ISSN: 2358-1271. Int. J. of Alive Eng. Educ. (IJAEEdu). (Online). Goiânia, v. 5, n. 1, p. 51-58, Jan./June 2018.

PETEEECS • EnAEn • EMC • UFG

Restructuring the Digital Systems Laboratory in Computer Engineering Course

Kollins Gabriel Lima[*]

[*]Departamento de Engenharia Elétrica e Computação, São Carlos, Brazil.
E-mail: kollins.lima@usp.br

Maximiliam Luppe[♦]

[♦]Departamento de Engenharia Elétrica e Computação, São Carlos, Brazil.
E-mail: maxluppe@sc.usp.br

Abstract

It's not difficult to find students of Computer Engineering, at the University of São Paulo, in São Carlos School of Engineering, complaining about the way that practical classes are given during the graduation. What is said is that these classes only reproduce results already seen in theory classes, that they are limited by a laboratory script and, at the end, nothing new is added to their knowledge, making them a little bit frustrated about the classes. In this paper, it's shown how PBL (Problem-based Learning) was used in Digital Systems Laboratory, a 4-semester discipline, to start a change in this situation. The main purpose of the project was to use all the knowledge in digital logic, already seen in theory classes by the students, to build the main modules of a basic processor: arithmetic logic unit (ALU), for combinational logic studies; register bank, for sequential logic studies; and control unity, used to study finite state machines. This new way to teach each topic of the discipline brought new challenges to the students, now free to solve them the way they wanted to, once only the specification of inputs and outputs were given. Using FPGA boards and circuit simulators for development, the students succeeded in their task and the modules were built and tested in a test platform, developed to allow the simulation of the entire processor. This project had a positive feedback, either reflected in grades (in laboratory and in related theoretical disciplines) and also in the student's motivation (although some difficulties was found) helping them to understand how different concepts seen during the graduation course are related. This feedback

Article history: Received on 30 November 2017. Accepted on 24 April 2018. Published online on 28 September 2018.

Digital Object Identifier (DOI): Pending number

is now helping the development of a new material for next classes, in order to make specifications clearer, correct bugs in the test platform and improve some features for debug.

Keywords: Computer Architecture, Digital Systems, Problem-based Learning.

1. Introduction

Many students of Computer Engineering course, in São Carlos School of Engineering, say that during the graduation, the practical classes offered don't meet their needs, mainly because it's stuck to a laboratory script, giving them a step-by-step solution with no need to think about what they are doing.This study tries to solve this issue through the implementation of PBL in a 4-semester discipline: the Digital Systems Laboratory.

The laboratory practices were focused on the use of Digital Systems concepts to build a processor. It can be found some successful approaches to build a simple processor in laboratory using discrete components[1,2], what helps the student to better understand each of the basic components of a processor, but it demands a simple architecture for that, otherwise it would take a considerable assembly time. Also, with a complex circuit, hardware errors may be frustrating to find and correct.

In this case though, it was used an FPGA board to implement a simplified version of a 32 bits MIPS processor, through circuit diagram schematics. This model of processor was chosen because it is also studied in Computer Organization at the same semester. That way, the student could learn the theory of MIPS and implement its modules in laboratory. To measure the results, the method was applied only in one of the three classes of laboratory, and it was used the grades of the students in the discipline of Computer Organization.

The purpose of this study is to apply Problem-based Learning (PBL) in Digital Systems Laboratory to motivate and help the students them understand different concepts of Digital Systems and Computer Organization, and how they are related each other. PBL is an active learning process that, through the use of real world problems, motivates learning and helps the students to connect the information, to give a meaning to it, so that way the learning is deeper[3]. It is expected that the students that were submitted to this new method improve its final grade in Computer Organization class.

2. Methodology

The course was divided into four practices. All the practices were divided into three steps: logic problem, logic circuit implementation, and report presentation. The focus of each practice was the logic circuits that compose MIPS, so that the students didn't have to worry about Computer Architecture questions. For each logic problem, it was expected the students to use Digital Systems concepts to solve it. Then, the solution proposed was implement and tested into an FPGA development kit, and a final report was presented. Figure 1 shows the architecture of the MIPS processor and the implemented modules[4].

A questionnaire to evaluate the opinion of the students was also carried out. Following the four practices are described.

