BER-BASED ADAPTIVE ADC-EQUALIZER BASED RECEIVER FOR COMMUNICATION LINKS

Rajan Narasimha, Naresh Shanbhag and Andrew Singer

Coordinated Science Laboratory/ECE Department
University of Illinois at Urbana-Champaign
1308 W Main St., Urbana, IL, USA, 61801.

ABSTRACT

This paper presents the architecture of a non-uniform reference level bit error-rate (BER)-optimal analog-to-digital converter (ADC) and equalizer, for high-speed communication links. Finite precision analysis demonstrates that the use of the BER-optimal ADC does not increase the equalizer complexity/power significantly. An adaptive algorithm referred to as the approximate minimum BER algorithm (AMBER) is proposed in order to determine the BER-optimal reference levels. Finite-precision analysis of AMBER indicates that reference levels represented with 9-bit precision is sufficient for a 3-bit BER-optimal ADC to achieve BER equal to that of a 4-bit conventional ADC. An architectural implementation of AMBER is also presented. The reference-level adaptation unit (**RL-UD**) has a full-adder (**FA**) complexity that is 76% over the conventional adaptive equalizer. The **RL-UD** block is clock-gated after convergence and hence does not present a power overhead. Thus, for high-speed links employing the flash ADC architecture, the proposed AMBER receiver represents a power savings of approximately 50% in the ADC.

1. INTRODUCTION

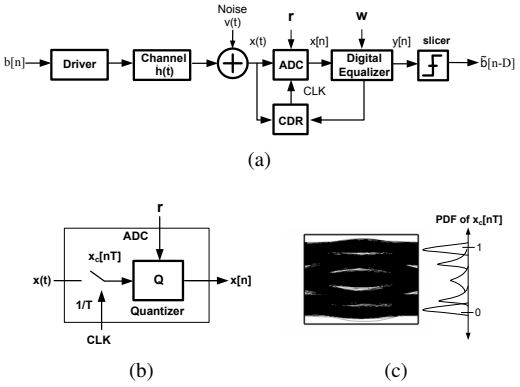

Fig. 1. Role of an ADC in a communication link: a) block diagram of a communication link, b) functional diagram of an ADC, and c) eye diagram and PDF of the sampled received signal $x_c[nT]$.

Figure 1(a) depicts a typical communication link, where the analog-to-digital converter (ADC) acts as a front end for subsequent digital processing. Traditionally, ADCs are designed to uniformly quantize the input signal, and optimized based on a signal fidelity criterion, such as signal-to-quantization-noise-ratio (SQNR) and signal-to-noise-plus-distortion-ratio (SNDR). Such an approach, where the ADC and subsequent digital processing are not jointly optimized, leads to over-designs. This issue is of great significance in high-speed links, where low-power ADCs are particularly difficult to design, and the effective number of bits (ENOB) usually does not exceed 6 [1–3]. Bit error-rate (BER) is a better metric to optimize for in communication links. Indeed, past work includes BER-optimal *equalizers* [4, 5] and BER-based *sampling phase adjustment* [5]. Note: the well-known trend of employing digitally-assisted ADCs [6, 7] do not minimize the BER.

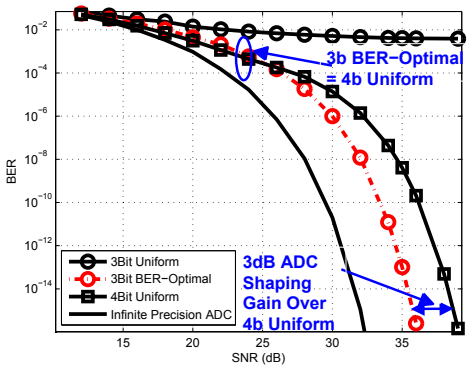

Fig. 2. Shaping gain offered by a BER-optimal ADC.

Recently, we proposed the co-design of the ADC and equalizer by optimizing the reference levels based on the BER metric [8]. The *ADC shaping gain* was defined as $S_G(BER) = SNR_{old}(BER) - SNR_{new}(BER)$ to quantify the reduction in SNR achieved via BER-optimal techniques. It was shown (see Fig. 2) that a 3-bit BER-optimal ADC offers significant shaping gains, i.e., $S_G(BER = 10^{-15}) >$ 30dB compared to a 3-bit uniform ADC. In addition, $S_G(BER = 10^{-15}) =$ 3dB when compared to a 4-bit uniform ADC. The ADC reference levels were computed using a gradient-descent on the BER cost function. This approach is computationally infeasible, as (a) it assumes knowledge of channel and equalizer coefficients, and (b) it is too complex to implement. An additional issue is the potentially increase in the equalizer complexity. This can occur if the BER-optimal ADC output is to be re-mapped/encoding in order operate using linear arithmetic operators.

