TIME-INTERLEAVED ANALOG-TO-DIGITAL CONVERTERS FOR DIGITAL COMMUNICATIONS

Tsung-Heng Tsai, Paul J. Hurst, and Stephen H. Lewis
Solid-State Circuits Research Laboratory
Department of Electrical and Computer Engineering
University of California, Davis
USA
ttsai@ece.ucdavis.edu

ABSTRACT
Techniques to overcome the errors caused by the offset, gain, and sample-time mismatches among time-interleaved analog-to-digital converters (ADCs) in high-speed digital communication systems are presented. The errors introduced by these mismatches are adaptively corrected using digital signal processing blocks. Sample-time errors are corrected by modifying the operation of the existing adaptive receive equalizer to reduce the hardware overhead. Simulations are presented that show that the gain, offset, and sample-time mismatches are corrected by the adaptive loops.

KEY WORDS
Analog circuits, digital signal processing, communication systems, calibration

1. Introduction

Digital processing of analog signals has become an attractive option in many applications, such as digital communications, storage channels, and waveform acquisition. A key advantage of digital processing blocks is that they can be easily scaled as CMOS technology advances. In a digital communication receiver, an 'all-digital' receiver requires a high-resolution analog-to-digital converter (ADC) to digitize the received signal. A block diagram of a candidate receiver is shown in Fig. 1. The ADC is followed by a digital automatic gain control (AGC) block, which amplifies the signal, an adaptive equalizer, which compensates for intersymbol interference (ISI) introduced by the channel, and a digital interpolated timing recovery (ITR) block, which uses interpolation to recover samples at the symbol rate from the slightly oversampled signal from the ADC [1]. The resulting signal is sliced to produce an estimate of the transmitted data. The resolution required in the ADC will be typically at least 10 bits, because it must not overload when receiving a strong signal and must give a reasonable output signal-to-noise-and-distortion ratio (SNDR) when receiving a weak signal. For example, if the input signal range is 40 dB, the peak-to-rms ratio of the receive signal is 2, and the desired SNDR is 20 dB, then an ADC with 11 bits of resolution is required. While high-resolution ADCs can be built, their conversion rates are limited.

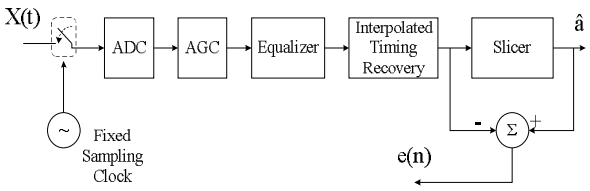

Fig. 1: Block diagram of a receiver.

To achieve high resolution at speeds near and above 1Gb/s, for example, multiple ADCs could be time-interleaved, as shown in Fig. 2 [2-10]. Here, multiple parallel ADCs are used in a parallel time-interleaved array. In this structure, each ADC channel samples the input in turn. With *M* ADC channels operating in parallel, the overall sampling rate increases by the factor *M* over that of a single channel. However, mismatches in gain, offset, and sample time among the time-interleaved ADCs generate undesired spectral components [2-10] and degrade the SNDR of the system.

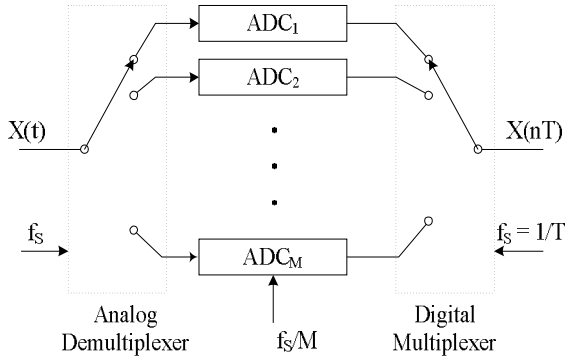

Fig. 2: Block diagram of the time-interleaved ADCs.

Digital background calibration has been demonstrated for correcting offset and gain mismatches [2-7]. A calibration signal added to the input of the ADC or a random chopping signal in the sample-and-hold

circuit has been used to remove the offset and gain mismatches. Previous work regarding sample-time mismatches in parallel channels was reported in [7-9]. In [9], a two-rank sample-and-hold (S/H) structure is used to reduce sample-time errors. Here, the first S/H must operate at the overall sampling rate of the converter, which will be difficult at high sampling rates with many ADC channels. In [8], linear interpolation is proposed and relies on oversampling the input signal to accurately estimate the corrected outputs. Without oversampling, the proposed method does not work well. In another approach [7], sample time errors in a two-channel ADC are detected and corrected digitally using a FIR filter. Sample-time errors are corrected by adaptively adjusting the coefficients of the FIR filter as a function of the detector signal. However, the hardware overhead increases as the required accuracy becomes higher. Also, the published technique works for only two channels.

