PAPER • OPEN ACCESS

Design of a dual-issue RISC-V processor

To cite this article: Zhang Hongsheng et al 2020 J. Phys.: Conf. Ser. 1693 012192

View the article online for updates and enhancements.

You may also like

- <u>Thioxanthone-containing blue thermally</u> activated delayed fluorescent emitter Natsuko Kanno, Yongxia Ren, Yu Kusakabe et al.
- <u>Thermally activated delayed fluorescence</u> <u>materials for organic light-emitting diodes</u> Xiaoning Li, Shiyao Fu, Yujun Xie et al.
- Implementation of a 32 bit RISC processor with floating point unit in FPGA platform

S Sushma, Smruthi Koushika Ravindran, Pavan Rajendar Nadagoudar et al.





DISCOVER how sustainability intersects with electrochemistry & solid state science research



This content was downloaded from IP address 3.133.160.156 on 28/04/2024 at 01:38

Design of a dual-issue RISC-V processor

Hongsheng, Zhang, Zekun Jiang^a, Yong Li

College of Electronic Engineering, Chongqing University of Posts and Telecommunications, Chongqing 400065, China

1693 (2020) 012192

jzk61010@hotmail.com, ^a2014214318@stu.cqupt.edu.cn

Abstract. A dual-issue 32-bit RISC-V processor is designed and reported. In order to evaluate the performance of the dual-issue processor, a single-issue processor based on the open source RISC-V instruction set architecture is first designed for reference and it is also the base of the dual-issue processor. The single-issue reference processor, which has a 5-stage pipeline, supports base integer instruction set, integer multiplication and division and compressed instructions, has passed the corresponding functional and compliance tests. The dual-issue processor extends the pipeline to dual-issue and introduces additional processing to solve data hazards in the pipeline. The evaluated result of the dual-issue processor shows that it has significant performance improvement than the single-issue processor.

1. Introduction

Designed by University of California at Berkeley, RISC-V is an open source instruction set architecture (ISA) for various application scenarios[1]. Since its release in 2014, RISC-V has received the attention and participation of many well-known universities and research institutions, and the ecosystem of RISC-V has gradually improved[2-3]. But we are aware that most RISC-V cores like Rocket Core, CAV6 and CV32E40P are designed with a single-issue architecture[4-6]. In this work, a single-issue core is designed first and then the core to is extended dual-issue for exploring the pipeline extension process and the performance difference between the single-issue and dual-issue processor.

2. Design of the Processor

2.1. The Single-Issue Processor

First, an in-order 5-stage pipeline single-issue processor is designed with the support of RV32IMC. The architecture of the processor is shown in Fig.1.



Fig.1 The architecture of the single-issue processor



Content from this work may be used under the terms of the Creative Commons Attribution 3.0 licence. Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.

The 5-stage pipeline in the processor is divided into instruction fetch (IF) stage, instruction decode (ID) stage, execute (EX) stage, memory access (Mem) stage and write back (WB) stage. In addition, the processor also has control unit and control and status register (CSR).

The processor is a 32-bit single-issue processor, which means that the maximum length of the data the IF stage should fetch from I-cache is 32-bit. In this design, the IF stage keeps fetching 32-bit data from I-cache to simplify the logic. The ID stage implements a 16-bit decoder and a 32-bit decoder. The two decoders work parallelly so that the decoding process can be done within one cycle. The ID stage also gives the IF stage a feedback to control the next instruction's start address. When a 16-bit instruction is detected in ID, the next program counter (PC) should be plus 2, and a flush signal will be triggered to clear the default PC value. When a 32-bit instruction appears, the PC should be plus 4. The ID stage also has a data hazard unit connected to the next three stages to eliminate data hazards in the pipeline. The EX stage implements the operation required by the RV32I and M instruction extension. The Mem stage connects to the cache to execute memory access instructions. The cache that the Mem stage connects is a memory model and it is not synthesizable. The Wb stage connects to the Regfile for writing back the data. The control unit monitors the pipeline and stalls or (and) flushes the pipeline if it is needed. The CSR unit monitors the exception and traps in the pipeline. And the registers that machine-level ISA requires are implemented in the CSR.

2.2. The extension of the Single-Issue Processor

Then the single-issue core to is extended a dual-issue core. The architecture of the extended processor is shown as Fig.2. The processor keeps the original 32-bit, 5-stage pipeline. And the function of each stage is also the same as the single-issue one. The original modules are reused as much as possible to reduce the extension cost and save the time for expansion.



