Short Papers

Segmented Bus Design for Low-Power Systems

J. Y. Chen, W. B. Jone, *Member, IEEE,* J. S. Wang, H.-I. Lu, and T. F. Chen

***Abstract*— This paper[1] proposes a bus-segmentation method that efficiently reduces the switched capacitance on the bus. The power consumed by the bus can, therefore, be substantially reduced. The basic idea of bus segmentation is to partition the bus into several bus segments separated by pass transistors. Highly communicating devices are located to adjacent bus segments, thus, most data communication can be achieved by switching a small portion of the bus segments. As a result, power consumption and critical path delay are both reduced. Experimental results obtained by simulating a delay model and a power model demonstrate that the proposed segmented bus system reduces bus power by about 60%–70% and improves critical bus delay by about 10%–30%.**

***Index Terms*— Bus communication, bus graph, bus segmentation, bus tree, low-power systems.**

I. Introduction

VERY large scale integration (VLSI) design for power optimization to satisfy the power budget is a very important research issue. Generally, a bus system consumes a very significant portion of power in the whole chip. For example, the bus wires dissipate about 15%–30% of total power in Alpha 21 064 and Intel 80386, respectively [1]. The work by [2] also demonstrates that at least 20%–35% of total power is dissipated by the bus wires in the case of quadrature mirror filter (QMF) filters. Thus, there have been much research work attempting to optimize the bus power from different design aspects. Basically, most of the work can be divided into three categories:

1) design of bus drivers/receivers to decrease the bus swing [3]–[5];
2) encoding techniques to reduce the bus-switching activity [6], [7];
3) bus structure redesign to take advantage of local communications [2].

In this paper, we propose a bus-segmentation method which changes the bus topology to reduce the power consumption without sacrificing the bus speed. The basic idea is to partition a single (and large) bus into several tree-structured bus segments that are separated by pass transistors. The segmented bus system is designed in such a way that heavily communicating devices are connected to the adjacent bus segments. Thus, by turning off some pass transistors, only part of the bus segments are involved in the data communication when two devices are performing data exchange. By this design, instead of charging or discharging the entire bus, only a small number of the bus segments are involved in each device communication, and power consumption can be greatly reduced. Since each data communication is performed through bus segments which are adjacent, the bus communication speed can be enhanced as well.

This work starts by building the bus graph model for a bus communication system. Each edge of the bus graph is weighted by a communication frequency if both devices connected by this edge have data exchange. Given the weighted bus graph, we apply the efficient Gomory–Hu method [8] to partition the single bus into a bus tree where each tree segment is connected by a single device. A tree-clustering method is then used to connect several devices into a bus segment if such segment sharing can further reduce the power consumption. In order to evaluate the effect of bus segmentation on the bus speed, a bus delay model is established to calculate the critical path delay of the new bus design. Results obtained by computer simulation demonstrate that the proposed bus-segmentation approach is very efficient. In most cases, about 70% of bus power can be saved and bus speed improvement is about 10%–30%.

II. The Bus-Graph Model

The power model of a circuit is generally expressed as

$$P = \sum C_i \cdot V_{dd}^2 \cdot f_i$$

where C_i is the load capacitance of circuit i, V_{dd} the operation voltage, and f_i the switching frequency. With the assumption of a fixed operation voltage, the power model can be simplified as

$$\sum (C_i \cdot f_i).$$

This paper concentrates on bus segmentation which leads to switched capacitance reduction for efficient power reduction.

The basic idea of bus segmentation is to divide the entire bus into several small bus segments, as shown in Fig. 1, such that data exchange among devices will result in minimum power consumption. It is assumed that the bus operation is *dynamic* and separated into two phases: the precharge phase

Manuscript received December 15, 1997; revised May 15, 1998.

J. Y. Chen, W. B. Jone, H.-I. Lu, and T. F. Chen are with the Department of Computer Science and Information Engineering, National Chung-Cheng University, Chiayi, Taiwan, R.O.C.

