Multiple-Width Bus Partitioning Approach to Datapath Synthesis

Arash Ahmadi Mark Zwolinski
Electronic System Design Group, School of Electronics and Computer Science, University of Southampton
Southampton, UK, SO17 1BJ
{aa03r, mz}@ecs.soton.ac.uk

Abstract—**A shared bus is a suitable structure for minimizing the interconnections costs in system synthesis. It has also been shown that the word-length of Functional Units has a great impact on design costs. A combination of both methods is used in this paper in the form of a partitioned shared bus structure, in which every partition has a different width and all the functional units connected to a bus partition have the same input/output word-lengths. Having controlled the group binding and word-length of the FUs as well as the other synthesis parameters, a high-level synthesis tool is introduced to implement DSP algorithms in digital hardware. The tool uses a Multi-Objective Optimization Genetic Algorithm to minimize the circuit area, delay, power consumption and digital noise by selecting an optimal grouping and word-length for each FU in a shared bus system. Results demonstrate that savings can be made in the overall system costs by applying this method.**

I. INTRODUCTION

The objective of this work is to present the concept of *Multiple-Width Bus Partitioning* (MWBP) and to show how it is used to find the optimum grouping scheme for *Functional Units* (FU) that share a section of the partitioned bus with the same Word-Length (WL). Each section is isolated from other FUs unless a trans-group data exchange happens. In other words, the feasible space for High level Synthesis (HLS), is broadened by two new dimensions: bus partitioning and WL. Accordingly, a Multiple-Objective Optimization (MOO) method is introduced and implemented using a Genetic Algorithm (GA) to optimize the circuit area, power consumption, accuracy and speed in this extended space.

One of the problems in implementing massively computational hardware, for instance a Digital Signal Processor (DSP), is choosing an appropriate word length for arithmetic units. Using a WL less than the worst-case assumption at different points in the system would save implementation costs [6]. To our best knowledge, this issue and related subjects have been investigated only in terms of their effects on FU implementations, independently of the target architecture or of the communication structure between FUs.

One commonly used on-chip communication structure is the shared bus architecture in which one or a small number of interconnections are shared among all the FUs as a data bus. The main advantages of the shared bus architecture include: simple topology, low cost, and extensibility. It is, however, slow and requires more control overhead in comparison to directly connected units. Segmentation of the shared bus is a simple but effective way to cut down the communication latency which also reduces the power consumption [13].

This work presents an application of WL optimization and bus partitioning to improve the circuit costs and optimization speed in datapath synthesis for hardware implementation of DSP algorithms. Previous work is reviewed in section two; section three explains the implementation method and details of the cost functions are explained in the section four. The GA method which is utilized is discussed in section five and finally results are reported in section six.

II. BACKGROUND

Kum and Sung [12] introduced several heuristic WL optimization methods to trade-off system area against Signal-to-Quantization-Noise Ratio (SQNR). In their technique, a reference system is designed without overflows or signal quantization effects and then a HLS is performed based on the minimum WL information, while the final WL optimization is conducted using the synthesized hardware models. There is no suggestion in that work to use the power consumption as an optimization objective.

Constantinides, Cheung, and Luk focused on developing algorithms for WL optimization [6],[7]. These methods employed analytical digital noise analysis which is more suitable for Linear Time Invariant (LTI) systems. Constantinides later extended the previous efforts to nonlinear components in a datapath by employing a small signal approach and investigated the effect of precision optimization on power reduction as a by-product of the WL optimization [5]. Again in this work power consumption was not an objective in the optimization heuristic.

Sulaiman and Arslan [14] presented a Multi Objective Genetic Algorithm (MOGA) for WL and power consumption in a Fast Fourier Transform (FFT) processor. The GA was used to find FFT coefficients which have optimum performance in terms of Signal to Noise Ratio (SNR) and power consumption. The results demonstrate that the GA can find solutions which are optimized for both objectives, but this work does not offer a general optimization method for DSP algorithms.

