Integrated Coursework for Computer Architecture and Digital Design using Field Programmable Gate Arrays and HDL

Kang Yi

School of Computer Science and Electrical Engineering,
Handong Global University, 791-708, Korea

yk@handong.edu

Abstract

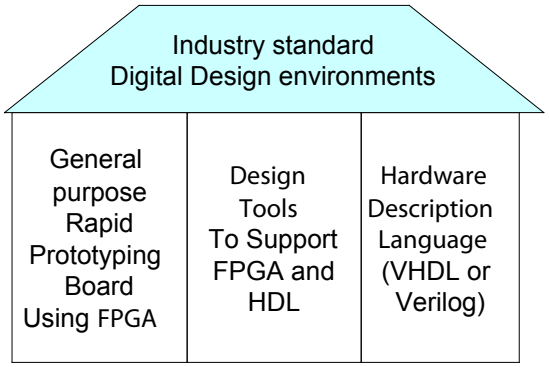
Field Programmable gate arrays(FPGA) are very economical and efficient tool to implement any digital hardware. Hardware Description Language(HDL) such as VHDL is also very efficient tool to describe complex digital circuits. The good news is the simulation synthesis software of industry standard is available on the internet at free of charge. In this paper we describe a course which integrates the learning experience of digital logic, computer architecture, digital system design, and embedded system application. Students should design their own instruction set architecture for RISC machine, design the proposed computer into digital hardware and finally demonstrate its working function using FPGA rapid prototyping board with a sorting program written in assembly language developed by the students. The author shares the teaching experience of this challenging coursework for ten years. In this paper the author also shares some trial-and-error experience in the past and current updates for teaching infrastructure and know-hows.

1. Introduction

Digital design subject is one of the most important topics in the modern CE (computer engineering) and EE (electronic engineering) major curriculum. This subject is covered by several courses such as logic design, computer organization and architecture, digital system design, embedded system design, and microprocessor design and applications. Since digital design subject requires less mathematics background than other EE subjects and gives students more tangible results at the earlier stage of EE study, it helps many students to find their interests in their major in the sophomore or junior level. Hands-on Experience of this subject is one of the key elements that attracts students as well as improves deep understanding of the course contents.

Modern digital circuit design heavily depends on the hardware description language such as VHDL, FPGA implementation, and their related design automation tools. HDL-based design method gives high flexibility and high productivity for the design job since it supports higher level abstraction of circuits behavior. FPGA (Field Programmable Gate Array) based design is another modern digital design trends. It provides implementation and verification platform for high complex digital system at relatively low cost. Students can implement their digital circuit design quickly by FPGA-based general purpose rapid prototyping board without any time consuming job for wiring components on breadboard. Many quality tools to support HDL-based design environments targeting FPGA implementation make the digital design job easy and shorten the design time. In addition to the modeling and simulation features of the design tools an automatic synthesis feature of the tools translates the verified design described in HDL into the circuits described at gate-level net-list and finally target technology (FPGA) dependent implementation information. Thus, HDL + FPGA + HDL-based design automation tools makes the complete and convenient design environments for digital design course hands-on experiences as shown in Fig 1.

