CORDIC-Based MMSE-DFE Coefficient Computation

Naofal Al-Dhahir *and Ali H. Sayed †

Abstract

A modular parallel architecture for a MMSE–DFE coefficient computation processor is presented. The architecture is based on **QR** factorization of a channel–and–noise–dependent data matrix and is implemented using CORDIC processors within a systolic array architecture. Implementation issues including the number of CORDIC stages and the bit precision required in a fixed–point implementation are investigated through computer simulations. The proposed architecture accommodates fractionally–spaced DFEs, co–channel interference, and multiple diversity paths.

1 Introduction

In many wireless packet data communication systems, a training sequence is embedded in each packet and used at the receiver to estimate the channel impulse response (CIR). The CIR estimate is then used to compute the optimal equalizer settings or the branch metrics of a maximum likelihood sequence estimator (MLSE) to mitigate inter-symbol interference (ISI) and noise.

On severe-ISI channels, such as those encountered in GSM–based digital cellular systems and the emerging broadband wireless services, the mean-square–error decision feedback equalizer (MMSE-DFE) has been demonstrated to be a high-performance receiver structure [4, 11, 12] and can have a much lower complexity than MLSE receivers (whose complexity grows exponentially with the channel delay spread). In particular, a channel-estimate-based MMSE-DFE has been shown to

*N. Al-Dhahir is with GE Corporate R& D Center, Niskayuna, NY 12309. Contact Author. Email : aldhahir@crd.ge.com, FAX (518) 387-4042

†A.H. Sayed is with the EE Dept., University of California, Los Angeles, CA 90095.

offer undeniable performance advantages over an adaptive LMS–based MMSE-DFE which suffers from slow convergence on severe-ISI channels (due to the large eigenvalue spread problem) [15, 17]. On the other hand, the CIR estimation problem is much better conditioned because of the "good correlation" property of the training sequence.

The main factor limiting the wide-spread use of channel-estimate-based MMSE-DFE's in practical systems has been the high-complexity in computing the optimal MMSE-DFE coefficients from the CIR coefficients in real-time which results in unacceptable power consumption, delay, and storage requirements for small-size, light-weight, and cost-effective wireless receivers.

In this paper, we present a low-complexity, numerically-stable, and parallelizable algorithm for computing the optimum coefficients of the MMSE-DFE from the CIR coefficients and the noise auto-correlation sequence. The main focus will be on the case of a single antenna with an uncorrelated input and white noise; however, we also show how to accommodate colored noise, co-channel interference, and multiple-antenna diversity reception.

The remainder of this paper is organized as follows. Section 2 presents the **QR**–factorization–based algorithm for computing the MMSE–DFE coefficients, investigates its robustness under fixed–point implementation, and estimates its computational complexity. A parallel systolic architecture for implementing the algorithm using CORDIC processors is given in Section 3. Extensions of the algorithm are discussed in Section 4 and the paper is concluded in Section 5.

2 QR-Based MMSE-DFE Coefficient Computation

We start in this section by describing the input–output model assumed. Then, we show how the optimum coefficient sof the finite–length MMSE–DFE can be computed from the **QR** factorization of a channel–and–noise–dependent matrix. This is followed by an investigation, through computer simulations, of the performance of the proposed MMSE–DFE computation algorithm with fixed–point implementation. The section is concluded by analyzing the computational requirements of the **QR**–based DFE computation algorithm.

2.1 Input-Output Model

We adopt the standard discrete–time representation of a linear additive–noise intersymbol interference (ISI) channel given by

$$\mathbf{y}_k = \sum_{m=0}^{\nu} \mathbf{h}_m x_{k-m} + \mathbf{n}_k \ , \tag{1}$$

where $\mathbf{h}_m \stackrel{def}{=} \left[\begin{array}{ccc} h_{l-1,m} & \cdots & h_{0,m} \end{array} \right]^t$, the notation $(.)^t$ denotes the transpose, is the m^{th} (vector) channel coefficient, assuming an oversampling factor of l, and ν is called the channel memory. Both the input sequence, $\{x_k\}$, and the noise sequence, $\{\mathbf{n}_k\}$, are assumed to be complex, zero–mean, and have (non–singular) auto–correlation matrices denoted by $\mathbf{R}_{xx}$ and $\mathbf{R}_{nn}$, respectively.

Over a block of N output symbols $\mathbf{y}_{k+N-1:k} \stackrel{def}{=} \left[\begin{array}{cccc} \mathbf{y}_{k+N-1} & \mathbf{y}_{k+N-2} & \cdots & \mathbf{y}_k \end{array} \right]^t$, we can write Equation (1) in vector notation as follows

$$\mathbf{y}_{k+N-1:k} = \mathbf{H}\mathbf{x}_{k+N-1:k-\nu} + \mathbf{n}_{k+N-1} \ ,$$

where $\mathbf{H}$ is a fully–windowed Toeplitz matrix whose first row is given by $\left[\begin{array}{ccccc} \mathbf{h}_0 & \mathbf{h}_1 & \cdots & \mathbf{h}_\nu & \mathbf{0}_{l\times(N-1)} \end{array} \right]$.

2.2 The Finite-Length MMSE-DFE

The MMSE–DFE consists of two filters : a fractionally–spaced feedforward filter that has (lN) taps and is denoted by the vector $\mathbf{w}^*$, and a symbol–spaced causal feedback filter that consists of $N_b(\leq \nu)$ taps denoted by $\{-b_1, -b_2, \cdots, -b_{N_b}\}$. For analytical convenience, we define the augmented vector $\tilde{\mathbf{b}}^* \stackrel{def}{=} \left[\begin{array}{ccccc} \mathbf{0}_{1\times\Delta} & 1 & b_1^* & \cdots & b_{N_b}^* \end{array} \right]$, where Δ $(0 \leq \Delta \leq N+\nu-1)$ is the decision delay , and $(.)^*$ denotes the conjugate transpose.

Consider the Cholesky factorization [5]

$$\mathbf{G} \stackrel{def}{=} \mathbf{R}_{xx}^{-1} + \mathbf{H}^*\mathbf{R}_{nn}^{-1}\mathbf{H} \stackrel{def}{=} \mathbf{R}^*\mathbf{R} \ , \tag{2}$$

where $\mathbf{R}$ is an upper–triangular matrix. We showed in [2] that the optimum FIR MMSE–DFE settings are given by

$$\tilde{\mathbf{b}}_{opt.} \ = \ \frac{\mathbf{R}^*\mathbf{e}_{(\Delta_{opt.}+1)}}{\mathbf{R}(\Delta_{opt.}+1, \Delta_{opt.}+1)} \tag{3}$$

$$\mathbf{w}^*_{opt.} = \frac{\mathbf{e}^*_{(\Delta_{opt.}+1)}\mathbf{R}^{-*}\mathbf{H}^*\mathbf{R}^{-1}_{nn}}{\mathbf{R}(\Delta_{opt.}+1, \Delta_{opt.}+1)} , \tag{4}$$

where $\mathbf{e}_i$ denotes the i^{th} unit column vector and $\mathbf{R}(i,j)$ denotes the (i,j) element of $\mathbf{R}$. The unbiased decision–point SNR of the MMSE–DFE is given by [8]

$$SNR_{MMSE-DFE,U} = |\mathbf{R}(\Delta_{opt.}+1, \Delta_{opt.}+1)|^2 - 1 . \tag{5}$$

