent value equal to the power of 2, and in practice the sum can be approximated with i_{max} values,

$$\theta = \sum_{i=0}^{\infty} \rho_i\theta_i \approx \sum_{i=0}^{i_{max}} \rho_i\theta_i, \qquad (10)$$

where $\rho_i = \{-1, +1\}$ and θ_i is constrained so that $\tan(\theta_i) = 2^{-i}$. [14]

The rotation in (9) is accomplished in a multistage manner, by a series of micro rotations. The micro rotations result in a series of intermediate results. The CORDIC implementation with i_{max} stages results from (9) as follows

$$\begin{bmatrix} \mathbf{r}^{[0]} \\ \mathbf{a}^{[0]} \end{bmatrix} = \kappa \begin{bmatrix} \mathbf{r} \\ \mathbf{a} \end{bmatrix}, \qquad (11)$$

76

$$\begin{bmatrix} \mathbf{r}^{[1]} \\ \mathbf{a}^{[1]} \end{bmatrix} = \begin{bmatrix} \mathbf{r}^{[0]} \\ \mathbf{a}^{[0]} \end{bmatrix} + \rho_0 2^0 \begin{bmatrix} -\mathbf{a}^{[0]} \\ \mathbf{r}^{[0]} \end{bmatrix}, \tag{12}$$

$$\vdots$$

$$\begin{bmatrix} \bar{\mathbf{r}} \\ \bar{\mathbf{a}} \end{bmatrix} = \begin{bmatrix} \mathbf{r}^{[i_{max}]} \\ \mathbf{a}^{[i_{max}]} \end{bmatrix} + \rho_{i_{max}} 2^{-i_{max}} \begin{bmatrix} -\mathbf{a}^{[i_{max}]} \\ \mathbf{r}^{[i_{max}]} \end{bmatrix}, \tag{13}$$

where $\kappa = \prod_{i=0}^{i_{max}} \cos(\theta_i)$ is a precomputed normalization constant and the sign of the micro rotation is determined by $\rho_i = \text{sgn}(r_k^{[i-1]})\text{sgn}(a_k^{[i-1]})$. [14]

The case of complex input data requires that the leading elements of two processed rows are made real. Thus, the typical step of the Givens approach can be replaced by a more complicated step involving three sub-steps as follows

$$\begin{bmatrix} \mathbf{r}' \\ \mathbf{a}' \end{bmatrix} = \begin{bmatrix} e^{-j\phi_r} & 0 \\ 0 & e^{-j\phi_a} \end{bmatrix} \begin{bmatrix} \mathbf{r} \\ \mathbf{a} \end{bmatrix}, \tag{14}$$

$$\begin{bmatrix} \bar{\mathbf{r}} \\ \bar{\mathbf{a}} \end{bmatrix} = \begin{bmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} \mathbf{r}' \\ \mathbf{a}' \end{bmatrix}, \tag{15}$$

where $\phi_r = \arctan\frac{\text{Im}(r_k)}{\text{Re}(r_k)}$, $\phi_a = \arctan\frac{\text{Im}(a_k)}{\text{Re}(a_k)}$ and $\theta = \arctan\frac{a'_k}{r'_k}$. The combination of four CORDIC elements can be applied to a supercell for complex data [14].

B. Squared Givens Rotations

The applied QR decomposition version in the SGR algorithm is different from that in (6) used in the CORDIC model. In the SGR algorithm, the factorization of a symmetric positive definite matrix $\mathbf{A}$ from (5) is expressed as follows

$$\mathbf{A} = \mathbf{Q}_A \mathbf{D}_R^{-2} \mathbf{U} \tag{16}$$

where $\mathbf{U} = \mathbf{D}_R \mathbf{R} \in \mathbb{C}^{M \times M}$ is upper triangular matrix, $\mathbf{D}_R = \text{diag}(\mathbf{R}) \in \mathbb{R}^{M \times M}$, $\mathbf{Q}_A = \mathbf{Q}\mathbf{D}_R \in \mathbb{C}^{M \times M}$. Matrix $\mathbf{Q}_A$ consists of the orthogonalized columns of the matrix $\mathbf{A}$. Now (5) can be written as follows

