Joint Equalization and Coding for On-Chip Bus Communication

Srinivasa R. Sridhara and Naresh R. Shanbhag
Coordinated Science Laboratory
University of Illinois at Urbana Champaign
1308 W Main St. Urbana IL 61801

Ganesh Balamurugan
Circuits Research, Intel Labs
Intel Corporation
Hillsboro OR 97124

Abstract

In this paper, we propose using joint equalization and coding to improve on-chip communication speeds by signaling at rates beyond the rate governed by RC delay of the interconnect. Operating beyond the RC limit introduces inter-symbol interference (ISI). We mitigate the effects of ISI by employing equalization. The proposed equalizer employs a variable threshold inverter whose switching threshold is modified as a function of past output of the bus. We demonstrate even higher speed-ups by combining equalization with crosstalk avoidance coding. Specifically, simulation results for a 10-mm 32-bit bus in 0.13-µm CMOS technology show that 1.28× speed-up is achievable by equalization alone and 2.30× speed-up is achievable by joint equalization and coding.

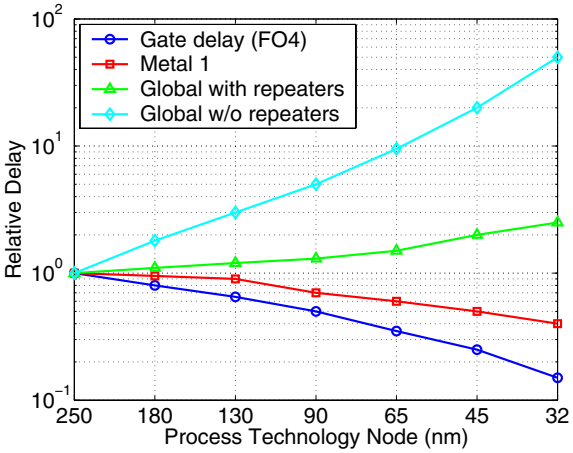


Figure 1. Gate and wiring delay versus feature size [1].

1. Introduction

Propagation delay of global buses acts as performance bottleneck in many system-on-chip and network-on-chip designs [1–5]. Figure 1 from the 2003 International Technology Roadmap of Semiconductors (ITRS) [1] shows the delay scaling trend with technology. While gate delay reduces with scaling, global wire delay increases. Repeater insertion mitigates the delay to some extent at the cost of additional power dissipation and chip area [1], but the gap between logic and interconnect delay is still expected to increase with scaling.

Coding, and in particular, cross-talk avoidance codes (CAC) [6–10], has emerged as an attractive technique for reducing delay *and* power. CAC reduces delay and power by eliminating worst-case coupling transitions in a bus. For example, the worst-case delay of a wire in a bus can be modeled as $(1+4\lambda)\tau_0$ [6], where λ is the ratio of coupling capacitance to bulk capacitance and τ_0 is the delay of a crosstalk-free wire. In CAC, bus delay is reduced from $(1+4\lambda)\tau_0$ to $(1+p\lambda)\tau_0$ by reducing the maximum coupling between adjacent wires, where $p = 1$, 2, or 3. The best CACs [9] have shown to provide a speed-up of 2.09×, and an energy-savings of 12% simultaneously. However, CACs have a large area overhead due to the need for additional bus wires.

On-chip signaling to date has focussed on satisfying the RC delay limit, which can be approximated by the Elmore delay of RC network composed of driver, interconnect, and receiver. Clocking a bus faster than the RC limit leads to intersymbol interference (ISI) [11]. ISI spreads out data pulses and reduces their peak magnitude resulting in bit errors at the receivers. In order to gain more insight into the behavior of ISI, consider the transmission of a lone pulse, i.e., a single 1 among a series of 0s or a single 0 among a series of 1s over an RC-dominated interconnect. Figure 2 plots the received pulses for a lone 1 with various pulse widths *PW* for a 10-mm interconnect with CMOS inverters as driver and receiver in a 0.13-μm CMOS process. We observe that the peak amplitude reduces with reduction in *PW*. Further, the peak amplitude goes below the decision threshold of 0.6 V for $PW < 2.9$ ns indicating that errors