The optimum delay, $\Delta_{opt.}$, that maximizes $SNR_{MMSE-DFE,U}$ is given by

$$\Delta_{opt.} = (argmax_{1\le i\le N+\nu}\{\mathbf{R}(i,i)\}) - 1 . \tag{6}$$

2.3 QR Factorization

To make the Cholesky factorization parallelizable, we perform a **QR** factorization [5] of the following augmented channel matrix

$$\tilde{\mathbf{H}} \stackrel{def}{=} \begin{bmatrix} \mathbf{R}_{xx}^{\frac{-*}{2}} \\ \mathbf{R}_{nn}^{\frac{-*}{2}}\mathbf{H} \end{bmatrix} = \mathbf{QR} , \tag{7}$$

which implies that $\mathbf{G} = \tilde{\mathbf{H}}^*\tilde{\mathbf{H}} = \mathbf{R}^*\mathbf{R}$. The following remarks are in order

- For the case of white input (or if the transmit filter shaping effect is absorbed in the channel impulse response) and white noise, we have $\tilde{\mathbf{H}} = \begin{bmatrix} \frac{1}{\sqrt{S_x}}\mathbf{I}_{N+\nu} \\ \frac{1}{\sqrt{N_0}}\mathbf{H} \end{bmatrix}$, where S_x and N_0 are the input energy and noise power spectral density per complex dimension, respectively. We focus on this case for the remainder of the paper, unless otherwise stated.

- **QR** factorization is numerically stable and requires less numerical precision than direct matrix inversion of an augmented auto–correlation matrix which is commonly used to compute DFE coefficients [14, 12]. In addition, it can be implemented in parallel which makes it attractive for VLSI implementation.

- It can be seen from (4) that computing the optimum feedforward filter requires

1. Calculating a row of $\mathbf{R}^{-*}$ which can be done by back substitution [5]. Denote this row by $\mathbf{s}$, then we have $\mathbf{R}\mathbf{s} = \mathbf{e}_{\Delta_{opt}+1}$. The size of this upper–triangular system of equations can be reduced from $(N+\nu)$ to $(\Delta_{opt}+1)$ since $\mathbf{s}(i) = 0$ for $(\Delta_{opt}+2) \leq i \leq (N+\nu)$.
2. The vector $\mathbf{s}$ is then multiplied by the fully–windowed Toeplitz matrix $\mathbf{H}^*$ and then scaled by 2 constants.

Figure 1 summarizes the computations needed to calculate the MMSE–DFE filters.

- For a time–varying channel, the CIR is continuously estimated, using a training sequence embedded in each block, the matrix $\tilde{\mathbf{H}}$ is updated, and the $\mathbf{QR}$ factorization is performed to update the MMSE–DFE filter settings.

The orthogonal transformation $\mathbf{Q}$ that triangularizes $\tilde{\mathbf{H}}$ can be implemented in a variety of ways, the most popular of which is Givens rotations [5] due to its suitability for parallel implementation. Givens rotations are a sequence of unitary transformations that rotate a vector to lie along the x–axis. Each of these transformations operates on a 2×1 vector and performs the rotation

$$\left[\begin{array}{cc} a_1 & a_2 \end{array} \right] \mathbf{\Theta} = \left[\begin{array}{cc} \pm\sqrt{|a_1|^2 + |a_2|^2} & 0 \end{array} \right] .$$

The rotation matrix $\mathbf{\Theta}$ has the form

$$\mathbf{\Theta} = \left[\begin{array}{cc} C & -S \\ S^* & C \end{array} \right] , \tag{8}$$

where the so–called Cosine and Sine parameters are defined by

$$C \stackrel{def}{=} \frac{1}{\sqrt{1+|\rho^2|}} \; ; \; S \stackrel{def}{=} \frac{\rho}{\sqrt{1+|\rho^2|}} ,$$

and the rotation angle is given by $\phi = \tan^{-1}\rho \stackrel{def}{=} \tan^{-1}\frac{b}{a}$.

2.4 Effects of Fixed-Point Implementation

In this subsection, we investigate the degradation in the decision–point SNR of the MMSE–DFE when its filters are computed in fixed–point using the Givens–rotations–based $\mathbf{QR}$ algorithm.

The two main reasons for implementing the algorithm on fixed–point rather than floating–point programmable DSP chips are the former's lower power consumption and smaller size both of which are attractive features for wireless applications.

The channel impulse responses assumed in the simulation are snapshots generated according to Jake's famed model for Rayleigh fading dispersive channels [16]. The baseline case is depicted in Figure 2 and is a 2–tap channel with an input SNR of 10 dB and 8 feedforward taps in the DFE. Under these conditions, a 10–bit representation results in less than 0.1 dB performance loss. To maintain this same performance level on a 4–tap channel (corresponds to a delay spread of $\approx 15\mu$sec at GSM bit rates), 12 bits of precision are needed as shown in Figure 3. The required bit precision also increases at higher SNR because of the increased dynamic range; Figure 4 shows that at an input SNR =30 dB, 16 bits of precision are needed. This shows that the required bit precision of the Givens–based MMSE–DFE coefficient computation algorithm for GSM scenarios is supportable by today's commercial fixed–point DSP chips. Next, we examine the delay and computational complexity requirements of the algorithm.

2.5 MIPS Estimate

Figure 5 shows a MIPS estimate of the MMSE–DFE coefficient computation algorithm as summarized in Figure 1. This estimate includes the number of complex multiplies only (assuming that a real add–multiply operation can be performed in one instruction), assumes that each complex multiply is equivalent to 4 real multiplies, and that the DFE coefficients are updated every 5 msec. This update interval was chosen to be approximately equal to the GSM frame length of 4.6 msec so that the CIR coefficients are held constant for the duration of the frame. It can be seen from the Figure that the MIPS estimate becomes quite substantial for severe–ISI channels and long DFE filters. As an example, for $\nu = 4$ (which corresponds to a delay spread of ≈ 15 microsec at the GSM bit rate) and $N = 30$, the complexity is around 60 MIPS. This estimate is for a *symbol-spaced* FFF (MIPS estimate increases approximately linearly with oversampling factor) and does not include the computations involved in computing the estimated CIR and input SNR and any I/O overhead. In addition, it does not include the complexity involved in computing the square roots (involved in the Sine and Cosine parameter calculation) which could be significant even if implemented using table look-ups. Even with all of these extra computational requirements excluded, Figure 6 clearly

shows that the time required to compute the DFE coefficients on a single 25–MIP processor could still become much longer than the coefficient update interval which would cause delays in applying the DFE weights to equalize the data (thus degrading performance) and increases storage requirements. For the GSM system, each 4.6 msec frame is shared by 8 users. Therefore, the processing time at the mobile should be upper–bound by the frame duration to allow for real–time processing; this restricts the number of MMSE–DFE coefficients that can be used.

