High-Performance VLSI Architecture of Adaptive Decision Feedback Equalizer Based on Predictive Parallel Branch Slicer (PPBS) Scheme

Meng-Da Yang, An-Yeu (Andy) Wu, *Member, IEEE*, and Jyh-Ting (Justin) Lai

Abstract—Among existing works of high-speed pipelined adaptive decision feedback equalizer (ADFE), the pipelined ADFE using relaxed look-ahead technique results in a substantial hardware saving than the parallel processing or Look-ahead approaches. However, it suffers from both the signal-to-noise ratio (SNR) degradation and slow convergence rate. In this paper, we employ the predictive parallel branch slicer (PPBS) to eliminate the dependencies of the present and past decisions so as to reduce the iteration bound of decision feedback loop of the ADFE. By adding negligible hardware complexity overheads, the proposed architecture can help to improve the output mean-square error (MSE) of the ADFE compared with the Relaxed Look-ahead ADFE architecture. Moreover, we show the superior performance of the proposed pipelined ADFE by using theoretical derivations and computer simulation results. A VLSI design example using Avant! 0.35- μm CMOS standard cell library is also illustrated. From the post-layout simulation results, we can see that the PPBS scheme requires only 38.4% gate count overhead, but it can help to reduce the critical path from 7.06 to 4.69 ns so as to meet very high-speed data transmission systems.

Index Terms—Adaptive decision feedback equalizer (ADFE), predictive parallel branch slicer (PPBS) scheme, relaxed look-ahead ADFE.

NOMENCLATURE

ADFE	Adaptive decision feedback equalizer.
BPSK	Binary phase-shift keying.
DFL	Decision feedback loop.
FBF	Feedback filter.
FFF	Feedforward filter.
LMS	Least mean-squared algorithm.
MSE	Mean square error.
MMSE	Minimum mean square error.
NEXT	Near-end crosstalk.
PAM5	5-level pulse-amplitude modulation.
PI-	Pipelined ADFE1 in [5].
PEADFE1	
PI-	Pipelined ADFE2 in [5], renamed as
PEADFE2	PP-ADFE in this paper.
PP-ADFE	Pre-Processing ADFE, which is the PI-PEADFE2 in [5].

PPBS
PPBS-ADFE
SNR
UTP-CAT5
WUB
WUF

Predictive parallel branch slicer.
ADFE based on PPBS scheme.
Signal-to-noise ratio.
Unshielded twisted pair category 5.
Weight-update block for FBF.
Weight-update block for FFF.

I. INTRODUCTION

ADAPTIVE decision feedback equalizer (ADFE) using least mean-squared (LMS) algorithm is a well-known equalization technique for magnetic storage and digital communication. The basic block diagram of traditional ADFE is depicted in Fig. 1, where ADFE is composed of two main FIR filters: the feedforward filter (FFF) and the feedback filter (FBF). The outputs from both filters are added together and fed into a slicer. The signal from the slicer output is the final equalized data. The basic function of FFF and FBF is to cancel the precursor and postcursor intersymbol interference (ISI), respectively. On the other hand, the WUC and WUD in the figure stand for the weight-update blocks for FFF and FBF, respectively. The detailed ADFE algorithm and structure are discussed extensively in [1]–[3].

Basically, the fine-grain pipelining of the ADFE is known to be a difficult problem for high-speed applications. This is due to the decision feedback loop (DFL). According to the Iteration Bound [3], the smallest clock period of ADFE is bounded by the DFL. Several approaches are proposed to solve aforementioned problems. For example, pipelining the ADFE can be achieved by precomputing all possible in DFL to open the DFL [4]. However, the parallel approach results in a large hardware overhead as it transforms a serial algorithm into an equivalent (in the sense of input-output behavior) pipelined algorithm. Another algorithm is proposed in [5], which is referred as PI-PEADFE1. It maintains the functionality instead of input-output behavior using the technique of Relaxed Look-ahead. Although the hardware overhead in this algorithm is small, it suffers from some performance degradation such as output signal-to-noise ratio (SNR) and convergence rate. Nevertheless, from VLSI implementation point of view, the second approach is suitable for low-cost VLSI designs.

PIPEADFE1 intends to cancel the first D_1 post-cursor ISI terms by the *feedforward filter* (FFF). However, we observe that it is not necessary to force the first D_1 taps of *feedback filter* (FBF) to zeros. In this paper, we employ the FFF to force the

Manuscript received July 10, 2002; revised July 18, 2003. This work was supported in part by the National Science Council, R.O.C., under Grant NSC 91-2218-E-002020.

The authors are with the Graduate Institute of Electronics Engineering, and also with the Department of Electrical Engineering, National Taiwan University, Taipei 106, Taiwan, R.O.C.

Digital Object Identifier 10.1109/TVLSI.2003.820521

Fig. 1. Architecture of the conventional ADFE.

first D_1 coefficients of FBF to the more appropriate fixed coefficients instead of zeros. Through our derivation, we propose a *predictive parallel branch slicer (PPBS)* scheme. It can be used to eliminate the dependencies of the present and past decisions in order to reduce the iteration bound of DFL. By doing so, we can significantly improve the output MSE in the slicer. In addition, the MSE performance of the proposed PPBS architecture is analyzed mathematically. The theoretical results and computer simulation results show that the output MSE of our proposed architecture is superior to PIPEADFE1. The VLSI implementations show that, based on $0.35\text{-}\mu\text{m}$ CMOS technology, the proposed PPBS scheme can help to reduce the critical path from 7.06 to 4.69 ns compared with conventional ADFE approach at the expense of reasonable hardware overhead.

