Synthesizing Optimal Filters for Crosstalk-cancellation for High-Speed Buses

Jihong Ren and Mark Greenstreet
Dept. of Computer Science, University of British Columbia
2366 Main Mall, Vancouver, BC, V6T 1Z4
{jihong,mrg}@cs.ubc.ca

ABSTRACT

We present practical algorithms for the synthesis of crosstalk cancelling equalizing filters. We examine designs optimized for the traditional l_2 metric and introduce an approach based on the l_∞ metric. We compare the two approaches for realistic buses with tight wire spacings. We show bandwidth improvements of up to a factor of 2 using crosstalk cancellation when compared with no filtering or independent preemphasis for each wire. Using l_∞ optimization, we achieve roughly 50% better performance than the l_2 methods. We are aware of only one other published description of crosstalk cancellation for high-performance buses [9]. We believe that our work is the first to show the advantages of l_∞ optimization and to consider crosstalk cancellation for more than just nearest neighbours for high-speed buses.

Categories and Subject Descriptors

B.4.3 [**Input/Output and Data Communications**]: Interconnections (Subsystems); B.4.4 [**Input/Output and Data Communications**]: Performance Analysis and Design Aids

General Terms

Design, performance.

Keywords

Buses, crosstalk, equalizing filters, optimal synthesis.

1. INTRODUCTION

With advances in integrated circuit fabrication technology the speed and integration levels of ICs have grown exponentially creating a corresponding demand for high-bandwidth for off-chip buses. Such bandwidth is especially critical for high-performance memory systems and inter-processor communication in shared memory multiprocessors. To meet this demand, designers are relying increasingly on equalizing filters and other on-chip signal processing techniques to maximize the utilization of off-chip interconnect.

High bit rates exacerbate the problems of crosstalk between wires in high-speed buses. With narrow wires and small line spacing, the coupling inductance and capacitance between adjacent lines approach the levels of self-inductance and capacitance. Short signal rise and fall times exacerbate coupling effects, making crosstalk a primary concern for present and future high-speed high-density circuit design. Traditional design methods reduce crosstalk by carefully controlling line geometry and arranging circuits to decrease the coupled line length. Some designs use differential signaling to reduce crosstalk at a cost of greater pin-count and doubling the size of the buses on the printed circuit board (PCB). While these methods might reduce crosstalk, they do not eliminate it. High performance PCB designs often require many revisions to produce a working design.

This paper explores the effectiveness of equalizing filters in crosstalk cancellation for high-bandwidth, digital communication. In practice, filter design is constrained by limitations of circuit speed, power consumption and the complexity of the filter. We show that practical filters can improve the bandwidths of PCB buses by substantial factors.

Figure 1 shows the structure of a typical channel with a pre-equalization filter for crosstalk cancellation. In this transmission system, a filter is assigned to each wire of the bus. Each filter takes the input signals on a wire and its adjacent wires as its inputs, and outputs a predistorted signal onto the wire. In later examples, we consider filters that use inputs for several neighbours in each direction. For a bus of width k_{bus}, the filter system can be viewed as a $k_{\mathsf{bus}} \times k_{\mathsf{bus}}$ network. Crosstalk is eliminated if the filter network is designed in a way that the concatenation of the filter network and the bus has an impulse response in the form of a diagonal matrix. This work is the first study that we have seen of the effectiveness of equalizing filters for cancelling crosstalk for high-speed buses. Preliminary results have been published in [6]. We show that crosstalk cancellation can double the bandwidth achievable on buses with tight wire spacings. We show that the l_∞ norm is a more appropriate measure for signal integrity in digital designs than the more commonly used l_2 norm, achieving roughly 50% improvements in bandwidth. We present a practical method for synthesizing optimal filters based on linear programming and present results from an implementation of our synthesis procedure.

Section 2 gives a brief survey of previous work in equalizing filter design. We formalize our linear models of the bus

DAC 2003 June 2–6,2003,Anaheim,California,USA.