Figure 1. Current Digital Design Environments

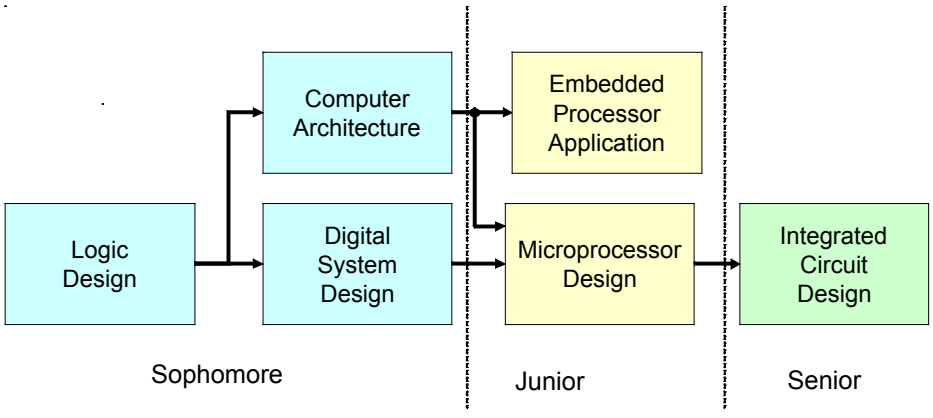


The good news is that FPGA companies such as Xilinx, Altera, and Actel give the industry standard HDL-based digital design tools at free of charge for education purpose. These tools provide wonderful opportunities for students to experience industry standard hardware design environments. According to other research [5,6] the tightly coupled laboratory environments of FPGA and HDL based digital design flow give the students good learning experience and results in high retention rate. And, many textbooks are developed for such education environments [1].

Since the year 1998 School of CSEE (Computer Science and Electrical Engineering) of HGU (Handong Global University) provides FPGA based digital design environments at various courses including logic design lab and computer design, and digital system design. For 10 years the school of CSEE of HGU has developed a series of courses for digital design using FPGA and HDL-based design method as shown in Fiure 2.

On the other hand, computer architecture courses are usually taught at conceptual level. In order to give students more concrete and firm understanding of computer organization the computer processor hardware design experience is definitely desirable. Though many computer architecture and organization textbooks [1,7] are written with some specific processor design examples, limited time of the computer architecture course for a semester period does not allow students to have a serious hardware design experience. In some computer architecture courses just HDL model simulation of CPU hardware is provided to students while in other courses hardware step by step execution experiments with pre-made board for a specific machine architecture is provided to students [8].

Figure 2. Digital Design Course Tree of School of CSEE of HGU



The school of CSEE of HGU has set up a separate course named microprocessor design to fill this gap between computer architecture course and real digital design experience. The microprocessor design course integrates the computer architecture knowledge and digital design experience especially focusing on the pipelined RISC processor design and implementation using HDL-based design methodology tightly coupled with FPGA implantation environments.

In this paper, the author presents the microprocessor course contents including the design experiments set up en-

vironments. The author summaries the history of trial and errors of running the course and describes the details of current microprocessor design course contents. In the last section evaluation and analysis of the course outcome is presented with some remarks.

2. Course History and Contents

The microprocessor design course title has been changed two times in the past: computer design(~ 2004), embedded processor design(2005~2007), and microprocessor design (2008 ~). The target processor for students to design has been changed two times. This course contents development period can be divided into three according to the requested final project target processor instruction set architecture and course operation principles.

At the early stage of the course development around in 1999 – 2002, the course name was computer design. A simple yet economic FPGA boards mounting Xilinx XC4K family were used. These boards and Xilinx Foundation design tool suits were donated to universities by Xilinx for education purpose. At this first stage, students are taught VHDL language and requested to design an 8-bit CPU. An instruction set architecture is given as shown in Figure 3 and Table 1. The internal CPU organization guideline (Figure 4) was also given to students. Students should implement the CPU using VHDL and demonstrate the function by a series of instruction execution on the real FPGA board as well as simulation. Students can confirm their understanding of computer through the hardware design experience. But, the limitations of this design experience are that processor ISA designed was far from current RISC architecture and the processor ISA had no direct relation with the popular computer architecture textbook of [7].

Figure 3. Instruction Format for the CPU Design Projects (1st stage of the course)

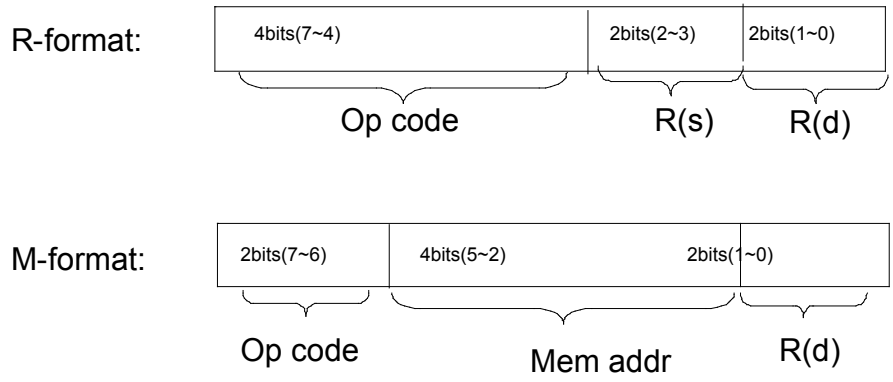


Figure 4. The given organization for the CPU (1st stage of course development)