The contributions of this paper are as follows: (a) we demon-

strate there is no increase in equalizer complexity required to achieve the ADC shaping gain estimated in [8], (b) we develop an Approximate Minimum BER algorithm (AMBER) to adapt the reference levels and thresholds, and (c) we propose a VLSI architecture to implement the algorithm, and demonstrate its feasibility through finite precision simulations.

Section. 2 reviews the work presented in [8]. Section 3 provides a brief description of the analysis and simulation methodology used in this paper. In Section. 4 we determine the precision required to represent the ADC reference levels, when a linear equalizer (LE) is employed at the back-end. Further, we show that the shaping gain can be utilized to reduce the coefficient precision necessary at the equalizer. An adaptive algorithm for the reference levels is developed in Section. 5 and an architecture to implement this algorithm is proposed and evaluated in Section 6.

2. PRELIMINARIES

In this section, we review the basic principles of the conventional/uniform and BER-optimal ADCs. Figure 1(a) depicts a communication link and the functionality of the ADC front-end. The signal distribution at the channel output for an ISI channel is multi-modal, as shown in Fig. 1(c). Assuming binary phase shift keying (BPSK) modulation, the transmitter sends pseudorandom sequence of bits $b[n] \in \{\pm 1\}$ through the channel. At the receiver, the ADC quantizes the signal, and the outputs are subsequently processed by a digital equalizer to eliminate ISI that results from the channel. A slicer following the equalizer makes a hard decision. With a slight abuse of notation, we refer to the BPSK symbols as bits in the sequel. As shown in Fig. 1(b), the ADC consists of a baud-rate sampler followed by a quantizer. At a given sampling time index n, the input $x_c[n] = x_c[nT]$ to the ADC is given by

$$x_c[n] = \sum_{i=0}^{M-1} h[i]b[n-i] + v[n] \tag{1}$$

where $b[n]$ is the transmitted bit, $h[i]$ the baud-rate sampled impulse response of the channel with memory M, and $v[n]$ is modeled as additive white Gaussian noise with variance σ^2.

The ADC has N reference levels $\mathbf{r} = \{r_k\}$ $(k = 1, \ldots, N)$ and $N-1$ thresholds $\mathbf{t} = \{t_k\}$ $(k = 1, \ldots, N-1)$, where N is equal to 2^{B_x}. The mapping between $x_c[n]$ and the quantized signal $x[n]$ is

$$\begin{aligned} x[n] &= r_1 \text{ if } x_c[n] \epsilon (-\infty, t_1] \\ &= r_N \text{ if } x_c[n] \epsilon (t_{N-1}, \infty) \\ &= r_k \text{ if } x_c[n] \epsilon (t_{k-1}, t_k] \text{ for } k = 2, \ldots, N-2. \end{aligned} \tag{2}$$

The output of the L-tap LE will be the convolution of ADC outputs and equalizer coefficients. The estimate of the transmitted symbol $b[n-D]$ is $\tilde{b}[n-D] = \text{sgn}(y[n])$. Here D is introduced to account for delay in the channel and equalizer.

2.1. Uniform ADC

In a uniform ADC, the quantization levels are spread evenly within the signal dynamic range. The minimum and maximum input amplitudes expected by this ADC are expressed as $-V_{max}$ and V_{max}, respectively. The quantizer step-size is $\Delta = \frac{2V_{max}}{N} = \frac{2V_{max}}{2^{B_x}}$. For sufficiently small quantization error, $q[n] = x_c[n] - x[n]$ is assumed to be a uniformly distributed random variable, bounded between $-\frac{\Delta}{2}$ and $+\frac{\Delta}{2}$ and independent of input. Quantization noise power σ_q^2 is given by $E[q^2[n]] = \frac{\Delta^2}{12}$. For uniform quantization, SQNR can be calculated from $6.02B_X + 4.8 - 20\log_{10}\frac{V_{max}}{\sigma_x}$, where each additional bit increases SQNR by 6dB.