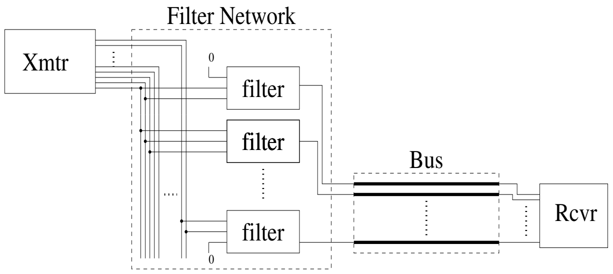


Figure 1: A typical channel with pre-equalizing filters for crosstalk cancellation.

and filter in section 3. Section 4 presents formulations for the l_2 and l_∞ synthesis problems based respectively on least-squares and linear programming optimizations. In section 5, we present efficiency and robustness issues related to the implementation of our l_∞ method, and section 6 presents the performance of filters designed with our methods providing a comparison of the l_2 and l_∞ approaches. Section 7 describes a hardware implementation framework suitable for filters designed in this paper.

2. RELATED WORK

Equalization has been used effectively to compensate for resistive effects of transmission lines [2, 3, 7]. With this technique and carefully chosen signaling methods, multi-Gb/s serial links have been built. Of these, the work most closely resembling ours is [7] , which includes a model of the distortions arising from an interleaved DAC as a multiple-input, multiple-output (MIMO) response function. Our work generalizes this by looking at buses that are naturally modeled as MIMO channels when crosstalk is considered.

To the best of our knowledge, [9] is the only design where an equalizing filter is used for crosstalk cancellation in the context of high speed buses. That paper describes a proprietary design and gives few details of how the filters are derived. This paper presents a novel method for designing the crosstalk cancelling filter and provides an evaluation comparing it with other design techniques.

For digital transmission, the worst-case eye height is a natural measure of signal integrity (see section 4.1). If a signal satisfies an eye-specification, then it should be acquired successfully by the receiver. If the signal passes through the interior of the eye, then an error may occur. All of the methods described above use an l_2 metric to measure signal integrity. While this leads to filters that minimize the average power of the error signal, they do not minimize the worst-case error. Our design method minimizes worst-case error, achieves greater eye height than filters designed with least-squares techniques and guarantees worst-case performance. To the best of our knowledge, our work is the first study of equalization filter design for high-speed digital transmission using an l_∞ objective function.

3. LINEAR FORMULATION

Buses have crosstalk, dispersive losses, reflections, and other effects that corrupt digital integrity, but all of these phenomena are linear processes. We model the bus as a time-invariant, linear system specified by its impulse response. This impulse response can be derived from the geometry of the interconnect using a field solver to extract lumped and distributed resistances, inductances, and capacitances. Alternatively, on-chip analog-to-digital converters can be used to estimate the impulse response based on a training sequence and/or on-going measurements during operation [1].

An effective filter requires a sample rate greater than the symbol rate because of the high frequency response of the bus. Implementing a filter where every filter output depends on the values received on every input wire requires large amounts of hardware and introduces substantial latency, especially for large buses. However, the largest contributions to crosstalk typically come from nearby wires; thus, we consider filters where each output is computed from the input value for the wire itself and each of its $k_{\mathsf{fir}} - 1$ closest neighbours in both directions. We write that the filter is $n_{\mathsf{fir}} \times k_{\mathsf{fir}}, r$ if the filter has n_{fir} taps, takes inputs from the wire itself and its $k_{\mathsf{fir}} - 1$ closest neighbours, and the filter sample rate is r times the symbol rate to compensate for the high frequency losses of the bus. For the sake of simplicity, we make the following assumptions. First, we note that the typical bus impulse response functions are extremely small for large values of time t, and we approximate bus response by a window of duration n_{bus}. Next, we consider buses where all wires have the same width, and all adjacent pairs of wires have the same spacing. This ignores effects arising near the edges of the bus and effectively considers buses with cylindrical topology. These assumptions allow us to simplify the models. Likewise, we assume that the the filter has the same symmetrical structure. However, the filter design methods presented in this paper can be adapted for more general cases.