Proceedings of the Sixth International Symposium on Quality Electronic Design (ISQED'05)
0-7695-2301-3/05 $ 20.00 IEEE

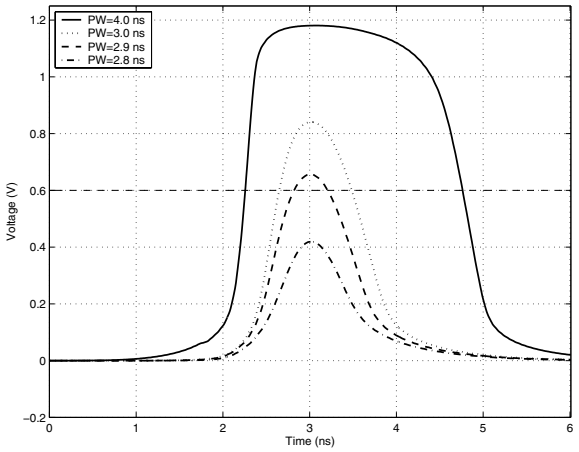


Figure 2. Reduction of peak amplitude due to ISI for a lone 1 in a 10-mm interconnect with pulse width PW.

will occur. Thus, current day systems employ larger pulse widths in order to ensure reliable signaling in the presence of DSM noise and clock jitter, thereby limiting the achievable data rate.

In this paper, we propose the use of equalization for signaling beyond the RC limit and demonstrate its utility in the context of global signaling in 0.13-μm CMOS process. For a 32-bit, 10-mm bus, we show that equalization alone provides a speed-up of 1.28×, with a 9.5% power overhead and 1% area overhead over an unequalized and uncoded bus. We further propose the use of joint equalization and coding where a speed-up of 2.3× is achieved with a 1.3% power savings and a 69% area overhead, where most of the area overhead is due to coding. To the best of our knowledge, this is the first work that proposes equalization and joint equalization and coding for on-chip signaling. Note that several chip input/output (I/O) signaling systems have begun to employ equalization for data rates greater than 3.125 Gb/s [12] and that equalization is expected to remain the work-horse for future I/O systems. Equalization and, more recently, joint equalization and coding have already been employed to enhance data rates in long-haul and macro networks. This work extends the applicability of equalization into the on-chip domain and is part of the general trend of retargetting communication-theoretic techniques to on-chip systems in order to trade-off delay/power with reliability.

2. Equalization

In communication and I/O signaling systems, equalizers are implemented using filters. These filters mitigate ISI by canceling the effect of past and future bits on the current bit.

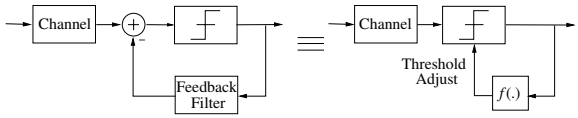
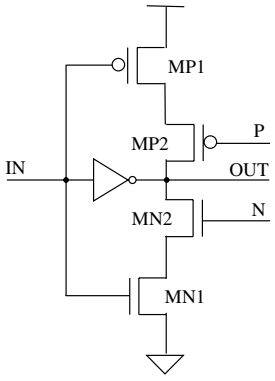


Figure 3. Decision-feedback equalization.



(b)

P	N	V_{th}
GND	VDD	V_{th}^{0}
GND	GND	V_{th}^{+}
VDD	VDD	V_{th}^{-}

Figure 4. Variable threshold inverter.

Figure 3 shows a decision-feedback equalizer (DFE) [11], where the effect of past bits (or decisions) on the current received pulse is canceled by employing a feedback filter. The output of the subtracter is fed to a slicer in order to determine the transmitted bit. This operation is equivalent to modifying the threshold of the slicer based on the past bits as shown in the figure. Such a DFE requires multiply-and-accumulate circuits and is impractical for on-chip buses. Instead, we modify the threshold based on past bits using a variable threshold inverter shown in Figure 4.