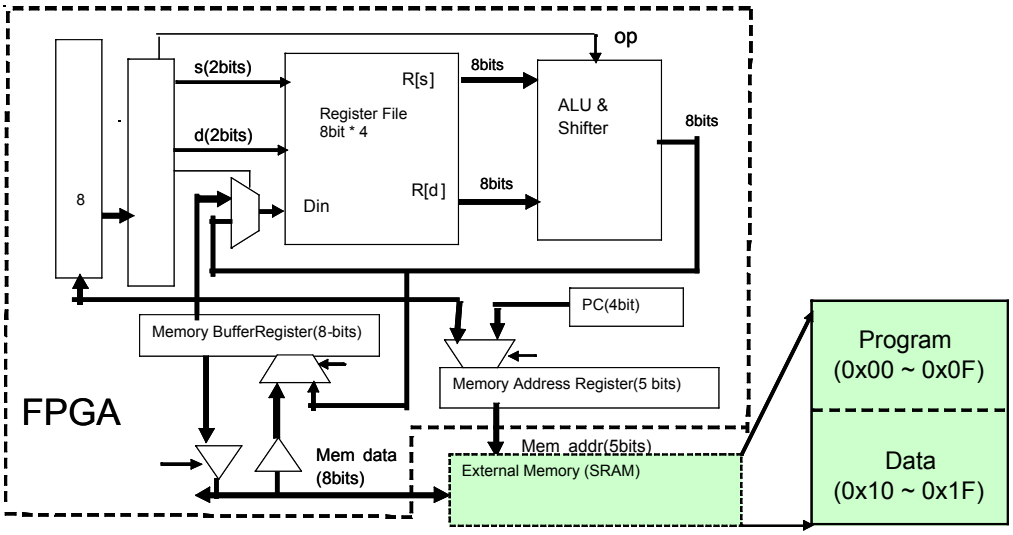


Table 1. Instruction Set for the CPU Design Projects (1st stage of the course)

Op-code	Meaning (RTL)	Instruction Name	Format
00	R[d] ← MEM ['1' & addr]	Load	M
01	MEM['1' & addr] ← R[d]	Store	M
1000	R[d] ← R[d] + R[s]	Addition	R
1001	R[d] ← R[d] – R[s]	Subtract	R
1010	R[d] ← R[d] AND R[s]	AND	R
1011	R[d] ← R[d] OR R[s]	OR	R
1100	R[d] ← R[d] XOR R[s]	XOR	R
1101	R[d] ← R[s] << 1	Shift Right	R
1110	R[d] ← R[s] >> 1	Shift Left	R
1111	HALT	Halt system	R

At the second stage of the course round in 2003 – 2007, a new FPGA boards (MP3100X) from Seloco [10] and Xilinx Webpack including ModelSim XE were used together. Xilinx Spartan2 FPGA was mounted on the FPGA board and its capacity was enough to implement 100K gates digital systems including 32-bit simple RISC machines. Students are given the 8-bit microprocessor ISA and its sample design and VHDL synthesizable code as a reference design. Students should design full pipelined 32-bit MIPS instruction set architecture with VHDL and implement their design on the FPGA board. But, they do not need to implement all the instructions. Only 12 basic instructions are required and additional points are given for additional implementation as shown in Table 2. Students should demonstrate the function and performance by a sorting program written in MIPS assembly language. Students are also given MIPS assembler and code converter into the data format for the download into the on-board SRAM memory as shown in Figure 5 and Table 2. If a pipeline hazards are completely solved by hardware approach additional points are given. Programs and data are stored in a SRAM and the execution results can be monitored by uploading the SRAM contents into computer file as shown in Figure 6.