4. FILTER SYNTHESIS

4.1 Measurements of signal integrity

The effects of distortion and noise are often depicted using eye diagrams. During each sampling interval, a binary signal should be either distinctly high or distinctly low. This allows the receiver to unambiguously determine the value of the bit that was transmitted. The signal can change between sample intervals. We also restrict how high (or low) the signal may go, otherwise, with scaling any eye opening can be made arbitrarily large. The eye height, *height*, is defined as

$$\mathit{height} \quad = \quad \min(h_{under}, 2 \cdot \mathit{target} - h_{over}) \qquad (1)$$

where h_{under} is the amount of undershoot at sample time, h_{over} is the amount of overshoot at the sample time and *target* is the target signal level. Eye width is defined as the time period that the signal received when a high signal transmitted is distinct from the signal received for a low value. The eye height and width are often used as an indication of signal integrity. In this paper, the goal of equalizing filter synthesis is to maximize the eye height for all input sequences.

4.2 Synthesis for l_2 optimality

With an ideal channel, the received signal would be simply a delayed version of the data from the sender. We write δ_0 to denote this delay. We use the peak of the Frobenius norm of the bus impulse response as the value of δ_0.

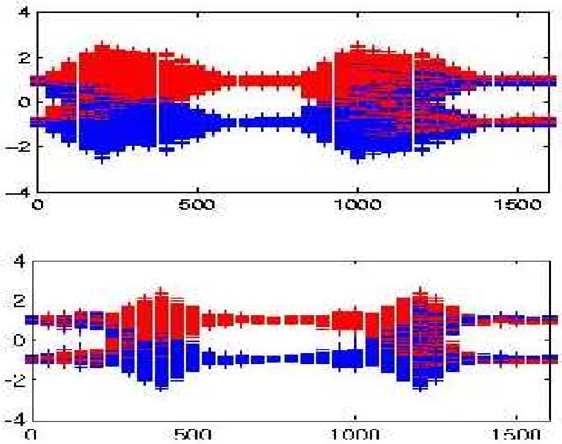


Figure 2: Eye diagram with (lower) and without (upper) a l_2 optimal equalizing filter (800ps/**bit**)

r	6.6 Ω/m
l	2.96e-7 H/m
c	1.69e-10 F/m
$l_{mr}(w)$	$0.528/(1.553 + (w-1)^{1.002})$
c_{mr}	0.022

Table 1: A Bus Model. $l_{mr}(w)$ is the relative mutual inductance between wires that are w wires away. These parameters correspond to microstrip lines 34.5 μm thick (1 oz copper), 75 μm wide with 225 μm separation between lines, running above a ground plane with a dielectric thickness of 100 μm, and a dielectric constant of ϵ_r 4.5.

A commonly used measure of the difference between the desired signal and the actual signal is the mean-square error [2, 4, 7]. We consider the case where the bits in the data stream are independent, evenly weighted, Bernoulli random variables. In this case, the cross-correlations are all zero, and we can exploit the symmetry of the bus and filter to compute the least-square error based on the the contributions from a single bit. Let $y(i, j)$ be the response of the filter and bus on wire i at time j when the input on wire 0 is one at bit-time 0 (i.e. tap times $0 \ldots r-1$), and 0 at all other times, and all other wires are 0 at all times. Note that y is a linear function of the filter coefficients. Likewise, let $y_0(i, j)$ be the response of an ideal channel with delay δ_0 to the same input:

$$y_0(i,j) \quad = \quad \begin{cases} 1, & \text{if } i = 0 \text{ and } \delta_0 \le j < \delta_0 + r \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

The least-squares optimization (LSQ) problem is

$$f_{\text{LSQ}} \quad = \quad \arg \min_{f \in \mathbb{R}^{n_{\mathsf{fir}} k_{\mathsf{fir}}}} \|y - y_0\| \tag{3}$$

As an example, figure 2 shows how an $8 \times 4, 4$ filter designed with the least-squares method improves eye height. Table 1 shows the parameters for the bus used in the examples throughout this paper. This bus is terminated at both ends with the characteristic impedance of the lines. The eye-diagram shows the received signal from the worst-case input (see section 4.3) concatenated 100 times with random input sequences. This example shows how the equalizing filter significantly improves signal integrity. We examine the performance of our filters more thoroughly in section 6.