In this paper, techniques to adaptively correct the errors caused by gain, offset, and sample-time mismatches are presented. A front-rank S/H is not required, which allows many ADC channels to be time interleaved. The offset and gain correction circuits are relatively simple. The correction of sample-time errors can be performed using a modified version of the equalizer, which is an existing block in most digital communication receivers. Hence, the offset, gain, and sample-time mismatches can be corrected with low hardware overhead.

2. Digital Calibration Algorithms

Digital adaptive correction loops based on the least mean-square (LMS) algorithm [11-12] are used to reduce the effects of offset, gain and sample-time mismatches. The loops operate to minimize the mean-squared value of the slicer error, which is $e[n]$ in Fig. 1. This paper focuses on a baseband receiver. The desired equalizer output is chosen to be a 3-level signal (e.g., a MLT3 signal as used in ethernet [13] or an equalized signal for a Viterbi detector as used in some magnetic recording channels [14-15]); however, the techniques described will work for any pulse-amplitude modulated signal (including a binary transmit signal). The transmit data is assumed to have been scrambled, so that it has zero mean and its autocorrelation function is an impulse. The receive input signal is sampled by the ADC at a rate that is somewhat faster than the baud rate, as required by the digital timing recovery block [1].

A linear model of the m^{th} channel ADC, ADC_m, with gain A_m (ideally 1), offset O_m (ideally 0), and sample-time error Δt_m (ideally 0) is shown in Fig. 3. Here, q_m is the quantization error, which is assumed to be small and uncorrelated with the input signal, which is a common assumption for a medium- to high-resolution ADC with a busy input. Here m is the channel index, while M is the total number of ADC channels, and $X_m(i)$ is the i^{th} sample output of this ADC channel. The time index for each ADC channel operating at a rate $1/MT$ is i, and n is the time index for samples at the overall sample rate of $1/T$. Therefore, sample index i in channel m corresponds to a sample index of $n = iM+m$. T will be assumed equal to the bit period in Sections 2A-2C for simplicity.

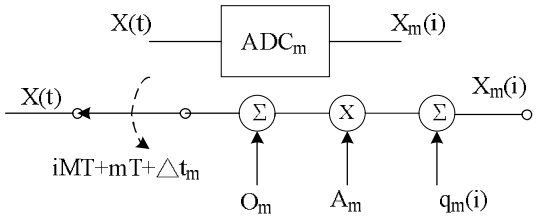

Fig. 3: Linear model of the m^{th} ADC.

A. Gain Mismatch Calibration

To correct the gain mismatches among the ADC channels, an independent AGC is used for each channel. For simplicity here, assume the ADCs are ideal except that they have different gains A_m (i.e., the ADC offsets and sample-time errors are zero). Further, assume the channel is ideal and the equalizer is not needed. The gain is adjusted using a digital multiplier operating on the ADC output samples. An adaptive feedback loop sets the gain K_m of the AGC in channel m to minimize its contribution to the mean-squared error (MSE) across the slicer. The block diagram of the AGC loop for channel m is shown in Fig. 4. Under the assumption that the transmitted data is random, the signal statistics at the slicer input are the same for all channels, and the total effective gain of each ADC channel after its AGC, the product A_mK_m, will be the same for all channels after convergence. With a perfect unity-gain communication channel, A_mK_m should equal 1. In this case, the gain K_m will converge to A_m^{-1}. In general, K_m will converge to compensate for the gain of ADC_m and any loss in the communication channel.

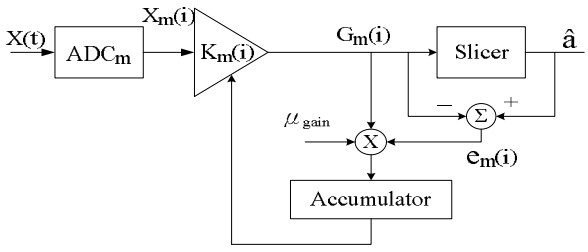

Fig. 4: Block diagram of the AGC loop for channel m.