2.2. BER-Optimal ADC

We propose quantization based on the detection criterion, by setting the levels $\mathbf{r}$ and thresholds $\mathbf{t}$ non-uniformly using the BER metric. In the system presented in Fig. 1(a), an error is made when $\tilde{b}[n] \neq b[n]$ (assuming $D = 0$), so BER is computed by averaging over all possible values of $y[n]$ and hence all vectors $\mathbf{x}[n] = [x[n]x[n-1]...x[n-L+1]]$ such that $\tilde{b}[n] = \text{sgn}(y[n]) = \text{sgn}(\mathbf{w}^T\mathbf{x}[n])$ produces an error at the slicer,

$$\begin{aligned} BER &= \mathbb{P}\{b[n] \neq \tilde{b}[n]\} \\ &= \sum_{y[n]} \left[\mathbb{P}\{y[n]\} \left(\frac{1 - b[n]\tilde{b}[n]}{2} \right) \right] \\ &= \sum_{\mathbf{x}_n} \left[\left(\prod_{j=0}^{L-1} \mathbb{P}\{x[n-j] = r_k\} \right) \left(\frac{1 - b[n]\tilde{b}[n]}{2} \right) \right] \end{aligned} \tag{3}$$

where $\mathbb{P}\{x[n-j] = r_k\}$ is given by

$$Q\left(\frac{t_{k-1} - x_c[n-j]}{\sigma} \right) - Q\left(\frac{t_k - x_c[n-j]}{\sigma} \right), \tag{4}$$

$\mathbb{P}\{\bullet\}$ signifies the probability of an event, and $Q(\bullet)$ is the Gaussian Q function.The equalizer output $\tilde{b}[n]$ is given by

$$\tilde{b}[n] = \text{sgn}\left(\sum_{j=0}^{L-1} w[j]x[n-j] \right). \tag{5}$$

A BER-optimal ADC is one where $\mathbf{r}$ and $\mathbf{t}$ are chosen to minimize (3).

3. ANALYSIS AND SIMULATION METHODOLOGY

In this section, we outline the analysis and simulation methodology used in the paper to estimate the performance of the techniques proposed. First, for the channel impulse response shown in Fig. 3, a minimum mean squared error (MMSE) LE ($\mathbf{w}_{opt}$) with three taps is obtained assuming a uniform ADC. Next, a numerical gradient descent approach is used to iteratively approximate the minimum BER thresholds and representation levels for the ADC. Equation (3) was then used to compute the BER analytically. Signal-to-noise ratio (SNR) was computed by SNR $= \sum_{i=0}^{M-1} \frac{h[i]^2}{\sigma^2}$. The results obtained through the analysis described above give bounds on the achievable shaping gain, as they assume a floating point representation of the reference levels and coefficients.

We verified our expressions via Monte-Carlo simulations and error counting for BER down to 10^{-5}. For lower BER we used Importance Sampling (IS) to obtain BER estimates. IS is a well known statistical tool to estimate the probabilities of rare events. For such events, it is not feasible to run a Monte-Carlo simulation to estimate the desired probability. IS relies on a combination of simulation and analysis to estimate the probability of interest. The probability distribution that governs the occurrence of the rare event is skewed, in

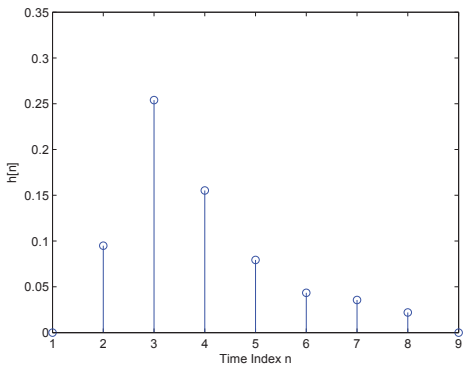

Fig. 3. Sampled impulse response of a high-ISI channel.

order to make it occur more often. The IS-estimator applies a correction factor to the conventional Monte-Carlo estimator in order to derive the statistics of the event based on the original distribution from the skewed distribution. In this paper, where it is desired to estimate $BER < 10^{-5}$, we alter the noise distribution by increasing its variance. This leads to more error events. A correction factor equal to the ratio of the original and altered distribution is applied to the Monte-Carlo estimator.