4.3 Synthesis for l_∞ Optimality

To ensure reliable communication of digital data, our objective is to maximize the eye height for the *worst-case* input pattern. An l_2 optimal solution may sacrifice *worst-case* error to reduce *average-case* error. Accordingly, better filters are possible by using the worst-case response as the objective function. This corresponds to an objective function using an the l_∞ norm. In this section, we show how this the resulting equalization filter synthesis problem can be formulated as a linear program.

Because the channel is a linear system, the output signal on wire i for the current bit is simply a summation of the effect on wire i at the current bit from:

- the input signal on wire i for the current bit, which is the signal expected to come through if there is no disturbance;
- the input signal on wire i at other bit times;
- the input signals on other wires for the current bit and other bit times.

To formulate the LP problem, we need to know the undisturbed output at the sampling point and the largest total disturbances at the sampling point. We then want to drive the undisturbed output value to the target minimizing the total disturbance. Without loss of generality, we optimize the response on wire 0 with the assumption that the filter is symmetric for all wires.

By the symmetry of the bus, the coupling of an input on wire i at time t to output wire 0 at time $t + \delta$ is the same as the coupling from wire 0 at time t to wire i at time $t+\delta$. Thus, we can compute all couplings to wire 0 based on the response to a 1 on wire 0 at bit-time 0 and zero values on all other wires and at all other times. As defined in the previous section, this response is y.

We optimize the eye height over a sampling window. This window starts after a delay of δ_0, the peak of the Frobenius norm of the bus impulse response and has a width of n_{tap} tap times. This corresponds to specifying an eye mask with width n_{tap} so that although we are optimizing eye height, our optimization procedure also guarantees worst-case eye width. Moreover, we found that for wider filters, optimizing only 1 tap often result in large overshoots because the optimizer only considers error at that single instant. The undisturbed output is given by $Y_1(\delta_0, 0) \ldots Y_1(\delta_0 + n_{\mathsf{tap}} - 1, 0)$. The disturbances at time $\delta_0 + m$ are given by $Y_1(\delta_0 + m + i\,r, j)$ where i and j are integers and both are not zero. We note that if a disturbance in response to an input with a value of $+1$ is positive (resp. negative), then the disturbance in response to the same input with a value of -1 will be negative (resp. positive) with the same magnitude. Thus, the worst-case disturbance is given by the sum of the absolute values of the disturbances arising from inputs with value $+1$. The linear programming formulation of the filter synthesis

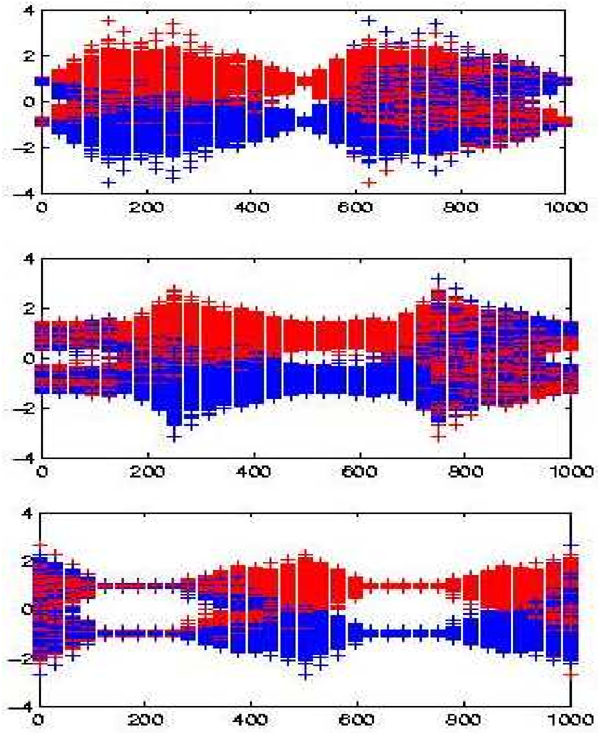


Figure 3: Eye diagram without filter (upper), with a 4*8 l_2 optimal equalizing filter (middle) and with a 4*8 l_∞ optimal equalizing filter (upper) (500ps/bit).