$$\begin{aligned} \mathbf{Q}_A^H \mathbf{A} \mathbf{X} &= \mathbf{Q}_A^H \mathbf{B} \\ \mathbf{D}_R \mathbf{Q}^H \mathbf{Q} \mathbf{D}_R \mathbf{D}_R^{-2} \mathbf{U} \mathbf{X} &= \mathbf{Q}_A^H \mathbf{B} \\ \mathbf{U}\mathbf{X} &= \mathbf{Q}_A^H \mathbf{B} \\ \mathbf{X} &= \mathbf{U}^{-1} \mathbf{Q}_A^H \mathbf{B}, \end{aligned} \tag{17}$$

where $\mathbf{X}$ is the desired coefficient matrix. [10]

The SGR algorithm is used to determine $\mathbf{Q}_A$ and $\mathbf{U}$ from $\mathbf{A}$ as in (16). The annihilation is done for one element at a time from appropriate pairs of rows as in (9). In the SGR algorithm, the selected pairs of rows $\mathbf{a}$ and $\mathbf{r}$ are first scaled as

$$\begin{aligned} \mathbf{u} &= r_k \mathbf{r} \\ \mathbf{a} &= w^{\frac{1}{2}} \mathbf{v}, \end{aligned} \tag{18}$$

where r_k is the kth element of $\mathbf{r}$ and given scalar $w > 0$. With the scaling in (18) only half of the multiplications and no square roots are required in the annihilation of a_k compared

to normal Givens rotations [10]. The rotation performed by the SGR algorithm is now

$$\begin{bmatrix} \bar{\mathbf{u}} \\ \bar{\mathbf{v}} \end{bmatrix} = \begin{bmatrix} 1 & wv_k \\ -\frac{v_k}{u_k} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \mathbf{v} \end{bmatrix}, \tag{19}$$

and $\bar{w} = wu_k/\bar{u}_k$. The relationship to (9) holds with representations

$$\begin{aligned} \bar{\mathbf{u}} &= \bar{r}_k \bar{\mathbf{r}} \\ \bar{\mathbf{a}} &= \bar{w}^{\frac{1}{2}} \bar{\mathbf{v}}. \end{aligned} \tag{20}$$

In the end of the annihilation process of the matrix $\mathbf{A} \in \mathbb{C}^{M \times M}$, we form an upper triangular matrix $\mathbf{U}$, i.e., $\mathbf{U} = \mathbf{D}_R \mathbf{R} = \text{diag}(\mathbf{R})\mathbf{R} \in \mathbb{C}^{M \times M}$. [10]

The desired coefficient matrix $\mathbf{X}$ is determined by calculating the inverse of matrix $\mathbf{U}$ and by multiplying both sides by $\mathbf{U}^{-1}$. The inversion of upper triangular matrix $\mathbf{U}$ can be performed using a stable algorithm listed in Table I [18]. It should be noted that inversion of upper triangular matrix $\mathbf{U}$ can also be calculated by back substitution algorithm. However, the algorithm listed in Table I is less complex in number of required operations [18], [19].

TABLE I

INVERSION OF TRIANGULAR MATRIX [18].

if $i = j$
$\quad \mathbf{U}_{ij}^{-1} = \frac{1}{\mathbf{U}_{jj}}$
else if $i < j$
$\quad \mathbf{U}_{ij}^{-1} = -\frac{1}{\mathbf{U}_{jj}} \sum_{m=1}^{j-1} \mathbf{U}_{im}^{-1} \mathbf{U}_{mj}^{-1}$
else if $i > j$
$\quad \mathbf{U}_{ij}^{-1} = 0$

IV. ARCHITECTURES

The architectural design of matrix operations in the literature is often based on systolic array structures with communicating processing elements (PEs) [11], [12]. The LMMSE detector coefficient matrix calculation in (4) requires several matrix operations such as matrix-matrix multiplications, QR decomposition, and back substitution or inversion of a triangular matrix. In this paper these operations have been implemented using systolic arrays.