4. FIXED REFERENCE LEVEL ADC, FIXED COEFFICIENT EQUALIZER: PRECISION REQUIREMENTS

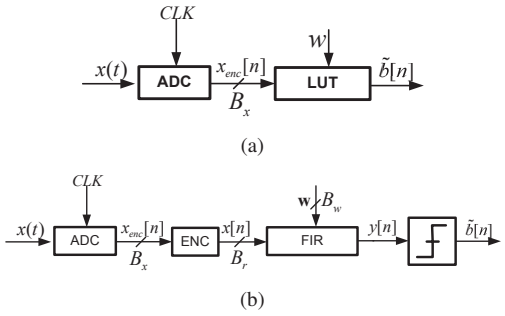

Fig. 4. Two equalization techniques: a) LUT-based non-linear, and b) linear equalizer (LE).

The representation of the ADC output values is tied to the implementation of the equalizer. Since the quantization levels are no longer equidistant, a change in digital output will correspond to different changes in analog input. The equalizer cannot operate directly on such digital outputs. To address this issue, two equalizer architectures are possible (Fig. 4): (1) look-up table (LUT)-based [8], and (2) a tapped-delay line LE. The LUT-based equalizer operates directly on the L ADC samples $x_{enc}[k]$ each with B_x bits in order to make the decision $\tilde{b}[k]$. The tapped-delay line LE first maps $x_{enc}[k]$ with B_x bits to $x[k]$ with B_r bits ($B_r > B_x$), so that the mapping $x(t) \mapsto x[k]$ is linear. Thus, a BER-optimal ADC *may* need an equalizer higher complexity than that needed by the conventional ADC. In this section, we show that this is not the case.

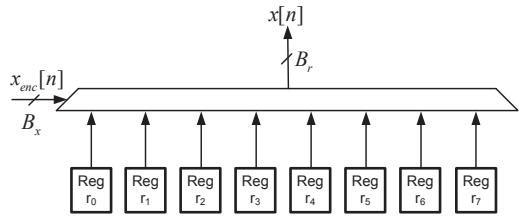

Fig. 5. ADC output encoder architecture.

Figure 5 shows a realization of the encoder. In this technique, the ADC output code (B_x bits) is used to select from one of $N = 2^{B_x}$ reference levels. The reference levels are stored in B_r bit registers. The BER estimates in [8] assumed floating point representation of the reference levels. To compare the complexity of the conventional equalizer ($\mathbf{EQ}_{conv}$(4b)) with that of the encoder and equalizer following the BER-optimal ADC ($\mathbf{EQ}_{opt}$(3b)), we look at the precision requirements for the ADC reference levels (B_r) and coefficients of the LE (B_w).

The high-ISI channel shown in Fig. 3 is used to evaluate the reference-level precision B_r necessary to achieve close-to-optimal performance. Note: B_r also determines the complexity of the equalizer and hence, is a critical parameter. To determine B_r for the 3-bit BER-optimal reference levels, we first consider the floating point equalizer case and consider two representative SNR points: low SNR (24dB) and high SNR (32dB). An estimate of B_r can be obtained through the following inequalities:

$$B_r \geq B_x + 1 \tag{6}$$

$$B_r \geq log_2 \left(\frac{\max_i(|r_i|)}{\min_j(r_{j+1} - r_j)} \right) \tag{7}$$

Constraint (6) is necessary to ensure that the ADC captures more information than a B_x bit uniform ADC. Constraint (7) is necessary to guarantee that the minimum difference in reference levels can be represented. At low and high SNR, $B_r = 4$ based on (6) and (7). The threshold precision is determined as $B_t = B_r + 1$, as the thresholds are placed equidistant from the reference levels. In the following discussion, BER_x (SNR_x) refers to the BER (SNR) when parameter x has a finite precision representation and all other parameters are floating point.