Figure 5. MIPS instruction set format

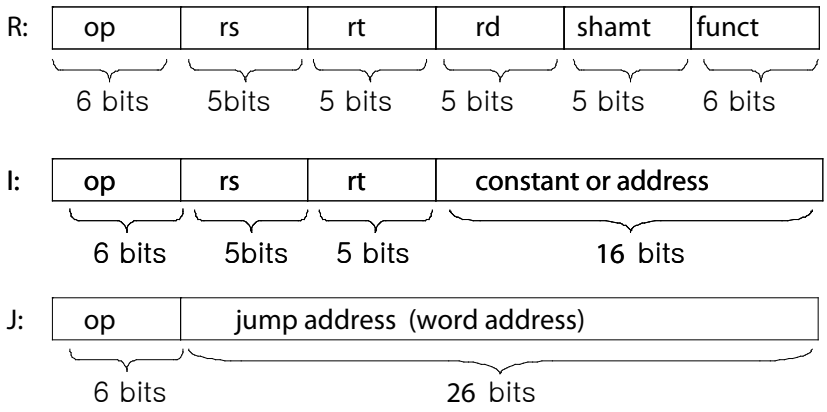


Figure 5. MIPS instruction set format

instructions	remarks		instructions	remarks
add rd, rs,rt	Add		beq rs, rt, imm16	Branch if equal
addi rt, rs, imm16	Add immediate		j imm26	Jump
and rd, rs, rt	And		lw rt, imm16(rs)	Load
slt rd, rs, rt	Set less than		sw rt, imm16(rs)	Store
nop	No-operation		Jr rs	Jump register

Figure 6. MIPS Assembler for programming and format conversion into memory file

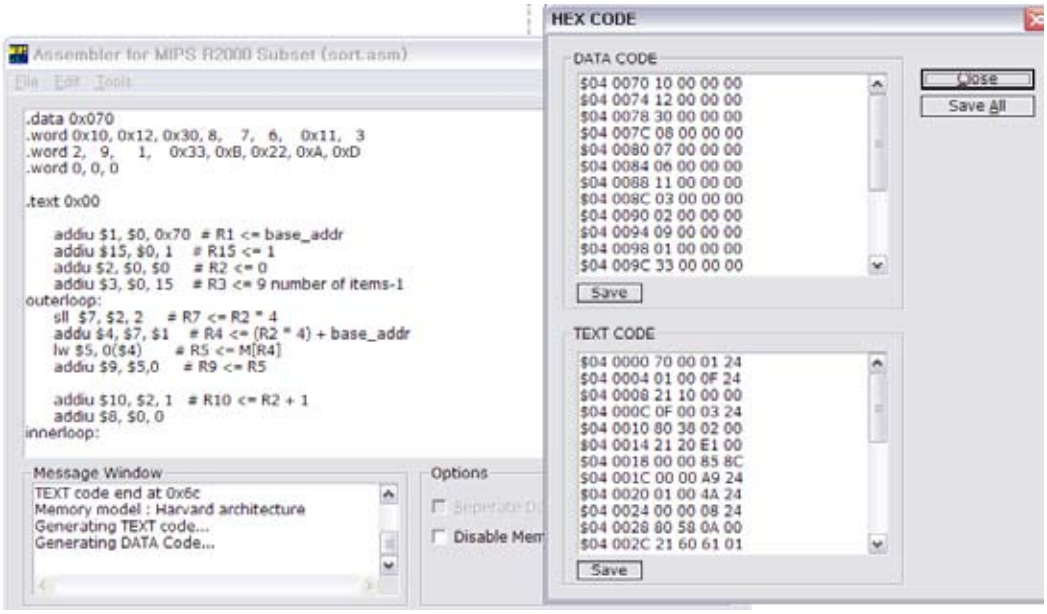
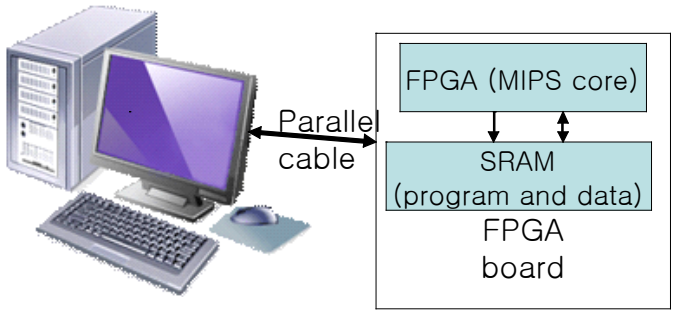