The selected architecture is highly dependent on the specific application. In MIMO-OFDM system, the detector coefficients are calculated separately for each subcarrier and the dimensions of the calculated coefficient matrices are dependent on the number of transmit antennas. Thus, the complexity of the required operations depends mainly of the number of subcarriers and the number of antennas. The coefficients need to be updated as the channel changes, i.e., according to the channel coherence time. In this case the use of adaptive algorithms, such as recursive least squares (RLS) or least mean square (LMS), would require separate detectors for each subcarrier, and, thus, such approach is not feasible for an
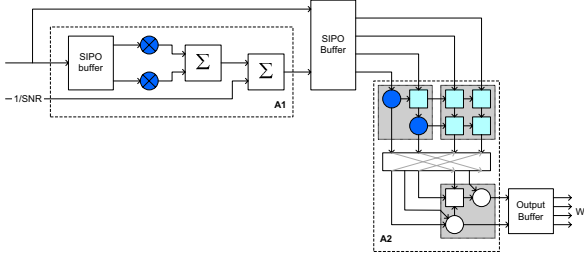
Fig. 2. The CORDIC based LMMSE detector architecture for 2×2 system.

Fig. 3. The CORDIC based LMMSE detector architecture for 4×4 system.

Fig. 4. The SGR based LMMSE detector architecture for 2×2 system.

OFDM system. In this paper, the detector is assumed to be used to calculate the solution in (4) for multiple subcarriers in the interval of channel coherence time.

The matrix-matrix multiplication can be implemented using a two dimensional systolic array architecture or a memory shared linear systolic array architecture [7]. The two dimensional array enables a fast and parallel dataflow whereas the linear array requires less resources in hardware implementation.

A traditional method for computing the QRD in literature is to use a simple and highly parallel triangular array architecture [11], [12]. Triangular array architecture enables simple data flow, high throughput with pipelining, and it is feasible for matrices with low dimensions, e.g., for 2×2 matrices. However, the architecture has certain drawbacks, such as a growing number of processing elements (PE's) needed with increasing matrix dimensions and, thus, lack of easy scalability. As an alternative structure a linear array architecture could be considered for larger systems. A derivation of linear QR array from triangular QR array has been presented, e.g., in [15], [20].

Both the algorithm for inversion of a triangular matrix [18] and the back substitution algorithm [7] can be implemented using a triangular array architecture. With increasing matrix dimensions, however, a linear array architecture could also be considered due to the growing complexity of triangular structure with increasing matrix dimensions. A linear array mapping of the triangular matrix inversion algorithm has been presented in [21].

A. CORDIC Based Detector Architecture

The CORDIC based LMMSE detector architecture for 2×2 system is illustrated in Figure 2. Matrix $\mathbf{A}$ from (5) is formed in part A1 using an array of complex multipliers and summation blocks. The matrices $\mathbf{A}$ and $\mathbf{B}$ from (5) are then input to part A2 which consists of two systolic arrays. In the upper CORDIC based systolic array of part A2 the calculation of the matrices $\mathbf{R}$ and $\mathbf{Q}^H\mathbf{B}$ from (8) is carried out. Then the lower systolic array applies the back substitution algorithm to form the desired matrix $\mathbf{X} = \mathbf{W}_p$.