The finite precision reference levels, $\mathbf{r}_{fx}$, and the corresponding thresholds $\mathbf{t}_{fx}$ were used to analytically estimate $BER_{r,t}$ from 3 and via Monte-Carlo simulations. From this analysis, the BER with fixed-point reference levels (BER_r) was found to be approximately equal to the BER of the conventional receiver when $B_r = 4$, i.e., $BER_r \approx BER$ for the high and low SNR cases. This implies that the coefficient precision will determine the complexity of the two equalizers This implies that the data precision is the same for the linear equalizers employed with the 4-bit uniform ADC and 3-bit BER-optimal ADC, i.e, the coefficient precisions for the two equalizers would determine which equalizer is more complex. We now determine the coefficient precision B_w for the conventional equalizer. From the floating-point BER estimates in Fig. 2, the SNR requirement at the equalizer output is determined as $SNR = Q^{-1}(BER)$ for the low and high SNR cases. We determine B_w such that the fixed-point SNR is within 0.1 dB of the floating-point SNR, resulting in $B_w = 5$ and $B_w = 6$ bits, respectively, for the low and high

SNR scenarios. Thus, for B_w starting from 6-bits and 5-bits respectively for the high and low SNR cases, and decreasing, we quantize w, and estimate $BER_{r,t,w}$ through analysis (3) and simulation. It was determined that at $B_w = 5$ and $B_w = 4$ bits for the high and low SNR cases, $BER^{opt}_{r,t,w} \approx BER^{unif}_{r,t,w}$. At this precision setting, the shaping gain offered by the BER-optimal ADC is leveraged to lower coefficient precision, and hence overall equalizer complexity.

The post-ADC digital processing units for $\mathbf{EQ}_{conv}$(4b) and $\mathbf{EQ}_{opt}$(3b) were synthesized using Nangate's 45nm cell library, at an operating frequency of 400 MHz. The results are summarized in Table 1. It is observed that at both low and high SNRs the area and power requirements of $\mathbf{EQ}_{conv}$(4b) and $\mathbf{EQ}_{opt}$(3b) are comparable. This implies that the ADC shaping gain and associated ADC power reduction can be realized without expending additional power for the equalizer.

Table 1. Power(μW)/Area(μm^2) Comparison

SNR (dB)	Low SNR (24)	High SNR (32)
$\mathbf{EQ}_{conv}$(4b)	75 (303)	90 (376)
$\mathbf{EQ}_{opt}$(3b)	82 (360)	101 (425)

5. ADC REFERENCE LEVEL ADAPTATION ALGORITHM

In this section, we develop AMBER algorithm for adapting reference levels, analogous to the development of AMBER for equalizer coefficients [4]. The coefficient update equation in AMBER algorithm is derived by considering the LMS algorithm and modifying it to perform updates *only* in the event of an error. The AMBER coefficient update equation is given as,

$$\mathbf{w}[k+1] = \mathbf{w}[k] + \mu_w I[k] \operatorname{sgn}(e[k]) \mathbf{x}[k] \tag{8}$$

where $\mathbf{w}[k] = \{w_0[k], w_1[k], \ldots, w_{L-1}[k]\}$ is the equalizer coefficient vector,

$$\begin{aligned} I[k] &= 1, \text{ if } \tilde{b}[k] \neq b[k] \\ &= 0, \text{ else} \end{aligned}$$

The probability of error, BER, given by (3), is a piecewise step function of $\mathbf{w}$ and $\mathbf{r}$. In order to develop an adaptation algorithm for reference levels, we assume that the equalizer coefficients are fixed and given by $\mathbf{w}$. The slicer input $y[k]$, and the slicer error $e[k]$ are given as,

$$\begin{aligned} y[k] &= \mathbf{w}^T \mathbf{x}[k] \\ e[k] &= b[k] - \sum_{j=0}^{L-1} w[j] x[k-j] \end{aligned} \tag{9}$$

Therefore,

$$E(e^2[k]) = E\left\{ (b[k] - \sum_{j=0}^{L-1} w[j] x[k-j])^2 \right\} \tag{10}$$

The gradient of the square error taken w.r.t the reference levels can be written as,

$$\begin{aligned} \frac{\partial E(e^2[k])}{\partial r_n} &= -2E\left\{ (b[k] - \sum_{j=0}^{L-1} w[j]x[k-j])(\sum_{j \in J_n[k]} w[j]) \right\} \\ &= -2E\left\{ e[k] \sum_{j \in J_n[k]} w[j] \right\} \end{aligned} \tag{11}$$

where, $J_n[k] = \{j : x[k-j] = r_n\}$. Approximating the gradient of MMSE from (11) by its stochastic counterpart and taking the sign of the error, $e[k]$, we get,

$$r_n[k+1] = r_n[k] + \mu_r \operatorname{sgn}(e[k]) \sum_{j \in J_n[k]} w[j] \tag{12}$$

where, μ_r is the step-size for reference-level update. The thresholds are updated as,

$$t_i = \frac{r_i + r_{i+1}}{2}, \text{ for } i = 1, \ldots, N \tag{13}$$

Based on (8), the signed LMS algorithm (12) for ADC reference levels can be modified as follows, to obtain AMBER algorithm for reference levels.

