Chalmers Publication Library

(article starts on next page)

Power-Efficient Time-Domain Dispersion Compensation Using Optimized FIR Filter Implementation

Christoffer Fougstedt[1], Alireza Sheikh[2], Pontus Johannisson[3], Alexandre Graell i Amat[2], and Per Larsson-Edefors[1],

[1] *Dept. of Computer Science and Engineering,* [2] *Dept. of Signals and Systems,* [3] *Photonics Laboratory,*
Chalmers University of Technology, SE-412 96 Göteborg, Sweden
chrfou@chalmers.se

Abstract: We investigate fixed-point aspects and time-domain ASIC implementations of CD compensation. An optimized implementation gives significant power dissipation reduction for short links, with further reduction if pulse shaping is considered.
OCIS codes: (060.1660) Coherent communications; (060.4510) Optical communications

1. Introduction

DSP-based chromatic dispersion (CD) compensation is a major cause of power dissipation in modern receiver ASICs [1, 2], and reduction of the CD power dissipation is highly desirable. Previous time- and frequency-domain evaluations compare complexity in terms of number of complex multiplications [3, 4]. This fails to take word-length requirements and internal switching activity into account, which significantly affects ASIC power dissipation. Published filter design methods include direct sampling (DS) of the inverse impulse response [5] and least-squares (LS) filter design [6]. However, these studies do not analyze the fixed-point performance which is critical to ASIC implementations. In the context of CD compensation filters for ASICs, we investigate finite word-length aspects and power-efficient implementation of time-domain filters that handle low-to-moderate amounts of dispersion. We implement parallelized CD compensation filters in VHDL, perform synthesis to hardware and power dissipation simulations, and show that our optimized filter design [7] can give significant power reductions.

2. Filter evaluation

A MATLAB-based coherent fiber-optic system model is used to evaluate CD compensation filter implementations, assuming two samples-per-symbol, polarization-multiplexed-QPSK (PM-QPSK) or PM-16-QAM modulation at 28 Gbaud, a wavelength of λ = 1550 nm, a dispersion parameter of D = 16 ps/(nm km), and root-raised-cosine (RRC) pulses with a roll-off factor of 0.25. Since CD compensation is in focus, no other channel impairments are modeled. Quantization of the input signal in the A/D-converter introduces resolution-dependent noise, resulting in a power penalty in the system. We investigated the impact of quantization noise by simulating a system with dispersion, A/D-quantization, and perfect CD compensation. Using linear-step quantization and taking the Gaussian-like distribution of the input signal into account when setting the maximum quantization level, we obtained a 0.1 dB power penalty when using 5-bit and 6-bit quantization for PM-QPSK and PM-16-QAM, respectively.

A CD compensation filter can be designed by direct sampling (DS) of the inverse of the continuous frequency response of the dispersion [5]. The possible filter length is limited by aliasing, as the frequency response is not band-limited before sampling [5]. Our simulations confirm that this design approach has a significant power penalty when compensating for low-to-moderate amounts of dispersion (<320 km fiber) [8]. Another approach is to design the filter by minimizing the deviation from the ideal response using the least-squares (LS) method [6]. Full-band LS (LS-FB) gives less penalty than DS and, according to our simulations, can be used to compensate for low amounts of dispersion by extending the filter length beyond the aliasing limit of the DS filter. The LS filter can also be designed taking the band-limited spectrum of the signal into account [6], and minimizing the deviation from the ideal filter within this band. An adjustment parameter ε was used to keep the LS problem well conditioned [6]. The quantization of filter coefficients, however, changes the frequency response of the CD compensation filter, which introduces a penalty. Higher resolution gives less penalty but increases filter power dissipation.