The architecture presented in Figure 2 does not require much control logic and the mapping of data flow is relatively easy. The applied architecture is feasible for systems with rather low matrix dimensions, i.e., 2×2 antenna system.
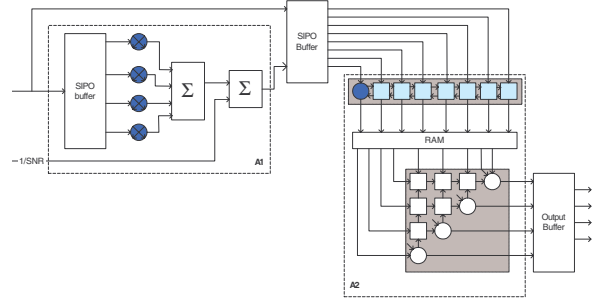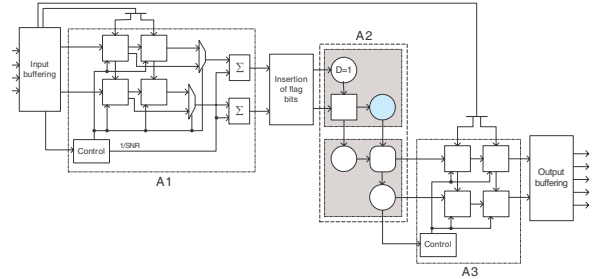
However, the complexity of the triangular array architecture grows dramatically with increasing matrix dimensions. Thus, a less complex architecture, such as linear array, should be used with higher matrix dimensions. The linear array requires more control logic and the overall delay for calculation of detector coefficients is higher, but the required complexity is less compared to the triangular array architecture. The CORDIC based LMMSE detector architecture for 4×4 system is illustrated in Figure 3. The QRD array is replaced with linear structure which is a less complex solution.

B. SGR Based Detector Architecture

The SGR based LMMSE detector architecture for the 2×2 system is presented in Figure 4. Two dimensional arrays are used for matrix multiplications and traditional triangular arrays for matrix inversion. The matrix $\mathbf{A}$ from (5) is calculated in part A1 using a two dimensional array. The matrix inversion by QRD and triangular matrix inversion is done in part A2 using triangular array architecture. The lower triangular array in part A2 also executes the calculation of $\mathbf{A}^{-1} = \mathbf{U}^{-1}\mathbf{Q}_A^H$. The two dimensional array in the A3 part calculates the matrix multiplication of terms $\mathbf{A}^{-1}$ and $\mathbf{B}$ in (17).

The architecture presented in Figure 4 is more suitable for systema with rather low matrix dimensions. A linear structure is also designed for systolic arrays with increasing matrix dimensions, e.g., 4×4 antenna system and larger. The SGR based LMMSE detector architecture for 4×4 antenna system is presented in Figure 5. A linear array architecture is applied for each part in Figure 5. The linear structure used for both matrix multiplications in parts A1 and A3 decreases the required number of processing elements from 16 to 4. Also the QRD
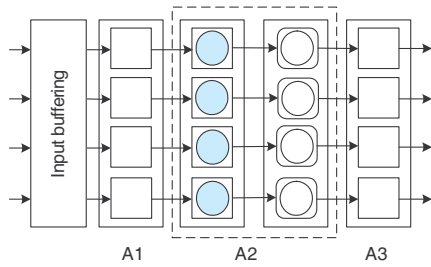
78

Fig. 5. The SGR based LMMSE detector architecture for 4×4 system.

and triangular matrix inversion arrays in part A2 are replaced with a linear array [15]. The linear array requires more control logic and the overall delay for calculation of detector coefficients is higher, but the complexity saving compared to a triangular array grows dramatically with increasing matrix dimensions. [15], [16], [21].

V. HARDWARE IMPLEMENTATION

An FPGA implementation of the detector architectures presented in Sections V-A and V-B has been done in a Xilinx Virtex-II XC2V6000 chip. Both implementations have been designed to be used with 66 MHz clock frequency, but the designs could also be modified for higher frequencies. The FPGA implementations of the detectors will be applied in Elektrobit Hiperlan-2 based OFDM testbed for 4G MIMO systems (EB4G) which consists of high-speed, FPGA-based programmable units. The EB4G supports configurations up to 4×4 MIMO and has flexible interfaces for digital and analog base band, IF and RF connections.