The rest of this paper is organized as follows. In Section II, we first review the relaxed lookahead pipelined ADFE architecture. In Section III, we derive the proposed PPBS-based ADFE architecture. The performance analysis and computer simulation results are presented in Section IV and V, respectively. In Section VI, we show the design example of the PPBS scheme is UTP-CAT5 channel as well as its VLSI implementation. Finally, we conclude our work in Section VII.

II. REVIEW OF PIPELINED ADFE ARCHITECTURE (PIPEADFE1)

In [5], the *Delayed LMS* [6] and the technique of *transfer delay relaxation* [7] are employed to develop the PIPEADFE1. Then, *sum relaxation* is applied to pipeline the updating circuit of ADFE. The equations describing the PIPEADFE1 (see Fig. 2) are summarized below.

$$\tilde{x}(n) = \mathbf{F}^T(n - D_4)\mathbf{Y}(n) + \mathbf{B}^T(n - D_4)\mathbf{D}(n) \quad (1a)$$

$$e(n) = x(n) - \tilde{x}(n) \quad (1b)$$

$$x(n) = Q[\tilde{x}(n)] \quad (1c)$$

$$\mathbf{F}(n) = \mathbf{F}(n - D_4) + \mu \sum_{i=0}^{LA-1} e(n - D_2 - i)\mathbf{Y}(n - D_2 - i) \quad (1d)$$

$$\mathbf{B}(n) = \mathbf{B}(n - D_4) + \mu \sum_{i=0}^{LA-1} e(n - D_3 - i)\mathbf{D}(n - D_3 - i) \quad (1e)$$

Fig. 2. PIPEADFE1 architecture proposed in [5].

where

- $Q[\bullet]$ BPSK slicer operation;
- $\tilde{x}(n)$ input to $Q(\bullet)$;
- $x(n)$ is the quantizer decision;
- $\mathbf{Y}(n) = [y(n), y(n-1), \dots, y(n-N_f+1)]^T$ is the vector of received samples;
- $\mathbf{D}(n) = [x(n-D_1-1), \dots, x(n-D_1-N_b)]^T$ is the vector of detected symbols;
- $\mathbf{F}(n)$ vector of FFF coefficients;
- $\mathbf{B}(n)$ vector of the rest coefficients in FBF.

III. NEW ARCHITECTURE OF PIPELINED ADAPTIVE DFE (PPBS-ADFE)

In this section, we will introduce the new architecture of pipelined ADFE, which is referred as PPBS-ADFE. For clarity, we will demonstrate the basic concept of PPBS-ADFE by using a simple example. In this example, we assume that the transmitted *binary phase-shift keying (BPSK)* signals are passed through the ISI channel, and the number of taps in FFF and FBF are N_f and N_b , respectively. In the most wirelined communication systems, the channel impulse response can be roughly estimated by the laboratory or field measurement. However, the practical channel impulse response is still different from the estimated channel impulse response. In this paper, we will exploit this feature to develop the PPBS-ADFE. According to the roughly estimated channel response, we can calculate the optimal coefficients, $\mathbf{V}_e$, for the first D_1 taps of the FBF in PPBS-ADFE, and fix these D_1 coefficients. Then, the decision of slicer at time instance n , $x(n)$, can be expressed as

$$\tilde{z}(n) = \mathbf{F}(n)^T \mathbf{Y}(n) + \mathbf{V}_e^T \mathbf{X}(n) + \mathbf{B}(n)^T \mathbf{D}(n) \quad (2)$$

$$x(n) = Q[\tilde{z}(n)] \quad (3)$$

where

- $\tilde{z}(n)$ input of slicer at time instance n ;
- $x(n)$ decision of slicer at time instance n , $x(n) \in \{+1, -1\}$;
- $\mathbf{Y}(n)$ vector of receiver samples in the input of FFF;
- $\mathbf{X}(n) = [x(n-1), x(n-2), \dots, x(n-D_1)]^T$ is the input vector for the first D_1 coefficients in FBF;
- $\mathbf{D}(n) = [x(n-D_1-1), x(n-D_1-2), \dots, x(n-N_b)]^T$ input vector for the rest coefficients in FBF;
- $\mathbf{F}(n)$ vector of FFF coefficients;

Fig. 3. The architecture of PPBS-ADFE.

$\mathbf{B}(n)$ vector of the rest coefficients in FBF;
 $\mathbf{V}_e$ vector of the first D_1 coefficients in FBF.

The algorithm can be considered as a constraint optimization problem. Mathematically, it can be expressed as

$$\text{Min} \left\{ \mathbf{E} (x(n) - \tilde{z}(n))^2 \right\}, \text{ subject to constraint } \mathbf{V}(n) = \mathbf{V}_e \quad (4)$$

where $\mathbf{E}(\bullet)$ denotes the expectation operator.

With the above settings, we can derive the *predictive parallel branch slicer (PPBS) scheme*. It is similar to the *lookahead computation* [4], and can be employed to eliminate the dependencies between $\tilde{z}(n)$ and $\mathbf{X}(n)$ in order to pipeline the DFL. Firstly, the inputs of PPBS, $b(n)$, can be written as

$$b(n) = \mathbf{F}(n)^T \mathbf{Y}(n) + \mathbf{B}(n)^T \mathbf{D}(n). \quad (5)$$

It also can be expressed as

$$b(n) = x(n) - \mathbf{V}_e^T \mathbf{X}(n) + \eta(n) \quad (6)$$

where $\eta(n)$ denotes the residual ISIs and noise components.