problem is:

$$f_{\mathrm{LP}} = \arg \min_{f \in \mathbb{R}^{n_{\mathsf{fir}} k_{\mathsf{fir}}}} \sum_{(i \neq 0) \vee (j < \delta_0) \vee (j \geq \delta_0 + r)} |y(i,j)| + \sum_{j=0}^{n_{tap}-1} |y(0, \delta_0 + j) - 1| \tag{4}$$

With this formulation, the value of the objective function at optimality tells us the worst-case eye height. The set of filter coefficients at the optimal vertex is the set of filter coefficients that generates least total disturbance and hence the best worst-case performance in the filter coefficient space. Thus, we can make signal integrity guarantees that aren't possible with the l_2 optimized filters. Furthermore, because we are optimizing for the best possible worst-case performance, we achieve better filters than those synthesized for average-case metrics (see section 6). As an example, figure 3 compares eye diagrams for a bus with no equalization, and with $4 \times 8, 4$ equalizing filters synthesized by the l_2 and l_∞ design methods. The bus parameters are those given in table 1. We examine the performance of our filters more thoroughly in section 6.

5. ALGORITHM IMPLEMENTATION

We implemented both the l_2 and l_∞ design methods directly in Matlab according to their formulation presented in section 4. In the LP problem formulation, disturbances from all other wires are considered even though inputs on wires far away cause very small disturbances. This wide range of responses is reflected in a wide range of magnitudes of responses to the various filter coefficients. As Matlab's LP solver, linprog() approaches the optimal vertex, the linear system becomes sufficiently ill-conditioned to prevent successful optimization. To overcome this, we implemented Mehrotra's interior-point, predictor-corrector algorithm [5] along with a simple model reduction technique. When the linear system for the interior point method becomes highly ill-conditioned, it is usually possible to identify many non-critical constraints. In particular, the linear program has two constraints for each absolute value term from equation 4. At optimality, exactly one constraint from each of these pairs will be tight. Near an optimal vertex, the tight constraint for such a pair can be identified because it has a much higher marginal cost than that of the other constraint. We replace the tight inequality constraint with an equality and eliminate the lower cost inequality. In practice, several rounds of model reduction may be performed before reaching the optimal vertex. Upon completion, we check complementary slackness to verify the correctness of our reductions. The simple LP solver we implemented using this technique succeeds for every filter design problem that we have attempted.

The size of the LP problem formulated grows with the bus width k_{bus}, the length of the bus impulse response n_{bus}, the size of the filter $n_{\mathsf{fir}} \times k_{\mathsf{fir}}, r$:

$$\begin{array}{ll} \text{number of variables} & = \\ \quad k_{\mathsf{fir}} n_{\mathsf{fir}} + n_{\mathsf{tap}} \lfloor \frac{r + n_{\mathsf{fir}} + n_{\mathsf{bus}}}{r} \rfloor k_{\mathsf{bus}} & \\ \text{number of constraints} & = \\ \quad 2 n_{\mathsf{tap}} \lfloor \frac{r + n_{\mathsf{fir}} + n_{\mathsf{bus}}}{r} \rfloor k_{\mathsf{bus}} & \end{array} \tag{5}$$

Note that the length of the impulse response of the bus, n_{bus}, and that of the filter, n_{fir}, both increase linearly with the oversampling rate r. For filter design for a 32-bit wide, 5cm long bus at 500ps bit time, LP problems formulated for filter sizes from $4 \times 1, 4$ to $12 \times 12, 4$ have approximately 500 to 800 variables and 1000 to 1300 inequality constraints, assuming $n_{\mathsf{tap}} = 2$. The constraints arise from the absolute value constructions in equation 4. Filter design for different filter sizes takes from 32s to 250s on a Linux PC with Intel P4 1.5GHz CPU and using at most 124 MB of memory.