The presence of co–channel interference, which is a performance–limiting impairment in TDMA digital cellular systems, further increases the computational complexity (c.f. Section 4) and hence the required time to compute the DFE coefficients. Finally, for many of the emerging broadband wireless services, the channel delay spread could last up to 100 symbol periods [4] resulting in prohibitively long processing delays in computing the DFE coefficients and hence degrading performance appreciably. This effect becomes even more pronounced on rapidly time–varying channels (e.g. high Doppler rates on trains), where the coefficient update interval is reduced to keep up with the channel variations resulting in a higher DFE computational complexity. Depending on a number of factors including time–to–market considerations, production volume/cost issues, power consumption and area restrictions, the additional processing power needed to compute the MMSE–DFE coefficients within the allowable time update interval can be attained either by partitioning the computations over multiple DSP processors or by an ASIC implementation. A key element in the successful implementation of the DFE computation algorithm using either approach is **parallelizability**.

3 Parallel Implementation

This section shows how the algorithm of Section 2 can be implemented in parallel using systolic arrays. In addition, the use of CORDIC processors to perform data triangularization is investigated by computer simulations.

3.1 Systolic Implementation

In this section, we present a *systolic* implementation of the MMSE–DFE coefficient computation processor where the rows of $\mathbf{R}$ (that determine the feedback filter) are computed in parallel. We

implement the orthogonal transformation $\mathbf{Q}$ as a 2×2 Givens rotation matrix whose entries are completely determined by the channel coefficients $\{\mathbf{h}_i\}_{i=0}^{\nu}$ and the input SNR. For each column in $\tilde{\mathbf{H}}$, we *pivot* on the element equal to $\frac{1}{\sqrt{SNR}}$ and annihilate the elements corresponding to the channel coefficients. We need to apply $N(\nu+1)$ Givens rotations to put $\tilde{\mathbf{H}}$ in upper–triangular form; only $(\nu+1)$ of these rotations are distinct. This systolic implementation is shown in Figure 7 for the case of $N+\nu = 5$. There are two types of cells : boundary cells (denoted by a circle) and internal cells (denoted by a rectangle). The functionality of these cells is defined in Figure 8. The number r represents the value stored in the cell. The boundary cells *compute* the rotation parameters needed to annihilate the lower–triangular portion of the incoming data while the internal cells *apply* these rotation parameters to the cell contents to compute the updated upper–triangular portion of the array. Each cell receives data from directions indicated in one clock cycle, performs specified arithmetic computations, and on the next clock cycle delivers the resulting output to its neighboring cells as indicated. In the sequel, we enumerate the steps involved in computing the DFE coefficients with the systolic array architecture for the case of white input and noise.

Algorithm (White Input and Noise Case)

Input :

CIR coefficients, Input SNR, number of feedforward filter taps.

Recursions :

1. Initialize the upper–triangular systolic array to $\frac{1}{\sqrt{SNR}}\mathbf{I}_{N+\nu}$.

2. Start feeding the array with the CIR coefficients in a time–skewed manner.

3. After all N rows of $\mathbf{H}$ have passed through the systolic array, the array cell contents will be the desired upper–triangular Cholesky factor $\mathbf{R}$.

4. Optimum feedback filter coefficients are given by the row of the systolic array that has the *largest* leading element.

5. As mentioned in Section 2.3, calculating the optimum feedforward filter involves solving a triangular system of equations by back substitution. This computation can be performed either on a DSP or by augmenting the triangular systolic array with a linear section as described in [6].

6. Freeze and apply DFE coefficients to buffered received data to equalize it.

7. Recompute CIR and DFE coefficients according to the desired update rate.

We conclude this section by summarizing the main advantages of the systolic implementation :

- Parallel implementation speeds up the DFE coefficient computation.
- Modular structure suitable for VLSI implementation, which results in lower power consumption and lower cost for high–volume production.
- All cells of systolic array are operated *synchronously*, i.e., using the same clock. Moreover, data is passed to neighboring cells only, hence, there is no need for a global bus.
- Data is fed to array in a skewed manner so that it arrives at the right cell at the right time. Therefore, computations are performed in a pipelined fashion and there is no need for storage.
- The systolic architecture is readily *scalable* which makes it straightforward to change the number of DFE coefficients.

Note that the boundary cells require computation of divisions and square roots. While these operations can be performed using table look–ups or polynomial approximations, they require a fairly large number of instruction cycles; higher throughput is achieved using a COordinate Rotation DIgital Computer (CORDIC) implementation.

3.2 CORDIC Implementation

A high–throughput implementation of the **QR** factorization is possible through CORDIC cells [7]. In this case, the rotation angle is decomposed into the weighted sum of pre–determined elementary rotation angles where rotation through each of them is implemented as simple shift–and–add operations. As in the systolic implementation, the CORDIC cells on the main diagonal of the upper–triangular array compute the rotation parameters (so–called *rotation mode*) and pass them to the internal CORDIC cells that apply these rotation to triangularize the data (so–called *vectoring mode*). For a comprehensive treatment of the theory, architectures, and applications of CORDIC processors, the reader is referred to [7] and the references therein.

In any CORDIC implementation, we need to determine the number of CORDIC stages and the bit precision required to achieve acceptable performance for the application of interest. In our simulations, we used $SNR_{MMSE-DFE,U}$, as defined in Equation (5), as a performance measure and we considered a "low dynamic range" scenario with $\nu = 1$ and input SNR = 10 dB and a "high dynamic range" scenario with $\nu = 3$ and input SNR = 30 dB. Figures 9 and 10 show that using 9 CORDIC stages results in the same performance as the Givens–rotations–based algorithm of Section 2.3. Assuming 9 stages, it can be seen from Figures 11 and 12 that 16 bits of precision result in less than 0.2 dB performance loss from the floating–point representation for GSM scenarios.

4 Extensions and Future Work

In this section, we show how the results of Section 3, which assumed single antenna reception and white noise, can be extended to include the effects of colored noise, co–channel interference, and diversity reception.

- **Colored Noise**

  A common model for colored noise is white noise passed through an all–pole prediction filter. In this case, $\mathbf{R}_{nn}^{\frac{-*}{2}}$ is an upper–triangular Toeplitz matrix whose coefficients are equal to the pole polynomial and $\mathbf{R}_{nn}^{\frac{-*}{2}}\mathbf{H}$ is also an upper–triangular matrix whose first row is equal to the length–$(N+\nu)$ impulse response of the convolution of the channel impulse response and the noise whitening filter. The matrix $\tilde{\mathbf{H}}$ is defined per Equation (7).

- **Co–Channel Interference**

  To increase spectral efficiency in TDMA digital cellular systems, frequency re-use is employed where multiple users (located in spatially–separated cells) use the same frequency at the same time. This results in co–channel interference (CCI) that limits performance.

  We showed in [1] that in the presence of CCI, the optimum DFE coefficients (for the case of uncorrelated input and noise) are computed from the Cholesky factorization :

$$\mathbf{G} = \mathbf{I}_{N+\nu} + \mathbf{H}_0^*(\frac{1}{SNR}\mathbf{I}_N + \sum_{i=1}^{m} SIR_i\mathbf{H}_i\mathbf{H}_i^*)^{-1}\mathbf{H}_0 \stackrel{def}{=} \mathbf{I}_{N+\nu_0} + \mathbf{H}_0^*\mathbf{G}_{CCI}^{-1}\mathbf{H}_0 \ ,$$

where $\mathbf{H}_i$ is the channel matrix of the i^{th} user ($1 \leq i \leq M$) and SIR_i is the signal to i^{th} interferer ratio. Then, a parallel implementation can be developed by performing a **QR** factorization of the matrix $\check{\mathbf{H}} = \begin{bmatrix} \mathbf{I}_{N+\nu} \\ \mathbf{R}_{CCI}\mathbf{H}_0 \end{bmatrix}$, where $\mathbf{R}_{CCI}$ is the upper–triangular Cholesky factor of the Toeplitz matrix $\mathbf{G}_{CCI}$.