According to (6), we still need to remove the first D_1 ISI terms in $b(n)$. There are 2^{D_1} branches inside the PPBS as shown in Fig. 3. In the PPBS, the past D_1 decisions, $\mathbf{X}(n)$, are not available. In the branch j , we assume the past decisions are, $\mathbf{T} = [a_{i,j} \ a_{2,j} \ \dots \ a_{D_1-j}]^T$, which is called the *tentative past*

decision. For convenience, we define a *mapping function*, ϕ , as follows.

$$j = \phi \left(\mathbf{T}_j = [a_{1,j}, a_{2,j}, \dots, a_{D_1-j}]^T \right) = \sum_{i=1}^{D_1} 2^{D_1-i} \left\lceil \frac{(a_{1,j} + 1)}{2} \right\rceil. \quad (7)$$

This mapping function can be interpreted as a one-to-one mapping from the index of branch to its corresponding tentative past decisions. Hence, the tentative decisions in the branch j , d_j , are as follows:

$$d_j = Q [b(n) + \mathbf{V}_e^T \mathbf{T}_j]. \quad (8)$$

The tentative estimation error of each branch in the PPBS, $e_{i,j}$, is as follows:

$$e_{t,j} = d_j - (b(n) + \mathbf{V}_e^T \mathbf{T}_j). \quad (9)$$

Since $\mathbf{V}_e$ and $\mathbf{T}_j$ are fixed, $\mathbf{V}_e^T \mathbf{T}_j$ can be pre-computed in advance. The function in (8) and (9) can be implemented using a fixed coefficient adder. Hence, the hardware overhead and the critical path of PPBS are quite small. Finally, the actual decision of PPBS, $x(n)$, is one of these tentative decisions, and $x(n)$ depends on the past decisions of PPBS, $\mathbf{M} = [x(n-1) \ x(n-2) \ \dots \ x(n-D_1)]^T$. Thus, $x(n)$ can be expressed as

$$x(n) = d_s, \quad s = \phi(\mathbf{M}). \quad (10)$$

The above mapping function can be implemented by using the 2^{D_1} -to-one multiplexer. Moreover, the PPBS also needs to output the estimation error corresponding the final decision of PPBS, $x(n)$, in order to update the FFF and the rest FBF coefficients. The estimation error $e(n)$ is computed as

$$e(n) = e_{t,S}, \quad s = \phi(\mathbf{M}). \quad (11)$$

This operation can also be implemented by the 2^{D_1} -to-one multiplexer.

The architecture of PPBS-ADFE with $D_1 = 3$ (Speedup factor = 4) is shown in Fig. 3. In general, the updating circuits applied in PPBS-ADFE, **WUF** and **WUB**, are similar to the PIPEADFE1 [5]. Since there are D_1 of extra delay elements in the DFL, the iteration bound of PPBS-ADFE can be reduced to $(T_m + 3T_a + T_s)/(D_1 + 1)$, where T_m is symbol multiplier delay, T_a is the adder delay, and T_s is the slicer delay. Then, the conventional retiming technique [1] can be applied to pipeline the DFL. Compared with the PIPEADFE1, the total overheads of hardware complexity in the PPBS-ADFE are merely 2^{D_1} adders and two 2^{D_1} -to-one multiplexer. Due to the inaccuracy of this roughly estimated channel impulse response, the output MSE of PPBS-ADFE will be degraded as the difference between the estimated and practical channel impulse response increases. The detailed relation between the output MSE and the inaccuracy of estimated channel impulse response will be analyzed mathematically in the next section. Moreover, it is obvious that the PPBS-ADFE becomes the PIPEADFE1 when the coefficients of the first D_1 taps, $\mathbf{V}_e$ in the FBF are zeros.

IV. PERFORMANCE ANALYSIS OF PPBS-ADFE

In this section, we will show the relationship between the output MSE of PPBS-ADFE and the inaccuracy of the roughly estimated channel-impulse response by the mathematical analyses and computer simulations. In the following analysis, the optimal coefficients (Wiener solution) for the roughly estimated channel-impulse response is denoted as $\mathbf{W}_e^T = [\mathbf{F}_e^T \mathbf{V}_e^T \mathbf{B}_e^T]$, where $\mathbf{F}_e^T$ represents the optimal coefficients of FFF and $\mathbf{V}_e^T$ and $\mathbf{B}_e^T$ denote the optimal coefficients of the first D_1 taps and the rest coefficients in FBF, respectively.

Similarly, the optimal coefficients for the practical channel impulse response is denoted as $\mathbf{W}_p^T = [\mathbf{F}_p^T \mathbf{V}_p^T \mathbf{B}_p^T]$, where $\mathbf{F}_p^T$ represents the optimal coefficients of FFF and $\mathbf{V}_p^T$ and $\mathbf{B}_p^T$ denote the optimal coefficients of the first D_1 taps and the rest coefficients in FBF, respectively.

Here, we also define the *inaccuracy index*,

$$\Gamma = \frac{(\mathbf{V}_e - \mathbf{V}_p)^H (\mathbf{V}_e - \mathbf{V}_p)}{\mathbf{V}_p^H \mathbf{V}_p} \quad (12)$$

to describe the inaccuracy of the first D_1 FBF coefficients in PPBS-ADFE. Clearly, the *minimum mean-square error (MMSE)* of PPBS-ADFE will be the same as the conventional (serial) ADFE when the fixed coefficients, $\mathbf{V}_e^T$, is equal to $\mathbf{V}_p^T$ ($\Gamma = 0$). Assume that the autocorrelation of the transmitted data is $\sigma_x^2 \mathbf{I}$, where σ_x^2 is the transmitted data variance. By following the analytical method in [1], the MSE of PPBS-ADFE,

given the first D_1 FBF coefficients, $\mathbf{W} = [\mathbf{F}^T \mathbf{V}_e^T \mathbf{B}^T]$, can be expressed as