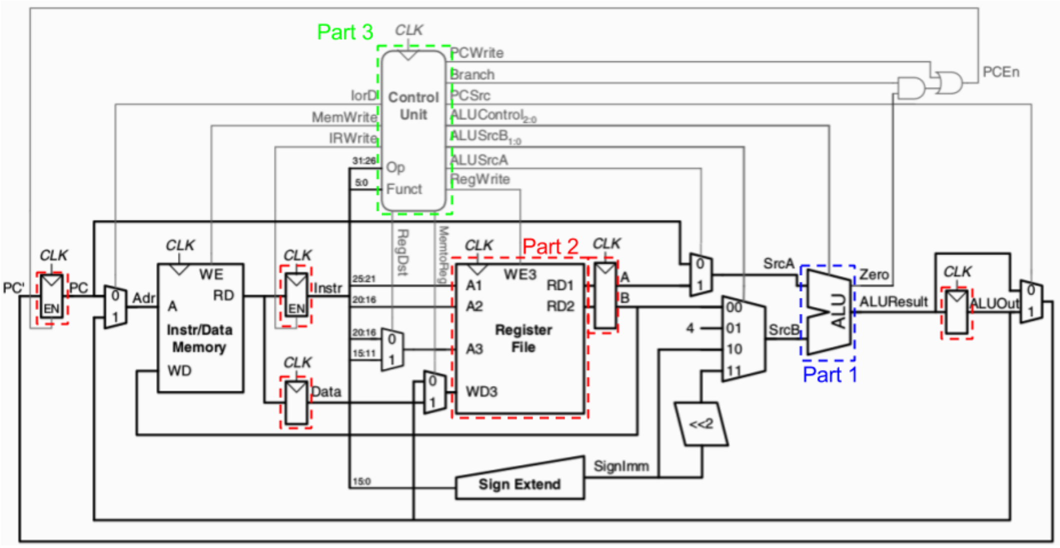

Figure 1.MIPS processor

2.1. Combinatorial Logic

An Arithmetic Logic Unit (ALU) was proposed and studied through the use switching functions. The ALU should have two inputs (A and B), an output S, a selector F and a zero flag. In this module, students were able to practice multiplexer, two complement adder and subtractor, and logic operations. Table 1 shows the specification of the circuit proposed.

Table 1. Specification on input and output ALU.

Selector (F)	Output (S)
000	A and B
001	A or B
010	A + B
011	Not used
100	A and(not B)
101	A or (not B)
110	A-B
111	Set on Less than(SLT)

An example of implementation is show in figure 2.

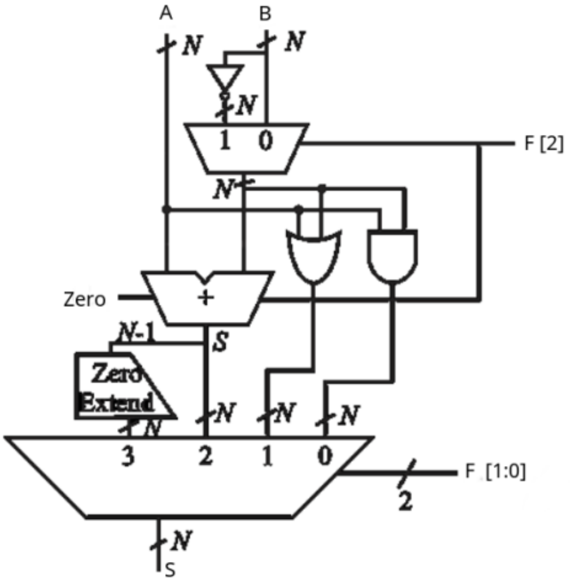

Figure 2. Possible implementation of ALU in practice 1.

2.2. Sequential Logic

The study of sequential logic was done based in the implementation of the Register Bank. Each register should contain: input, output, clock, clock enable, and asynchronous set/reset. The register bank should contain 32 registers, one data input and two data outputs (as well as one write selector, one write enable, two read selectors, clock and clock enable). Figure 3 shows an example of implementation. Like in the other parts, design decisions was free for every component (registers, mux, decoders, etc.)

