A Method of Distributed Controller Design for RTL Circuits

Chris Papachristou

Computer Engineering Department
Case Western Reserve University
Cleveland, Ohio 44106

Yusuf Alzazeri

Computer Engineering Department
Case Western Reserve University
Cleveland, Ohio 44106

Abstract

This paper describes a design or redesign technique to reduce the control path delay and thus improve the performance of an RTL circuit. The basic idea is to replace an existing centralized controller of an RTL circuit with a distributed controller structure which is made up of multiple local controllers. By placing local controllers close to the datapath resources they control reduces the control wire length which in turn increases the circuit performance, especially at higher frequencies where interconnect delays are dominant. The approach begins with a datapath partitioning into distinct functional blocks, local controller extraction algorithm, followed by and hierarchical floor planning. Two local controller styles are presented, the communicating and the and non-communicating local controller, CCL and NCCL, respectively.

Introduction

The critical path delay in a RTL circuit with a datapath and a controller is the register to register data flow path with the longest delay. It consists generally of three delays: controller delay, control wire delay and datapath delay. The critical path delay determines the circuit clock cycle. So reducing the critical path delay increase the circuit performance. Traditionally, techniques are used at different design level to reduce the critical path delay. At the RTL level, module generation techniques produce faster modules to improve performance in the critical path. Logic minimization methods are used at the logic level to reduced the number of gate levels in the critical path. Smaller devices are used at the physical level to reduce capacitance which in turn reduce critical path delay. These techniques reduce the datapath and/or the controller delays but not necessarily the wiring delay. However, at high frequency digital system, the interconnect wiring delay is the dominant factor in the circuit delay including the critical path delay. In an RTL circuit, long wires occurs between the controller and datapath. A control wire delay have been shown in [1] to be the slowest segment of the overall critical path delays. A control path starts at the controller state register, traverses the controller output logic and the wiring connecting to the datapath control points.

A controller normally modeled as a finite state machine. Much work has been done on finite state machine decomposition [1, 2, 3]. All of these methods try to reduce the area or reduce the delay within the finite state machine. We propose a different decomposition method based on RTL information with the objective, to reduce the control path delay, including control wires delays which effects circuit performance. Eppling [1] did some work on reducing the control path delay by decomposing a centralized controller into multiple local controllers. He used a layout model and a wire length extraction technique, followed by clustering of of control points into local groups.

Our method transforms an existing or a new RTL circuit into one with distributed controller structure. In our approach we partition the RTL based controller output into groups of control points. This requires partitioning of the datapath around its constituent ALUs, so that all control points that enable an ALU operation are placed in the same partition. The centralized controller output is split into control points which resulted from the datapath partitioning. Multiple local controllers are generated each controlling one partition or functional block. During the layout phase, each local controller is placed physically close to its corresponding functional block, shortening wire lengths and thus reducing delays, especially at high frequencies. This is the main motivation of our work.

Experimentation and Results

Our CAD software consists of the datapath partitioning algorithm and the distributed controller design algorithms. We have implemented some of the high-level synthesis benchmarks using the COMPASS commercial CAD system. For each example, we generated the conventional datapath and centralized controller by SYNTEST [4, 5], then we partitioned the datapath into functional blocks. For every functional block we generated an NCLC local controller and a CLC local controller. The three designs were synthesized using the COMPASS ASIC Synthesizer. We used the COMPASS floor planner to generate hierarchical layouts based on macro cells. The entire circuit layout was done by placing these macro cells and performing block routing among them. COMPASS "interconnect" was used to extract the control wires RC factors for timing analysis. The parasitic capacitance was included in the critical path delay calculation. COMPASS QTV is used to compute the critical path delay at 50 MHz and 500MHz for the three designs.

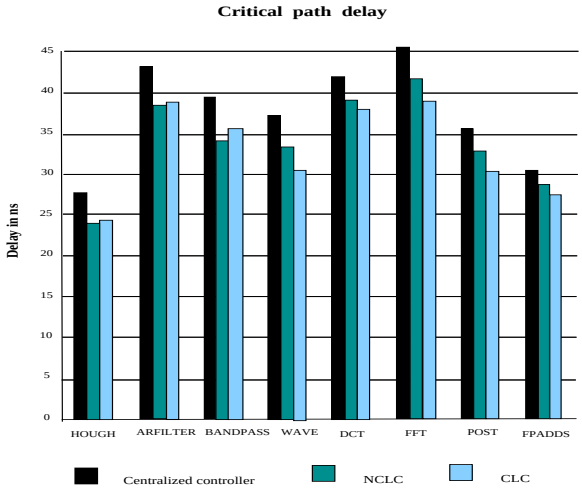

Figure 1: Critical path delay at low frequency (50MHz).