$$\begin{aligned} J_e(\mathbf{W}) &= J_{\min,p} + (\mathbf{W} - \mathbf{W}_p)^T \mathbf{R}_p (\mathbf{W} - \mathbf{W}_p) \\ &= J_{\min,p} + (\mathbf{W} - \mathbf{W}_p)^T \\ &\quad \times \begin{bmatrix} \mathbf{R}_{YY} & \mathbf{P}_{YX} & \mathbf{P}_{YD} \\ \mathbf{P}_{YX}^T & \sigma_x^2 \mathbf{I} & \mathbf{0} \\ \mathbf{P}_{YD}^T & \mathbf{0} & \sigma^2 \mathbf{I} \end{bmatrix} (\mathbf{W} - \mathbf{W}_p) \\ &= J_{\min,p} + \mathbf{k}^T \mathbf{R}_e \mathbf{k} + \mathbf{r}^T \begin{bmatrix} \mathbf{P}_{YX}^T & \mathbf{0} \end{bmatrix} \mathbf{k} \\ &\quad + \mathbf{k}^T \begin{bmatrix} \mathbf{P}_{YX} \\ \mathbf{0} \end{bmatrix} \mathbf{r} + \sigma^2 \mathbf{r}^T \mathbf{r} \end{aligned} \quad (13)$$

where

$$\begin{aligned} \mathbf{R}_e &= \mathbf{E}\{\mathbf{Y}(n) [\mathbf{D}(n)]^T \mathbf{D}(n) \mathbf{Y}(n)^T\} \text{ is the autocorrelation of the PPBS-ADFE;} \\ \mathbf{R}_{YY} &= \mathbf{E}\{\mathbf{Y}(n) \mathbf{Y}(n)^T\} \text{ is the autocorrelation of FFF input samples;} \\ \mathbf{P}_{YX} &= \mathbf{E}\{\mathbf{Y}(n) \mathbf{X}(n)^T\} \text{ is the crosscorrelation of } \mathbf{Y}(n) \text{ and } \mathbf{X}(n); \\ \mathbf{P}_{YD} &= \mathbf{E}\{\mathbf{Y}(n) \mathbf{D}(n)^T\} \text{ is the crosscorrelation of } \mathbf{Y}(n) \text{ and } \mathbf{D}(n) \\ \mathbf{k}^T &= (\mathbf{F} - \mathbf{F}_p)^T (\mathbf{B} - \mathbf{B}_p)^T; \\ \mathbf{r}^T &= (\mathbf{V}_e - \mathbf{V}_p)^T. \end{aligned}$$

Since $\mathbf{R}_e$ is the positive-definite matrix, the Cholesky factorization can be applied to factorize $\mathbf{R}_e$. That is, $\mathbf{R}_e = \mathbf{L} \mathbf{L}^T$, where $\mathbf{L}$ is the lower triangular matrix. Then, we can reformulate (13) as

$$\begin{aligned} J_e(\mathbf{W}) &= J_e(\mathbf{k}) \\ &= J_{\min,p} + \left(\mathbf{k}^T \mathbf{L} + \begin{bmatrix} \mathbf{r}^T \mathbf{P}_{YX}^T & \mathbf{0} \end{bmatrix} (\mathbf{L}^T)^{-1} \right) \\ &\quad \times \left(\mathbf{L}^T \mathbf{k} + \mathbf{L}^{-1} \begin{bmatrix} \mathbf{P}_{YX} \mathbf{r} \\ \mathbf{0} \end{bmatrix} \right) - \mathbf{r}^T \begin{bmatrix} \mathbf{P}_{YX}^T & \mathbf{0} \end{bmatrix} \\ &\quad \times \mathbf{R}_e^{-1} \begin{bmatrix} \mathbf{P}_{YX} \\ \mathbf{0} \end{bmatrix} \mathbf{r} + \sigma^2 \mathbf{r}^T \mathbf{r} \\ &= J_{\min,p} + \mathbf{H}^T \mathbf{H} + \mathbf{r}^T \\ &\quad \times \left(\sigma_x^2 \mathbf{I} - \begin{bmatrix} \mathbf{P}_{YX}^T & \mathbf{0} \end{bmatrix} \mathbf{R}_e^{-1} \begin{bmatrix} \mathbf{P}_{YX} \\ \mathbf{0} \end{bmatrix} \right) \mathbf{r} \\ &= J_{\min,p} + \mathbf{H}^T \mathbf{H} + \mathbf{r}^T \mathbf{G} \mathbf{r} \\ &= J_e(\mathbf{H}) \end{aligned} \quad (14)$$

where

$$\begin{aligned} \mathbf{G} &\triangleq \sigma_x^2 \mathbf{I} \begin{bmatrix} \mathbf{P}_{YX}^T & \mathbf{0} \end{bmatrix} \mathbf{R}_e^{-1} \begin{bmatrix} \mathbf{P}_{YX} \\ \mathbf{0} \end{bmatrix} \\ \mathbf{H} &\triangleq \mathbf{L}^T \mathbf{k} + \mathbf{L}^{-1} \begin{bmatrix} \mathbf{P}_{YX} \mathbf{r} \\ \mathbf{0} \end{bmatrix}. \end{aligned} \quad (15)$$

Since $\mathbf{r}$ is fixed, the MMSE of the PPBS-ADFE, $J_{e,\min}$, can be expressed as

$$J_{e,\min} = \min_{\mathbf{k}} \{J_e(\mathbf{W})\} = \min_{\mathbf{H}} \{J_e(\mathbf{H})\}. \quad (16)$$