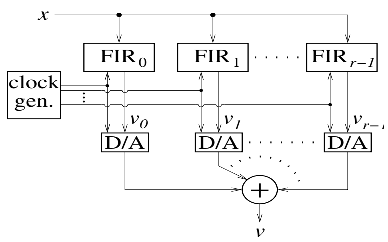
6. COMPARISON OF L_2 OPTIMAL FILTERS AND L_∞ OPTIMAL FILTERS

To test our filters, we used a variation of our l_∞ design method to find the worst-case input for each filter. Given the filter coefficients, we can determine the disturbance caused by each input wire and bit-time. The worst-case disturbance occurs when all of the individual disturbances have the same sign. This is easily achieved by setting the sign of each input bit appropriately. In this section, all testing results are obtained with input sequences that are concatenations of the worst-case input sequence and pseudo-random input sequences, unless otherwise indicated.

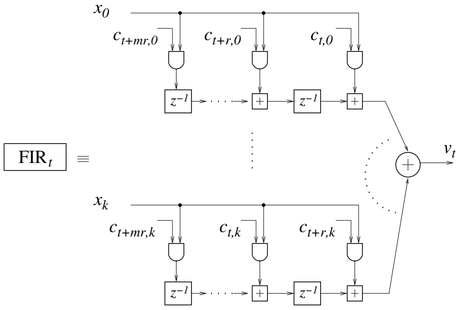
To simplify design and yet achieve reasonable crosstalk cancellation, an important question is what's an appropriate size for the filter. In general, larger filters achieve better crosstalk cancellation at an increased cost for the hardware implementation.

Taps	*Width*	*5 cm bus* *Min bit time* (ps)		*20 cm bus* *Min bit time* (ps)
		l_∞	l_2	l_∞
4	1	681	687	2718
4	3	550	550	2174
4	5	405	512	1554
4	8	349	525	1400
4	12	349	550	1362
6	1	689	687	2718
6	3	549	568	2174
6	5	389	456	1554
6	8	342	525	1362
6	12	342	550	1362
8	1	680	687	2718
8	3	549	550	2000
8	5	389	475	1554
8	8	351	525	1362
8	12	342	550	1362
12	1	680	687	2718
12	3	549	550	1812
12	5	389	456	1549
12	8	342	525	1365
12	12	342	525	1365
no filter		687		2722

Table 2: Performance of equalizing filters with different sizes for a 32-bit bus. The lower the minimum bit time, the better the performance.



(a) FIR filter with interleaved DAC



(b) Filter for neighbour Pair

Figure 4: An implementation of an equalizing filter

Table 2 shows simulation results with the filter length and width varied, assuming 4 taps per bit. Note that a filter of width 1 corresponds to independent pre-emphasis on each wire. To evaluate the performance of an equalizing filter, the maximum operating frequency (minimum bit time) at which the height of the eye is around 50% and eye width is over 25% is used. The lower the minimum bit time, the better the performance. In these simulations, design parameters are the same as in table 1. Table 2 shows that:

- Both l_2 optimal filters and l_∞ optimal filters effectively improve the maximum bit rate of the bus.
- l_∞ optimal filters have better performance than l_2 optimal filters for nearly every configuration considered.

