FPGA BASED INDIVIDUAL COMPUTER ARCHITECTURE LABORATORY EXERCISES

Othman Ahmad
Faculty of Engineering, Universiti Malaysia Sabah

Email: othman58@ums.edu.my

ABSTRACT

Computer Architecture is the study of digital computers towards designing, building and operating digital computers. Digital computers are vital for the modern living because they are essential in providing the intelligences in devices such as self-driving cars and smartphones. Computer Architecture is a core subject for the Electronic (Computer) Engineering course at the Universiti Malaysia Sabah that is compliant to the requirement of the Washington Accord as accredited by the Engineering Accreditation Council of the Board of Engineers of Malaysia (EAC). An FPGA (Field Programmable Gate Array) based Computer Architecture Laboratory had been developed to support the curriculum of this course. FPGA allows a sustainable implementation of laboratory exercises without resorting to poisonous fabrication of microelectronic devices and installation of integrated circuits. An FPGA is just a configurable and therefore reusable digital design component. Two established organisations promoting computer engineering curriculum, ACM and IEEE, encourages the use of FPGA in digital design in their latest recommendation and together with the EAC, emphasises the grasp of the fundamentals for each student. The laboratory exercises are individual exercises where each student is given a unique assignment. A laboratory manual is provided as a guide and project specification for each student but overall the concept of the laboratory exercise is a student-centred one. Each student is allowed to pace their effort to achieve the sessions of the laboratory exercises starting from session one to session ten. A quantitative analysis of the effectiveness of these laboratory sessions is carried out based on the numbers of students completing the laboratory sessions. These sessions start from an 1:FPGA tutorial to implementations of features of a microprocessor of 2:Immediate Load, 3:Immediate Load to Multiple Registers, 4:Addition, 5:Operation Code, 6:Program Memory, 7:Jump, 8:Conditional Jump, 9:Register to Register and 10:Input-Output. The results of three batches of students show that within the time limits of a one credit hour course, students had managed to complete some aspects of the implementation of a simple microprocessor.

Keywords: Computer Architecture; Microprocessor; FPGA; Laboratory Exercises

1.0 INTRODUCTION

Digital computers are important for the modern living. They are used in more and more devices such as smartphones and self-driving cars. It is vital for a nation to learn how to construct digital computers in order to be competitive against all other nations. It is not surprising that China had developed the fastest super computer by using their own microprocessors. Previously, the fastest digital computers tend to rely on microprocessors from the U.S.A. but starting from 2016, China had developed the fastest super computer using their own design and manufacturing. (Feldman, 2017).

Initially, simulators are used in Computer Architecture courses (Nikolic, et al., 2009). The topics coverage criteria in the survey were established using the IEEE Curriculum

Guidelines for Undergraduate Degree Programs in Computer Engineering in 2004 (ACM and IEEE Computer Society, 2005) (Nelson, et al., 2004). The latest one was published in 2016 (Association for Computing Machinery (ACM) and IEEE Computer Society, 2016). There is not much change in Computer Architecture so we may still use the same criteria used in the survey.

The survey concluded that simulators EDCOMP, HASE, ISE Design Suite, JHDL, M5, and Quartus II met most of the criteria. Quartus II is an HDL simulator but can be used to program an FPGA as well. FPGA is an electronic device that can be programmed at the logic gate level. Programming FPGA is similar to fabricating a microelectronic device. Whereas in custom designed microelectronic device, the logic gates are hard wired to each other, an FPGA allows the wirings to be reprogrammed at will. The survey was on the suitability of simulators for teaching Computer Architecture in a laboratory environment but not on the development of actual laboratory sessions. Nevertheless, it should be obvious that simulators will not be able to compete with FPGA development boards that can actually implement any digital circuit. The latest FPGA boards are able to implement entire microprocessors and peripherals so would be the best tool to teach Computer Architecture.