J. S. Wang is with the Department of Electrical Engineering, National Chung-Cheng University, Chiayi, Taiwan, R.O.C.

Publisher Item Identifier S 1063-8210(99)01551-6.

[1]See the Guest Editorial of the Special Section on Low-Power Electronics and Design of the IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 6, pp. 518–519, Dec. 1998.

1063–8210/99$10.00 © 1999 IEEE

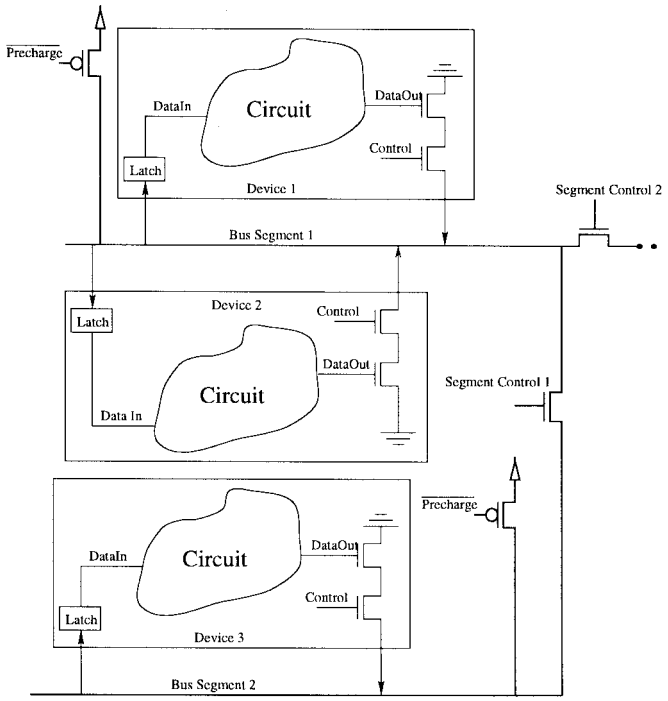

Fig. 1. Bus segmentation.

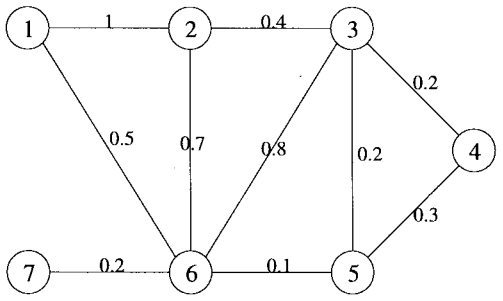

Fig. 2. A graph with edge weights.

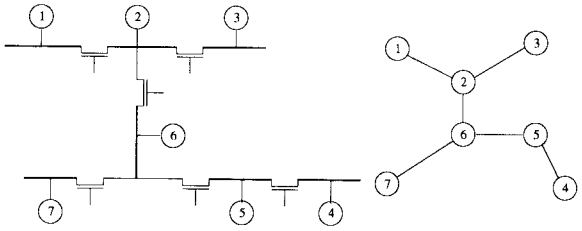

Fig. 3. An example for bus segmentation.

and evaluation phase. In the precharge phase, the precharge pMOS is turned on and the entire bus is charged to high voltage. During the bus evaluation phase, the devices on the bus will communicate with each other by enabling the control signal (of the sender) and the input latch (of the receiver). Now, if the communication signal is one, the bus remains in high voltage. If signal 0 will be propagated through the bus, however, the bus is thus discharged. Since power dissipation occurs when a signal 0 is to be propagated through the bus, only the communication frequency for signal 0 must be considered. To further simplify the investigation, we assume that signal 0 and signal 1 will be propagated through the bus with equal likelihood. Thus, we can only concentrate on the communication frequency as far as low power design is concerned. More general cases with logic value considered can be accomplished by computer simulations.