Since $\mathbf{H}^T \mathbf{H} \geq 0$, we can have $J_{e,\min} = J_e(\mathbf{H} = \mathbf{0}) = J_{\min,p} + J_{\text{loss}}$. Basically, J_{loss} can be interpreted as the MSE degradation of PPBS-ADFE. Here, we define $S_i = (J_{\text{loss}}/\Gamma)$ in order to describe how the *Inaccuracy Index* of the estimated channel impulse response, Γ , affects J_{loss} . Since $\mathbf{G}$ is the Hermitian matrix. S_i can be reformulated into the following form:

$$S_i = \left(\frac{\mathbf{r}^T \mathbf{G} \mathbf{r}}{\mathbf{r}^T \mathbf{r}} \right) \mathbf{V}_p^T \mathbf{V}_p = \left(\frac{\mathbf{r}^T \mathbf{Q}^T \mathbf{D}_e \mathbf{Q} \mathbf{r}}{\mathbf{r}^T \mathbf{r}} \right) \mathbf{V}_p^T \mathbf{V}_p, \quad (17)$$

where $\mathbf{D}_e$ is the diagonal matrix, and $\mathbf{Q}^T \mathbf{Q} = \mathbf{1}$. From (17), it implies the PPBS-ADFE is most sensitive to the inaccuracy of the estimated channel impulse response when $\mathbf{r} = c\mathbf{q}_{\text{Max}}$, where $\mathbf{q}_{\text{Max}}$ is the eigenvector corresponding to the maximum eigen-value of $\mathbf{G}$, λ_{MAX} . Therefore, we have the bound of $J_{e,\text{loss}}$ as

$$S_{\text{MAX}} \cdot \Gamma = \lambda_{\text{MAX}} \cdot \Gamma \geq J_{e,\text{loss}} \geq S_{\text{MIN}} \cdot \Gamma = \lambda_{\text{MIN}} \cdot \Gamma. \quad (18)$$

S_i can be also interpreted as the sensitivity index of the PPBS-ADFE. As long as Γ is under the tolerated range, the output MSE of PPBS-ADFE can be lower than PIPEADFE1. This tolerated range will be explained in the following simulations.

V. COMPUTER SIMULATIONS AND HARDWARE COMPARISON

In this section, we will show that the output MSE of the proposed PPBS-ADFE can be lower than PIPEADFE1 by the simulation results when the estimated channel impulse response is accurate enough. In addition, we will show the requirement of Γ to grantee that the output MSE of PPBS-ADFE is lower than PIPEADFE1.

A. Simulation I

In the Simulation I, the practical channel impulse response we employ in our simulations is, $h = [0.04, 0.05, 0.07, 0.21, 0.5, 0.72, 0.36, 0.21, 0.03, 0.07]$, which is obtained by the typical response of a good-quality telephone channel [2]. Assume that the number of taps in FFF and FBF are $N_f = 12$ and $N_b = 7$, respectively, and the input SNR is equal to 18 dB. Using the method in [1], we can calculate the optimal coefficients the first D_1 FBF coefficients for the conventional ADFE, $\mathbf{V}_p^T = [-1.1321, -0.9955, -0.4725]$. Hence, we assume the first D_1 FBF coefficients of the PPBS-ADFE, $\mathbf{V}_e^T = [-0.5661, -0.4978, -0.2362]$, which is obtained by the optimal solution of the roughly estimated channel impulse response in advance. From (12), the inaccuracy of $\mathbf{V}_e^T$, Γ , is equal to 0.25. The learning curves of conventional ADFE, PIPEADFE1, and PPBS-ADFE are depicted in Fig. 4(a). In Fig. 4(a), the three horizontal lines show theoretical MSE bounds of the conventional ADFE, PIPEADFE1, and PPBS-ADFE, respectively. Next, we change the input SNR to 24 dB and repeat the simulation, and its simulation results are shown in Fig. 4(b). Based on the simulation results shown in Fig. 4(a) and (b), it illustrate that the output MSE of the proposed PPBS-ADFE is lower than the PIPEADFE1 when $\mathbf{r} = [0.5661, 0.4978, 0.2362]$ in this example. The FFF of PIPEADFE1 intends to cancel all precursor ISI terms and the first D_1 postcursor ISI terms. This can be indicated in Fig. 5, where the combined channel and FFF impulse response for PIPEADFE1 with $D_1 = 3$ is shown. In PPBS-ADFE, the FFF does not intend to eliminate all the first D_1 postcursor ISI terms. Hence, the burden of the FFF in PPBS-ADFE can be alleviated. This phenomenon can be indicated in Fig. 6, where the combined channel and FFF impulse response for PPBS-ADFE with $D_1 = 3$ is shown. It is obvious that the FFF forces the first D_1 postcursor ISI's seen by FBF to $\mathbf{V}_e = [-0.5661, -0.4978, -0.2362]$ instead of canceling the first D_1 ISI terms.

(a)

(b)

Fig. 4. Learning curves of the conventional ADFE, PIPEADFE1, and PPBS-ADFE with (a) the input SNR = 18 dB and (b) the input SNR = 24 dB.

B. Simulation II

In the Simulation II, we will show the requirement of Γ to guarantee that the output MSE of PPBS-ADFE is lower than PIPEADFE1. From (17), we can see that, as long as $\Gamma \leq J_{e,\text{loss,PIPEADFE1}}/S_{\text{MAX}}$, the output MSE of PPBS-ADFE is always superior to the PIPEADFE1. We will show this property by a simple example. Basically, we will apply the same parameters and the channel environment like the Simulation I except $\mathbf{r}$. In this example, we will consider the three types of $\mathbf{r}$, which are $\mathbf{r}_1$, $\mathbf{r}_2$, and $\mathbf{r}_3$. ($\mathbf{r}_1$, $\mathbf{r}_2$, and $\mathbf{r}_3$ are $c_1\mathbf{V}_p$, $c_2\mathbf{q}_{\text{Max}}$, and $c_3\mathbf{q}_{\text{Min}}$, respectively, where c_1 , c_2 , c_3 are scalar) Then, we will observe their maximum achievable MSE at different Γ by the simulation results and the theoretical results. By the method in [1], the minimum achievable MSE of the PPBS-ADFE is

$$\text{MSE}_{\infty, \text{PIPEADFE2}} = \frac{J_{e,\text{min}}}{1 - \sum_{i=1}^{N_f+N_b-D_1} \mu\lambda_i/(2 - \mu\lambda_i)} \quad (19)$$

Fig. 5. Combined channel and the FFF impulse response of PIPEADFE1.