- **Diversity Combining**

  Combining the outputs of several diversity paths prior to equalization is an effective scheme for combatting frequency–selective fading [11, 10]. A block diagram of post–combining diversity is given in Figure 13. The input–output model of Section 2 still applies by expanding each channel coefficient $\mathbf{h}_i$ from being an $l \times 1$ vector to an $lM \times 1$ vector where M is the number of diversity branches. In other words, the first block row of the $lMN \times (N+\nu)$ block Toeplitz channel matrix $\mathbf{H}$ consists of lM channel impulse responses that represent the l sampling phases of the M diversity paths.

- **Blind Channel Identification**

  An interesting direction for future research is to develop an architecture for combining the DFE coefficient computation processor with a *blind* channel estimator such as the one proposed in [13] to avoid the training overhead in non–blind schemes. The main computational task in these blind schemes is an eigen–decomposition of a correlation matrix which can also be performed in parallel using CORDIC processors [7].

5 Conclusions

We have shown how the coefficient calculation of a channel–estimate–based fractionally–spaced finite–length MMSE–DFE can be implemented in parallel for dispersive and noisy linear channels. The parallel implementation is made possible by relating the optimum DFE coefficients to the **QR** factorization of a data matrix that depends on the CIR coefficients and input SNR.

The main computational task is triangularization of a data matrix, which can be performed online using CORDIC processors pipelined within a systolic array architecture. For scenarios typical of the GSM standard, 9 CORDIC stages and 16 bits of precision were found to results in very small

performance loss from a floating–point processor executing the Givens–based **QR** algorithm. The parallel implementation speeds up the DFE coefficient computation procedure (by a factor equal to the sum of the channel memory and the number of feedforward filter taps), is modular and suitable for VLSI implementation, and scalable with the number of DFE taps.

References

[1] N. Al-Dhahir. "A Computationally-Efficient FIR MMSE-DFE for CCI-Impaired Dispersive Channels". *IEEE Transactions on Signal Processing*, pages 32–41, January 1997.

[2] N. Al-Dhahir and J.M.Cioffi. "Efficient Computation of the Delay-Optimized Finite-Length MMSE-DFE". *IEEE Transactions on Signal Processing*, 44(5):1288–1292, May 1996.

[3] N. Al-Dhahir and J.M.Cioffi. "Efficiently-Computed Reduced-Parameter Input-Aided MMSE Equalizers for ML Detection : A Unified Approach". *IEEE Transactions on Information Theory*, pages 903–915, May 1996.

[4] S. Ariyavisitakul and L.J. Greenstein. "Reduced-Complexity Equalization Techniques for Broadband Wireless Channels". *"IEEE Journal on Selected Areas in Communications"*, pages 5–15, January 1997.

[5] G. Golub and C. Van Loan. *"Matrix Computations"*. The Johns Hopkins University Press, 1989. Second Edition.

[6] S. Haykin. *"Adaptive Filter Theory"*. Prentice Hall, 1991. 2nd Edition.

[7] Yu. Hu. "CORDIC-Based VLSI Architectures for Digital Signal Processing". *IEEE Signal Processing Magazine*, pages 16–35, July 1992.

[8] J.M.Cioffi and G. Dudevoir and M.V. Eyuboglu and G.D. Forney, Jr. "Minimum Mean-Square-Error Decision Feedback Equalization and Coding - Parts I and II". *IEEE Transactions on Communications*, October 1995.

[9] T. Kailath and A.H.Sayed. "Displacement Structure: Theory and Applications". *SIAM Review*, 37(3):297–386, September 1995.

[10] S. Lin and V.K. Prabhu. "Optimum Diversity Combining with Finite-Tap Decision Feedback Equalization in Digital Cellular Mobile Radio". In *International Conference on Communications*, June 1997.

[11] N. Lo, D.D.Falconer, and A. Sheikh. "Adaptive Equalization and Diversity Combining for Mobile Radio using Interpolated Channel Estimates". *IEEE Transactions on Vehicular Technology*, pages 636–645, August 1991.

[12] N. Lo, D.D.Falconer, and A. Sheikh. "Adaptive Equalization for Co-Channel Interference in a Multipath Fading Environment". *IEEE Transactions on Communications*, pages 1441–1453, Feb./March/April 1995.

[13] L.Tong, G. Xu, and T.Kailath. "Blind Identification and Equalization Based on Second-Order Statistics : A Time Domain Approach". *IEEE Transactions on Information Theory*, 40(2):340–349, March 1994.

[14] S. Qureshi. "Adaptive Equalization". *Proceedings of the IEEE*, 73(9), September 1985.

[15] P. Shukla and L. Turner. "Channel-Estimation-Based Adaptive DFE for Fading Multipath Radio Channels". *IEE Proceedings - I*, 138(6), December 1991.

[16] W.C.Jakes. *"Microwave Mobile Communications"*. IEEE Press, 1974.

[17] R. Ziegler, N. Al-Dhahir, and J.M.Cioffi. "Nonrecursive Adaptive Decision-Feedback Equalization from Channel Estimates". In *Proceedings of IEEE Vehicular Technology Society* 42^{nd} *VTS Conference*, volume 2, pages 600–603, Denver, May 1992.

Bibliography

- **Naofal Al-Dhahir**

  Received his B.Sc. degree from Kuwait University with first–class honors in 1989 and M.Sc. and Ph.D. degrees from Stanford University in 1990 and 1994, respectively, all in Electrical Engineering. He was an Instructor at Stanford University during Winter 1993.

  Since August 1994, he has been with GE Corporate R & D Center in Schenectady, NY. His research interests are in combined equalization and coding transceivers, fast signal processing algorithms in communications, and satellite communication systems. He holds two US patents on power control schemes for satellite communications systems and is a Senior Member of the IEEE.

- **Ali H. Sayed**

  Received the degree of Engineer in Electrical Engineering from the University of Sao Paulo, Brazil in 1987, and was the first-place graduate of the School of Engineering. In 1989 he received, with distinction, the M.S. degree in Electrical Engineering from the University of Sao Paulo, Brazil. In 1992 he received the Ph.D. degree in Electrical Engineering from Stanford University. From September 1992 to August 1993 he was a Research Associate at Stanford University, after which he joined, as an Assistant Professor, the Department of Electrical and Computer Engineering at the University of California, Santa Barbara. Since July 1996 he has been an Associate Professor in the Department of Electrical Engineering at the University of California, Los Angeles. He is the co-director of the Adaptive and Nonlinear Systems Laboratory.