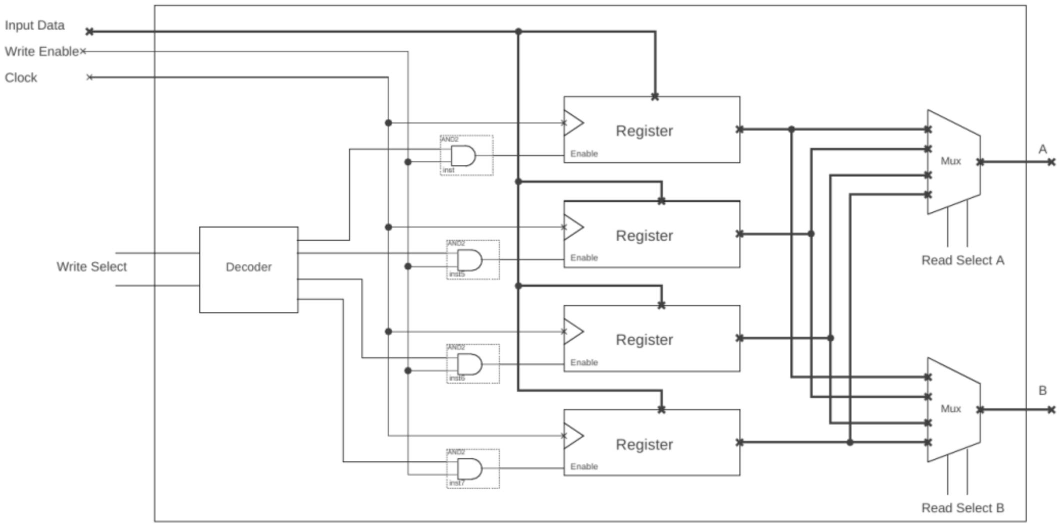

Figure 3. Simplified implementation of register bank,with only 4 registers.

For the Program Counter (PC), instead a register, it was proposed the implementation of a counter, with parallel load for jumps, shown in figure 4.