As shown in Fig. 1, devices 1 and 2 are connected to the same bus segment, but separated from others by pass transistors. When device 1 sends a signal to device 2 (intra-segment communication), both pass transistors connected to bus segment 1 will be turned off. Thus, the signal only flows within the small bus segment, instead of the entire bus, so this avoids charging or discharging the unnecessary part of the entire bus system. The power saved can be very significant if most data exchanges are performed by intra-segment communication. However, pass transistors must be turned on for inter-segment communication, e.g., segment control 1 of Fig. 1 must be turned on if device 1 sends a signal to device 3. The number of pass transistors to be turned on for each inter-segment communication depends on the topological relationship among the devices and the bus architecture. Thus, it is important to organize the bus architecture such that most data exchanges will be performed within the bus segments.

The first step to solve the bus-segmentation problem is to characterize the hardware structure that can be mapped into a graph, called a "bus graph." Thus, each bus configuration can be represented as a weighted undirected graph $G = (V, E)$ where V is the set of devices connected to the bus; and there exists a weighted edge $e = (i, j)$ if devices i and j have signal exchange. The weight associated with edge (i, j) specifies the communication frequency between devices i and j. For example, Fig. 2 gives the bus graph for a bus connecting seven devices. Multiple-bus systems can be modeled similarly. Two assumptions are made to simplify further discussions: 1) the length (and thus, the wire capacitance) of each bus segment is proportional to number of devices connected to the bus segment and 2) only one sender and one receiver are involved at each data transfer. More general cases can be handled with slight modification to the graph model. Finally, the bus-segmentation problem can be modeled to partition the bus graph into a bus tree such that the total switched capacitance and, thus, the power consumption, by bus switching is minimized.

To further simplify the problem, we assume that exactly one device is connected to each bus segment of the bus tree. A possible bus-tree implementation for the bus graph of Fig. 2 is shown in Fig. 3(a) (Fig. 3(b) gives the corresponding bus tree). For example, devices 1 and 6 have communication frequency 0.5, and their communication can be accomplished by routing signals from device 1 to device 2, and then to device 6. The switched capacitance between devices i and j is the product of $\text{weight}_G(i, j)$ and the capacitance $C(i, j)$ required to be charged or discharged, when devices i and j perform signal exchange. For example, the switched capacitance between devices 1 and 6, $C(1, 6)$, can be roughly estimated by $0.5 \cdot C(1, 6) = 0.5 \cdot 3 \cdot C_s = 1.5 \cdot C_s$, where C_s is the capacitance introduced by each bus segment (including

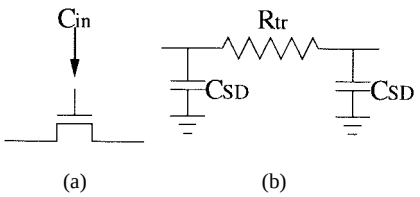

Fig. 4. (a) The nMOS schematic. (b) The nMOS *RC* model.

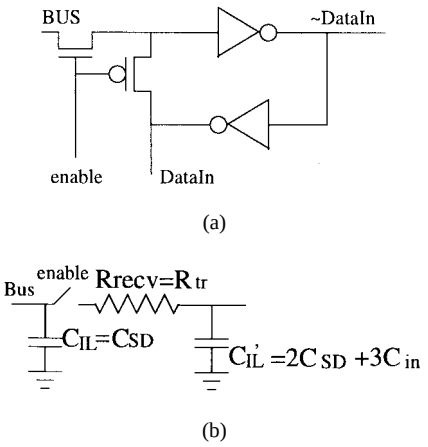

Fig. 5. The input latch model.

the wire and device capacitances). The general case where more than one device is connected to each bus segment will be discussed in the following section by the tree-clustering technique.

If device 1 is sending a logic 0 to device 6, as shown in Fig. 3, for example, then several components will be discharged during the evaluation phase. These components include: 1) the input/output switches for devices 1, 2, and 6; 2) the bus wires for devices 1, 2, and 6; 3) the pass transistors which must be turned on, i.e., those between device pairs (1, 2) and (2, 6); and 4) pass transistors which are turned off, i.e., those between device pairs (2, 3), (5, 6), and (6, 7). Consequently, the capacitance cost of a signal communication between devices i and j can be estimated by the formula

$$n \cdot C_w + n \cdot C_d + C_{T_{\text{on}}} \cdot n' + C_{T_{\text{off}}} \cdot n'' + \alpha$$

where the notation is defined as follows.