$$r_n[k+1] = r_n[k] + \mu_r I[k] \operatorname{sgn}(e[k]) \sum_{j \in J_n[k]} w[j] \tag{14}$$

To evaluate the algorithm, the equalizer coefficients are initialized to $\mathbf{w} = \mathbf{w}_{opt}$. In practice, $\mathbf{w}$ can be initialized to $[0 \ldots 0]^T$, and the coefficients can be updated using either LMS or AMBER algorithm with the reference levels uniformly spaced in this period. Once the equalizer coefficients converge, the coefficient adaptation can be turned off. The reference-levels are then updated by choosing a very small step-size μ_r owing to the high sensitivity of the BER to reference-levels. The step size necessary was determined through simulations. To evaluate AMBER, a few sets of initial conditions (each uniformly spaced but differing from the others in the full scale value $2V_{max}$) were experimented with and the best performance gains quantified. For the high-ISI channel shown in Fig. 3, Fig. 6

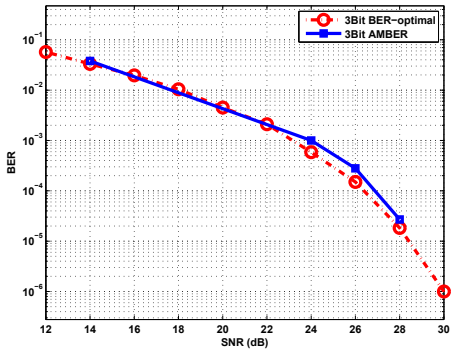

Fig. 6. Performance evaluation of AMBER algorithm for reference-level updates.

shows a comparison between AMBER performance and the performance bound calculated using optimal reference-level settings (Sec-

tion 2) for a 3 bit BER-optimal ADC. The reference levels were initially set to uniformly spaced levels spanning half the full-dynamic range, i.e, $-\frac{V_{max}}{2}$ to $\frac{V_{max}}{2}$. The performance was evaluated based on the threshold values at the end of 4×10^5 bit periods.

Figure 6 demonstrates that using AMBER, it is possible to achieve the shaping gains predicted in [8]. We carried out simulations up to 28 dB, as we were limited by simulation run length. The reference levels get updated only when there is an error; hence it is difficult to determine if the reference levels have converged at high SNR.

6. ADAPTIVE RECEIVER ARCHITECTURE

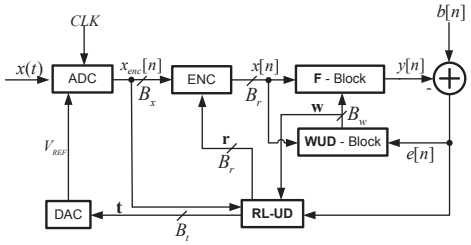

Fig. 7. Adaptive receiver architecture with a LE.

In this section, we present an architecture (Fig. 7) to realize the algorithm developed in Section. 5. The ADC reference levels are set

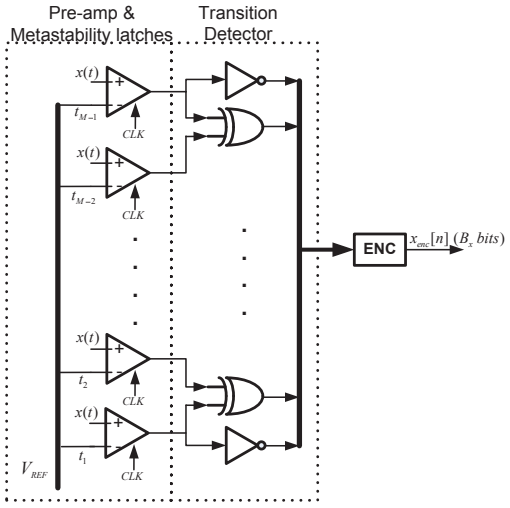

Fig. 8. ADC architecture.

by a digital-to-analog converter (DAC). The ADC consists of pre-amps and metastability latches (Fig. 8) that compare the input signal with the thresholds and generates a thermometer code. A transition detector generates a transition code 1 corresponding to the transition index. This code is then converted to a B_x bit index $x_{enc}[n]$. The index is mapped at the encoder (ENC) to a B_r bit digital representation of the reference levels as illustrated in Fig. 5. The index also feeds into the reference-level update unit (**RL-UD**). The **F**-block and **WUD** blocks realize the conventional filtering (FIR) and coefficient weight-adaptation blocks, respectively.