  Dr. Sayed has over 110 journal and conference publications. He is also co-editor of a volume in press by SIAM. Dr. Sayed is a recipient of a 1994 NSF Research Initiation Award, and of the 1996 IEEE Donald G. Fink Prize Award. He is an Associate Editor for the IEEE Transactions on Signal Processing and a member of the Editorial Board of the SIAM Journal on Matrix Analysis and Applications. He was also a co-editor of a special issue of the journal Linear Algebra and Its Applications on the topic of Fast Algorithms for Control, Signals, and Image Processing. He is a member of IEEE, SIAM, and ILAS.

List of Figures

1 Block Diagram of Proposed DFE Coefficient Computation Processor 16

2 Effect of Fixed-Point Implementation of Givens–Based **QR** Factorization on $SNR_{MMSE-DFE,U}$ with $N = 8$, $\nu = 1$, and SNR=10 dB 17

3 Effect of Fixed-Point Implementation of Givens–Based **QR** Factorization on $SNR_{MMSE-DFE,U}$ with $N = 16$, $\nu = 3$, and SNR= 10 dB 17

4 Effect of Fixed-Point Implementation of Givens–Based **QR** Factorization on $SNR_{MMSE-DFE,U}$ with $N = 16$, $\nu = 3$, and SNR=30 dB 18

5 MIPS Estimate for Computing DFE Coefficients with 5 msec Update Rate . . 18

6 Time Required to Compute DFE Coefficients with 25-MIP Processor 19

7 Systolic Array Implementation for **QR** Factorization of $\tilde{\mathbf{H}}$ 20

8 Systolic Array Cell Definitions for **QR** Factorization of $\tilde{\mathbf{H}}$ 21

9 Variation of $SNR_{MMSE-DFE,U}$ versus the Number of CORDIC Stages with $N = 8$, $\nu = 1$, and SNR= 10 dB 22

10 Variation of $SNR_{MMSE-DFE,U}$ versus the Number of CORDIC Stages with $N = 16$, $\nu = 3$, and SNR= 30 dB 22

11 Effect of Fixed-Point Implementation of CORDIC-Based **QR** Factorization on $SNR_{MMSE-DFE,U}$ with $N = 8$, $\nu = 1$, and SNR= 10 dB 23

12 Effect of Fixed-Point Implementation of CORDIC-Based **QR** Factorization on $SNR_{MMSE-DFE,U}$ with $N = 16$, $\nu = 3$, and SNR= 30 dB 23

13 Block Diagram of Post-Combining Diversity and Decision Feedback Equalization 24

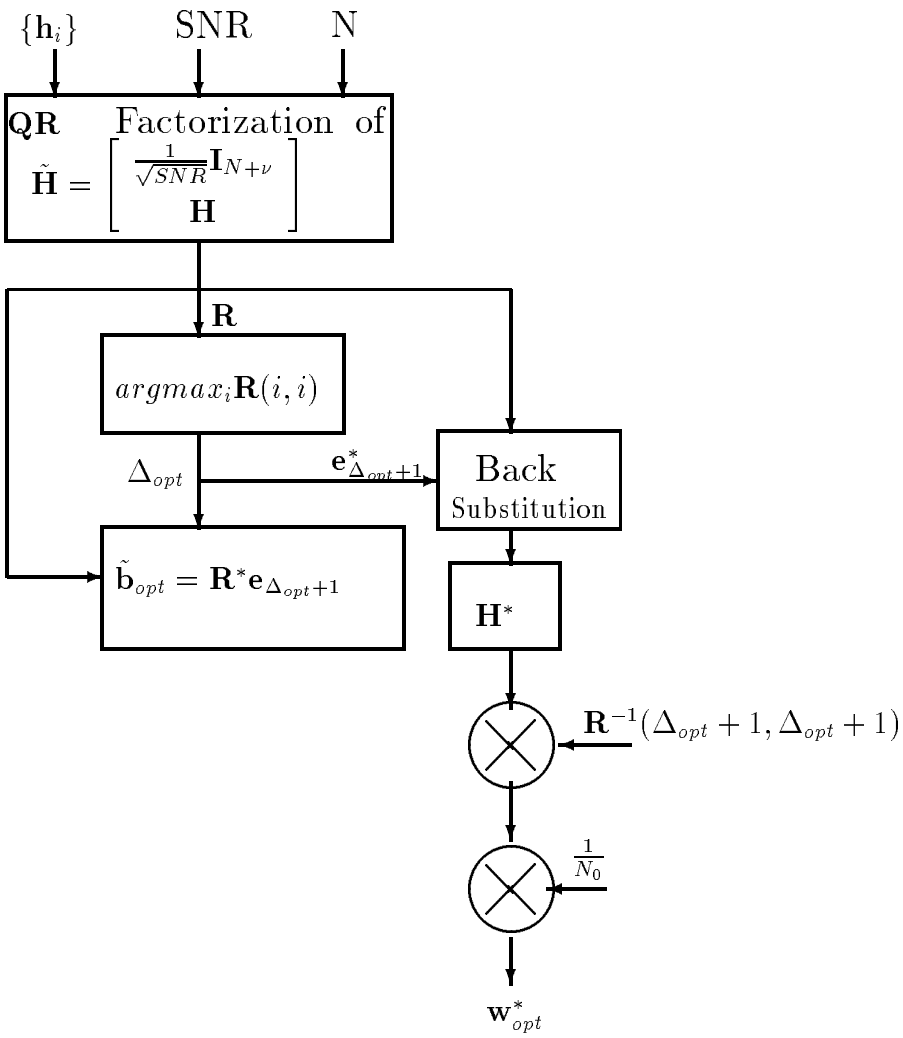


Figure 1: Block Diagram of Proposed DFE Coefficient Computation Processor

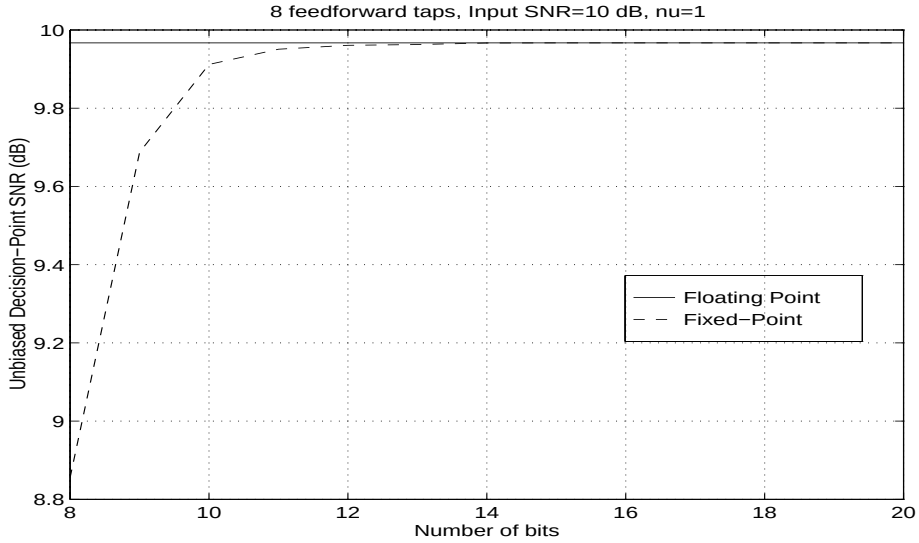


Figure 2: Effect of Fixed-Point Implementation of Givens–Based **QR** Factorization on $SNR_{MMSE-DFE,U}$ with $N = 8$, $\nu = 1$, and SNR=10 dB