- C_w: wire capacitance of a bus segment.
- C_d: input/output capacitance of a device which is turned off. We have shown how a device is connected into a bus segment in Fig. 1 where each device has an input latch to receive data from the bus, and two output nMOS transistors to send data to the bus. Basically, an nMOS transistor can be modeled as shown in Fig. 4, where C_{in} is the gate input capacitance and C_{SD} is the source/drain to substrate capacitance. Additionally, an input latch is shown in Fig. 5. Now, if the latch enable signal is turned off, the capacitance seen from the bus is only an nMOS transistor, which is turned off. Thus, only one C_{SD} is contributed to the capacitance calculation as far as the latch is concerned. The total input/output capacitance of an inactive device is $2C_{\text{SD}}$, one contributed by the latch and the other by the nMOS for output control. During the evaluation phase, the input/output capacitance of the inactive device will be discharged, even though the device is not involved in the data communication.
- $C_{T_{\text{on}}}$: pass transistor capacitance when it is turned on. This value equals $2C_{\text{SD}}$.
- $C_{T_{\text{off}}}$: capacitance seen from one side of a turned-off pass transistor. In this case, only one diffusion capacitance is counted so the value of $C_{T_{\text{off}}}$ is C_{SD}.
- n: number of bus segments charged or discharged for communicating devices i and j.
- n': number of pass transistors which must be on for communicating devices i and j.
- n'': number of pass transistors which must be off to isolate the unused bus segments for communicating devices i and j.
- α: extra capacitance (other than C_d) introduced by the input latch of the receiving device and the output control transistors of the sending device.

Suppose n bus segments are traversed for the purpose of communication between devices i and j. Since the bus topology is a tree, the number of pass transistors that must be turned on is $n-1$. Therefore, we have $n' = n-1$. Furthermore, the number of gates that must be turned off is, at most, $|V|-n$, where $|V|$ is the total number of devices on the bus tree. Note that the pass transistors which are not directly connected to the communication path do not have to be turned off since the control lines of the unused devices prohibit bus discharging. Thus, the value of n'' is, at most, $|V| - n$. By the above discussion, the capacitance cost formula can be upper bounded by $k_1 \cdot (n-1) + k_2$ where k_1 and k_2 are the following constants:

$$\begin{aligned}(C_w + C_d) \cdot n + C_{T_{\text{on}}} \cdot (n-1) + C_{T_{\text{off}}} \cdot n'' + \alpha \\ \le (C_w + C_d + C_{T_{\text{on}}} - C_{T_{\text{off}}}) \cdot (n-1) \\ + (C_w + C_d + C_{T_{\text{off}}} \cdot |V| - C_{T_{\text{off}}} + \alpha) \\ = k_1 \cdot (n-1) + k_2.\end{aligned}$$

For example, in Fig. 3, the capacitance cost for device 1 to send a logic 0 to device 6 can be estimated by assigning 3 to n. Note that the capacitance introduced by device connections (i.e., the input latch and output control transistors) are considered by the term $n \cdot C_d$. However, in Fig. 3, the sending device 1 and receiving device 6 each contributes more than C_d. Consequently, the constant α is added to take this factor into account.

III. The Bus-Segmentation Problem

By the above discussion, we know that, in order to achieve low-power design for a bus organization, it suffices to minimize the number $(n-1)$ of traversed bus segments for each pair of data communication. We need the following definitions to formally define the bus-segmentation problem:

Definition 3.1: The distance of two nodes i, j, denoted $\text{dist}(T, i, j)$, on a tree T is the number of edges traversed from i to j on T.

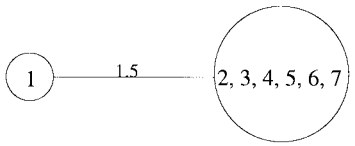

Fig. 6. The tree of super nodes after obtaining the minimum 1–2 cut.