Fig.2 The architecture of the single-issue processor

2.3. Changes in the Dual-Issue Processor

(1) IF and ID Stage

Compared with the single-issue processor, the length of data the IF stage fetches is doubled to 64 bits. With the extended processor supporting mixed 16/32-bit instruction decoding, IF stage may encounter 16/16-bit instructions, 16/32-bit instructions, 32/16-bit instructions, and 32/32-bit instructions when fetching instructions. The four situations correspond to PC+4/+6/+6/+8. Another improvement is that the original core needs a flush when a 16-bit instruction is detected but the dual-issue core do not require flushing the pipeline.

The ID stage is shown as Fig.3, has tripled in scale. For realizing the decoding of 16/32-bit mixed instructions, three 16-bit instruction decoders and 32-bit instruction decoders are implemented in this design. The three 16-bit instruction decoders read bits 0-15, 16-31 and 32-47 of the 64-bit instruction

sent by ID stage, and the three 32-bit instruction decoders read bits 0-31, 16-47 and 32-63. All the decoders are hard-wired to realize fast 16/32-bit mixed instruction decoding with simple selection logic.



Fig.3 ID stage of the dual-issue processor

(2) EX Stage

The EX stage is shown as Fig.4. The number of execution unit (EXU) in the EX stage is increased to two. Due to the increase of the EXUs in the EX stage, data hazard in the stage will occur. To solve the data hazards, a data hazard detection unit is introduced in the EX stage, which can lead the output of the EXU 0 as the input of the EXU 1. The data hazard detection unit has a negative impact on the timing performance, but the impact is not great.



Fig.4 EX stage of the dual-issue processor

(3) Data Hazards in the Dual-Issue Pipeline

The data hazard unit in the ID stage is designed for dealing with the read after write (RAW) data hazard. It receives the current instruction's source register's data and address information from the decoding unit, and it also receives the target register's address and data information from the EX stage, the Mem stage, and the Wb stage. When the address of the source register in the ID stage is the same as the destination register address of the instruction in the next three stages, it is proved that a RAW data hazard in the pipeline has occurred. The data in the latter three stations will be fetched to the ID stage and replaces the origin data read from the Regfile.

The data hazard unit located in the ID stage cannot handle if the data hazard appears in one stage, because data hazard unit cannot get the instruction's result in the ID stage. For this reason, an extra data hazard unit is added in the EX stage to control the second ALU's operand. But if load is the first instruction in EX stage, both data hazard unit cannot get the result until the Mem stage reads from the cache. In this situation, there will be an instruction separation which is described below.

(4) The instruction separation

In this design, the back end can only process one instruction under certain circumstances, so it needs to match the speed of the front end. The speed of the front end in this processor is always greater than or equal to the speed of the back end. Therefore, for saving area, it is not necessary to implement a buffer in front of the EX stage, separating the two issued instructions before sending to the EX stage is a better way. In this design, the instructions are judged in the decoding unit, and an additional control signal is added. The registers between ID and EX stage performs the separate operation of the two instructions, and the two instructions are sent separately with nop. The waveform is shown as the Fig.5.



3. The analysis

3.1. Functional Verification

The design uses Icarus Verilog to verify the correctness by running the official test program. Each test program contains a number of functions for the test instruction. If all the functions are running correctly, it will jump to the pass function, and if an error occurs in any test function, it will jump to the fail function. Therefore, the assertion can be used to determine whether the program is executed correctly and finally jump to the pass function section correctly. Fig.6 shows part of the functional test results, and the left side is the result of the single-issue processor, the right side is the dual-issue processor's result. Both the processors can pass the tests, and the dual-issue processor is noticeably faster.

3.2. The Performance

The total cycles for the two processors to run the test program are summed up, and the result is shown as Fig.7. The result shows that the dual-issue processor is about 80% faster than the single-issue processor. However, it should be noted that the cache in this design is a behavioral model, so the negative impact on performance caused by the cache read time and cache misses in the memory access instruction is ignored in the simulation results. In actual situations, due to these negative effects and the existence of branch instructions, the actual performance improvement will be lower.

3.3. The Synthesis

Both the processors are synthesized by Synopsys Design Compiler with a 0.13um library. The cache is removed before the synthesis. The result is shown as Fig.8. The maximum clock speed of the dual-issue core is 17% slower than the single-issue core, which is shown as Fig.8(a). The power is about the same

1693 (2020) 012192 doi:10.1088/1742-6596/1693/1/012192

between the two processors and the result is shown as Fig.8(b). And the area of the dual-issue processor is about 58% larger which is shown as Fig.8(c).

