Available at
www.ElsevierComputerScience.com
POWERED BY SCIENCE DIRECT®
Signal Processing 85 (2005) 1301–1313
SIGNAL PROCESSING
www.elsevier.com/locate/sigpro

Asynchronous LMS adaptive equalization ☆

Jan W.M. Bergmans[a,*], Maria Yu Lin[b], David Modrie[c], Rob Otte[d]

[a]Eindhoven University of Technology, Bldg. EH3.06, P.O. Box 513, 5600 MB Eindhoven,The Netherlands
[b]Data Storage Institute, Singapore
[c]Philips Innovative Technology Solutions N.V., Leuven, Belgium
[d]Philips Lighting, Eindhoven, The Netherlands

Received 27 March 2003; received in revised form 15 July 2004

Abstract

Digital data receivers often operate at a fixed sampling rate $1/T_s$ that is asynchronous to the baud rate $1/T$. A digital equalizer that processes the incoming signal will also operate in the asynchronous clock domain. Existing adaptation techniques for this equalizer involve an error sequence e_k that is produced in the synchronous clock domain, and converted to the asynchronous domain via an inverse sampling-rate converter. Several disadvantages of this approach may be avoided by means of an alternative topology that is developed and analyzed in this paper. Numerical results for an idealized optical storage channel serve to illustrate the merits of the approach.

Keywords: Adaptive equalization; Digital recording; Optical storage; LMS; Timing recovery; Partial-response techniques

1. Introduction

Receivers for digital transmission and storage systems are often realized with the aid of digital IC technology. To profit optimally from the rapid advances of this technology, analog-to-digital conversion is ideally performed early on in the receiver. A common baseband topology is depicted in Fig. 1.

A received signal $r(t)$ is applied to an analog low-pass filter (LPF) which suppresses out-of-band noise. The LPF output is digitized by an analog-to-digital converter (ADC) which operates at a crystal-controlled free-running frequency $1/T_s$ that is high enough to prevent aliasing. The ADC output is applied to an equalizer (EQ) which serves to condition intersymbol interference (ISI) and noise. The equalizer operates at the sampling rate $1/T_s$, i.e. asynchronously to the baud rate $1/T$. A sampling-rate converter (SRC) produces an equivalent synchronous output which serves as

☆This work was supported by Philips Research Labs, Eindhoven, The Netherlands.
*Corresponding author. Tel.: +31 40 247 4438; fax: +31 40 246 6508.
E-mail address: J.W.M.Bergmans@tue.nl (J.W.M. Bergmans).

0165-1684/$ - see front matter © 2005 Published by Elsevier B.V.
doi:10.1016/j.sigpro.2004.12.006

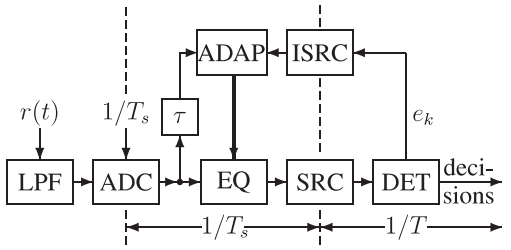

Fig. 1. Baseband receiver with asynchronous equalizer. Asynchronous and synchronous clock domains are indicated with the symbols $1/T_s$ and $1/T$, respectively.

the input of a bit detector (DET). The SRC forms part of a timing-recovery loop which is not depicted explicitly. Rather than placing the equalizer before the SRC as in Fig. 1, it would also be possible to reverse their order. That would, however, cause the latency of the equalizer to contribute to the overall delay of the timing-recovery loop, thus significantly lowering its stability margin and attainable acquisition speed. Also, the sampling rate $1/T_s$ can be lower than the baud rate $1/T$ whenever the channel has negative excess bandwidth. This is so, for example, in most optical storage systems. In such cases the asynchronous equalizer can have fewer taps and lower operating speed than its synchronous counterpart, thereby lowering complexity and power dissipation.

To cope with variations of the system parameters, the equalizer often needs to be adaptive. Existing asynchronous adaptation techniques are extensions of LMS and involve cross-correlation of the tap sequences with a suitable 'asynchronous' error sequence (see, for example, [1,2]). To this end, a synchronous error sequence e_k may be extracted from the bit-detector and converted to the asynchronous clock domain via an inverse sampling-rate converter (ISRC; see Fig. 1). Unlike the SRC, the ISRC must handle a wide-band input and can hence be comparatively complex. Prior to cross-correlation, the error and tap sequences must be matched both in sampling rate and in phase. The first condition is met via the ISRC. The second one requires that the total latency of SRC, bit-detector, error formation circuit, and ISRC be matched by delaying the tap sequences accordingly. Delayed tap sequences may be derived (through the block that is labelled 'τ' in Fig. 1) from the equalizer input. Delay matching then boils down to a proper choice of the delay τ. Both ISRC and delay matching add to the complexity of the solution. Delay matching, moreover, may not be accurate because of the time-varying nature of the latency of SRC and ISRC. As a result, adaptation performance may degrade.

This paper develops and analyzes an alternative adaptation topology, first proposed in [3], that overcomes these disadvantages (Fig. 2). The basic idea is to do the cross-correlation in the synchronous clock domain instead. To this end, the equalizer input is converted to the synchronous domain by means of an auxiliary SRC, and a fractional shift register (FSR) produces synchronous versions of the tap sequences. These are correlated with the synchronous error sequence e_k to produce tap update information, which is converted into tap values via a bank of integrators. To close the adaptation loop, the outputs of this bank must be converted to the asynchronous clock domain. Since tap values change only slowly with respect to both $1/T$ and $1/T_s$, this inverse sampling-rate conversion can be done in the simplest conceivable manner, namely via a bank of latches (or, equivalently, zeroth-order interpolation). As a result, the topology of Fig. 2 effectively avoids the two problems that were mentioned above, at the cost of an auxiliary SRC that can be identical to the main SRC. This cost will often be modest, and can, in fact, be lowered further by permitting the auxiliary SRC to differ from the main SRC. This possibility is elaborated in [4], yet is beyond the scope of the present text. Added complexity of the FSR tends to be negligible.

The remainder of this paper is organized as follows. Section 2 describes the system model and nomenclature, and Section 3 introduces the new adaptation topology in more detail. Section 4 analyzes the steady-state equalizer settings. Section 5 discusses the design of the FSR, while Section 6 analyzes the maximum sampling-rate range that a fixed FSR can handle. Section 7 presents numerical results that illustrate the steady-state solution of the adaptation loop *vis a vis* the minimum