Fig. 6. Combined channel and FFF impulse response of PPBS-ADFE.

where μ is the step size, λ_i is the i th eigenvalue of $\mathbf{R}_e$. The simulation results (dash line) and the theoretical lower bounds (solid line) for each case are shown in Fig. 7. The PIPEADFE1 is the case when $c_1 \mathbf{V}_p = 0$. Assume the MSE degradation of the PPBS-ADFE, $J_{\text{loss,PPBS-ADFE}} = S_i \cdot \Gamma$, is always lower than the MSE degradation of PIPEADFE1, $J_{\text{loss,PIPEADFE1}}$. Based on the simulation results shown in Fig. 7 and (17), the maximum inaccuracy index, Γ , must be smaller than $J_{\text{loss,PPBS-ADFE}}/S_{\text{MAX}} = 0.1442$ in order to guarantee that the output MSE of PPBS-ADFE is lower than PIPEADFE1. In other words, $|c_2 \mathbf{q}_{\text{MAX}}|$ must be small than $|0.6 \cdot \mathbf{q}_{\text{MAX}}| = |[-0.3751 \ 0.4244 \ -0.1979]|$. However, we may loosen the requirements in the practical situation since $\mathbf{V}_e$ is not always in the direction of $\mathbf{q}_{\text{MAX}}$. In practical, since $\mathbf{V}_e$ are often the combination of all orthogonal eigen-vector of $\mathbf{G}$, the designers can store more than one $\mathbf{V}_e$ in the ROM, and select one value for applications. Then, we can achieve the lower output MSE.

C. Comparisons of Hardware Complexity

Next, we consider the comparisons of hardware complexity between PIPEADFE1 and PPBS-ADFE under the same speedup factor, $S = D_1 + 1$. Since the fixed coefficient adders and slicers in the PPBS can be implemented using an adder, we treat these two components as one slicer. The comparisons

Fig. 7. Output MSE versus the inaccuracy index, Γ .

between the PIPEADFE1 and PPBS-ADFE with speedup factor = S are listed in Table I.

There is still another ADFE structure, PIPEADFE2 proposed in [5]. We name it as *Pre-Processing ADFE (PP-ADFE)* in our discussion. In above simulations, we only compare our PPBS-ADFE with PIPEADFE1, but did not directly compare it to PP-ADFE due to the following reasons:

- From hardware point of view, PIPEADFE1 is the best and the easier way to relax the critical path of DFL by just inserting the delays. The proposed PPBS-ADFE is *almost the same as* PIPEADFE1 except the modification of the slicer part.
- The PP-ADFE is a totally different architecture from PIPEADFE1 and the better converge rate is from the usage of the extra filter, Pre-Processing unit (PP). Strictly speaking, it is not a direct modification from PIPEADFE1 since it employs approximations to derive the Pre-Processing scheme. Moreover, the output SNR of PP-ADFE is *the same as* PIPEADFE1 since they both force the first D_1 tap in FBF to zeros. Compared with PIPEADFE1 and PP-ADFE, our PPBS-ADFE will achieve better output SNR when we can estimate the channel characteristic in advance, which is discussed in Section IV.

In Table I, we compare the hardware complexities about aforementioned four types of ADFE: Series ADFE, PIPEADFE1, PP-ADFE, and PPBS-ADFE. For our design, there are two issues, which will result in the hardware overhead. One is large D_1 problem and the other is the large constellation problem. We explain both issues as follows:

- *D1 problem*: The hardware complexity would be very high due to the large D_1 . However, for most applications, the D_1 is small (typically 1 to 4). The reason is that the speedup factor is $(D_1 + 1)$, and usually we need not to speed up the circuits too much.
- *Large constellation problem*: The PPBS-ADFE is suitable for low constellation modulation such as BPSK, MLT-3, PAM-3, PAM-5 and QPSK, because the PPBS-ADFE with high constellation modulation will require a large hardware complexity overhead. Although the PPBS-ADFE requires additional 2^{s-1} slicer, the BPSK slicer is just a constant adder, which is very small and has no timing issue.

TABLE I
COMPARISON TABLE OF HARDWARE COMPLEXITIES OF VARIOUS ADFE SCHEMES

	<i>Serial ADFE</i>	<i>PIPEADFE1</i>	<i>PP-ADFE</i>	<i>PPBS-ADFE</i>
Mult. In FFF	N_f	N_f	N_f	N_f
Mult. In WUF	N_f	N_f	N_f	N_f
Mult. In FBF	N_b	$N_b - S + 1$	$N_b - S + 1$	$N_b - S + 1$
Mult. In WUB	N_b	$N_b - S + 1$	$N_b - S + 1$	$N_b - S + 1$
Mult. In PP	0	0	S-1	0
Total adders	$2(N_f + N_b) - 1$	$2(N_f + N_b - S) + 1$	$2(N_f + N_b - S) + 1$	$2(N_f + N_b - S) + 1$
BPSK Slicer	1	1	1	2^{S-1}
2^{S-1} -to-1 MUX	0	0	0	2

In summary, the overhead of PPBS-ADFE is acceptable in low constellation applications.