Note that width = 1 is separate pre-emphasis for each line. With width = 1, l_2 optimal filters and l_∞ optimal filters have similar performance. The performance of the system without filter (width = 0) and systems with pre-emphasis filters (width = 1) are similar (681ps vs. 687ps). With crosstalk cancellation (width > 1), the performance of the bus is greatly improved with nearly twice the bit-rate of a bus with separate pre-emphasis for each line. This shows that for this bus with relatively tight line-spacing, crosstalk is the dominant signal integrity issue, and equalizing filters are an effective method for reducing crosstalk disturbances. For filter widths greater than 3, the l_∞ optimal filters are significantly better than their l_2 counterparts. This suggests that while l_2 design and l_∞ design are comparably well suited for designing filters for independent pre-emphasis, the l_∞ design method is much better suited for crosstalk cancellation. Furthermore, notice that filter width has lower return margin as it goes up. This is because wires far away create little crosstalk, and taking them into account brings little benefit. A $4 \times 8, 4$ l_∞ optimal filter is a good choice in terms of performance and cost.

Table 2 also shows simulation results with the filter length and width varied for buses 20 cm long. Long buses suffer more from its series resistance and also larger coupling effect thus they operate at much lower speed. Nevertheless, observations similar to those that we made for buses 5cm long also can be made here. For example, $4 \times 8, 4$ l_∞ optimal equalizing filters improve the bandwidth with almost twice the bit-rate of a 20cm long bus with or without separate pre-emphasis for each line.

7. HARDWARE IMPLEMENTATION

Once the size and the oversample rate of the filter are decided, filter implementation can be typical FIR designs. The filter synthesis problem amounts to determining the values of the filter coefficients. We do this using the optimization methods described in the previous sections. These coefficients can then be loaded into the FIR hardware using a scan-chain.

In this section, we present a look-up based implementation of our filters. Notice that in previous sections, for simplicity of presentations of the algorithm, the assumption of symmetric buses and filters was made. Generalizing the case where each wire has a separate model is straightforward. In particular, this would allow us to compensate for the parasitic inductances and capacitances of bonding wires, lead frames, solder bumps, etc. This is clearly an important area for future work. For this reason, here, we present a filter implementation that doesn't assume the symmetric filters.

As shown in figure 1, the filter can be implemented as a separate filter for each wire that receives inputs for the data to be transmitted on the wire itself and a few of its nearest neighbours. Figure 4 shows an hardware implementation framework suitable for the filters described in this paper. It uses an interleaved DAC as described in [8]. The clock generator produces phases for enabling each DAC. A current summing circuit combines the DAC outputs to produce the filter output, v. For simplicity, we show a design where the interleaving factor for the DAC is the same as the oversampling rate of the filter, r. By using a separate filter for each DAC, the DACs are incorporated into the channel, and filter coefficients can be adjusted to compensate for variations between the DACs.

Because the filter is linear, we can compute the contributions to the output arising from each input separately. The output of a FIR filter for a single DAC channel includes a delay, z^{-t} to align its output with the clock phase of its DAC. Figure 4(b) shows an implementation of a FIR filter for a single DAC channel. The filter coefficients, $c_{t,j}$ correspond to the contributions of an input on wire j after a delay of t tap times:

$$c_{t,j} = \sum_{g=0}^{\min(t,r)} F(t-g,j) \tag{6}$$

The values of x are either 0 or 1, thus the multiplications are simple AND gates.

The total hardware required for our filters is acceptable. For example, if we consider an oversampling rate of r and a general filter that is n_{fir}-tap long and takes k input signals, the filter has $k(n_{\mathsf{fir}}+r-1)$ output coefficients and requires $k(n_{\mathsf{fir}}+r-1)-r$ adders to compute the input for the DACs for each wire. For example, with $n_{\mathsf{fir}}=8$ and $k=9$, $r=4$, and 8-bit data paths for 8-bit DACs, our design requires 760 one-bit full-adders. Thus, filter for each output pad can be constructed with about 30000 transistors. Notice that this is a straightforward implementation without any effort being put in to reduce the number of transistors. With careful design, we believe the transistor count can be reduced further. For a chip with 100-200 million transistors and a few hundred high-speed I/Os, our filters can double the output bandwidth for a few percent of the total chip area. Furthermore, the filter's latency is very small. The design shown here requires an adder-tree of depth $1+\log_2(k)$. This is four adders for the example design. Thus, it is reasonable to estimate that the filter adds less than $1ns$ to the latency of the channel for an implementation in a 0.13μ process. The significant bandwidth advantages, the acceptable per-pad transistor count, and the low added latency demonstrate that crosstalk cancelling filters are a practical way to use on-chip computation resources to improve chip-to-chip signal integrity and bandwidth.