There are studies, on the other hand, which introduce methods to improve the speed or power consumption of the communication on the shared buses, including bus splitting.

According to Hsieh and Pedram [11], the segmented bus architecture compared to a monolithic bus architecture showed a considerable energy saving . The proposed heuristic used a maximum weight matching algorithm and combinatorial search; however it ignored some HLS parameters and assumed a fixed set of allocated FUs.

Seceleanu et al [13], reported resource allocation on a segmented bus platform in which the optimal solution was formalized as an organizational problem, and where the objective was to minimize the maximal weighted traffic between the system devices. In contrast to our work, they focused on applications of bus partitioning

1-4244-0921-7/07 $25.00 © 2007 IEEE.

in which coarser-grained units are involved and HLS techniques are out of the scope of their work.

Our study offers a combination of the WL optimization method with shared bus partitioning in which the delay cost is reduced considerably and also the number of wires, power consumption and optimization time of the WL optimization are decreased as a side effect.

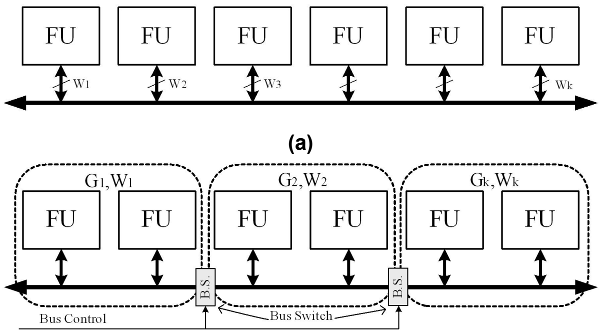
III. IMPLEMENTATION

In this study, a design method is proposed that starts from behavioral specification of the target system and produces a synthesizable Register Transfer Level (RTL) representation. The resulting design is constructed hierarchically using a shared bus with a flexible structure to match a variety of applications. Moreover, it is very modular and manageable for the synthesizer and optimizer [1].

The majority of HLS methods split the target design into two parts: controller and datapath. The controller is a state machine which manages the sequence of operations and controls the datapath blocks and the datapath does the computation.

Normally, the datapath is synthesized using multiplexers and switches for the required interconnections between FUs, registers and other modules. The synthesizer works in a similar manner to a software compiler in which high level specification is translated to a low-level implementation. The basic differences between software and hardware, however; demand more attention to the hardware target. In our method, the pre-defined target architecture fills this gap.

From a synthesis point of view, on the other hand, this target architecture is a restriction in that it forces the synthesizer to map every design to a pre-defined structure which dominates the feasible solution space in favor of the optimization performance. Accordingly, the datapath structure is constrained as depicted in Figure 1. Figure 1-a shows the case of an unpartitioned bus, in which every FU might have a different WL from the others, whereas Figure 1-b presents a set of grouped FUs which are connected to the bus segments. Unlike [12], where FUs are grouped in the final design, here FU grouping and bus partitioning are performed during allocation but before scheduling, which increases the synthesis flexibility and the possibility of the better results. We describe this approach as *Multiple-Width Bus Partitioning* (MWBP).



(a)

(b)

Figure 1. a) FUs with different I/O are connected to a single shared bus b) multiple-width partitioned shared bus

IV. COST FUNCTIONS

The costs of the design can be divided into three parts: those of datapaths; controllers; and interconnections. Since the design space is extended by WL and bus partitioning here, the effect of these two new costs must be evaluated on each part individually. The controller part is not dependent on the WL or system bus partitioning and so it should be considered as a constant value in the cost function but the effect on the two other parts must be investigated. Having focused on WL, it is shown in [3] and [2] that accuracy, area and power consumption costs are dramatically dependent on the WL and execution delay is a function of the WL in the case of sequential units (for example sequential multipliers). Bus partitioning, on the other hand, influences costs by adding bus switches and their control wires. Area, delay and power consumption of the bus switches are straightforward for inclusion in the cost functions and based on the fact that the number of bus switches is neglect able in comparison to the FUs, their effect on interconnection costs is not considered [4]. Therefore, the cost model is as given in Equation (1),

