Single-Issue 1500MIPS Embedded DSP with Ultra Compact Codes

Li-Chun Lin, Shih-Hao Ou, *Tay-Jyi Lin*, Siang-Den Deng, and *Chih-Wei Liu*

Department of Electronics Engineering
National Chiao Tung University, Taiwan

***Abstract* –The performance of single-issue RISC cores can be improved significantly with multi-issue architectures (i.e. superscalar or VLIW) by activating the parallel functional units concurrently. However, they suffer high complexity or huge code sizes. In this paper, we borrow some ideas from old vector machines and propose a novel DSP architecture with very compact codes. In our simulations, the DSP has comparable performance to a 5-issue VLIW core with identical computing resources. However, its code sizes are greatly reduced. The DSP core has been implemented in the TSMC 0.13um CMOS technology, where the operating frequency is 305MHz and the core size is 1.45×1.4 mm^2 including 12KB on-chip memory.**

I. Introduction

Today's multimedia applications have grown rapidly in most embedded systems, especially in portable devices. To give consideration to both the flexibility and the performance, the general-purpose single-issue micro-processor, i.e. RISC, is one of the most popularly adopted solutions. However, the single-issue micro-processors can't provide sufficient computing power and its low hardware utilization makes the designers find other ways to improve the performance. Multi-issue architectures, superscalar and VLIW, are commonly used to raise the resource utilization.

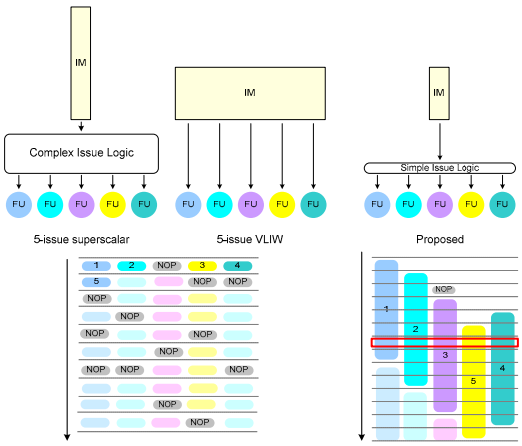

Fig. 1 Illustration of multi-issue and the designed processor

Fig.1 illustrates the configuration and program execution of both the multi-issue processors. Superscalar dynamically schedules multiple instructions to be executed concurrently but is criticized for its dramatically-growing complexity of the issue and decoding logic. VLIW schedules multiple instructions statically at compile time and thus simplifies the design complexity, but it is notorious for its poor code density [1][2]. Both the large issue and decoding logic in superscalar and large code sizes in VLIW imply more power consumption which is undesirable in most embedded systems. In this design paper, we borrow some ideas from the old vector machine, where a single instruction can perform a sequence of equal operations for different data, and then design a compact DSP with improved single-issue logic to address the high design complexity and poor code density incurred by the superscalar and the VLIW processors respectively. As Fig. 1 shows, the DSP core achieves the peak performance as that of the VLIW while keeping the issue logic simple as that of a RISC core. Our implementation results show that the DSP core can achieve 300MHz/1500MIPS peak performance with 38mW average power consumption in the TSMC 0.13 CMOS 1P8M technology. Moreover, the simulation results show that our DSP can have comparable performance to several commercial DSPs but with greatly reduced code sizes.

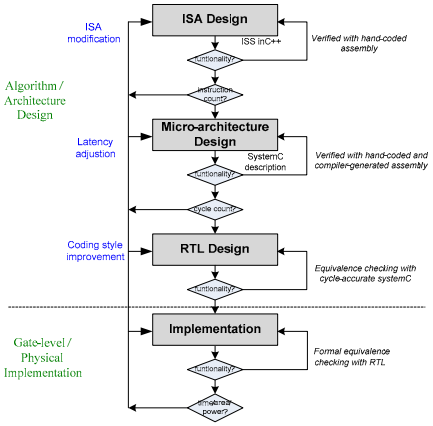

Fig. 2 Design flow

Fig. 2 outlines the design flow of the proposed DSP core. How we design the ISA and micro-architecture of the DSP core are explained in Section II, and the implementation flow with results are shown in Section III. Additionally, the performance evaluation of the proposed DSP is summarized in Section II. Finally, we give a brief conclusion of this paper in Section IV.