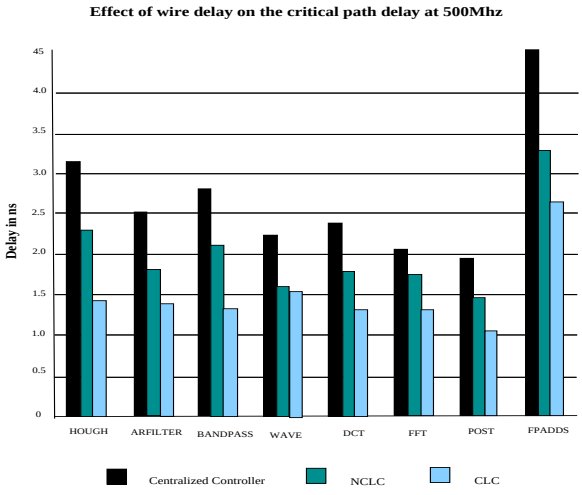

Figure 2: Critical path delay at high frequency (500Mhz)

Eight design examples were used: Discrete Cosine Transform (DCT), WAVE Filter, BANDPASS Filter, ARFILTER, HOUGH, Fast Fourier Transform (FFT), Postage Price Calculation (POST) and Floating-Point Addition (FPADDS). The chart in Figure 1 shows the critical path delay for all eight examples under the three design styles assuming frequency of 50MHz.

The results in Figure 1 suggest strongly that the distributed controller design has less critical path delay than the centralized controller.

The chart in Figure 2 shows the effect of wire delay on the critical path delay at 500MHz. The reduction in the critical path delay between the centralized and distributed controller is about 10-30%.

All the experimental data are summarized in the table at end of this section. Columns 1 and 2 summarize the results of Figures 1 and 2, respectively. Column 3 provides the number of transistors for each design.

Summary

Our method has been implemented in C on Unix SPARC stations. The critical path delay was calculated for circuits operating at normal and high frequencies and with normal and reduce gate size. The interconnect parasitic capacitance was included in the critical path delay.

We compared two distributed controller implementations, the Non-Communicating Local Controller (NCLC) design and the Communicating Local Controller (CLC) design with the centralized controller design . We have reported very encouraging results of critical path delay reduction up to 12% for the distributed controller circuits operation at 50 MHz frequency and about 25% for circuits operating at 500MHz. One important conclusion from our results is that distributed controller design followed by hierarchical floor planning provides an effective technique for boosting digital circuit performance.

Experimental Data

Applications	Controller	1	2	3
HOUGH	Centralized	2692	27.53	3.18
	NCLC	1858	24.8	2.27
	CLC	1795	24.83	1.43
ARFILTER	Centralized	5446	42.95	2.50
	NCLC	3669	38.43	1.80
	CLC	3715	38.58	1.40
BANDPASS	Centralized	6961	39.22	2.69
	NCLC	4766	34.31	2.08
	CLC	4831	36.04	1.35
WAVE	Centralized	5898	37.33	2.24
	NCLC	4448	33.07	1.60
	CLC	4663	29.93	1.54
DCT	Centralized	8843	41.93	2.42
	NCLC	8173	39.39	1.75
	CLC	8694	37.95	1.32
FFT	Centralized	10176	45.64	2.12
	NCLC	9868	41.82	1.73
	CLC	9410	39.31	1.32
POST	Centralized	6381	35.80	1.94
	NCLC	6022	32.61	1.48
	CLC	5596	31.04	1.16
FPADDS	Centralized	6213	31.52	4.53
	NCLC	10497	28.56	3.33
	CLC	6751	27.58	2.66

References

[1] J. Eppling and D. Thomas, "Multiple Controller Synthesis for Reducing Control Path Delays", *SRC TECHCON*, Sept. 1993.

[2] S. Devadas and R. Newton, " Decomposition and Factorization of Sequential Finite State Machine", *IEEE Trans. on CAD*, Vol. 8, pp. 1206-1217.

[3] P. Ashar and S. Devadas, "em Optimum and Heuristic Algorithms for an Approach to Finite State Machine Decomposition", *IEEE trans. on CAD*, Vol. 10, pp. 296-310.

[4] H. Harmanani, C. Papachristou, S. Chiu and M. Nourani, "SYNTEST: An Environment for System-Level Design for Test," *European Design Automation Conference (EURO-DAC)*, Sept. 1992.

[5] C. Papachristou, H. Harmanani and M. Nourani, "An Approach for Redesign in Data Path Synthesis", *DAC-93*, June 1993.