In Figure 4, the threshold voltage of the inverter is controlled by using signals P and N. When P = GND and N = VDD, the threshold voltage is V_{th}^{0}, which is the nominal threshold voltage of the inverter. When P = GND and N = GND, the pull down path is off and the threshold voltage increases to V_{th}^{+}. Similarly, when P = VDD and N = VDD, the pull up path is off and threshold voltage decreases to V_{th}^{-}. As shown in the figure, a weak inverter is required to ensure that the output of the inverter is never floating. The relative sizing of transistors MP1, MP2, MN1, and MN2 determines the values of V_{th}^{+} and V_{th}^{-}. Further, there exists a trade-off between the range of variability in the threshold voltage and susceptibility of the inverter to DSM noise due to change in the threshold voltage. We will address this issue in greater detail in Section 2.2.

2.1. Equalizer for a single wire

For a single wire, worst-case delay occurs whenever the wire switches. Therefore, the threshold voltage of the in-

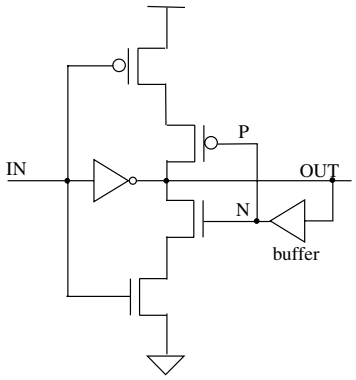


Figure 5. Equalizer for a single wire.

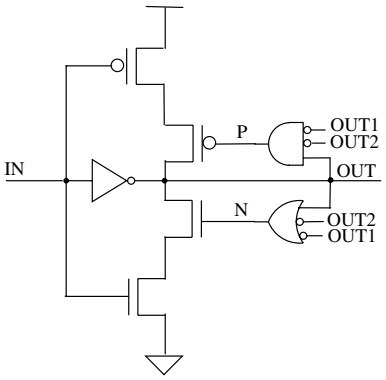


Figure 6. Equalizer for buses.

verter should be adjusted such that it anticipates a transition on the wire. If OUT = VDD (IN = GND), then the threshold voltage should be lowered (P = VDD, N = VDD) such that the delay of a GND $\rightarrow$ VDD transition on the wire is reduced. Similarly, if OUT = GND, then the threshold voltage should be increased (P = GND, N = GND) such that the delay of a VDD $\rightarrow$ GND transition on the wire is reduced. Therefore, the control signals P and N are

$$\mathrm{P} = \mathrm{N} = \mathrm{OUT}. \tag{1}$$

Figure 5 shows the equalizer circuit for a single wire. A buffer is used to drive P and N nodes from OUT. The buffer is designed such that its delay is greater than delay from IN to OUT. Thus, the OUT node is able to charge or discharge before P and N signals change.

2.2. Equalizers for buses

In a bus, the worst-case delay occurs when both neighbors of a switching wire switch in the opposite direction. In other words, the middle wire in a set of three wires will have the worst-case delay when the wires transition from 101 to 010 or vice versa. Therefore, the equalizer for buses needs to utilize the past bits of all three wires when adjusting the threshold voltage for the middle wire. This is achieved by modifying the threshold voltage for the middle wire only when the past output of the set of three wires is either 101 or 010. When the past output is 101, the threshold voltage is raised to V_{th}^{+} and when the past output is 010, the threshold voltage is lowered to V_{th}^{-}. If the past output is neither 101 nor 010, then the threshold voltage is V_{th}^{0}. The above settings of threshold voltage can be obtained by setting P = VDD when the past output is 010 and P = GND otherwise, and by setting N = GND when the past output is 101 and N = VDD otherwise. Therefore,

$$\mathrm{P} = \overline{\mathrm{OUT1}} \cdot \mathrm{OUT} \cdot \overline{\mathrm{OUT2}} \tag{2}$$

and

$$\mathrm{N} = \overline{\mathrm{OUT1}} + \mathrm{OUT} + \overline{\mathrm{OUT2}}, \tag{3}$$

where OUT1 and OUT2 are the outputs of the two adjacent wires.