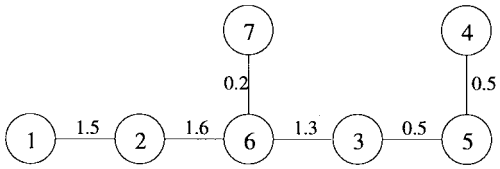

Fig. 7. The final topology of the super nodes.

Definition 3.2: The linear arrangement cost of a spanning tree T corresponding to a weighted graph G, denoted $\mathrm{cost}_{\mathrm{LA}}(T, G)$, is

$$\sum_{e=(i,j)\in E} \mathrm{weight}_G(e) \cdot \mathrm{dist}(T, i, j).$$

Finally, the bus-segmentation problem can be defined as follows: Given a weighted undirected graph $G = (V, E)$, the bus-segmentation problem is to identify a spanning tree T whose linear arrangement cost $\mathrm{cost}_{\mathrm{LA}}(T, G)$ is minimum.

An optimal solution can be found for our bus-segmentation problem using a method proposed by Gomory and Hu [8]. That is, given a weighted and undirected graph, the Gomory–Hu algorithm can be applied to find a tree whose linear arrangement cost is minimum and the entire process is guaranteed to be finished in polynomial time. The algorithm is illustrated using the example in Fig. 2 where a weighted undirected graph is given. In the beginning of the algorithm, all nodes are clustered in a super (single) node. We arbitrarily select two nodes, e.g., 1 and 2, in Fig. 2 to be disconnected. Since the minimum cut set which uncouples nodes 1 and 2 lies on edges (1, 2) and (1, 6), the super node can be divided into two super nodes (1) and (2–7) with weight 1.5 (Fig. 6). By repeatedly applying the above process, we can finally have the graph transformed into a weighted tree (Fig. 7). It can be proven that the tree generated has the minimum linear arrangement cost using polynomial computing time.

IV. Tree Clustering

In the previous section, we have derived the optimal solution for the construction of bus segments if each segment is allowed to have only one device attached. However, how is the power cost changed if more than one device is attached to a bus segment? To answer this question, the bus segment architecture must be investigated. Suppose bus segments S_1 and S_2 are adjacent and separated by a pass transistor; furthermore, device D_i is connected to S_i. Now, if S_1 and S_2 are attached into a single (but longer) bus segment, i.e., D_1 and D_2 share the same bus segment, then the capacitance switched for communications involving D_1 or D_2 might be increased or decreased. For example, if D_1 communicates with D_2, then the switched capacitance is reduced because of the elimination of the pass transistor. However, if D_1 communicates with a device other than D_2, then the switched capacitance might be increased because the signal unnecessarily charges longer bus wire due to the removal of the pass transistor between S_1 and S_2.

To further lower the power cost, we give a heuristic algorithm which incrementally (and locally) improves the switched capacitance. The algorithm arbitrarily selects an edge e in the bus tree already derived and tries to merge the corresponding two end-nodes (devices) of e into one generalized node (the same bus segment). If this merging results in smaller power cost, then these two nodes will be melded into the same cluster. Otherwise, the algorithm will give up the merging process for e. This process is repeated until merging two generalized nodes for any edge will not further reduce the switched capacitance. Note that the power cost is derived by enumerating the (weighted) switched capacitance for each pair of data communication using the formula given in Section II.