The filters evaluated in Sec. 4 are designed to achieve a power penalty below 0.25 dB, including the penalty caused by A/D-conversion. Our simulations show that in the case of the band-limited LS filter and short word-length coefficients, the performance improves significantly for $\varepsilon = 10^{-3}$, compared to the suggested $\varepsilon = 10^{-14}$ [6]. Using a stringent approach, we have introduced a constraint on the out-of-band gain [7]: This LS filter design with constrained optimization (LS-CO) allows a tradeoff between in-band filter error and fixed-point penalty caused by the out-of-band frequency response and allows an optimized FIR filter implementation.
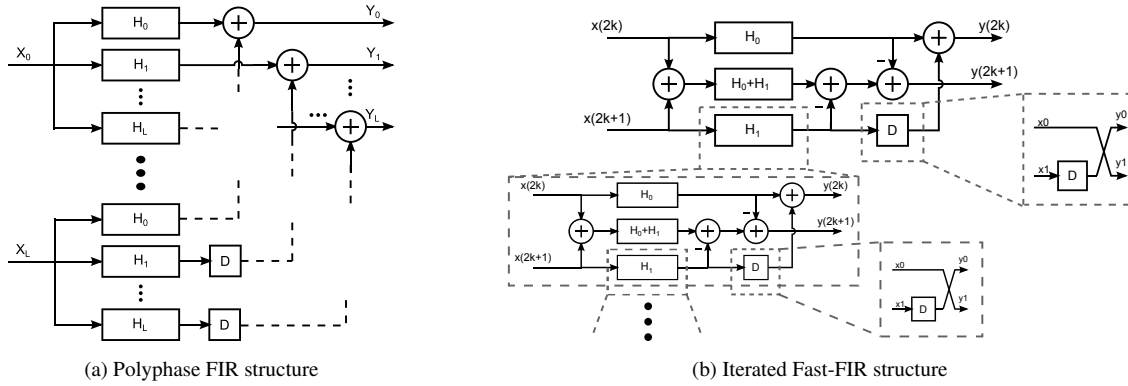
(a) Polyphase FIR structure   (b) Iterated Fast-FIR structure

Fig. 1: Block diagrams illustrating the polyphase implementation and the iterated application of a Fast-FIR algorithm.

3. FIR implementation for ASICs

At a throughput of 56 Gsample/s, CMOS-based DSP circuits need to be extensively parallelized. The implementations in this work consistently use 64-way parallelization, corresponding to a clock rate of 875 MHz.

3.1. Parallel polyphase FIR

Polyphase decomposition can be applied in order to decompose an FIR filter into a block-processing parallel structure [9]. An L-parallel FIR filter with N taps can be created by decomposing the impulse response into L subfilters, each consisting of every L-th tap. Each parallel lane consists of L filters of length N/L and the total complexity is NL. At constant clock rate, complexity increases linearly with increased throughput. The filter response is zero-padded when the tap count is not an integer multiple of the parallelization factor, and the zero-filters and coefficients are removed. Unnecessary operations are pruned away when synthesizing the filters. The general principle of the parallel polyphase filter is illustrated in Fig. 1a.

3.2. Fast-FIR

Fast-FIR filters exploit common subexpressions in the parallelized filter to reduce the number of multiplications [9]. Common expressions in the subfilters are merged and a pre-processing adder tree is introduced. Fast-FIR reduces the number of complex multiplications at the expense of an increase in number of additions and higher internal word lengths that are needed to prevent overflow. The pre-processing adder tree causes the required subfilter input word length to increase depending on which expression the filter implements, leading to an increase in power dissipation for each multiplication. A complexity analysis based on comparing the number of multiplications can therefore be misguiding. Here, an iterated application of a two-parallel Fast-FIR algorithm is used to generate 64-way parallel filters. The algorithm is applied along with proper unfolding of the delay elements. The iterated application and delay unfolding is illustrated in Fig. 1b.

3.3. Hardware implementation

Due to the complex structure of the parallelized filters, we created a MATLAB-based tool for generating VHDL descriptions from filter structure specifications. The tool utilizes pipelining of the adder structures to reduce both critical path length and adder tree switching activity. The implemented filters are synthesized using Synopsys Design Compiler and a commercially available 65 nm general-purpose standard-V_T library, characterized at 1.1 V, nominal corner, and 25°C. Retiming is used to optimize register placement of the inferred pipelining registers.

As a reference, Overlap-Save FFT/IFFT configurations employing 128-point transforms and 64-point overlap, giving a 64-way parallel operation, are implemented to support up to 64 taps. The radix-2 transform blocks are generated using Spiral [10] with an internal word length of 13 and 14 bits for QPSK and 16-QAM, respectively, and internal scaling to prevent overflow. Since the two FFT/IFFT reference implementations 1) give 0.3-0.4 dB penalty and 2) lack tap multiplications, the reference power dissipation values are lower bounds for frequency-domain implementations.