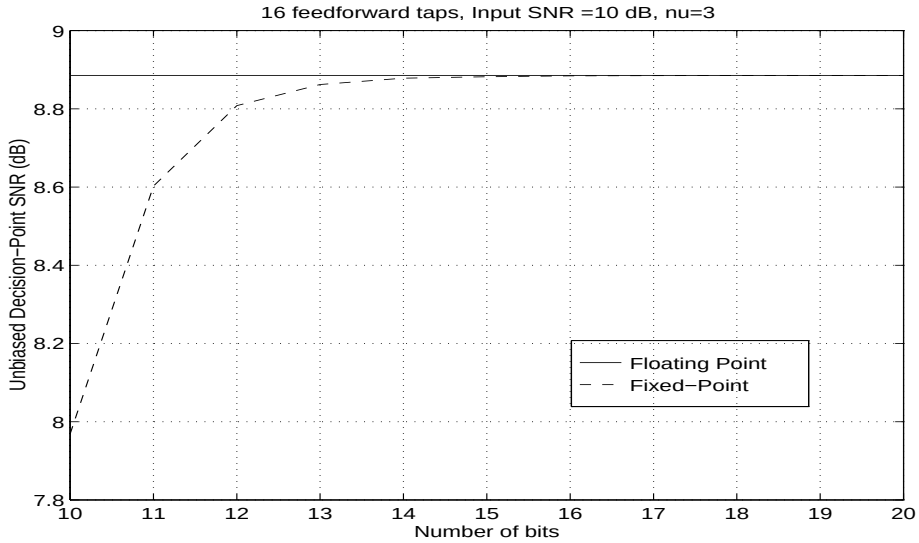


Figure 3: Effect of Fixed-Point Implementation of Givens–Based **QR** Factorization on $SNR_{MMSE-DFE,U}$ with $N = 16$, $\nu = 3$, and SNR= 10 dB

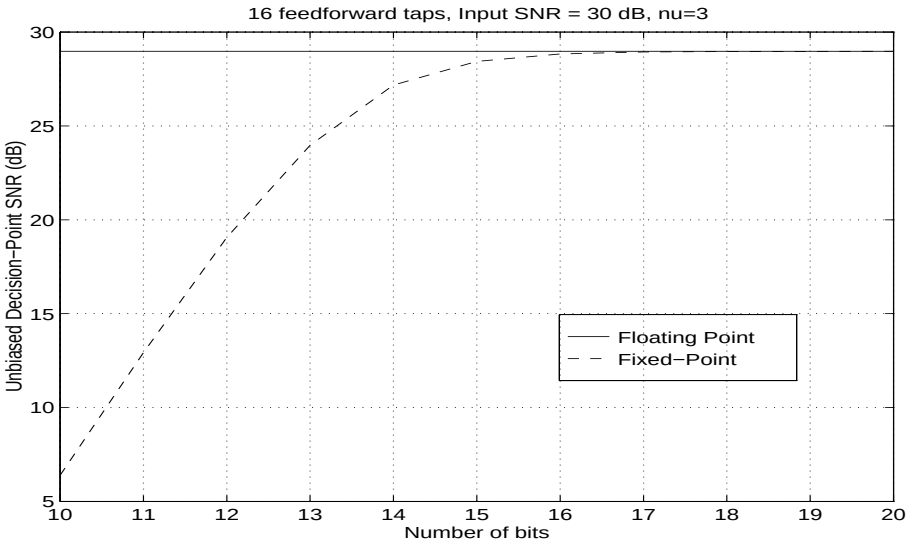


Figure 4: Effect of Fixed-Point Implementation of Givens–Based **QR** Factorization on $SNR_{MMSE-DFE,U}$ with $N = 16$, $\nu = 3$, and SNR=30 dB