The CORDIC based detector was implemented in handwritten very high speed integrated circuits (VHSIC) hardware description language (VHDL) and functionally verified in ModelSim. The SGR based LMMSE detector architecture was developed and simulated in System Generator for DSP software tool from Xilinx. The tool provides high-level abstractions for Matlab Simulink environment that can be automatically compiled into VHDL. The tool also enables the importing of HDL modules into the Simulink-based design and co-simulating them using ModelSim.

A. CORDIC Based Detector

The CORDIC based QRD array is shown in Figure 6. The array contains two types of cells, the round vectoring cells and the square rotating cells. The round boundary cell performs so the vectoring operation, i.e., it computes the angles needed for annihilation of the incoming data samples. Two real CORDIC blocks are needed for complex implementation. The boundary cell sends the angle values to the inner square cells in the same row. The inner square cell calculates the new rotated sample values based on the angle values given from the boundary cell. Three real CORDIC blocks are needed for each block using complex numbers. The complexity of the CORDIC-array is determined by the number of CORDIC iterations and word length used [9], [14].
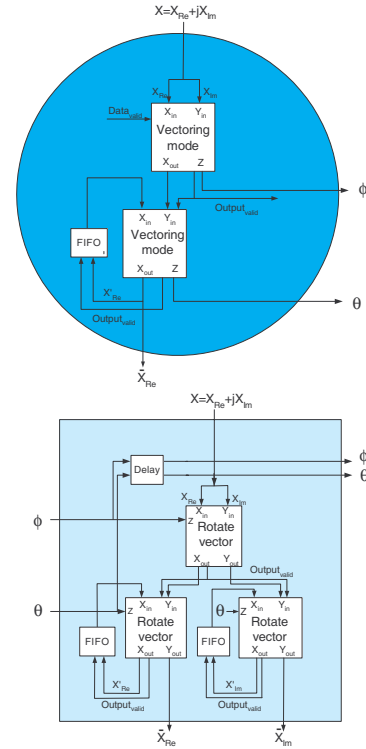

Fig. 6. Hardware realization of the CORDIC vectoring and rotating cells.

The back substitution array cells are illustrated in Figure 7. The triangular array structure includes two different types of cells. The boundary round cell performs a complex by real division operation. The division operation in the boundary cell is implemented using a reciprocal divider from Xilinx IP core library and two real multipliers. The inner cell contains a complex multiplication and an arithmetic subtraction operations. The overall complexity of the back substitution array is relatively low compared to the QRD array and it is dominated by the reciprocal divider blocks.

B. SGR Based Detector

The systolic array architecture for the SGR algorithm includes three different kind of cells as shown in Figure 4 for 2×2 system and in Figure 5 for 4×4 system. The data and the control signal flow and timing are omitted from the figures. In the architecture the round boundary cell is only a delay element except for the last darkened cell and the main operations of the SGR algorithm are executed in the square internal cell. It should be noted that all the cells in the linear array in Figure 5 include both the boundary cells and the square internal cell. Hardware realizations of the last round boundary cell and the square internal cell are presented in Figure 8. Each cell consists of arithmetic blocks such as divider, multipliers, adders, multiplexers, and registers. The darkened blocks and the bold lines depict complex signal representation. The complexity of the SGR array is dominated
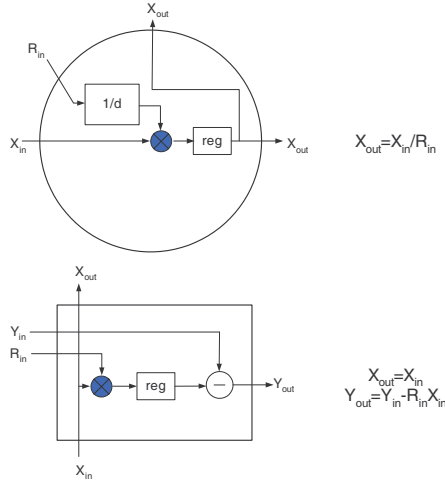
79