2.0 FPGA BASED LABORATORY SESSIONS

Nowadays there are more and more FPGA based Computer Architecture laboratory sessions (Schoeberl, 2016) (Institute of Computer Engineering, Austria, 2016) (Department of Computing, Imperial College London, 2016). However, all of them appear to concentrate on using pre-built microprocessors. As early as 1996 (Li & Chu, 1996), Li and Chu had reported the use of FPGA boards in hardware exercises for Computer Architecture at the University of Aizu, Japan where students were reported to had implemented a pipelined RISC processor using Xilink FPGA chip. Jong Hyuk Lee et al., reported a class project building a five-stage pipelined 32-bit MIPS design with experiments on the Altera DE2 board (Lee, et al., 2012). Although this project requires the implementation of a real CPU instead of utilizing simulators or just experimenting with ready-made complete CPU models, it is a group effort instead of individual achievements.

3.0 COMPUTER ARCHITECTURE LABORATORY SESSIONS AT UMS

The Computer Architecture course at the Universiti Malaysia Sabah (UMS) is accredited by the Engineering Accreditation Council (EAC) of the Board of Engineers Malaysia (BEM). The accreditation process follows the Washington Accord and documented by the Graduate Attributes (International Engineering Alliance, 2013). The curriculum is targeted towards achieving Outcome Based Education (OBE). Each student is assessed towards their competence in each of the Program Outcomes. There are 12 Program Outcomes that must be achieved, corresponding to the 12 knowledge profiles of the Washington Accord Graduate Attributes (WA1 to WA12).

The course outcome that is relevant to the laboratory exercises is the ability of students to describe fundamental knowledge on Computer Architecture supporting the Program Outcome 1 or WA1: Apply knowledge of mathematics, natural science, engineering fundamentals and an engineering specialization as specified in WK4 to the solution of complex engineering problems. WK4 is engineering specialist knowledge that provides theoretical frameworks and bodies of knowledge for the accepted practice areas in the engineering discipline; much is at the forefront of the discipline.

The whole course is supported by lectures, tutorials and assignments. The laboratory exercises are just supporting the curriculum of the course. The emphasis of the laboratory had been the fundamentals of Computer Architecture, specifically the design and implementation of a central processing unit (CPU) or microprocessor if implemented as a single integrated circuit. Once a student is able to design and build a microprocessor by himself, he should be able to develop his skills further towards designing and implementing pipelined and multicore processors.

The laboratory exercises are divided by sessions. Session 1 is to familiarise students with logic design using HDL and schematics. Session 2 is the implementation of the immediate load. Session 3 is the implementation of immediate load to several registers. Session 4 is the implementation of adding immediate data. Session 5 is the implementation of the opcode which chooses between adding and loading immediate data. These sessions are just continuations of the previous sessions. Session 6 continue with session 5 by using a program memory to store the instructions instead of loading the instructions from switches. Session 7 is the additional implementation of a jump instruction. Session 8 is the implementation of the conditional jump instruction. Session 9 is the implementation of the register to register instructions. Session 10 is the implementation of input and output instructions. An eBook had been made available at Smashwords (Ahmad, 2014).

In order to reduce the chance of plagiarism, during session 1, each student must create project files with their own unique names. In session 2, the immediate data are located at different bit locations in the instruction format based on the last digit of their matriculation numbers. To reduce the chance of copying further, the opcodes are varied to use positive or negative logic. Session 1 is based on a tutorial supplied by the manufacturer of the FPGA board, Terasic DE2-115. There is no example circuit provided from session 2 to 4. However, from session 5 to 8, students are provided with reference circuit diagrams. Students need to understand their assigned instruction patterns and modify the provided circuit diagrams. They were free to decide for themselves how they rearrange the other unallocated bits or methods of displaying the outputs. They were encouraged to discuss with classmates and teach others within reasonable restraints. Each student will be verified to see if they had actually completed their claimed sessions with demonstrations and interviews.

4.0 ANALYSING THE EFFECTIVENESS OF THE LABORATORY EXERCISES

The laboratory exercises had been carried out for three batches. Students are marked based on various criteria. The most important criterion is the highest session that each student can achieve. The other criteria are so subjective that they will not reveal much definitive information. For the same reason, qualitative surveys were also not conducted, apart from the mandatory customer satisfaction surveys. In this analysis, only the verified achievement of the highest sessions will be used as the data for the analysis. With three batches, we should have sufficient data to come up with some useful conclusions.