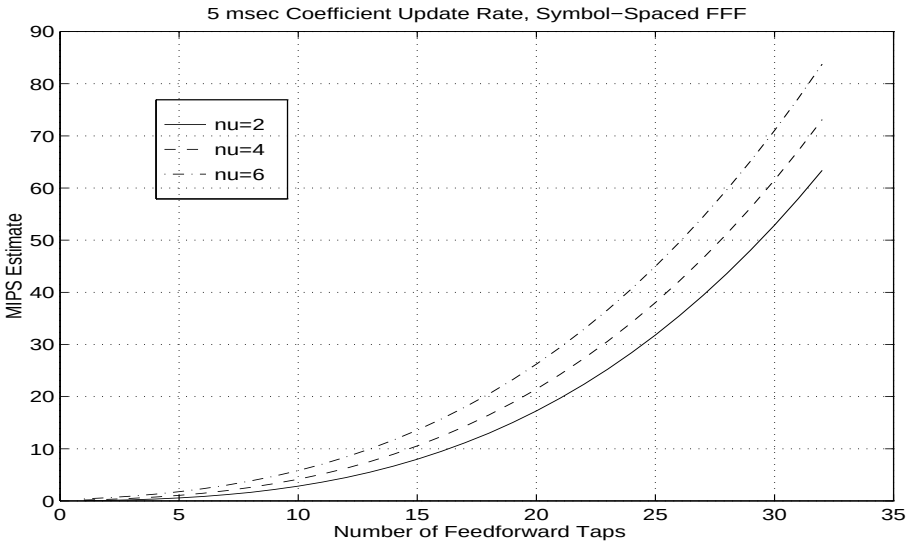


Figure 5: MIPS Estimate for Computing DFE Coefficients with 5 msec Update Rate

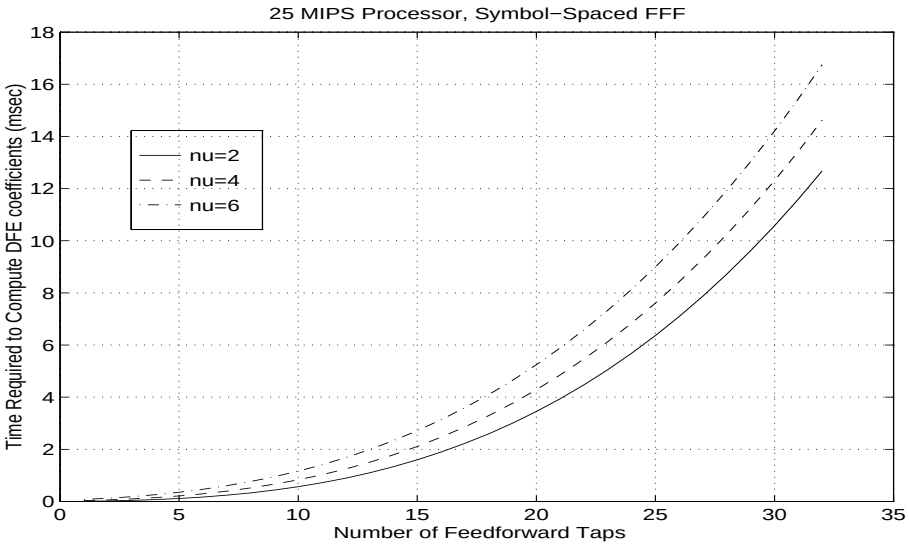


Figure 6: Time Required to Compute DFE Coefficients with 25-MIP Processor

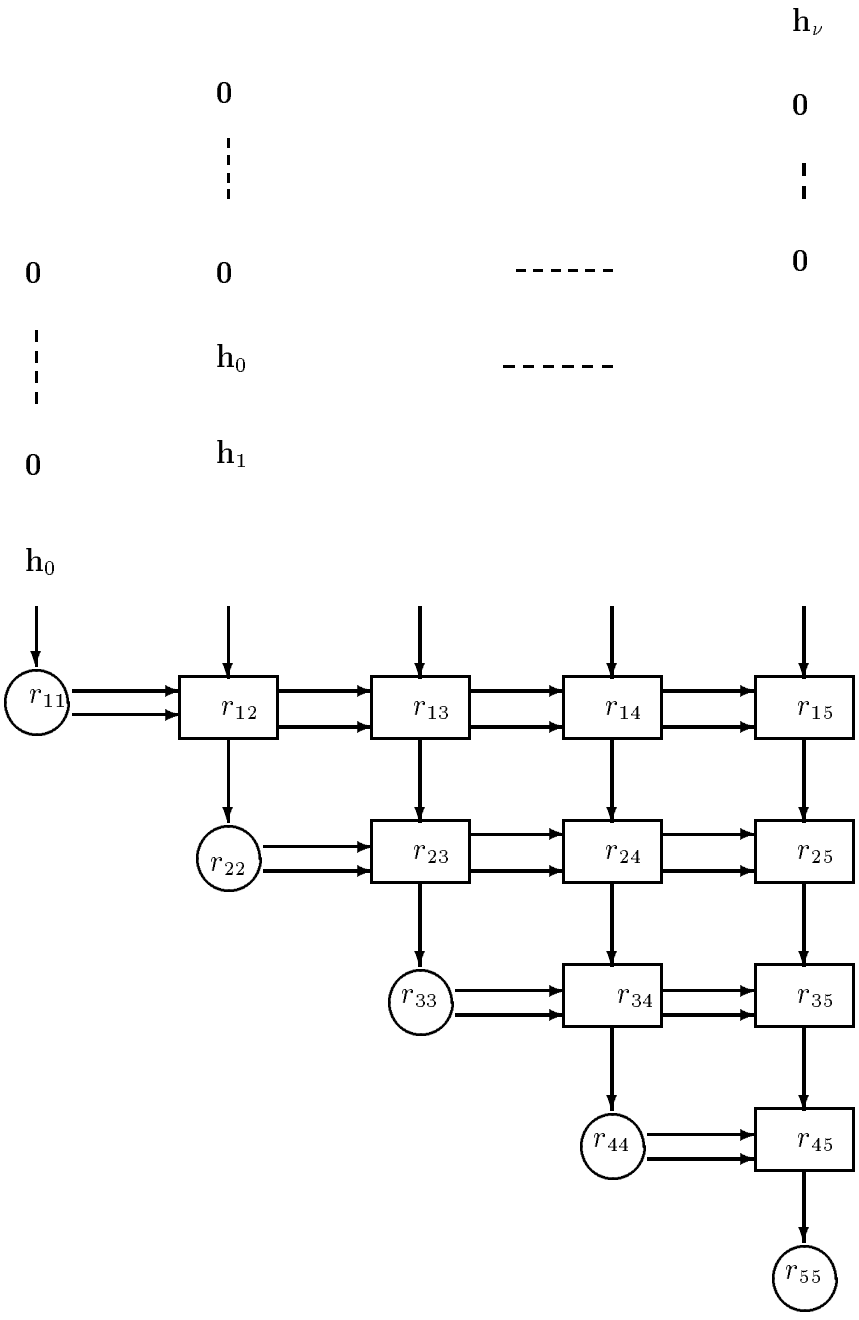
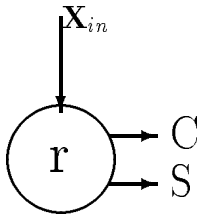


Figure 7: Systolic Array Implementation for $\mathbf{QR}$ Factorization of $\tilde{\mathbf{H}}$

Boundary Cell



Initialization

$r = \frac{1}{\sqrt{SNR}}$, C=1 , S=0

Functionality

IF $\mathbf{X}_{in} = 0$

r=r , C=1 , S=0

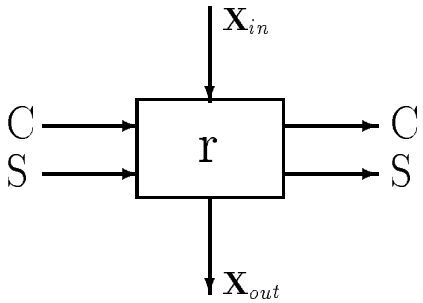
ELSE

$r_1 = \sqrt{r^2 + |\mathbf{X}_{in}|^2}$

$C = \frac{r}{r_1}$ $\qquad$ $S = \frac{\mathbf{X}_{in}}{r_1}$

$r = r_1$

Internal Cell



Initialization

r=0, C=1, S=0

Functionality

$\mathbf{X}_{out} = C.\mathbf{X}_{in} - S.r$

$r = S^*.\mathbf{X}_{in} + C.r$

Figure 8: Systolic Array Cell Definitions for **QR** Factorization of $\tilde{\mathbf{H}}$

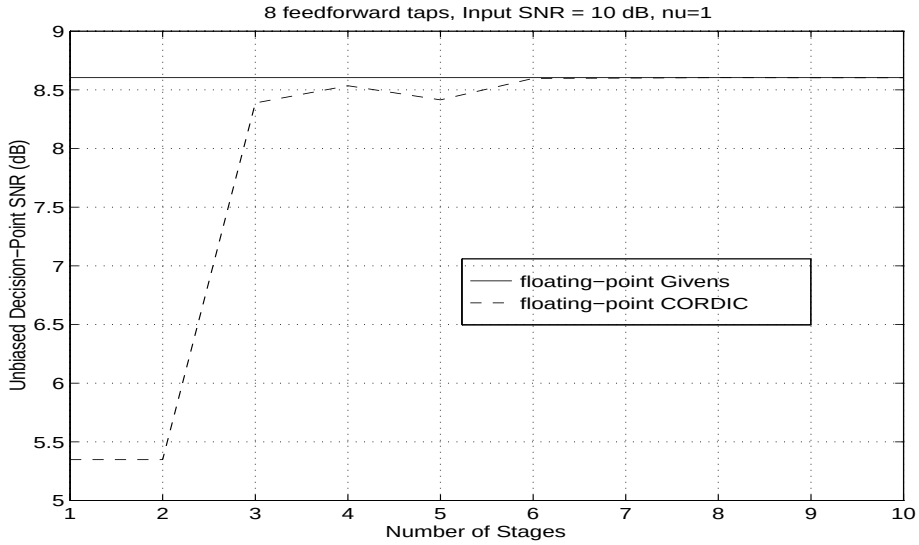


Figure 9: Variation of $SNR_{MMSE-DFE,U}$ versus the Number of CORDIC Stages with $N = 8$, $\nu = 1$, and SNR= 10 dB