II. Processor Design

A. *Instruction-set Architecture Design*

The ISA characterizes the processors and has large impact on the design effort and implementation complexity. In order to retain the generality and simplicity of the proposed DSP core, we first analyze several common DSP algorithms to design the ISA. Additionally, we develop the high abstraction model, i.e. instruction set simulator (ISS), of the designed DSP to verify the hand-coded assembly and to evaluate processor performance. The proposed 16-bit DSP core is a load/store machine with a scalar and a vector register file. The instructions are categorized into three types: (1) scalar, (2) vector-scalar and (3) vector-vector according to whether the operands are from/to the scalar or vector register file. Vector instructions are able to perform specific operation for several times, 8 in our case, while scalar instructions perform single operation. The instruction set covers the arithmetic, the load/store, and program flow control instructions. The arithmetic instructions include most commonly used DSP operations like addition/subtraction, logic operations, comparison and etc. Besides, the full/half precision multiply-and-accumulate (MAC) instructions are also supported to perform 16-bit multiplication and accumulation. A separate 40-bit accumulator is embedded to store the results of the MAC functional unit, The load/store instructions are used to make memory reference and the program control instructions are responsible for program flow control like branch, jump. The lengths of instruction encoding are 16-bit and 32-bit for those without/with immediates.

1-4244-0630-7/07/$20.00 ©2007 IEEE.

B. *Micro-architecture Design & Performance Evaluation*

The micro-architecture design begins after the ISA is finalized. First, three orthogonal design parameters, (1) **functional unit configuration**, (2) **vector length** and (3) **number of vector register file**, are chosen to explore the design space. In our exploration, additional one ALU and one load/store unit are added to a baseline RISC core (1x ALU, 1x load/store unit, and 1x MAC) to increase the IPC (instruction per cycle). On the other hand, the performance gain will be saturated when the number of vector registers and the vector length are great than 8 and 8 respectively. As a result, the designed DSP is finally equipped with **16** 16-bit scalar registers and **8** 128-bit vector registers. Each vector register contains **8** 16-bit elements [3].

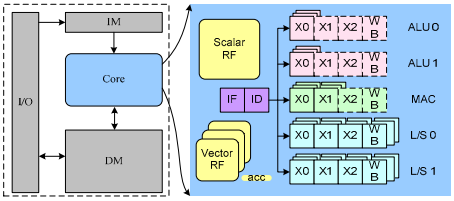

Fig. 3 Core architecture

Secondly, we factorize the micro-operation of instructions and then balance the pipeline stages. The merging and subdividing of operations are performed to achieve the target performance. As shown in Fig. 3, the DSP has 6-stage pipeline similar to the classical scalar RISC. In addition, the datapath of this DSP core is almost the same as the conventional single-issue RISC. The only slight difference lies in the instruction dispatcher. A minor modification for the instruction dispatcher of the conventional RISC enables decoding of vector instructions. Once the vector instruction is decoded, it activates the corresponding functional unit for several cycles without any additional instruction fed. Therefore, 5 FUs of the DSP core can concurrently work like multi-issue processors while the design complexity is greatly reduced.