Figure 7. FPGA board and computer connection for the Lab



But, there is a missing point in this design experience. That missing part is students have no chance to see and understand other processor architecture such as x86 and a processor design from [11]. In order to mitigate such a problem, students were requested to study and present other processors' ISA such as SPARC, Pentium PRO and x86, PowerPC, old machine architecture like VAX. But, it took too much time for students to present with power-point file and there were too much wrong information in the presented contents confusing the audience students because of unprepared presentation and unclear understanding. The presenters should have been trained for formal verbal presentation skills and reading understanding of official data documents. Because of this problem ISA case study was regarded as not proper for this course objects within a limited time and resources.

At the 3rd stage of the course development, many changes are introduced. Students are requested to design their

own instruction set architecture instead of designing a given instruction set architecture such as MIPS. And, design is performed by teams instead of individual. The constraint is that the ISA should be a 16-bits RISC processor and the proposed ISA should be complete and minimal to fit in given FPGA chip on the MP3100 training board. In order to prove their completeness each team has to show working assembly programs corresponding to a couple of given C code programs. Final demonstration of hardware execution using FPGA board should cover a set of missions such as finding the average value of N numbers stored in main memory, sorting numbers in ascending orders, moving data from a memory location to a specific memory location, and arithmetic operation on big numbers whose size if more than 32 bits. The figure 8 and Table 3 shows one of the instruction set architectures proposed by students in 2008 microprocessor design class. The ISA has 14 instructions and 3 information formats. The student's datapath design for this processor architecture is shown in Figure 9 which is similar to that in the textbook of [7]. All the teams should write their own program in machine language or assembly language for demonstration. Since every processor has different instruction set architecture students should design their own assembler if it is needed. Finally, all the team has to present their instruction architecture and highlight the special feature of their design verbally and by written documents.

Figure 8 16-bit RISC Instruction format of students' design for final project

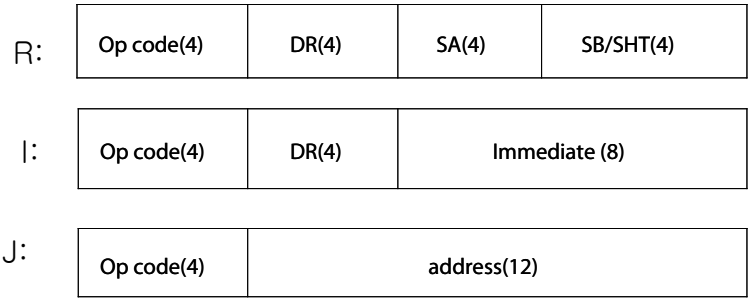
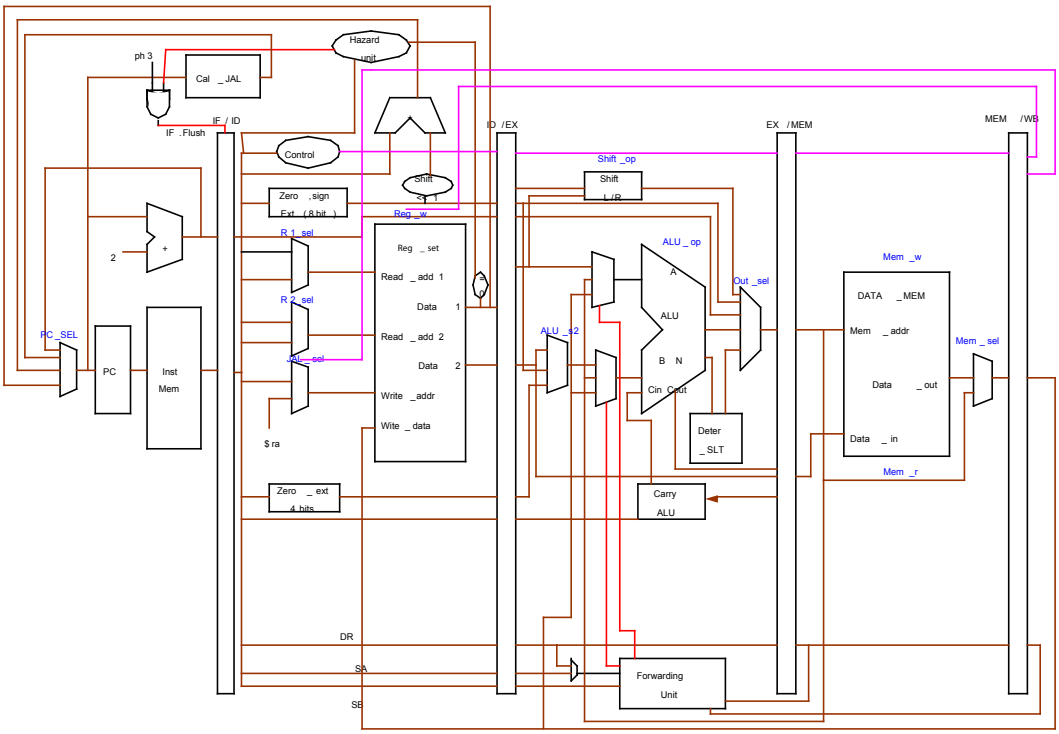