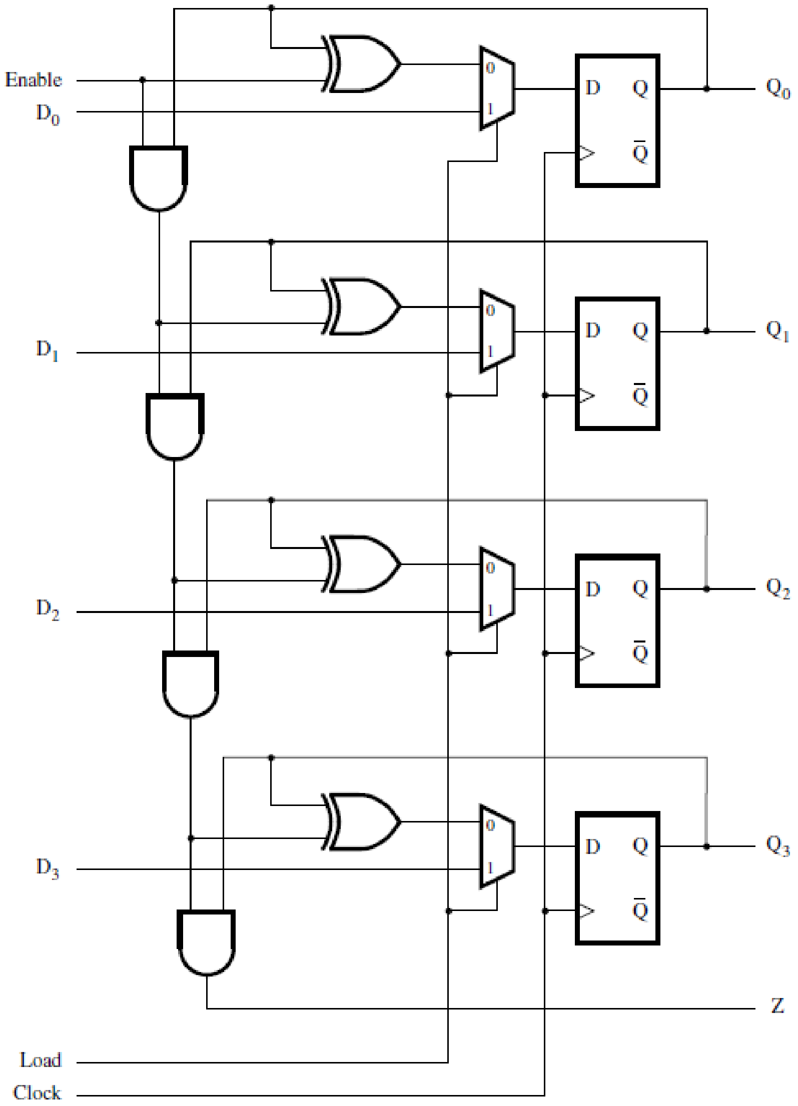

Figure 4. Counter with parallel-load proposed

2.3. Finite State Machine

The Control Unit was composed by a Finite State Machine (FSM) and an ALUControl. The use of a FSM allowed joining both combinatorial and sequential logic. It was used concepts of Digital System to build a Moore Machine, a FSM where the output depends only on the current state. There were 12 states, allowing the decoding of some basic instructions, like Load/Store, Conditional Branch, Jumps, Add immediate and R-type instructions. The input of FSM is the OPCODE of the instruction (read from memory in PC position) and its outputs are shown in figure 5.

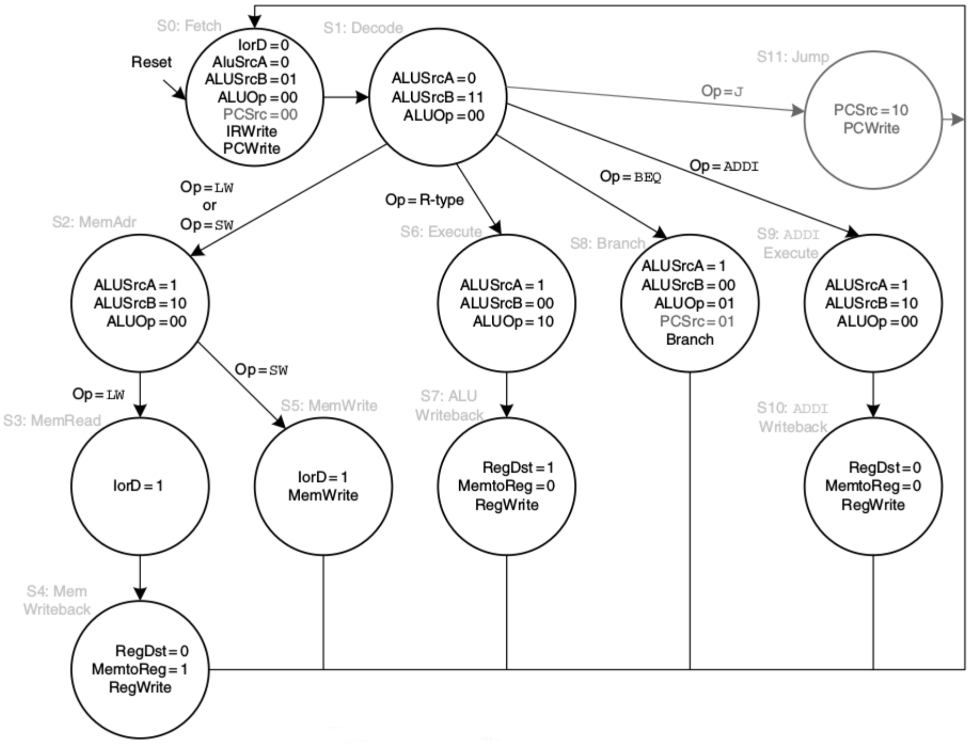

Figure 5. FSM proposed for practice 3.

To built the ALUControl, it was used the specifications shown in table 2. The output of ALUControl was used as control input of ALU, from practice 1. As input, it takes the FUNCTION field of the instruction and the ALUOp output from the FSM.

Table 2. Specification of ALUControl.

$\text{ALUO}_{p1:0}$	$\text{Funct}_{5:0}$	Output(ALU selector - F)
00	-	010
01	-	110
10	100100	000
10	100101	001
10	10000	010
10	10001	110
10	101010	111
11	-	-

2.4. Final Assembly

At last, it was created a test platform containing all the MIPS organization (as showed in figure 1) and some slots for the modules developed by students: Register Bank, Control Unit and the ALU, so that each student could integrate their own implementation of these components and test the whole processor using real MIPS. Figure 6 shows the processor running in a FPGA board.

The test platform also contained a program memory. The program to test the components was stored in this memory and it could be changed or customized with any of the instructions that could be decoded by the Control Unity. During the execution of the program, informations like progress, input and output data and PC value could be seen in the LED matrix, 7 segment displays and LCD display.