The digital update equation for the variable gain K_m is

$$K_m(i+1) = K_m(i) + \mu_{gain} \cdot G_m(i) \cdot e_m(i) \qquad (1)$$

$$e_m(i) = \hat{a}(iM+m) - G_m(i) \qquad (2)$$

where i is a sample-time index for channel m. $e_m(i)$ is the slicer error associated with decisions $\hat{a}(iM+m)$ made every M sample periods using the output of channel m. The key to this algorithm is that the feedback loop forces the error e_m to converge to a small value with a mean of zero.

B. Offset Calibration

The goal of the offset calibration system is to cancel the DC offset in each channel. If the offset is cancelled in each channel, then offset mismatches are eliminated and all channel offsets are zero. For simplicity, assume the gain mismatch has been corrected by the loop described above, and the sample-time errors are zero. Also, assume the channel is ideal and the equalizer is not needed. The offset cancellation loop for channel m is shown in Fig. 5, and the corresponding LMS update equation is:

$$OC_m(i+1) = OC_m(i) + \mu_{offset} \cdot e_m(i) \qquad (3)$$

In steady state, OC_m has been adjusted so that the error e_m has zero mean. The error e_m is the difference between the slicer output decision and the slicer input. Since the transmitted data was scrambled, the data decisions output by each channel have zero mean; therefore, the slicer input (which is the output of the offset cancellation loop) should have zero mean. Therefore, any DC offset in channel m has been cancelled by the loop in Fig. 5.

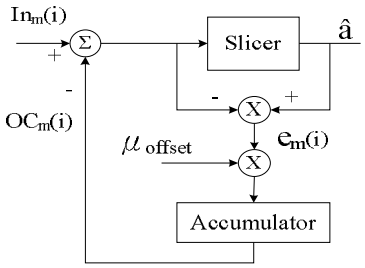

Fig. 5: Block diagram of the offset cancellation system.

C. Sample-Time Error Calibration

As shown in Fig. 3, sampling by ADC_m ideally occurs at times $iMT+mT$, but the actual samples are taken at the times $iMT+mT+\Delta t_m$, where Δt_m is the constant sample-time error of the m^{th} ADC. Since the gain and offset mismatches can be removed by the correction loops in Sections A and B, gain and offset will be ignored in this section for simplicity. Furthermore, assume for simplicity here that T is the bit period. (This assumption will be removed in the next section.)

To explain the sample-time correction concept, consider a system with a two-channel ADC (M=2). A sample-time error in a channel is equivalent to a time shift of the input $X(t)$ before a sampler with zero sample-time error in that channel. The effect of sample-time errors in the channels can be corrected by adding a filter $F_m(z)$ in each channel [16]. This is shown in Fig. 6, where each ADC is simply shown as a sampler. The corrected output of the 2 ADC channels is also combined to form a signal $Y(n)$, which is ideally the input signal $X(t)$ uniformly sampled at a rate $1/T$. The filters $F_m(z)$ would have to adapt to correct for the unknown Δt_m.

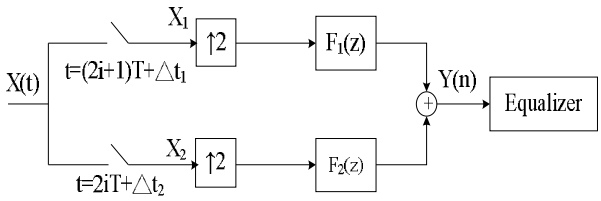

Fig. 6: Sample-time correction.

In Fig. 6, the signal $Y(n)$ is fed into an adaptive FIR equalizer that adapts its coefficients to remove any ISI. Through extensive block-diagram or equation manipulations, it can be shown that the filters $F_m(z)$ and the FIR equalizer in Fig. 6 can be replaced by an adaptive FIR equalizer that has two sets of coefficients; one set is used to produce the equalizer output samples at odd sample times and the other set used for even sample times. This leads to the system shown in Fig. 7. The two sets of equalizer coefficients can be adapted using the LMS algorithm to minimize the mean-square value of the slicer error e, which is due to both sample-time errors and ISI. Such an equalizer requires more taps than an equalizer that only removes ISI.