Table 3. Instruction Set for Figure 8

Instruction	Format		Instruction	format
ADD DR, SA, SB	R		JAL imm12	J
SUB DR, SA, SB	R		LD DR, SA, imm4	R
LSR DR, SA, SHT	R		ST DR, SA, imm4	R
LSL DR, SA, SHT	R		LDI DR, imm8	I
NAND DR, SA, SB	R		ADI DR, imm8	I
SLT DR, SA, SB	R		AUI DR, imm8	I
JR SA	R		BNZ DR, imm8	I

Figure 9. the datapath for the processor described by Figure 8 and Table 3.

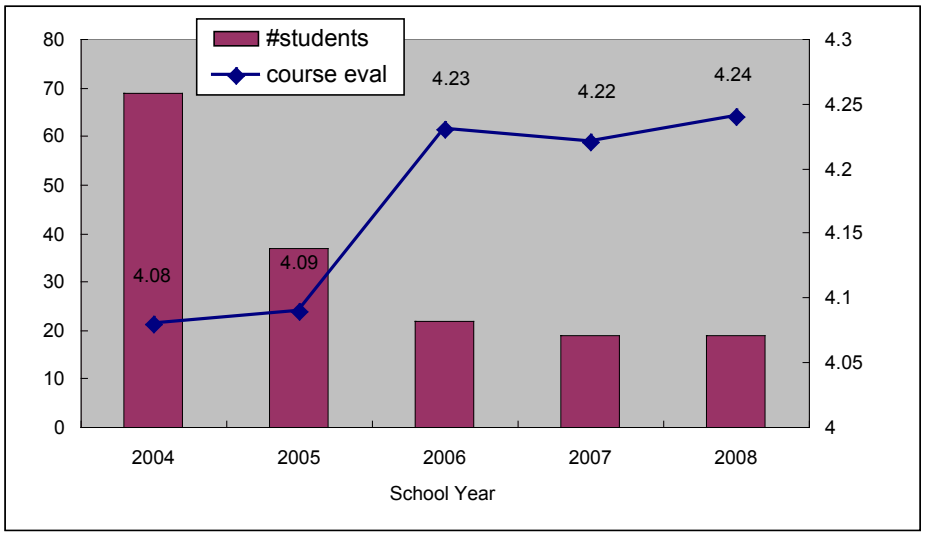


3. Course Evaluation & Discussion

Two types of assessment are used here to estimate the effectiveness of the microprocessor design course remodeling efforts. One is the students' response to evaluate the overall course quality. At the end of every semester web-based survey is performed for each course by university's office of academic affairs anonymously. The other type of assessment is the ratio of the number of students who successfully completed the final processor design project over the total number of registered students for the course.