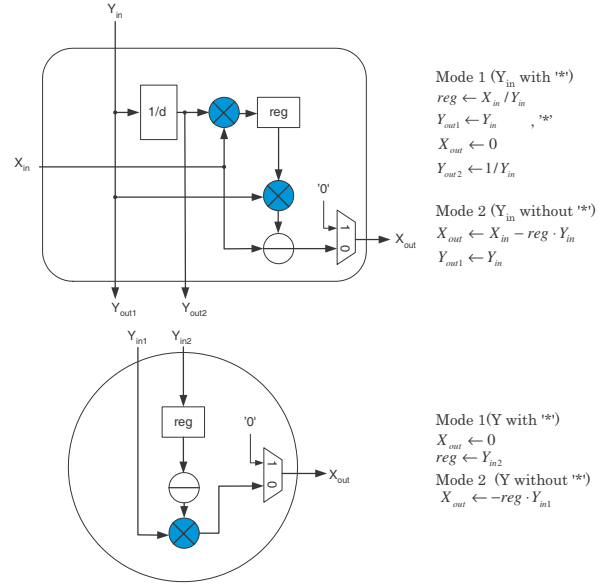Fig. 7. Hardware realization of the Back substitution array cells.

$X_{out} = X_{in}/R_{in}$

$X_{out} = X_{in}$
$Y_{out} = Y_{in} - R_{in} X_{in}$

Fig. 8. Hardware realization of the SGR array cells.

Mode1 (Y_{in} with '*')
$Y_{out1} \leftarrow Y_{in} \cdot Y_{in} + X_{in} \cdot X_{in}$
$X_{out} \leftarrow 0$
$r_k \leftarrow Y_{in}, v_k \leftarrow X_{in}$
$w_{out} \leftarrow \dfrac{Y_{in} \cdot Y_{in}}{Y_{out1}}$
$Y_{out2} = 1/Y_{out1}$

Mode2 (Y_{in} without '*')
$X_{out} \leftarrow -\dfrac{v_k^*}{r_k} \cdot Y_{in} + X_{in}$
$Y_{out1} \leftarrow r_k \cdot Y_{in} + v_k \cdot X_{in}$

Mode1 (X_{in} with '*')
$Y_{out} \leftarrow w_{in} \cdot X_{in} \cdot X_{in}$
$r_k^* \leftarrow X_{in}$

Mode2 (X_{in} without '*')
$Y_{out} \leftarrow w_{in} \cdot r_k^* \cdot X_{in}$

Mode 1 (Y_{in} with '*')
$reg \leftarrow X_{in}/Y_{in}$
$Y_{out1} \leftarrow Y_{in}$, '*'
$X_{out} \leftarrow 0$
$Y_{out2} \leftarrow 1/Y_{in}$

Mode 2 (Y_{in} without '*')
$X_{out} \leftarrow X_{in} - reg \cdot Y_{in}$
$Y_{out1} \leftarrow Y_{in}$

Mode 1(Y with '*')
$X_{out} \leftarrow 0$
$reg \leftarrow Y_{in2}$

Mode 2 (Y without '*')
$X_{out} \leftarrow -reg \cdot Y_{in1}$

Fig. 9. Hardware realization of the triangular matrix inversion array cells.

TABLE II
CORDIC BASED DETECTOR, DEVICE UTILIZATION FOR XC2V6000.

Resource	2×2 (Usage %)	4×4 (Usage %)	Available
CLB Slices	11910 (35.2%)	16805 (49.7%)	33792
Block RAMs	6 (4.2%)	101 (70.1%)	144
Block Multipliers	20 (13.9%)	44 (30.5%)	144

executed in the square internal cell.

C. Area Reports and Comparison of Designs

The area reports include only the LMMSE coefficient matrix calculation in (4), which dominates the total complexity of the LMMSE detector. The filtering operation, i.e., a matrix-vector multiplication, is omitted from the area reports due to similar design in both detector implementations.