$$F_{Total}(\vec{X}) = F_{Controllers} + F_{Interconc} + F_{Datapath}(\vec{X}), \qquad (1)$$

where F is the cost function and $\vec{X}$ is the set of MWPB parameters. An important point which needs to be reiterated here is that the proposed cost models are functions to evaluate different designs during optimization, which means their ability to map feasible design space individuals into a set of distinct cost values are more important than their precision. In the following subsections, brief descriptions of the cost models are presented. All the relations and values are based on basic cells in the ST 1.2μm technology using the Synopsys tools, more details are provided in [2]

V. OPTIMIZATION ALGORITHM

The implemented synthesizer tool employs an Elite-preserving, Vector Evaluated GA (VEGA) optimization algorithm [9]. The genetic operators are extracted from a standard GA procedure which includes selection by roulette wheel, crossovers and mutation [10] for variable length and integer genomes. The resultant genes represent the number of the groups, FU binding, the bus-width in each group and number of each FU in every group, as shown in Figure (2).

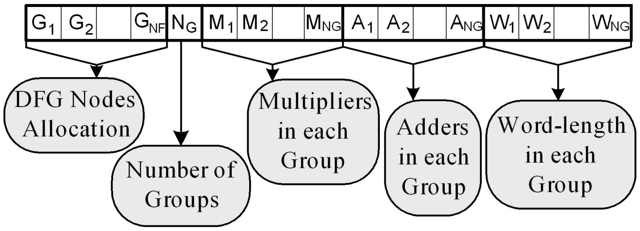


Figure 2. Genomes structure in the applied GA

In Figure (2) the general format of the genes is depicted. Every gene has sections which are (from left to right in Figure (2)): DFG assignment to the groups; number of groups; number of FU0 (multiplier for instance) in each group; number of FU1 (ALU for instance) in each group and so on for other FU types and the last section is the WL of each bus segment. Clearly values in this genome are integers and the gene length is variable as are the minimum and maximum numbers of the FUs. It must be noted that the gene length in Figure (2) might be different for individuals in every generation, (gene-length = $N_{DFG} + 1 + 3N_G$ where N_{DFG} is number of nodes in the DFG and N_G is number of the bus segments).

After assignment of the nodes of the DFG to groups (bus segments) by genes, RC allocation and scheduling [8] is applied to evaluate each individual's fitness. Population size, number of iterations, percentage of crossover and mutation and their probabilities are chosen to achieve the optimal points in the shortest time.

It can be easily proved that the minimum of a linear combination of the basic cost functions is Pareto Optimal [9]. Accordingly, a fitness function of Weighted Tchebycheff method [9] is used with the other basic costs (area, delay, energy and noise) to find the optimal points in the constrained feasible space by an implemented VEGA method.

VI. RESULTS

A number of case studies have been implemented in ST 1.2µm technology using this method. 8-point Digital Cosine Transform (DCT) as, Design I, 5-order Elliptic Filter, as Design II, and an RGB to YCbCr converter, as Design III, are used to illustrate this method of datapath optimization in comparison with other implementation techniques.

Table (1) provides the evaluation results of design optimizations. In this table, four different binding cases are assumed (with 2, 3, 4 and 5 Add/Subtract and Multiplier FUs) for each benchmark. At first, every design is implemented with the assumption of a fixed and uniform WL (W=16) for all its FUs and then their design costs (area, power consumption, delay and variance of digital noise) are calculated as the reference values for optimizations.

In the second step, a general WL optimization method, as introduced in [2], is applied for each design. Clearly, the results of this WL-optimized synthesis suggest reduction in the costs. As it obtainable from table, the computational accuracy most often is traded with improvements in other costs. As discussed in [2], improvements are possible without losing accuracy.