zenk > single_NF5 > NF5 > output > Report > 🧧 Report_tunct.txt						> Report > 🗉 Report_funct.txt					
rv32ui-p-andi.log						rv32ui-p-andi.log					
ISA_test	Pass	when	clk=	178,	PC=000002d8	ISA_test	Pass	when	clk=	93,	PC=000002d8
rv32ui-p-auipc.log						rv32ui-p-auipc.log					
ISA_test	Pass	when	clk=	31,	PC=00000160	ISA_test	Pass	when	clk=	17,	PC=00000160
rv32ui-p-beq.log						rv32ui-p-beq.log					
ISA_test	Pass	when	clk=	311,	PC=000003d8	ISA_test	Pass	when	clk=	165,	PC=000003d8
rv32ui-p-bge.log						rv32ui-p-bge.log					
ISA_test	Pass	when	clk=	347,	PC=00000438	ISA_test	Pass	when	clk=	186,	PC=00000438
rv32ui-p-bgeu.log						rv32ui-p-bgeu.log					
ISA_test	Pass	when	clk=	372,	PC=0000046c	ISA_test	Pass	when	clk=	199,	PC=0000046c
rv32ui-p-blt.log						rv32ui-p-blt.log					
ISA_test	Pass	when	clk=	311,	PC=00003d8	ISA_test	Pass	when	clk=	165,	PC=000003d8
rv32ui-p-bltu.log						rv32ui-p-bltu.log					
ISA_test	Pass	when	clk=	336,	PC=0000040c	ISA_test	Pass	when	clk=	178,	PC=0000040c
rv32ui-p-bne.log						rv32ui-p-bne.log					
ISA_test	Pass	when	clk=	315,	PC=00003dc	ISA_test	Pass	when	clk=	168,	PC=000003dc
rv32ui-p-fence_i.log						rv32ui-p-fence_i.log					
ISA_test	Pass	when	clk=	464,	PC=000001f4	ISA_test	Pass	when	clk=	239,	PC=000001f4
rv32ui-p-jal.log						rv32ui-p-jal.log					
ISA_test	Pass	when	clk=	27,	PC=0000016c	ISA_test	Pass	when	clk=	16,	PC=0000016c
rv32ui-p-jalr.log						rv32ui-p-jalr.log					
ISA_test	Pass	when	clk=	98,	PC=000001ec	ISA_test	Pass	when	clk=	55,	PC=000001ec
rv32ui-p-lb.log						rv32ui-p-lb.log					
ISA_test	Pass	when	clk=	228,	PC=00000364	ISA_test	Pass	when	clk=	143,	PC=00000364
Fig. 6 Test result of the two processor											
			-	-5.0	i est i estant o						



Fig.7 Cycles for running test programs of the two processor



1693 (2020) 012192 doi:10.1088/1742-6596/1693/1/012192



Fig.8 The result generated by Synopsys Design Compiler

4. Conclusion

In this work, a single-issue processor is designed, and then it is extended to a dual-issue processor. The extended processor benefits from being able to process more instructions at the same time, and its performance has been significantly improved. However, the negative impact of the memory system on performance is ignored in this article. If there is a memory-bound bottleneck, the performance gains brought by dual-issue may be significantly reduced. In the process of extension, although the expanded processor is still a in-order processor, the most noteworthy thing in the extension process is still the problem of data hazards and instruction conflict in the pipeline. The processor in this design can further improve performance by implementing dynamic branch prediction, expanding the back-end, and supporting out-of-order execution.

References

- Waterman, A., Lee, Y., Patterson, D.A., Asanovi, K.The RISC-V instruction set manual, volume I: user-level ISA, version 2.1[C].Technical report UCB/EECS-2016-118, EECS Department, University of California, Berkeley, May 2016.
- [2] Hruska, Joel.RISC rides again: New RISC-V architecture hopes to battle ARM and x86 by being totally open source[J]. ExtremeTech.com,2014.
- [3] Silei Lei.A Survey of RISC-V Architecture Open Source Processors and SoC[J]. Microcontrollers & Embedded Systems, 2017,17(02):56-60+76(in Chinese).
- [4] Krste Asanović, Rimas Avižienis, et al. The Rocket Chip Generator, Technical Report UCB/EECS-2016-17, EECS Department, University of California, Berkeley, April 2016.
- [5] F. Zaruba and L. Benini.The Cost of Application-Class Processing: Energy and Performance Analysis of a Linux-Ready 1.7-GHz 64-Bit RISC-V Core in 22-nm FDSOI Technology[J]. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 27, no. 11, pp. 2629-2640, Nov. 2019.
- [6] D. Rossi, F. Conti, et al.Pulp: A parallel ultra low power platform for next generation iot applications[J].2015 IEEE Hot Chips 27 Symposium (HCS), pp. 1-39, Aug 2015.