All in all, LS-FB filters, LS-CO filters and reference FFT/IFFT circuits for time- and frequency-domain translation were implemented and synthesized for the comparison presented in Sec. 4.
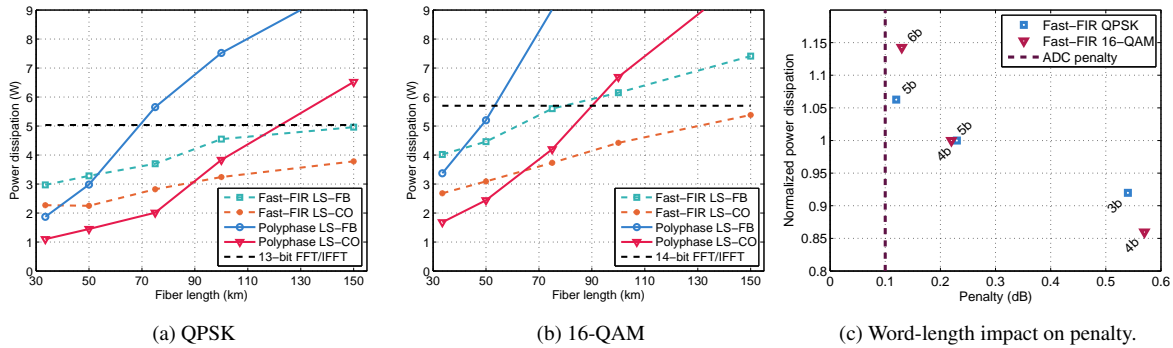
Fig. 2: Power dissipation for different fiber lengths, and impact of word length on power dissipation and penalty.

4. Results

LS-FB filters required 5- and 6-bit coefficients for PM-QPSK and PM-16-QAM, respectively, to fulfil the penalty requirement of <0.25 dB. Using LS-CO filters, the penalty requirement is met with 4- and 5-bit coefficients, respectively. Netlists of all synthesized filters were simulated and verified using test vectors with dispersion. The resulting switching activities were annotated to the netlists and power estimation was performed using Design Compiler.

In comparison to LS-FB, an LS-CO implementation gives significant hardware savings since shorter coefficients and fewer taps are required for <0.25 dB of penalty. Fast-FIR scales significantly better with the number of coefficients compared to polyphase, although the overhead due to the pre- and post-processing trees and increased word-length requirement results in a higher power dissipation for short fibers. The resulting power dissipation for equalization of one PM-QPSK and one PM-16-QAM channel is shown in Fig. 2a and 2b, respectively. An LS-CO filter gives an average power dissipation reduction of approximately 25% and 30% for PM-QPSK and PM-16-QAM, respectively, when employing Fast-FIR. For the polyphase structure, a band-limited LS-CO filter can reduce power dissipation by 50% compared to a LS-FB filter. The power dissipation reduction is higher than expected when only considering the reduction in number of taps [7], as the coefficient word length can be reduced by one bit. The time-domain implementation dissipates less power than the reference FFT/IFFT-implementation up to approximately 150 km for PM-16-QAM, and dissipates 1.2 W less power at 150 km for PM-QPSK.

An LS-CO Fast-FIR filter for 100 km of fiber was synthesized with 3–5-bit and 4–6-bit coefficients for PM-QPSK and PM-16-QAM, respectively. As shown in Fig. 2c, the 4-bit and 5-bit coefficients provide a good tradeoff between power dissipation and power penalty for PM-QPSK and PM-16-QAM, respectively.

5. Conclusion

We have shown that time-domain filters are power efficient for low dispersion applications. A Fast-FIR-based filter increases the span where it is viable to perform CD compensation in time domain. Furthermore, we have shown that optimizing the filter for the pulse-shaped signal's spectrum is a very efficient implementation approach; since they are less sensitive to quantization, these filters need fewer taps and shorter word length [7].

References

[1] D. Crivelli *et al.*, "Architecture of a single-chip 50 Gb/s DP-QPSK/BPSK transceiver with electronic dispersion compensation for coherent optical channels," IEEE Trans. on Circuits Syst. I, Reg. Papers **61** (2014).

[2] B. Pillai *et al.*, "End-to-end energy modeling and analysis of long-haul coherent transmission systems," J. Lightw. Technol. **32** (2014).

[3] M. Kuschnerov *et al.*, "DSP for Coherent Single-Carrier Receivers," J. Lightw. Technol. **27** (2009).

[4] J. Geyer *et al.*, "Efficient frequency domain chromatic dispersion compensation in a coherent Polmux QPSK-receiver," in "Opt. Fiber Commun. Conf. (OFC)," (2010).

[5] S. J. Savory, "Digital filters for coherent optical receivers," Opt. Express **16** (2008).

[6] A. Eghbali *et al.*, "Optimal least-squares FIR digital filters for compensation of chromatic dispersion in digital coherent optical receivers," J. Lightw. Technol. (2014).

[7] A. Sheikh *et al.*, "Dispersion compensating filter design optimized for robustness and power-efficiency," in "Signal Proc. in Photonic Comm. (SPPCom)," (2015).

[8] T. Xu *et al.*, "Chromatic dispersion compensation in coherent transmission system using digital filters," Opt. Express **18** (2010).

[9] K. K. Parhi, *VLSI Digital Signal Processing* (Wiley-Interscience, 1999).

[10] G. Nordin *et al.*, "Automatic generation of customized discrete Fourier transform IPs," in "Design Automation Conf. (DAC)," (2005), pp. 471–474.