In the third step, the proposed method (MWBP) is used to synthesis the benchmarks. In completion of Table (1), Table (2) presents the MWBP bus partitioning and FU grouping results with their WLs for each bus segment. In comparison with the ordinary WL optimization, MWBP trades average bus width with bus switches. In other words, MWBP uses bus switches to reduce the average wire length in the final implementation of the systems. Since wire costs are not included in the cost evaluation procedure, having sub-optimal results in area and power consumption are expected, as confirmed by the results in Table (1).

On the other hand, however, even regardless of wire cost effects in the final design evaluation; simulation results show that there is a considerable improvement in system latency (14% to 60%) because of the bus partitioning method, which is a valuable achievement.

According to Table (1), MWBP is more effective in the case of more complicated designs, in terms of the number of FU and data communication between them. This means that by increasing the number of FUs in binding of the implemented benchmark, the cost of bus switches reduces as a proportion of the overall cost. Table (2) also supports this premise in which partitioned buses with different WLs are found as the optimal point in the case of designs with bigger number of FUs.

VII. CONCLUSION and REMARKS

This paper presents a methodology for datapath synthesis based on a multiple-width shared bus, and which uses models of power consumption, circuit area delay and output noise and their relationship with the FU grouping, binding, allocation and WL. Examination of the results demonstrates a considerable improvement in design latency cost when this structure is employed for synthesis and optimization instead of generally WL optimization method. In future work, the area and power cost of the interconnections will be addressed and their effects on optimization results will be examined.

Table 1. Optimization results for different number of FUs[1]

		Binding I			Binding II		
		2* , 2+			3* , 3+		
	Costs	NP[2],W=16	NP, optimized[3]	Optimized[4]	NP,W=16	NP, optimized	Optimized
Design I	Area	16608	14907	15344	24912	21529	22610
	Delay	185	169	150	148	136	74
	Noise	3.07E-7	8.87E-6	1.14E-6	3.07E-7	9.55E-6	1.14E-6
	Energy	17957.9	16468.1	14766.2	18089	14633.9	14479.1
Design II	Area	16608	14926	15344	24912	21351	22326
	Delay	115	107	89	107	101	73
	Noise	2.06E-7	6.504E-7	3.03E-7	2.06E-7	9.79E-7	1.00E-6
	Energy	8439.67	6653.97	7198.52	8637.91	6217.07	6782.98
Design III	Area	16608	14532	15344	24912	22586	22610
	Delay	88	80	68	71	65	32
	Noise	6.33E-8	2.53E-7	2.68E-7	6.33E-8	2.38E-7	2.68E-7
	Energy	9686.87	7471.45	7650.02	9784.31	7573.95	7528.61

[1] Costs are in μm^2 for Area, in µWatt/Hz for Energy, digital noise variance in the output (σ^2) for Noise and number of clock cycles for.

[2] NP stands for Non Bus Partitioned

[3] Optimized with different WL for each FU

[4] Bus partitioned and WL optimized, groups binding and WLs are explained in Table 2

Table 1. Continued

		Binding III			Binding IV		
		4* , 4+			5* , 5+		
	Costs	**NP,W=16**	**NP, optimized**	**Optimized**	**NP,W=16**	**NP, optimized**	**Optimized**
Design I	Area	33216	31193	31265	41520	39713	36661
	Delay	138	130	63	132	124	48
	Noise	3.07E-7	6.49E-6	7.63E-7	3.07E-7	9.74E-7	3.96E-6
	Energy	18260.3	17273	15769.6	18485.4	16423	13227.6
Design II	Area	33216	28814	29131	41520	37137	36370
	Delay	101	97	66	101	90	65
	Noise	2.06E-7	2.00E-6	1.09E-6	2.06E-7	1.79E-6	1.02E-6
	Energy	8916.79	6564.84	6844.9	9168.79	6827.91	7151.67
Design III	Area	33216	30496	32361	41520	37690	38589
	Delay	59	55	28	59	52	25
	Noise	6.33E-8	1.65E-7	1.49E-7	6.33E-8	1.87E-7	1.945E-7
	Energy	9801.11	8052.9	8027.14	9959.03	8150.73	8124.72