8. CONCLUSION

We presented a new use for equalizing filters to improve the bandwidth of off-chip buses. While equalizing filters have been used to compensate for dispersive losses for high-speed digital links [2, 3, 7], and for crosstalk cancellation for lower bandwidth telephone subscriber loop problems [4], their use for crosstalk cancellation appears to be new. Rambus has alluded to using filters that consider each wire with one neighbour on each side [9], but we haven't seen any designs for wide filters. Our work indicates that wide filters offer significant performance advantages.

Whereas least squares optimization is the predominant method for equalizing filter design, we have presented an alternative approach based on linear programming. LP based design has clear advantage that the objective function corresponds directly to the eye-height measure of signal integrity. Thus, our methods allows us to make strong claims about the worst-case performance, and our design procedure can produce the worst-case inputs as feedback for the designer.

Our l_∞ optimal filters outperform their l_2 counterparts by margins of up to 50% at no extra cost in the hardware. Our experiments indicate that wide filters designed by our l_∞ design method are very effective at cancelling crosstalk, whereas l_2 optimal filters show little advantage beyond relatively narrow designs. Thus, we expect our l_∞ design approach to dominate as crosstalk becomes a progressively more severe design issue.

Moreover, scaling trends of the VLSI technology favor this approach. Long buses cost more and support lower data rates. The cost of the bus justifies added circuitry on the chip. The lower data rate provide more time for the filtering operations. Furthermore, improvements in chip fabrication are producing smaller and faster circuits for implementing the filter while buses remain big and slow. This also contributes to the favorability of adding more sophisticated equalizing filters.

9. REFERENCES

[1] S. Crozier, D. Falconer, and S. Mahmoud. Least sum of squared errors (LSSE) channel estimation. *Radar and Signal Processing, IEE Proc. F*, 138:371–378, 1991.

[2] W. Dally and J. Poulton. Transmitter equalization for 4-GBPs signaling. *IEEE Micro*, 1:48–56, 1997.

[3] A. Fiedler, R. Mactaggart, J. Welch, and S. Krishnan. A 1.0625Gbps transceiver with 2x-oversampling and transmit signal pre-emphasis. In *Proc. of ISSCC97*, pages 238–239, 1997.

[4] M. Honig, K. Steiglitz, and B. Gopinath. Multichannel signal processing for data communications in the presence of crosstalk. *IEEE Trans. on Communications*, 38:551–558, 1990.

[5] J. Nocedal and S. Wright. *Numerical Optimization*, pages 395–417. Springer Series in Operations Research, Springer Press, 1999.

[6] J. Ren and M. Greenstreet. Equalizing filter design for crosstalk cancellation. In *IEEE Computer Society Annual Symp. on VLSI*, pages 272–274, 2003.

[7] V. Stojanovic, G. Ginis, and M. Horowitz. Transmit pre-emphasis for high-speed time-division-multiplexed serial-link transceiver. *IEEE Trans. on Communications*, 38:551–558, 2001.

[8] C. Yang, V. Stojanovic, S. Modjtahedi, M. Horowitz, and W. Ellersick. A serial-link transceiver based on 8-GSamples/s A/D and D/A converters in 0.25-μm CMOS. In *IEEE Int'l. Conf. on Communications*, pages 1934–1939, 2002.

[9] J. L. Zerbe, R. S. Chau, C. W. Werner, W. F. Stonecypher, H. J. Liaw, G. J. Yeh, T. P. Thrush, S. C. Best, and K. S. Donnelly. A 2Gb/s/pin 4-PAM parallel bus interface with transmit crosstalk cancellation, equalization and integrating receivers. In *IEEE Int'l. Solid State Circuits Conf.*, pages 430–432, 2001.