Figure 6 shows the equalizer circuit for buses. As shown in the figure, the control signals P and N are obtained by using AND and OR gates, respectively. Once again, the delays of the control logic gates are designed to be greater than IN-to-OUT delay so that the OUT node is able to charge or discharge before P and N change. In an equalized bus, all non-boundary wires will have the equalizer as receivers, while the boundary wires can use CMOS inverters. This is because a boundary wire has only one neighbor and, hence, experiences much lower crosstalk.

Modifying the threshold voltage of the receiver increases its susceptibility to noise, in general. We argue next that the robustness of the proposed equalizer is not reduced in any way if crosstalk noise is the dominant noise source. For example, in a crosstalk noise dominated scenario, the middle wire in a set of three wires will experience a glitch whenever a 000 $\rightarrow$ 101 transition occurs. If the receiver threshold for such a wire is lower than the nominal, then the potential for an error increases. The proposed equalizer lowers (raises) the threshold when the past output is 010 (101). A glitch will be introduced if and only if the middle wire remains quiet and one or both of the adjacent wires make a transition from 0 $\rightarrow$ 1 (1 $\rightarrow$ 0). However, such a glitch will make the middle wire voltage to increase (decrease) beyond VDD (GND). Hence, such a glitch will not affect the robustness of the circuit. However, the modification of threshold voltage can make the bus prone to errors due to other sources of noise. Thus, the proposed technique is an example of the fundamental trade-off between delay and reliability.

The proposed equalizer can reduce the bus delay to $(1+3\lambda)\tau_0$ at best. This is because the threshold remains at the nominal level for all past outputs except 010 and 101. Therefore, when the past output is 110, the threshold volt-

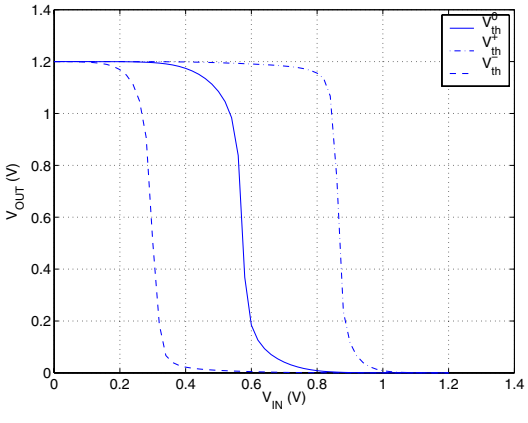


Figure 7. DC transfer characteristics of the equalizer.

age is V_{th}^0. If a $110 \rightarrow 101$ transition occurs, then the middle wire has a delay of $(1+3\lambda)\tau_0$. Since the proposed equalizer mitigates the effect of ISI only after the worst-case transition has occurred, its performance is bounded by $(1+3\lambda)\tau_0$.

The worst-case bus delay can be also reduced to $(1+3\lambda)\tau_0$ by employing crosstalk avoidance codes referred to as forbidden overlap codes (FOC) [10]. In FOC, a codeword having bit pattern 010 does not transition to another codeword having the bit pattern 101 and vice versa. This is achieved by employing additional bus wires and codec circuits. For example, 40 wires are required to implement a coded 32-bit bus [10]. Compared to FOC, equalization requires very small area overhead. Equalization does not require additional wires and the receiver circuits require small area overhead compared to FOC as we show in Section 4.

3. Joint Equalization and Coding

Crosstalk avoidance codes can also reduce the bus delay to $(1+2\lambda)\tau_0$ by either avoiding opposing transitions on adjacent wires [8] or avoiding specific data patterns [7]. Crosstalk avoidance codes proposed in [8] reduce the worst-case delay by ensuring that a transition from one codeword to another codeword does not cause adjacent wires to transition in opposite directions. This constraint on data transitions is known as forbidden transition condition [9]. A forbidden transition code (FTC) satisfies the forbidden transition condition.

The forbidden transition condition ensures that a transitioning wire has its neighbors either quiet or switching in the same direction. When at least one of the neighbors switches in the same direction, the delay of the wire is less than $(1+2\lambda)\tau_0$. The worst-case delay of $(1+2\lambda)\tau_0$ occurs only when a transitioning wire has both its neighbors quiet. This implies that the worst-case delay with FTC can occur in all possible states of the past outputs of the neighbors. Therefore, the equalizer for the coded bus needs to consider only the past bit of the wire. Hence, the equalizer for a coded bus is same as the equalizer for a single wire shown in Figure 5. Thus, crosstalk avoidance coding provides speed-up and simplifies the equalizer by eliminating opposing transitions in adjacent wires.