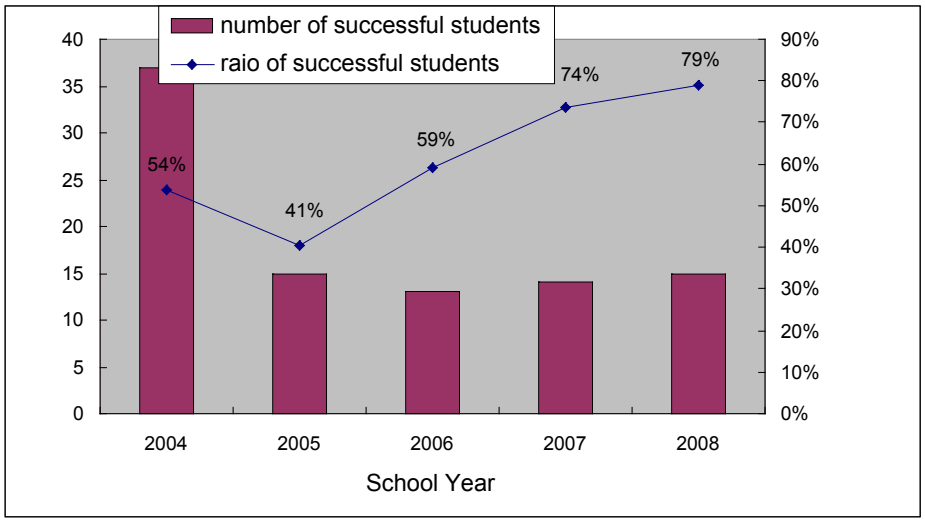
The Figure 10 shows the course evaluation response from the students for the question "Overall, was the course beneficial?" The author analyzed the course evaluation data from 2004 to 2008 because only those data is available from the school. In the figure the course evaluation points increases as the school year changes from 2004 to 2008. In 2004 the evaluation average score is 4.08 while in 2008 the evaluation average score is 4.24 resulting in 4% improvement over 4 past years. But, note the number of total registered students for the course decreased from 69 in year 2004 to 22 in year 2008. Thus, it is not clear whether the course evaluation improvement is the result from course itself enhancement or small-sized class effects. Thus, the 2nd type of assessment is performed. Figure 11 shows the ratio of number of students successfully demonstrated their performance in the class by final projects over the total number of registered students for the microprocessor design course. It shows a consistent improvement of student's design performance since year 2005. About 47% improvements are observed in year 2008, compared to the ratio of successfully completed students year 2004. The dramatic improvements of the number of students who successfully completed the final design projects can be considered as the proof of consistent course improvement. Therefore, the direction of the course enhancement from design problems for a given ISA to design problem for their own processor ISA is considered to be a proper and right decision.

Figure 10. The course evaluation from students and the number of registered students



Therefore, we can infer the course enhancement direction is right in terms of the number of students successfully completing projects. The author assumes the reason for the success of chance even more challenging subjects is that enhanced students' understanding and motivation level due to the designing their own proposed ISA design and implementation results in more chance of successful completion.

Figure 11. The ratio of students successfully completing their final design project



4. Conclusion

This paper describes an integrated course for computer architecture knowledge and digital system design experience. The course was developed and changed constantly towards more challenging topics and more area of topics. In the beginning it just covers HDL-based digital design and simple 8 bit processor design, changing the focus on MIPS processor design and implementation, and finally shifting the stress to the ISA design and its hardware representation plus software design for language translation. The course assessments show a kind of constant improvements of the course operation in terms of student's responses and ratio of successfully finishing students.

References

01. Roth, C. and L. John, “Digital System Design Using VHDL”, 2nd Ed, Thompson, 2008.
02. http://xilinx.com, Xilinx ISE 10.1Software Manual, Xilinx Inc., 2008.
03. http://mycad.com, MP3100X User Guide, Seloco Inc.
04. http://xilinx.com , Modelsim XE, Memtor Graphics Corp.
05. Mihaela Radu, C. Cole, M. A. Dabacan, S. Sexton“Extensive Use of Advanced FPGA Technology in Digital Design Education”, ASEE Annual Conference & Exposition, 2008.
06. Ronald Hayne, “Beyond VHDL Simulation to On-Chip Testing”, ASEE Annual Conference & Exposition, 2008.
07. David A. Patterson, John L. Hennessy, ”Computer Organiation and Design 4th Ed.”,, Elsevier Science, 2008.
08. Soonhwan Ro, “Practical Exercise for Efficient education of Computer System Architecture”, ASEE Annual Conference & Exposition, 2008.
09. XESS Corp. , http://www.xess.com
10. Seloco, http://seloco.com/mycad_/mycad_en/sub02/sub028.html