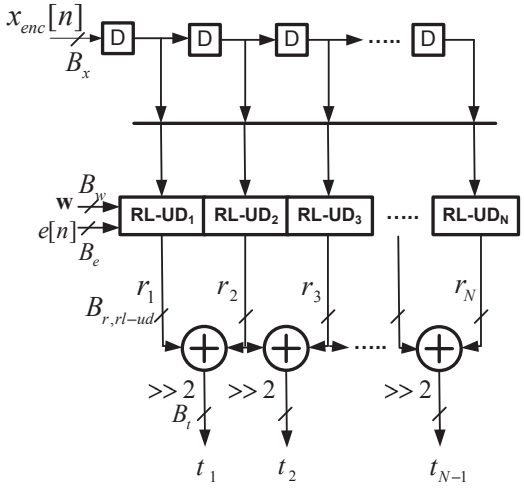

Fig. 9. **RL-UD** architecture.

The **RL-UD** block (Fig. 9) consists of a L-long delay line that buffers the ADC output. This delay line is shared by the update units corresponding to the individual reference levels (**RL-UD**$_1$ . . .**RL-UD**$_N$). The error signal $e(n)$ and coefficients **w** are inputs to this block. To implement (14), $e[n]$ can be quantized to 1-bit by taking its sign, i.e, $B_e = 1$. **RL-UD**$_1$ . . .**RL-UD**$_N$ generate the reference levels r_1 to r_N, which are added and 1-bit right shifted to generate the ADC thresholds (13). The reference levels and thresholds are represented using $B_{r,rl-ud}$ and B_t bits respectively, where $B_t = B_{r,rl-ud} + 1$. The B_r most significant bits (MSBs) of $B_{r,rl-ud}$ are fed into the encoder.

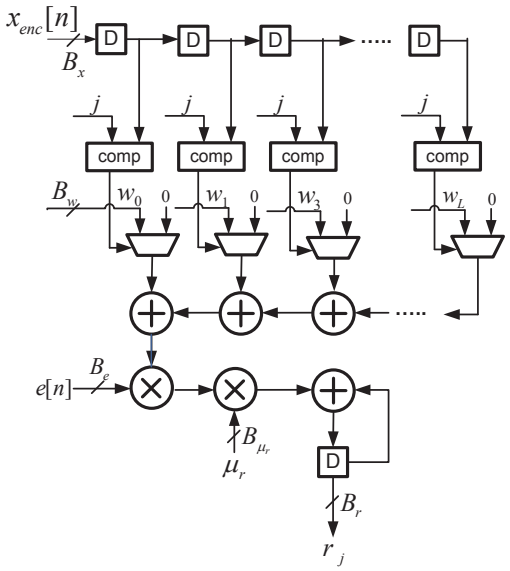

Fig. 10. Reference level r_j update block.

Figure 10 shows a direct-mapped architecture for updating reference level j as described in (14). The index j is compared against the indices in this tapped delay line to generate the control signals to the muxes. The equalizer coefficients ($\mathbf{w}$, B_w bits) are inputs to the muxes. Each mux outputs a 0 when its control signal is 0 and the corresponding coefficient when 1. The equalizer coefficients corresponding to the selected indices ($J_n[k]$) are thus summed, multiplied by the step-size (μ_r) and the sign of the error to generate the update term. This is then added to the stored reference-level to generate the updated reference-level. A sorting network might be necessary to guarantee monotonicity in the levels. We now demonstrate through fixed point simulations, that the architecture proposed in this section is implementable. For this purpose, we consider the following precision allocation: (a) $B_r = 4, B_w = 5$ as determined for the high-SNR case in Section 4. (b) $B_{r,rl-ud}$ is determined by the stopping criterion as follows:

$$B_{r,rl-ud} = log_2 \left(\frac{V_{max}}{\min |(\mu_r \sum_{j \in J_n[k]} w[j])|} \right) \qquad (15)$$

For the step-size used to implement AMBER in Section 5, $B_{r,rl-ud} = 9, B_t = 10$ (15). With these precision values, AMBER was implemented in fixed point.