Table 2. Design configurations after MWBP optimization

		Binding I			Binding II			Binding III			Binding IV		
	Groups	**W**	*	+	**W**	*	+	**W**	*	+	**W**	*	+
Design I	1	14	1	1	14	2	2	14	2	2	14	3	2
	2	14	1	1	14	1	1	14	1	1	13	1	2
	3	-	-	-	-	-	-	15	1	1	13	1	1
Design II	1	14	1	1	14	1	1	13	2	2	13	2	2
	2	14	1	1	13	1	1	14	1	1	15	2	1
	3	-	-	-	13	1	1	13	1	1	13	1	2
Design III	1	14	1	1	14	2	2	15	2	2	14	3	2
	2	14	1	1	14	1	1	14	1	1	15	1	1
	3	-	-	-	-	-	-	15	1	1	14	1	2

REFERENCES

[1] A. Ahmadi and M. Zwolinski, "Area Word-Length trade Off in DSP Algorithm Implementation and Optimization," presented at IEE/EURASIP Conference on DSPenabledRadio, 2005.

[2] A. Ahmadi and M. Zwolinski, "Word-Length Oriented Multiobjective Optimization of Area and Power Consumption in DSP Algorithm Implementation," presented at The International Conference on Microelectronics, 2006.

[3] G. Caffarena, G. A. Constantinides, P. Y. K. Cheung, C. Carreras, and O. Nieto-Taladriz, "Otpimal Combined Word-Length Allocation and Architectural Synthesis of Digital Signal Processing Circuits," *IEEE Transactions on Circuits and Systems II: Express Briefs*, 2005.

[4] P. Christie and D. Stroobandt, "The interpretation and application of Rent's rule," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 8, pp. 639 - 648, 2000.

[5] G. A. Constantinides, "Perturbation analysis for word-length optimization," presented at Proceedings of the 11th Annual IEEE Symposium on Field-Programmable Custom Computing Machines (FCCM'03), 2003.

[6] G. A. Constantinides, P. Y. K. Cheung, and W. Luk, *Synthesis and Optimization of DSP Algorithms (Fundamental Theories of Physics S.)*: Kluwer Academic Publishers, 2004.

[7] G. A. Constantinides, P. Y. K. Cheung, and Wayne Luk, "Optimum and heuristic synthesis of multiple word-length architectures," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 13, pp. 39 - 57, 2005.

[8] G. De Micheli, *Synthesis and Optimization of Digital Circuits*: McGraw-Hill Education, 1994.

[9] K. Deb, *Multi-Objective Optimization Using Evolutionary Algorithms*: John Wiley & Sons, 2001.

[10] D. A. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning* Addison-Wesley Professional 1989.

[11] C. T. Hsieh and M. Pedram, "Architectural Energy Optimization by Bus Splitting," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 21, pp. 408 - 414, 2002.

[12] K.-I. Kum and W. Sung, "Combined word-length optimization and high-level synthesis of digital signal processing systems," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 20, pp. 921 - 930, 2001.

[13] T. Seceleanu, V. Leppanen, J. Suomi, and O. Nevalainen, "Resource allocation methodology for the segmented bus platform," presented at IEEE International SOC Conference, 2005.

[14] N. Sulaiman and T. Arslan, "A Multi-objective Genetic Algorithm for On-chip Real-time Optimisation of Word Length and Power Consumption in a Pipelined FFT Processor targeting a MC-CDMA Receiver," presented at 2005 NASA/DoD Conference on Evolvable Hardware, 2005.