The device utilization with CORDIC based LMMSE detector implementation for 2×2 and 4×4 systems has been listed in Table II. The CORDIC algorithm is implemented with $i_{max} = 10$ iterations and 16 bit internal word length. It should be noted that the number of iterations and word length may be decreased depending on the required accuracy. A design with $i_{max} = 7$ iterations and 12 bit internal word length would require approximately 30% less slices in synthesis. The latency for the 2×2 and 4×4 coefficient calculation is 685 and 3000 clock cycles, respectively.

The device utilization with SGR based LMMSE detector implementation for 2×2 has been listed in Table III. The architecture for 4×4 system has not yet been implemented in an FPGA. However, the estimate for the required complexity is approximately in the same ratio as with CORDIC based implementation. The SGR algorithm has been implemented using maximum internal word length of 19 bits. The latency for the 2×2 coefficient calculation is 415 clock cycles.

It can be noted from the area reports in Tables II and III that the CORDIC based design requires more slices and less

by the complex reciprocal divider block which is executed in the square internal cell. The complex reciprocal divider was brought from Xilinx IP core library to System Generator design as a black box.

The triangular matrix inversion array cells are presented in Figure 9. The array includes two different kinds of cells. It should be noted that the linear array architecture cells include the operations of both cells. The design is based on [18] with certain simplifications. The round cells do not need to calculate the reciprocal division operation, because the required operation is already calculated in another cell and can be directed to the cell as Y_{in2}. This decreases the complexity of the array significantly. The complexity of the array is dominated by the reciprocal divider block which is

Resource	2×2	Available	Utilization
CLB Slices	6305	33792	18.7 %
Block RAMs	8	144	5.6 %
Block Multipliers	59	144	41.0 %

block multipliers compared to the SGR based design. This is due to normal arithmetic applied in the SGR algorithm and the rotation based arithmetic applied in CORDIC algorithm. Also the required word lengths with fixed-point arithmetic are a little higher with SGR based design.

D. Comparison of Design Methodologies

The System Generator tool provides high-level abstractions for the Matlab Simulink environment that can be automatically compiled into VHDL. It was noted that the System Generator is a good tool for implementation of direct data flow applications. It provides a graphical user interface (GUI) with ready blocks which makes it easy to start with for persons unexperienced with VHDL design. The tool also enables the importing of HDL modules to the design and verification can be done using Matlab/Simulink generated test vectors. It was noted that the generated VHDL is rather good compared to handwritten VHDL. However, the design of control logic is not very simple with the tool.

There are still some benefits in writing the VHDL code by hand. The handwritten VHDL code is optimal in the sense that one can be sure what is got as an output. However, the optimality of the design is still highly dependent on the expertise of the designer. The learning time for an unexperienced person for such an approach may be rather long. The design time for architecture and hardware implementation was approximately the same with both design methods.

VI. CONCLUSIONS

Two FPGA implementations of a LMMSE detector were considered based on the CORDIC and SGR algorithms for MIMO-OFDM systems, where the detector complexity and the number of required operations depend mainly of the number of subcarriers and the number of antennas. The detector architecture solutions were presented and compared for 2×2 and 4×4 antenna systems. A fast and parallel architecture was considered for lower dimensional systems, and a less complex architecture with easy scalability and time sharing PEs was considered for larger systems.

The FPGA hardware implementations for both detectors were presented and the computational complexity of each implementation was evaluated and compared. The CORDIC based implementation was found to require more slices and less block multipliers compared to SGR based design. This is due to the normal arithmetic applied in the SGR algorithm and rotation based arithmetic applied in CORDIC algorithm.

The SGR based detector was designed using the System Generator for DSP tool and the CORDIC based detector was designed using handwritten VHDL. The two design methods were compared during the work. It can be noted that the System Generator based flow is useful especially if the designers are unexperienced with VHDL design.

REFERENCES

[1] L. J. Cimini, "Analysis and simulation of a digital mobile channel using orthogonal frequency division multiplexing," *IEEE Trans. Commun.*, vol. 33, no. 7, pp. 665–675, July, 1985.