TABLE I. EXPERIMENTAL RESULTS

	TI C55x [4]		Proposed DSP	
	Cycle counts	Code sizes	Cycle counts	Code sizes
FIR filtering	$N*(T+2)/2+18$	99	$\mathbf{N*(T+9)+6}$	**54**
Vector add.	$N+7$	66	$\mathbf{3*N/2+5}$	**40**
Vector max.	$N/2+15$	99	$\mathbf{N+15}$	**56**
8-point 1D DCT	67	94	**82**	**50**

*code sizes in bytes

We have hand-coded several DSP algorithms on our DSP, including *N*-sample/*T*-tap FIR filtering, *N*-element vector addition and maximum, and 8-point discrete cosine transform (DCT). The cycle counts and code sizes are listed in Table 1, where the performance of TI's C'55x DSP is included for reference. Note that C'55 has two MAC units and thus has better performance. However, the proposed vector issue logic can have linear performance gain once incorporating MMX-like SIMD parallel functional units, and our DSP will outperform TI's C'55x DSP with identical code sizes. When compared to a 5-issue VLIW processor the code size can be even reduced by a factor of 8 [3].

III. Silicon Implementation

A. *Modular RTL Design*

The synthesizable RTL design is conducted right after the ISA/micro-architecture design. We make the one-to-one mapping from the modular micro-architecture to the RTL. Several DSP kernels serve as test-bench to verify the linted and fully synthesizable RTL codes with the SystemC model by HDL simulator ***Verilog-XL***. We further account for the corner cases by handcrafting assembly codes to increase the code coverage. The resulting statement coverage, branch coverage, state coverage and arc coverage are all 100%.

B. *Incremental Synthesis & Physical Implementation*

The implementation phase involves the RTL incremental synthesis and the physical design by means of the ***Synopsys Design Compiler*** and the ***Cadence SoC Encounter*** respectively. The functional verification of the synthesized gate-level net-list is through the formal equivalence checking with the verified RTL model. Besides, the timing/area/power consumption estimation is made by the popular estimation tools like ***PrimeTime***, ***PrimePower***, ***nanosim***. We also implement the mechanism for chip testability. If the performance/cost specification is not met, our solution is going back to micro-architecture design to refine it or back to RTL design to improve code quality as shown in Fig. 2.

C. *Results*

The proposed DSP core has been implemented on silicon with TSMC 0.13µm CMOS 1P8M technology. Table II summarizes the chip specification and Fig. 4 shows the chip layout. The DSP core can operate at 305 MHz while consuming only 38mW average power, and the core size is 1.45 x 1.4mm^2.

TABLE II. CHIP SEPCIFICATION

Technology	**TSMC 0.13um 1P8M CMOS**
Core size	1.45 x 1.4 mm^2
On-chip memory size	8KB DM & 4KB IM
Transistor/Gate Count	269,739
Max. frequency	305 MHz
Power dissipation	38 mW

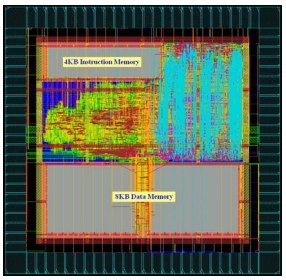

Fig. 4 Chip layout

IV. Conclusion

The paper summarizes the design and implementation of a single-issue DSP core with compact codes. We start from the definition of ISA design, the exploration of micro-architecture to the synthesizable RTL implementation. In our simulations, the designed DSP core has comparable performance to a 5-issue VLIW with similar resources but definitely reduced code sizes. Its silicon implementation in the TSMC 0.13µm CMOS technology achieves 305MHz operating frequency while consuming only 38mW. We are going to utilize some techniques such as banking to address the complex vector register file. Besides, the spatial vectorization, i.e. SIMD, will also be investigated in the future.

References

[1] J. L. Hennessy and D. A. Patterson, *Computer Architecture – A Quantitative Approach*, 3rd Ed., Morgan Kaufmann, 2002

[2] C. Kozyrakis, and D. Patterson, "Vector versus superscalar and VLIW architectures for embedded multimedia benchmarks," in *Proc. Micro*, 2002, pp. 283-293

[3] Shih-Hao Ou and et al., "A compact 300MHZ/1500MIPS DSP with improved single-issue unit," in *Proc. VLSI Design/CAD Symposium*, 2006

[4] TMS320C55x DSP Generation. [Online]. Available: http://www.ti.com