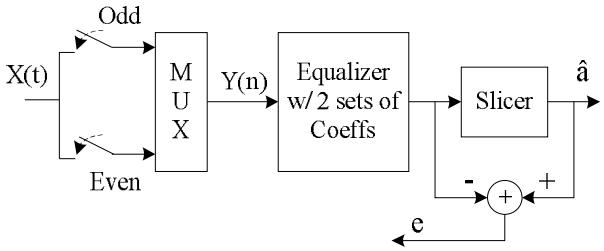

Fig. 7: Sample-time correction using two sets of equalizer coefficients.

This concept can be extended to M ADC channels, as shown in Fig. 8. In this general case, an FIR equalizer with M sets of coefficients is needed to both cancel ISI and correct for the M sample-time errors in the M ADC channels.

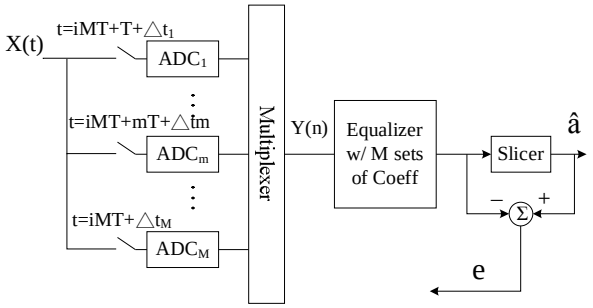

Fig. 8: Sample-time correction - general case with M ADCs.

In Fig. 8, the j^{th} equalizer coefficient of the m^{th} set of equalizer coefficients, $C_{m,j}$, can be adapted by:

$$C_{m,j}[i+1] = C_{m,j}[i] + \mu_{eq} \cdot Y(iM+m-j) \cdot e(iM+m) \quad (4)$$

Each set of coefficients is updated after every M^{th} decision. The equalizer with M sets of coefficients not only eliminates the ISI introduced by the communication channel, but also corrects the sample-time errors at the same time. Furthermore, the equalizer can be implemented by swapping coefficient sets each bit period, but only one input delay line and one set of hardware multipliers are required.

D. Calibration of All Channel Mismatches

In the previous subsections, each calibration approach was considered independently. In practice, all the correction loops must operate together to cancel the gain, offset, and sample-time errors. The correction loops are placed as shown in Fig. 9. Initially, ignore the ITR and Linear Interpolator blocks. Gain correction follows each ADC. The gain update equation is as given in (1), but $e_m(i)$ is a downsampled values of the error $e(n)$ that would be taken from the slicer in Fig. 9, since a slicer does not exist at the AGC_m output as in Fig. 4. Also, to assure proper adaptation, signal G_m is delayed by an amount equal to the latency through the equalizer and ITR before it is multiplied by e_m as in (1). The output of the gain correction blocks are combined to give samples $Y(n)$ that are fed into the equalizer, which corrects for sample-time errors and cancels ISI in the received signal. Eqn. 4 is used to adapt the M sets of equalizer coefficients, using the slicer error and the samples $Y(n)$. The output of the equalizer is then multiplexed into M separate signal paths, and the offset cancellation scheme of Fig. 5 is applied to each channel.

The offset cancellation is performed after the equalizer rather than before to avoid potential interaction between the offset cancellers and the equalizer. For example, if the offset cancellation blocks were before the equalizer, and the equalizer had a zero at DC, then the average DC offset of the channels could not be cancelled since the equalizer would effectively break the forward path of the correction loop at DC.

To avoid interaction between the equalizer and gain-correction blocks, the center equalizer coefficient in each coefficient set is fixed and equal to 1. Therefore, the equalizer coefficients cannot adjust to correct the gain errors.

Now, consider the ITR and Linear Interpolation blocks in Fig. 9. These blocks implement fully digital timing recovery in the proposed receiver. The ADC array samples $X(t)$ at a fixed frequency $1/T_S$, which is slightly higher than the symbol rate $1/T$. Each ADC_m samples at a rate of $1/MT_S$. The input to all blocks except the slicers are samples at a rate of $1/T_S$ or $1/MT_S$. A digital interpolated timing recovery (ITR) loop is used to recover the desired samples [1]. In the ITR, T-spaced ITR output samples are generated by interpolating between T_S-spaced samples using the phase information from a phase-tracking loop, which tracks the phase differences between the desired and actual samples. A training sequence is needed to estimate the initial phase offset. The ITR scheme used here is explained in detail in [1].