On the other hand, the complexity of the PP-ADFE is almost the same as that of Serial ADFE. However, we have to highlight the increased hardware in the PP scheme. Basically, the multipliers in FBF are very simple, because the input signals are in “symbol space”. For example, only 3-bit multipliers are needed in PAM-5 constellation. On the contrary, the multipliers of FFF and PP are full multiplier. For example, if the output signal of A/D converter is 10 bit and the word length of weights is 15, the size of multiplier is as large as 10×15 . Therefore, the hardware complexity of PP is very high, and thus, the PP-ADFE has much higher overhead to be implemented in real chips. To sum up, the PPBS-ADFE is a modified version from PIPEADFE1 while keeping the advantage of PIPEADFE1.

VI. DESIGN EXAMPLE AND VLSI IMPLEMENTATION OF PPBS-ADFE

A. UTP-CAT5 Design Example

In the design example, we will apply the PPBS-ADFE to Gigabit Ethernet application where the channel is a typical Unshielded Twisted Pair Category 5 (UTP-CAT5) channel. We assume that the receiver can operate in the optimal sampling phase. The channel model we considered in this example is illustrated in Fig. 8. Before we begin the design example of the PPBS-ADFE, we will introduce some characteristic of the UTP-CAT5 cables.

- The channel impulse responses of the UTP-CAT5 with different cable length are quite different. The typical channel impulse response, $h(n)$, with cable length = 25, 50, and 100 m are shown in Fig. 9. Based on the observation in Fig. 9, we discover the cable impulse responses of UTP-CAT5 cable with different cable length have the similar shape.
- The *signal-to-near-end crosstalk (NEXT) ratio (SNR)* of the longer cable is much lower than the shorter one since the longer cable introduces the larger attenuation.

Assume we transmit 5-level pulse-amplitude-modulation (PAM5) over the UTP-CAT5 cables, which's cable length range from 0 to 100 m. It is obvious that it results in the worst

Fig. 8. Channel model.

output MSE of the ADFE when we transmit PAM5 signal over 100-m UTP-CAT5 cable. Hence, we can apply the estimated channel-impulse response of 100-m UTP-CAT5 cable to obtain $\mathbf{V}_e$ in order to minimize the worst case output MSE (SNR) of the PPBS-ADFE. Here, we assume $N_f = 8$, $N_b = 14$, and $D_1 = 2$. It is clear that Γ is equal to zero when the cable length is 100 m. The input SNR and the output SNR of the conventional ADFE, PIPEADFE1 and PPBS-ADFE with different cable length are shown in Fig. 10. From Fig. 10, we can make the following observations:

- The output SNR of the conventional ADFE, PIPEADFE1, and PPBS-ADFE increase when the cable length decreases. In addition, all approaches suffers less SNR degradation when the cable length is less than 100 m.
- The output SNR of PPBS-ADFE is superior to the PIPEADFE1 with cable length of UTP-CAT5 ranging from 100 to 10 m. Since the channel impulse response of 0-m UTP-CAT5 cable is close to a delta function, the optimal first D_1 taps for 0-m UTP-CAT5 cable are almost zeros. Hence, the output SNR of the PIPEADFE1 will be higher than the output SNR of PPBS-ADFE. However, the PIPEADFE1 cannot perform well when the first D_1 post-cursor ISI terms are significantly large. Hence, the PPBS-ADFE can be employed to resolve this problem.
- The design example demonstrates the PPBS-ADFE design is very effective for UTP-CAT5 cables. By using the same methodology, we can apply the PPBS-ADFE to other receiver designs of wirelined communication systems.

B. VLSI Implementation of PPBS-ADFE

In this section, we implement the PPBS-ADFE in the Section VI by using Avant! 0.35 μm standard cell library (see Fig. 11), where the zero-force LMS is used to decrease the

Fig. 9. Impulse response of UTP-CAT5 with: (a) cable length = 25 m; (b) cable length = 50 m; (c) cable length = 75 m; (d) cable length = 100 m (over - sampling rate = 8).

Fig. 10. The input SNR and the output SNR of the conventional ADFE, PIPEADFE1, and PPBS-ADFE with different cable length.

dynamic range of weights of the FFE [8]. For comparisons purposes, we also apply the same wordlength assignment and the same arithmetic units to implement the conventional ADFE.

Fig. 11. Chip layout of the PPBS-ADFE.

The synthesis results of PPBS-ADFE and the conventional ADFE, which are obtained from the *Synopsys Design Compiler* are depicted in Table II. Here, we summary the VLSI implementation results of the conventional ADFE and PPBS-ADFE by the following descriptions.

- The synthesis results of the conventional ADFE and PPBS-ADFE are shown in Table II. In our VLSI implementation of PPBS-ADFE, the total overhead of PPBS-ADFE is about 70%. The overheads coming from the PPBS, which contains 25 PAM5 slicers, are about 38.4%. The overheads coming from the other modules are almost the pipelined registers.
- Because the physical delay of flip-flop is not zero, the critical path of PPBS-ADFE must be re-expressed as $(T_m + 3T_a + T_s)/(D_1 + 1) + T_D$, where T_D is the physical delay of flip-flop. In the VLSI implementations of the conventional ADFE and PPBS-ADFE, the Timemill simulation shows that the critical paths of the conventional ADFE and PPBS-ADFE are 7.06 ns and 4.69 ns, respectively. Hence, the throughput rate of PPBS-ADFE can be 1.5 times faster than the conventional ADFE at reasonable hardware overhead. It demonstrate the effectiveness of the proposed design.

VII. CONCLUSIONS

In this paper, a new pipelined PPBS-based ADFE using the roughly estimated channel impulse response is presented. To make sure PPBS-based ADFE can be applied to the cases where the pre-estimated channel is not exactly the real channel, we employ the concepts of the sensitivity and inaccuracy indexes to provide a systematical analysis. Compared with the relaxed lookahead ADFE algorithm in [5] through the simulations and analyses, we show that the output MSE of the proposed algorithm can be improved by adding negligible overhead of hardware complexity. Therefore, the proposed PPBS-based ADFE can achieve both goals of high SNR and high operation speed.