[2] A. Goldsmith, S.A. Jafar, N. Jindal, and S. Vishwanath, "Capacity limits of MIMO channels," *IEEE Journal on Selected Areas in Communications*, vol. 21, no. 5, pp. 684–702, June, 2003.

[3] G.J. Foschini, "Layered space–time architecture for wireless communication in a fading environment when using multi-element antennas," pp. 41–59, Aug. 1996.

[4] G. J. Foschini, G. D. Golden, R. A. Valenzuela, and P. W. Wolniansky, "Simplified processing for high spectral efficiency wireless communication employing multi-element arrays," *IEEE J. Select. Areas Commun.*, vol. 17, no. 11, pp. 1841–1852, Nov. 1999.

[5] H. Bölcskei, D. Gesbert, and A. J. Paulraj, "On the capacity of OFDM based spatial multiplexing systems," *IEEE Trans. Commun.*, vol. 50, no. 2, pp. 225–234, Feb., 2002.

[6] H. Yang, "A road to future broadband wireless access: MIMO-OFDM-based air interface," *Communications Magazine, IEEE*, vol. 43, no. 1, pp. 53–60, Jan., 2005.

[7] G. H. Golub and C. F. Van Loan, *Matrix Computations, 3rd ed.*, The Johns Hopkins University Press, Baltimore, 1996.

[8] S. Haykin, *Adaptive Filter Theory*, Prentice Hall, Englewood Cliffs, NJ, USA, 2nd edition, 1991.

[9] J.E. Volder, "The CORDIC Trigonometric Computing Technique," *IRE Trans. on Electronic Computers*, vol. EC-8, no. 3, pp. 330–4, 1959.

[10] R. Döhler, "Squared Givens Rotation," *IMA Journal of Numerical Analysis*, vol. 11, pp. 1–5, 1991.

[11] W.M. Gentleman and H.T. Kung, "Matrix triangularization by systolic array," *Proc. SPIE, Real-time signal processing IV*, vol. 298, pp. 19–26, Bellingham, Washington, 1981.

[12] S.Y. Kung, *VLSI Array Processors*, Prentice-Hall, 1987.

[13] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*, Prentice-Hall, Englewood Cliffs, NJ, USA, 1993.

[14] Y. H. Hu, "CORDIC-based VLSI architechtucres for digital signal processing," *IEEE Signal Processing Mag.*, vol. 9, no. 3, pp. 16–35, July 1992.

[15] F. Edman and V. Öwall, "An FPGA implementation of a matrix inversion architecture for multiple antenna algorithms," in *Nordic Radio Symposium*, Oulu, Finland, 2004.

[16] R. W. M. Smith R. L. Walke and G. Lightbody, "Architectures for adaptive weight calculation on ASIC and FPGA," in *33rd Proc. on Asilomar Conference*, Oct. 1999, vol. 2, pp. 1375–1380.

[17] M. Karkooti, J. R. Cavallaro, and C. Dick, "FPGA Implementation of Matrix Inversion Using QRD-RLS Algorithm," in *Proc. on 2005 Asilomar conference*, Pacific Grove, USA, Oct. 30 - Nov. 2. 2005.

[18] A. El-Amawy amd K.R. Dharmarajan, "Parallel VLSI algorithm for stable inversion of dense matrices," *IEE Proceedings*, vol. 136, 1989.

[19] M. Myllylä, M. Vehkaperä, and M. Juntti, "Complexity Evaluation of MMSE Based Detector for LST Architectures," in *In Proc. IEEE International Workshop on Convergent Technologies (IWCT'05)*, Oulu, Finland, 6-10 June, 2005.

[20] G. Lightbody, R. Walke, R. Woods, and J. McCanny, "Linear QR Architecture for a Single Chip Adaptive Beamformer," *Journal of VLSI Signal Processing Systems*, vol. 24, no. 1, pp. 67–81, 2000.

[21] F. Edman and V. Öwall, "Implementation of a scalable matrix inversion architecture for triangular matrices," *14th Proc. on PIMRC'03*, vol. 3, pp. 2558–2562, 7-10 Sep. 2003.