The error signal e' generated by the slicer is T-spaced and asynchronous to the samples in the correction loops. A T_S-spaced synchronous error signal e is needed for the LMS equations used in the correction loops [Equations (1), (3), and (4)]. Error e is found from e' by linear interpolation. Since an accurate error signal for the LMS updates is not required, a linear interpolation is sufficient. The inaccuracy of this simple linear interpolation does not affect the convergence of the correction loops.

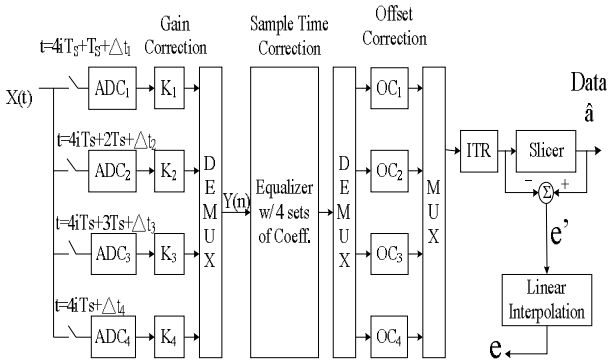

Fig. 9: System block diagram.

3. Simulation Results

Simulations were carried out on the system in Fig. 9, which has four time-interleaved ADCs. The system simulated is a read channel; the ADC input signal is the output of a Lorentzian channel model [12]

$$h(T) = \frac{T}{p \cdot PW_{50}} \frac{1}{1+\left(\frac{2t}{PW_{50}}\right)^2} \tag{6}$$

with a 50% pulse width (PW_{50}) of $3T$. The simulations use 10-bit ADCs, a 25-tap adaptive equalizer, and a 30-tap adaptive digital interpolation filter in the ITR. The signal is equalized to a 3-level signal, so that the resulting slicer input signal could be processed by a Viterbi detector [14-15] to achieve a lower bit-error rate (BER) than a simple slicer can achieve.

Table I gives the comparisons of the overall MSE before and after the calibration loops are enabled for cases with only gain errors, only offset errrors, only sample-time errors, and finally for the case where all errors exist and all calibration loops are operated.

Type of Error	MSE	
	Calibration Off	Calibration On
±10% Gain	–22.9 dB	–44.2 dB
±10% Offset	–33.1 dB	–43.3 dB
±5% Sample-Time	–22.8 dB	–43.0 dB
All errors present	–8.9 dB	–42.9 dB

Table 1: Comparison of MSEs with calibration Off and On.

Fig. 10 shows the plots of the MSE (=E[$e(n)$]^2) without and with the calibration loops running for the case with all errors. In this simulation, large channel mismatches were used to clearly show the operation of the correction loops. There are ±10% peak gain mismatches, peak DC offsets that are ±10% of the ADC full scale voltage, and ±5% peak sample-time errors introduced into the four ADC channels. Without calibration, as shown in Fig. 10(a), the channel mismatches result in a very poor MSE of -8.9 dB, which would lead to an unacceptable BER. After calibration, the MSE is a very low -42.9 dB. This MSE would fall far below the channel noise that exists in practical read channels, which is often -20 to -30 dB below the signal.

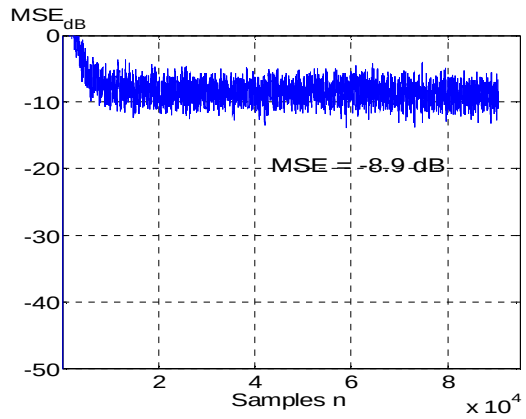

Fig. 10(a): MSE without calibration.

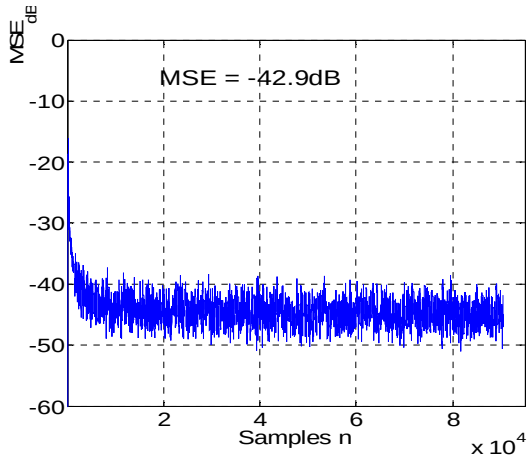

Fig. 10(b): MSE with calibration.