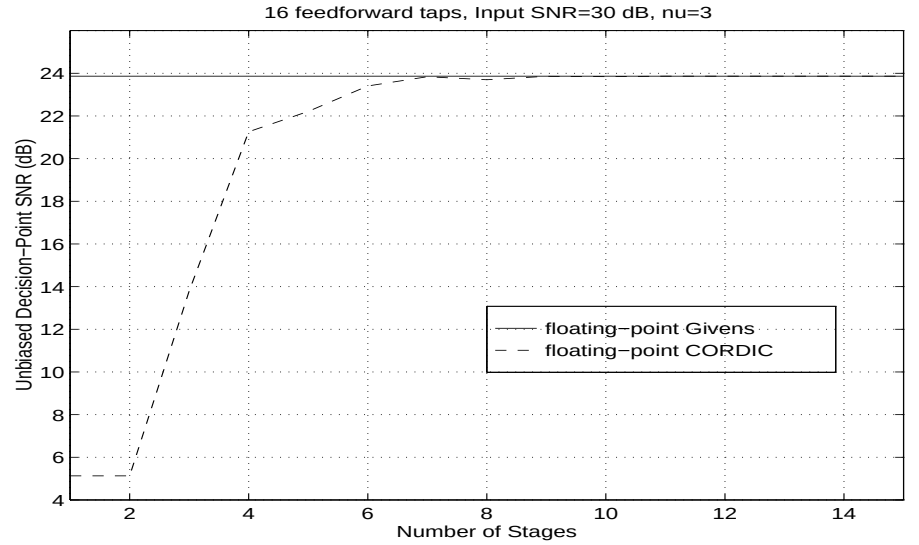


Figure 10: Variation of $SNR_{MMSE-DFE,U}$ versus the Number of CORDIC Stages with $N = 16$, $\nu = 3$, and SNR= 30 dB

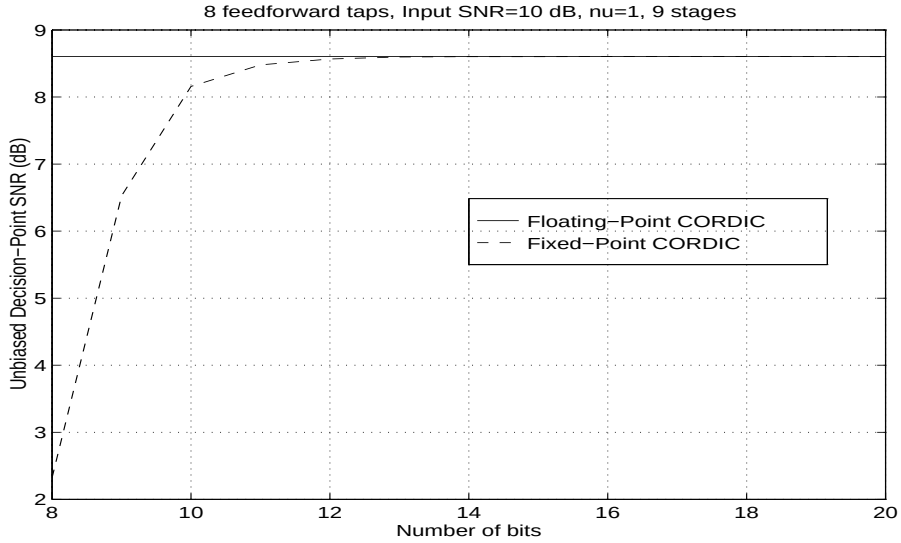


Figure 11: Effect of Fixed-Point Implementation of CORDIC-Based **QR** Factorization on $SNR_{MMSE-DFE,U}$ with $N = 8$, $\nu = 1$, and SNR= 10 dB

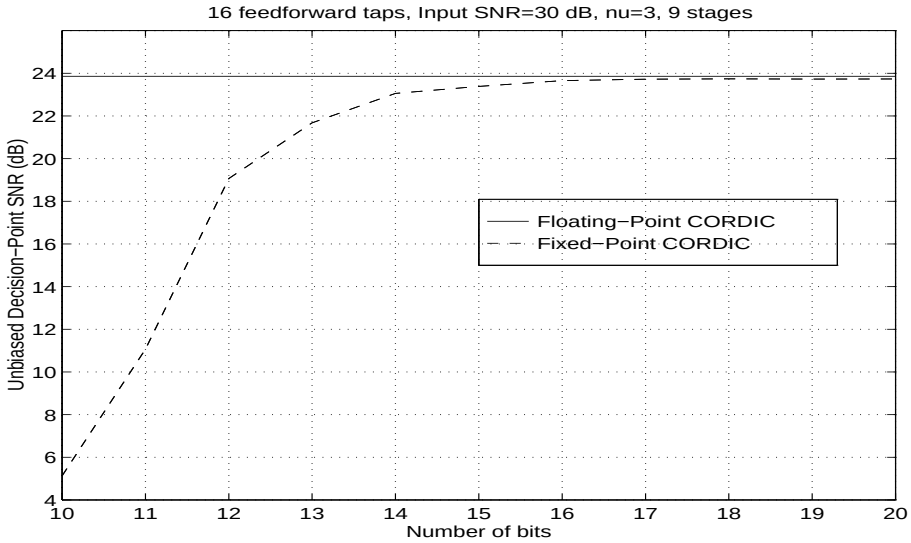


Figure 12: Effect of Fixed-Point Implementation of CORDIC-Based **QR** Factorization on $SNR_{MMSE-DFE,U}$ with $N = 16$, $\nu = 3$, and SNR= 30 dB

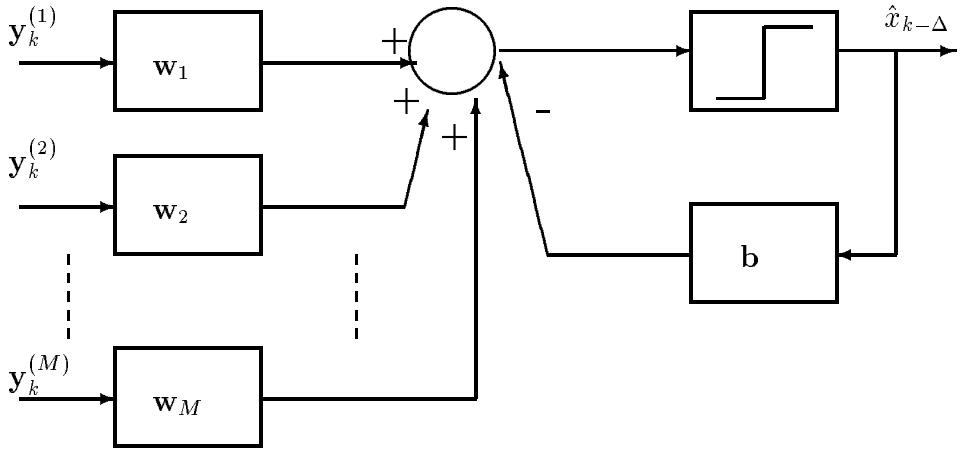


Figure 13: Block Diagram of Post-Combining Diversity and Decision Feedback Equalization