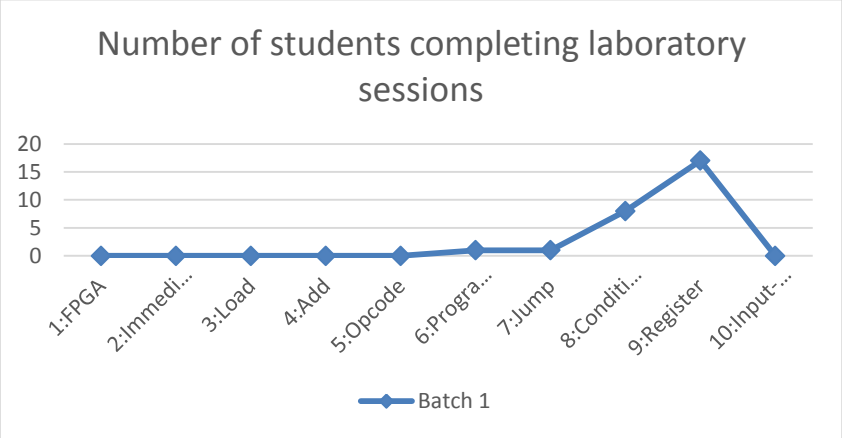


Figure 1: Batch 1

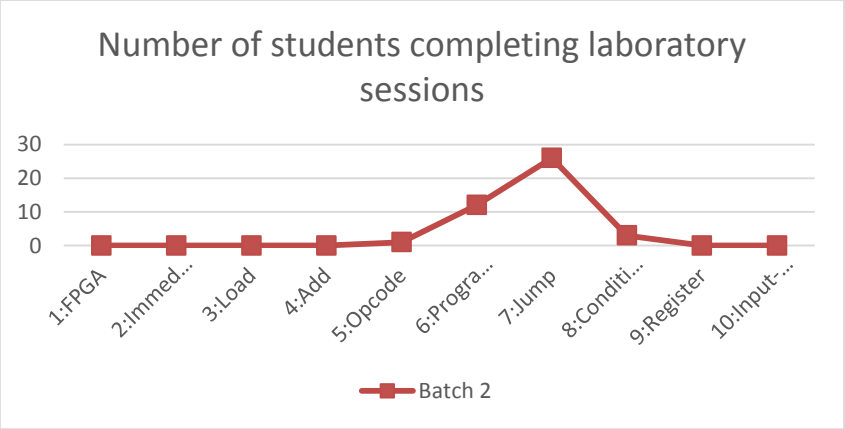


Figure 2: Batch 2

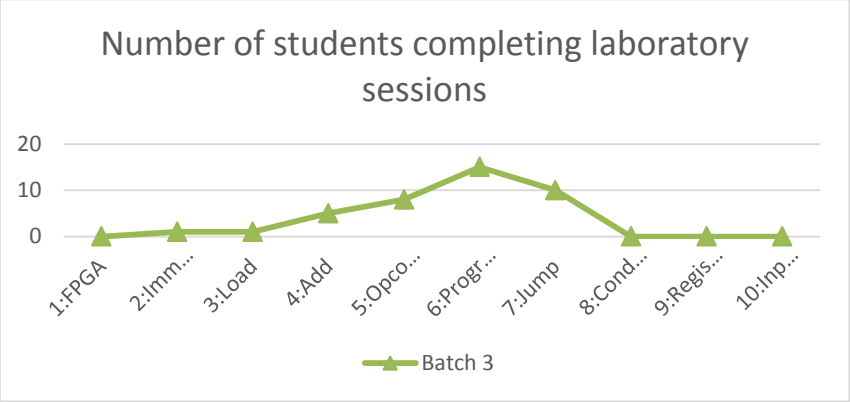


Figure 3: Batch 3

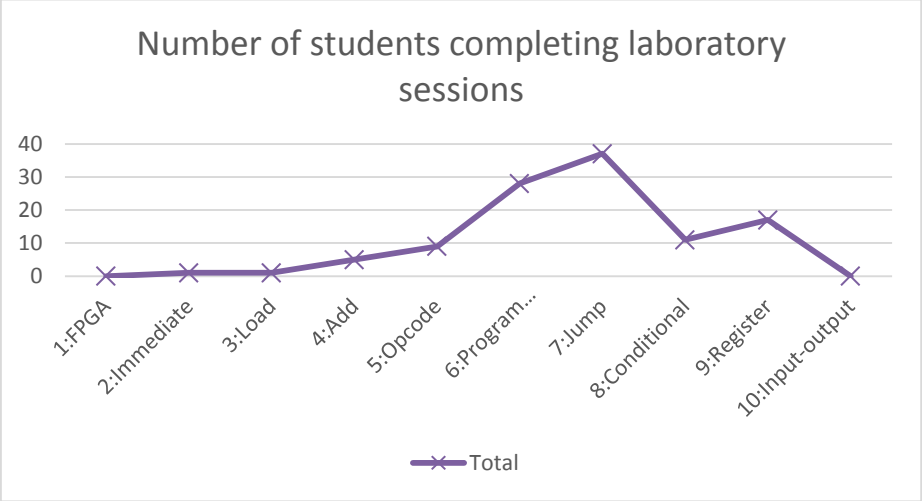


Figure 4: Total number of students only

Batch 1 was conducted in 2013. Batch 2 was conducted in 2015 and batch 3 in 2016. Then the results in Figures 1, 2, 3, 4 are analysed. There does not seem to be a definite pattern apart from the peaks. There is some resemblance of a normal distribution curve where the peaks indicate the mean. In Figure 1, the mean is Session 9. In Figure 2, the mean is Session 7. In Figure 3, the mean is Session 6.

The performance of students is worst in the latest laboratory session in 2016. One reason could be the differences in the supervision load as the number of students vary. Batch 1 has the smallest number of students at 27. Batch 2, 42. Batch 3, also 42 but there was no laboratory demonstrator to help supervise the students.

Another reason is the quality of students at these batches. The laboratory sessions are voluntary. It is up to the students to determine the level of work that they intend to pursue. Batch 1 has the best students because a few of them were Tanzanian scholarship holders. The local students were also good. In fact, one of them was deemed as the best student for the entire UMS. They helped to encourage other students to accomplish at a higher level. For batch 2, a student from batch 1 had become the demonstrator for the laboratory sessions.

The third reason could be the marking scheme. They vary from batch to batch. In batch 1, students do not really had a good idea as to how they were marked. They had to assume that they need to complete all the laboratory sessions in order to achieve the highest grade of A. In batch 2 and 3, the marks are based on session numbers multiplied by 10, and 20 more marks from other assessments. It means that the highest that need to be covered is just 8, assuming that they can get high marks for the 20 other marks. There is less incentive for them to pursue laboratory session 8.

These variations can be smoothed by just considering the total number of students as in Figure 4. There is an anomaly in the spike in session 9, which is register to register instruction. The largest number of students is session 7, which is the implementation of the jump instruction. This is the beginning of programming and not usually covered in detail in textbooks. On average, achieving session 6 should be reasonable towards a fundamental understanding of the implementation of a microprocessor. Session 6 is the storing of instructions in a program memory, the beginning of the stored program concept.

Figure 5 shows a sample of the schematic diagram of the implementation of Laboratory Session 9 compared to the schematic which is given in the laboratory manual, which was up to Session 8 only, the implementation of a conditional jump instruction. The laboratory manual only shows the descriptions of a Mano RISC described in the textbook by Mano (Mano & R. Kime, 2008).