4. Conclusion

Parallelism can increase the effective throughput of a system. Unfortunately, time-interleaved ADC channels introduce errors such as gain, offset, and timing mismatches. As a result, the noise floor is raised and the overall SNDR is reduced. Techniques for minimizing these sources of error in a receiver have been described in this paper. All the corrections are performed by digital signal processing blocks, which can scale to take advantage of evolving CMOS technologies. The correction of sample-time errors is achieved by modifying the existing equalizer block to reduce hardware overhead. The techniques presented can be applied to many high speed communication receivers. While 4 ADCs were used in the simulations, these correction schemes will work with any number of ADC channels.

4. Acknowledgements

This work was supported by NSF Grant CCR-9901925 and by UC MICRO Grant 03-051.

References:

[1] Z. Wu, J. M. Cioffi, & K. D. Fisher, A MMSE interpolated timing recovery scheme for the magnetic recording channel, *IEEE International Conference on Communications, 3*(6), 1997, 1625-1629.

[2] C. S. G. Conroy, D. W. Cline, & P. R. Gray, An 8-b 85-MS/s parallel pipeline A/D converter in 1-µm CMOS, *IEEE Journal of Solid-State Circuits, 28*(4), 1993, 447-454.

[3] A. Petraglia & S. K. Mitra, Analysis of mismatch effects among A/D converters in a time-interleaved waveform digitizers, *IEEE Trans. Instrum. and Meas., 40*(5), 1991, 831-835.

[4] K. C. Dyer, D. Fu, S. H. Lewis, & P. J. Hurst, An analog background calibration technique for time-interleaved analog-to-digital converters, *IEEE Journal of Solid-State Circuits, 33*(12), 1998, 1912-1919.

[5] D. Fu, K. C. Dyer, S. H. Lewis, & P. J. Hurst, A digital background calibration technique for time-interleaved analog-to-digital converters, *IEEE Journal of Solid-State Circuits, 33*(12), 1998, 1904-1911.

[6] N. Kurosawa, H. Kobayashi, K. Maruyama, H. Sugawara, & K. Kobayashi, Explicit analysis of channel mismatch effects in time-interleaved ADC systems, *IEEE Transactions on Circuits and Systems I, 48*(3), 2001, 261-271.

[7] S. M. Jamal, D. Fu, S. H. Lewis, & P. J. Hurst, A 10-bit 120 Msample/s time-interleaved analog-to-digital converter with digital background calibration, *IEEE Journal of Solid-State Circuits, 37*(12), 2002, 1618-1627.

[8] H. Jin & E. K. F. Lee, A digital-background calibration technique for minimizing timing-error effects in time-interleaved ADCs, *IEEE Transactions on Circuits and Systems II, 47*(7), 2000, 603-613.

[9] K. Poulton, J. Corcoran, & T. Hornak, A 1-GHz 6-bit ADC system, *IEEE J. Solid-State Circuits, 22* (6), 1987, 926-927.

[10]K. Y. Kim, N. Kusayanagi, & A. A. Abidi, A 10-b 100-MS/s CMOS A/D Converter, *IEEE Journal of Solid-State Circuits, 32*(3), 1998, 302-311.

[11]B. Widrow & S.Stearns, *Adaptive signal processing* (Englewood Cliffs, NJ: Prentice-Hall, 1985).

[12]J. W. M. Bergmans, *Digital Baseband Transmission and Recording* (Dordrecht, The Netherlands: Kluwer Academic Publishers, 1996).

[13]C. E. Spurgeon, Ethernet: the definitive guide (O'Reilly & Associates, 2000).

[14]R. W. Wood, D. A. Petersen, Viterbi detection of class IV partial response on a magnetic recording channel, *IEEE Transactions on Communication, 34*(5), 1986, 454-461.

[15]K. Han & R. R. Spencer, Performance and implementation of adaptive partial response maximum likelihood detection, *IEEE Transactions on Magnetics*, 34(5), 1998, 3806-3815.

[16]S. M. Jamal, D. Fu, M. P. Singh, P. J. Hurst & S. H. Lewis, Calibration of sample-time error in a two-channel time-interleaved analog-to-digital converter, *IEEE Transactions on Circuit and System I, 51*(1), 2004, 130-139.