TABLE II
SYNTHESIS RESULTS OF THE CONVENTIONAL ADFE AND PPBS-ADFE

	FFE Filter	FFE Update	FBE Filter	FBE Update	Zero Force	PPBS	Total
Gate count of the conventional ADFE	6,205	3,455	3,633	3,949	1,857	0	19,133
Gate count of PPBS-ADFE	9,127	4,890	4,322	4,534	2,321	7,571	32,839
Overhead	15.6%	7%	3.3%	3%	2.7%	38.4%	70%

Moreover, we also verify the proposed pipelined PPBS-based ADFE architecture by implementing a prototyping chip based on Avant! standard cell library. The target application is Gigabit Ethernet and the relative channel is UTP-CAT5 cable. Without loss the SNR in long cable case, the delay of critical path can be improved from 7.06 to 4.69 ns compared with conventional ADFE schemes. To sum up, we provide a novel alternative approach for the design of high-speed pipelining ADFE when the output MSE is critical.

REFERENCES

- [1] S. Haykin, *Adaptive Filter Theory*, 2nd ed. Englewood Cliffs, NJ: Prentice-Hall, 1991.
- [2] J. G. Proakis, *Digital Communications*, 4th ed. New York: McGraw-Hill, 2001.
- [3] K. K. Parhi, *VLSI Digital Signal Processing System*. New York: Wiley-Interscience, 1999.
- [4] —, "Pipelining in algorithm with quantizer loops," *IEEE Trans. Circuits Syst.*, vol. 38, pp. 745–754, July 1991.
- [5] N. R. Shanbhag and K. K. Parhi, "Pipelined adaptive DFE architectures using relaxed look-ahead," *IEEE Trans. Signal Processing*, vol. 43, pp. 1368–1385, June 1995.
- [6] G. Long, F. Ling, and J. G. Proakis, "The LMS algorithm with delayed coefficient adaptation," *IEEE Trans. Acoust. Speech, Signal Processing*, vol. 37, pp. 1397–1405, Sept. 1989.
- [7] B. Widrow and S. D. Stearns, *Adaptive Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1985.
- [8] R. He, N. Nazari, and S. Sutardja, "A DSP based receiver for 100BASE-T PHY," in *Proc. IEEE Int. Solid-State Circuits Conf. (ISSCC)*, sec. 19, 2001.

Meng-Da Yang received the B.S. and M.S. degrees from National Taiwan University in 2000 and 2002, respectively, all in electrical engineering.

He is currently an engineer with MediaTek, Hsin-chu, Taiwan, R.O.C., working on advanced optical storage IC design. His research interests include low-power/high-performance VLSI architectures for DSP and communication applications, adaptive signal processing, optical read channel applications, and error-correcting coding.

An-Yeu (Andy) Wu (S'91–M'96) received the B.S. degree from National Taiwan University in 1987, and the M.S. and Ph.D. degrees from the University of Maryland, College Park, in 1992 and 1995, respectively, all in electrical engineering.

During 1987–1989, he served as a signal officer in the Army, Taipei, Taiwan, R.O.C., for his mandatory military service. During 1990–1995, he was a graduate teaching and research assistant with the Department of Electrical Engineering and Institute for Systems Research at the University of Maryland, College Park.

From August 1995 to July 1996, he was a Member of Technical Staff at AT&T Bell Laboratories, Murray Hill, NJ, working on high-speed transmission IC designs. From 1996 to July 2000, he was with the Electrical Engineering Department, National Central University, Taiwan, R.O.C. He is currently with the Graduate Institute of Electronics Engineering and Department of Electrical Engineering, National Taiwan University, Taiwan, R.O.C. His research interests include low-power/high-performance VLSI architectures for DSP and communication applications, adaptive/multirate signal processing, and reconfigurable broadband access systems and architectures.

Dr. Wu is currently serving as an Associate Editor for *EURASIP Journal of Applied Signal Processing*, and is a Guest Editor for a special issue on "Signal Processing for Broadband Access Systems: Techniques and Implementations" in the same journal. Since July 2003, he is an Associate Editor for the IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION (VLSI) SYSTEMS. He has served on the technical program committees of the IEEE International Conference on Image Processing (ICIP), IEEE Workshop on Signal processing Systems (SiPS), IEEE Asia-Pacific Conference on SSICs (AP-ASIC), IEEE International Symposium on Intelligent Signal processing and Communication Systems (ISPACS), IEEE International Conference on Multimedia & Expo (ICME), and IEEE International SOC Conference (formerly IEEE International ASIC/SOC Conference).

Jyh-Ting (Justin) Lai was born in Taiwan, R.O.C., in 1972. He received the B.S. degree from the National Chiao Tung University, Taiwan, R.O.C., and the M.S. degrees from National Taiwan University, in 1995 and 1997, respectively, all in electrical engineering. He is currently working toward the Ph.D. degree in the Graduate Institute of Electronics Engineering, National Taiwan University, in the field of DSP-based high-performance VLSI designs.

During 1998–1999, he served as second lieutenant personnel officer in the Army, Taiwan. In 1999, he served as a technical officer in the Directorate General of Telecommunications, where he joined the communication liberalization team to deal with the communication regulations. During 2000–2003, he was Vice Manager of the Mixed Signal Division, Faraday Technology Corporation, Hsin-chu Industrial Science Park, Taiwan, R.O.C., where he led Gigabit Ethernet and other serial-link projects. His current research interests are in the area of VLSI implementation of DSP algorithms, adaptive filtering, and digital communication systems.