4. Simulation Results

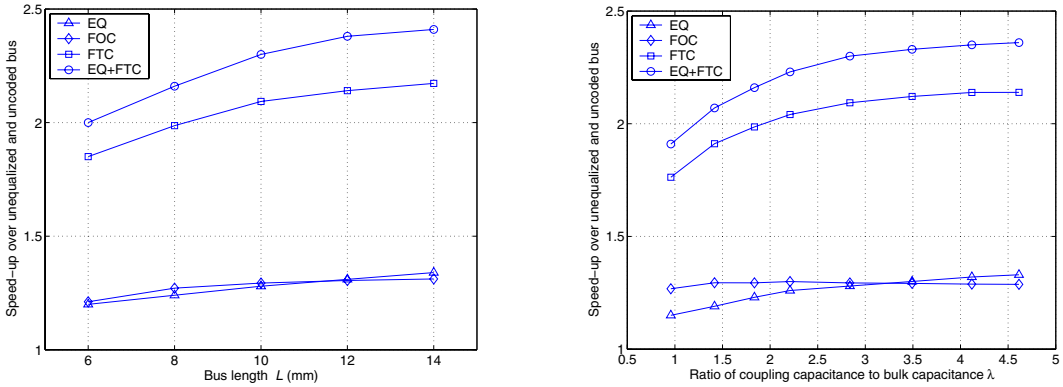
In this section, we present simulation results to demonstrate the improvement in delay achieved by employing equalization and coding. The achieved improvements vary with bus length L, ratio of coupling capacitance to bulk capacitance λ, and the process technology. We quantify the improvements in a standard 0.13-μm CMOS technology for various values of L and λ.

We consider a metal 4 bus with L in the range of 6 mm to 14 mm with minimum width of 0.2 μm and minimum spacing of 0.2 μm. For a given bus geometry, the value of λ depends on the metal coverage in upper and lower metal layers. We vary λ between the following two extreme scenarios. First, 100% metal coverage is assumed in metal layers 3 and 5, resulting in $\lambda = 0.95$. Second, all the bulk capacitance is assumed to be from metal 4 to the substrate, resulting in $\lambda = 4.6$.

Encoder and decoder for CAC are synthesized using Synopsys Design Compiler with a 0.13-μm CMOS standard cell library and optimized for speed. The overhead required in terms of area, delay, and energy dissipation is obtained from synthesized gate level netlists.

The baseline for comparison is an unequalized and uncoded bus with optimally-sized CMOS inverters as drivers and receivers. The equalizer sizing is obtained by a simulation based methodology using HSPICE that optimizes the bus delay. Buses are modeled as RLC transmission lines in HSPICE. Figure 7 shows the DC transfer characteristics of the equalizer designed for a 10-mm bus. A wide range variability in threshold voltage is achievable by appropriately sizing the transistors of the variable threshold inverter. If the output of wire has a slow ramp due to large capacitance, then significant speed-up is achievable by appropriately modifying the threshold.

Figure 8 plots the speed-up achieved by equalization (EQ), forbidden overlap coding (FOC), forbidden transition coding (FTC) and joint equalization and FTC (EQ+FTC). Speed-up is defined as the ratio of the delay of unequalized and uncoded bus to the delay of the proposed bus. Note that the delay of the proposed bus includes the delay of equalizer, encoder, and decoder. The speed-ups achieved increase in all cases with L as shown in Figure 8(a). The effectiveness of equalization improves with L as longer buses have



(a) (b)

Figure 8. Speed-up over unequalized and uncoded 32-bit bus: (a) as a function of bus length L at $\lambda = 2.8$ and (b) as a function of λ at $L = 10$ mm.