TABLE I
Simulation Results

Original Bus		Segmented Bus	
power	critical delay	power	critical delay
2088.38 fF	9.67781 ns	719.046 fF	6.77183 ns

V. Simulation Results

In this section, to evaluate the performance of the proposed bus-segmentation method, experiments are conducted using computer simulation. Given a bus graph, the corresponding bus tree is constructed using the Gomory–Hu algorithm. The generalized Elmore delay (GED) model [9] and the switched-capacitance power model are then applied to estimate the critical path delay and power consumption for the resulting bus structure. The bus system is assumed to be fabricated using an industrially available 0.6-μm technology[2] with the die size equal $4200 \cdot 4200$ μm^2. We further assume that seven devices are connected to a single bus and the length of each bus segment is roughly estimated to 600 μm. All R, C parameters are estimated based on the 0.6-μm technology and are then applied to the GED and power models. Both critical path delay and power consumption can thus be derived. We emphasize again that, instead of extracting parameters from a realistic design implementation, the experiment is conducted using R, C parameters estimated from a TSMC 0. 6-μm technology.[2]

We use a bus system whose corresponding graph is shown in Fig. 2, and the bus graph is then transformed into the bus tree given in Fig. 7 using the Gomory–Hu algorithm. It is interesting to compare the power consumption and critical path delay for the original (single) and segmented bus systems. As described above, the power consumption is estimated by the weighted capacitance, while the critical path delay is analyzed by the GED model. The R, C parameters used do not encourage tree clustering, thus each bus segment connects only one device. The results are very encouraging, as shown in Table I. It can be found that both power consumption and critical path delay have been dramatically improved.

[2]0.6-μm Logic CMOS Process (0.6U-TW-14L-DS), TSMC, Hsin-Chu, Taiwan, R.O.C.

VI. Conclusion

This paper proposes a bus-segmentation method for lowering the power dissipation on system buses. If the highly communicating devices are clustered and separated from the rest of devices, the bus capacitance required to be charged/discharged becomes smaller and the power dissipation can thus be saved. By computer simulation, we have found that the bus delay can also be improved. Our example demonstrates that the bus-segmentation method results in a significant bus power reduction of magnitude of about 65% and a bus performance improvement of about 30% (using models). Recently, we have implemented the bus-segmentation technique into an 8-b microcontroller, and the chip layout is then simulated using Powermill. Results show that about 25% of power can be saved with about 2.6% of hardware overhead. Another set of data will be released after the chip has been fabricated. The results can be even more attractive if the number of data bits is large since more power can be saved for bus switching (but extra power consumed by the controller is nearly the same).

References

[1] D. Liu and C. Svensson, "Power consumption estimation in CMOS VLSI chips," *IEEE J. Solid-State Circuits*, vol. 29, pp. 663–670, June 1994.

[2] R. Mehra, L. M. Guerra, and J. M. Rabaey, "A partitioning scheme for optimizing interconnect power," *IEEE J. Solid-State Circuits*, vol. 32, pp. 433–443, Mar. 1997.

[3] Y. Najagome, K. Itoh, M. Isoda, K. Takeuchi, and M. Aoki, "Sub-$1-v$ swing internal bus architecture for future low-power ULSI's," *IEEE J. Solid-State Circuits*, vol. 28, pp. 414–419, Apr. 1993.

[4] G. C. Cardarilli, M. Salmeri, A. Salsano, and O. Simonelli, "Bus architecture for low-power VLSI digital circuit," in *Proc. Int. Symp. Circuits Syst.*, vol. 4, May 1996, pp. 21–24.

[5] R. Golshan and B. Haroun, "A novel reduced swing CMOS bus interface circuit for high-speed low-power VLSI systems," in *Proc. Int. Symp. Circuits Syst.*, vol. 4, May 1994, pp. 351–354.

[6] M. R. Stan and W. P. Burleson, "Coding a terminated bus for low power," in *Proc. 5th Great Lake Symp. VLSI*, Mar. 1995, pp. 70–73.

[7] ———, "Bus–invert coding for low-power I/O," *IEEE Trans. VLSI Syst.*, vol. 3, pp. 49–58, Mar. 1995.

[8] R. E. Gomory and T. C. Hu, "Multi-terminal network flows," *J. Soc. Industrial Appl. Math.*, vol. 9, no. 4, pp. 551–569, Dec. 1961.

[9] K. Yamakoshi and M. Ino, "Generalized Elmore delay expression for distributed *RC* tree networks," *Electron. Lett.*, vol. 29, pp. 617–618, Apr. 1993.