Table 2. Finite precision AMBER: Performance

SNR (dB)	24 $B_w = 4$	28 $B_w = 5$	32 $B_w = 5$
3-b fixed	1×10^{-3}	6.7×10^{-5}	2×10^{-6}
3-b adaptive	1×10^{-3}	2.9×10^{-4}	2×10^{-5}
4-b fixed	1×10^{-3}	2.6×10^{-4}	2×10^{-5}

It can be inferred from Table 2, that 3-bit AMBER and a 4-bit uniform ADC have similar BER. This implies that the adaptation algorithm proposed in this paper can be implemented in practice and the *shaping gains* predicted in [8] are realizable. We now compare the complexities of conventional LMS algorithm for coefficient adaptation and AMBER algorithm for reference level adaptation, in terms of full-adders (**FA**s). The multiplication with adaptation step size is assumed to be implemented using shift operations and the WUD accumulator precision is assumed to be twice the coefficient precision in the **F**-block.

LMS-Coefficients
Multiplier: $(B_w B_x + L - 1)B_x L$
Adder: $2B_w L$
Total: $((B_w B_x + L - 1)B_x + 2B_w)L$

AMBER-Reference levels
Adder: $(LB_{x_{enc}} + B_{r,rl-ud} + (L-1)B_w)N$
Total: $(LB_{x_{enc}} + B_{r,rl-ud} + (L-1)B_w)N$

For the precision values B_w, B_x and $B_{r,rl-ud}$ determined in this section, this implies that for AMBER, the **FA** complexity is 76% over the conventional adaptive equalizer. However, since the **RL-UD** block is clock gated after convergence, this does not present a power overhead. Therefore, for high-speed links emplying the flash ADC architecture, the proposed AMBER receiver represents a power savings of approximately 50% in the ADC.

7. CONCLUSION

In this paper, we proposed an adaptive algorithm and architecture (AMBER) for designing BER-optimal ADCs with a linear equalizer. In addition, we demonstrated that AMBER-based receivers provide a power savings of 50% in the ADC alone with no increase in equalizer complexity. Future research will be directed towards determining optimal detection algorithms and architectures to maximally leverage the benefits of ADC shaping gain, and improve the robustness of BER-based approaches.

Acknowledgement

The authors would like to thank Neerav Mehta (M.S., ECE, University of Illinois at Urbana-Champaign) for his initial contributions on the subject.

8. REFERENCES

[1] H.-M. Bae, J. Ashbrook, J. Park, N. Shanbhag, A. Singer, and S. Chopra, "MLSE receiver for electronic dispersion compensation of OC-192 fiber links," *IEEE Journal of Solid State Circuits*, vol. 41, no. 11, pp. 2541–2554, 2006.

[2] M. Harwood et al., "A 12.5 Gb/s SerDes in 65nm CMOS using a baud-rate ADC with digital RX equalization and clock recovery," in *IEEE International Solid-State Circuits Conference*, 2007.

[3] P. Schvan et al., "A 24GS/s 6b ADC in 90nm CMOS," in *IEEE International Solid-State Circuits Conference*, 2008.

[4] C.-C. Yeh and J. R. Barry, "Adaptive minimum bit-error rate equalization for binary signaling," *IEEE Transactions on Communication*, vol. 48, no. 7, pp. 1226–1235, 2000.

[5] E. H. Chen et al, "Near-optimal equalizer and timing adaptation for I/O links using a BER-based metric," *IEEE Journal of Solid-State Circuits*, vol. 43, no. 9, pp. 2144–2156, 2008.

[6] A. Meruva and B. Farahani, "A 14-b 32MS/s pipelined ADC with novel fast-convergence comprehensive background calibration," in *IEEE International Symposium on Circuits and Systems*, 2009, pp. 956–959.

[7] P. Nikaeen and B. Murmann, "Digital compensation of dynamic acquisition errors at the front-end of high-performance A/D converters," *IEEE Journal of Selected Topics in Signal Processing*, vol. 3, no. 3, pp. 499–508, 2009.

[8] M. Lu, A.Singer, and N. Shanbhag, "BER-Optimal analog-to-digital converters for communication links," in *International Symposium on Circuits and Systems*, 2010.