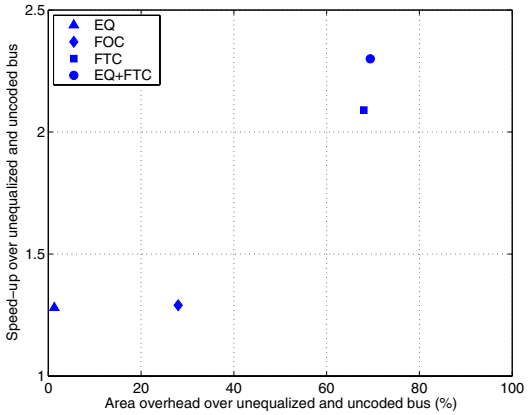


Figure 9. Speed-up vs. area overhead for a 10-mm 32-bit bus with $\lambda = 2.8$.

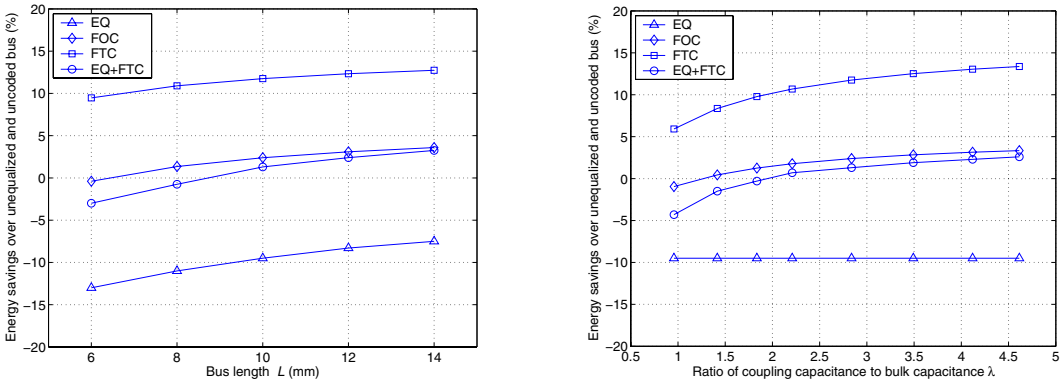
slower ramps. The effectiveness of coding improves with L as the coding delay forms smaller portion of the total bus delay for longer buses.

The speed-ups achieved increases with λ as shown in Figure 8(b). This is because the benefits of employing equalization and coding increase with crosstalk, which in turn increases with λ. For 10-mm bus at $\lambda = 2.8$, equalization, FOC, FTC, and joint equalization and FTC achieve speed-ups of $1.28\times$, $1.29\times$, $2.09\times$, and $2.30\times$, respectively. As expected, equalization and FOC provide approximately the same speed-up as both reduce the bus delay to about $(1+3\lambda)\tau_0$. However, equalization provides speed-up at the cost of a small area overhead, while coding requires large area overhead as shown next.

Figure 9 shows the trade-off between speed-up and area overhead for a 10-mm 32-bit bus. The area overhead for equalization is due to more complex receiver circuits. However, coding requires additional wires along with encoder and decoder circuits. This wiring overhead along with codec circuits translates into 28% area overhead over the uncoded bus for FOC as shown in the figure. The area overhead for equalization is about 1%. Thus, equalization provides the same speed-up as FOC but at a much lower area overhead. Higher speed-ups are achieved by trading-off more area via FTC. The use of FTC alone provides a speed-up of $2.09\times$ with 68% area overhead. The speed-up improves to $2.3\times$ when FTC is combined with equalization and the area overhead increases to 69%. Thus, equalization is an effective technique to improve throughput.

We evaluate the total power overhead next. The average energy per bus transfer is computed assuming that the data is spatially and temporally uncorrelated and that "0" and "1" are equally likely to appear. The energy dissipation in drivers, receivers, and equalizers is obtained by HSPICE simulation using the above data statistics.