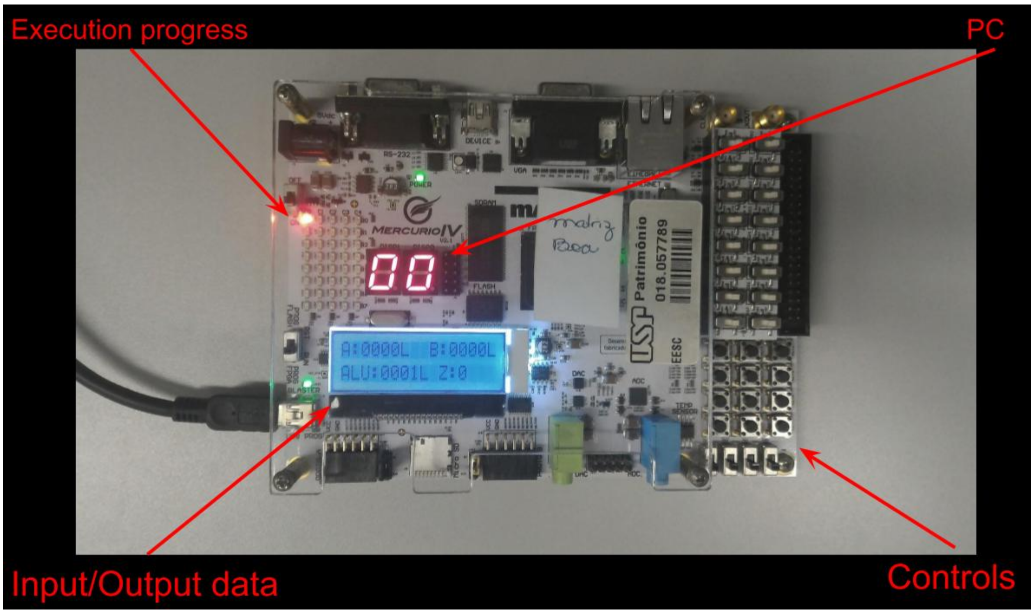

Figure 6. Test platform running the complete MIPS processor.

3. Results

Feedback from the students showed that, although some difficulties, mostly related to the use of the new softwares, it was motivating and it helped them to understand different concepts and how they are connected. The grade of the students in the discipline of Computer Organization was compared. It was possible to notice that there is a slight difference between those who had the PBL experience and the others:

- Final Grade: grades 6.2 % higher than the others;
- 71 % of the 19 students reached the passing grade; and
- 60 % of the others students (in other classes in the same semester) reached the passing grade.

This was the first time this methodology was applied in this laboratory and new classes will bring more data about how it can improve practical classes.

4. Conclusions

The PBL was well accepted and motivate the students to understand different concepts of Digital Systems and Computer Organization, and for this reason, it looks like a promising method. New materials are being elaborated to the next semester, including more examples, figures and theory to make things clearer.

Acknowledgment

We'd like to acknowledge Professor Eduardo Marques for gently provide his students grades in the Computer Organization discipline, so that we were able to analyze the impact of this experiment in a related course.

Reference

1. FILHO, E. E.; RIBEIRO, L. R. C. Innovating in the teaching of business administration: an experiment with problem-based learning(PBL). *Cadernos EBAPE.BR*, v. 6, 2018. Available in: <https://http://www.scielo.br/scielo.php?script=sci_arttext&pid=S1679-39512008000500004>. Accessed on: 12 sept. 2018.
2. FEINBERG, D.A Simple and Affordable TTL Processor for the Classroom. *Computer Science Education*, v. 17, 2007. Available in: <https://pt.scribd.com/document/377555968/A-Simple-and-Affordable-TTL-Processor-for-the-Classroom>. Accessed on: 12 sept. 2018.
3. PILGRIM, R. A. Design and Construction of the Very Simple Computer (VSC):A Laboratory Project for an Undergraduate Computer Architecture Course. *SIGCSE,ACM*, p 151-154, 1993. Available in: <http://www.homebrewcpu.com/p151-pilgrim.pdf>. Accessed on: 12 sept. 2018.
4. HARRIS, D.M; HARRIS, S.L. *Digital Design and Computer Architecture*. Burlington: Morgan Kaufmann, 2012.