Figure 10(a) plots the energy savings compared to the unequalized and uncoded 32-bit bus as a function of L at $\lambda = 2.8$. The required power overhead for equalization reduces with L as the equalizer power will account for a smaller portion of the total power for longer buses. Note that equalization on its own does not reduce power because the equalizers are complex inverters. Coding provides energy savings along with speed-up by reducing the coupling component of energy [9]. However, as mentioned earlier, coding has a large area penalty. Joint equalization and FTC has small energy savings or overhead as the two effects cancel each other out. The energy savings improve with λ for coding and joint equalization and coding as shown in Figure 10(b). For 10-mm bus at $\lambda = 2.8$, equalization requires 9.5% energy overhead, while FOC, FTC, and joint equalization and FTC provide 11.3%, 2.4%, and 1.3% energy savings, respectively.



(a) (b)

Figure 10. Energy savings over unequalized and uncoded 32-bit bus : (a) as a function of bus length L at $\lambda = 2.8$ and (b) as a function of λ at $L = 10$ mm.

5. Conclusions

We have proposed joint equalization and coding as a method to counter the trend of increasing gap between gate and interconnect delay. We have shown that equalization provides 1.28× speed-up at the cost of small power overhead, while joint equalization and coding provides 2.30× speed-up at the cost of large area overhead and no power penalty. Technology scaling will improve the performance of the proposed techniques. The increasing value of coupling capacitance will increase crosstalk and, hence, benefits of employing equalization and crosstalk avoidance coding will improve. Further, codec delay, area, and power dissipation will reduce with scaling and, hence, the performance of the coded bus will improve.

The equalizer proposed in this paper employs just the past bits of three adjacent wires to adjust the threshold of the receiver. More complex equalizers can be envisaged that utilize several past outputs and/or outputs of non-immediate neighbors. Technology scaling will enable and necessitate the use of such receivers in the future to achieve high-speed on-chip bus communication.

6 Acknowledgments

This work was supported by the MARCO-sponsored Gigascale Systems Research Center and Intel Corporation.

References

[1] *International Technology Roadmap for Semiconductors*, Semiconductor Industry Association, 2003. [Online]. Available: http://public.itrs.net

[2] R. Ho, K. W. Mai, and M. A. Horowitz, "The future of wires," *Proc. IEEE*, vol. 89, pp. 490–504, Apr. 2001.

[3] L. Benini and G. D. Micheli, "Networks on chips: a new SoC paradigm," *IEEE Computer*, vol. 35, pp. 70–78, Jan. 2002.

[4] D. Pamunuwa, L.-R. Zheng, and H. Tenhunen, "Maximizing throughput over parallel wire structures in the deep submicrometer regime," *IEEE Trans. VLSI Syst.*, vol. 11, pp. 224–243, Apr. 2003.

[5] A. Nalamalpu, S. Srinivasan, and W. P. Burleson, "Boosters for driving long onchip interconnects - design issues, interconnect synthesis, and comparison with repeaters," *IEEE Trans. Computer-Aided Design*, vol. 21, pp. 50–62, Jan. 2002.

[6] P. P. Sotiriadis and A. Chandrakasan, "Reducing bus delay in submicron technology using coding," in *Proc. ASP-DAC*, 2001, pp. 109–114.

[7] C. Duan, A. Tirumala, and S. P. Khatri, "Analysis and avoidance of cross-talk in on-chip buses," in *Proc. Hot Interconnects*, 2001, pp. 133–138.

[8] B. Victor and K. Keutzer, "Bus encoding to prevent crosstalk delay," in *Proc. ICCAD*, 2001, pp. 57–63.

[9] S. R. Sridhara, A. Ahmed, and N. R. Shanbhag, "Area and energy-efficient crosstalk avoidance codes for on-chip buses," in *Proc. ICCD*, 2004, pp. 12–17.

[10] S. R. Sridhara and N. R. Shanbhag, "Coding for reliable on-chip buses: fundamental limits and practical codes," in *Proc. VLSI Design*, 2005.

[11] J. G. Proakis, *Digital Communications*. New York, NY: McGraw-Hill, 2000.

[12] J. L. Zerbe *et al.*, "Equalization and clock recovery for a 2.5-10-Gb/s 2-PAM/4-PAM backplane transceiver cell," *IEEE J. Solid-State Circuits*, vol. 38, pp. 2121–2130, Dec. 2003.