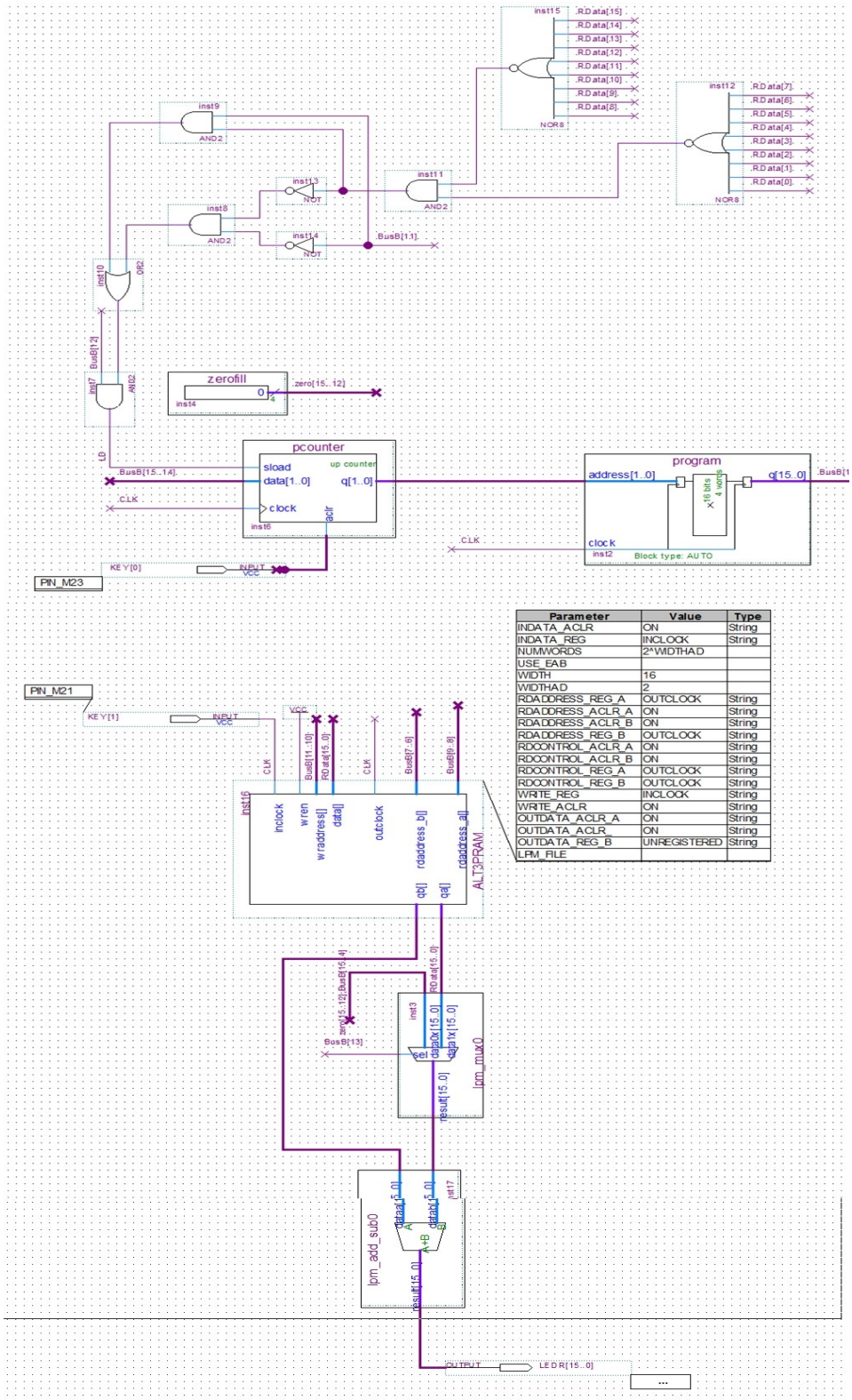
5.0 SUGGESTED IMPROVEMENTS

The laboratory session uses the concept of student centred learning. Students were only guided towards achieving their targets. This requires highly motivated students. Concrete steps should be taken to increase the interest of students in basic Computer Architecture. Computer Engineering is not a popular subject in Malaysia but graduates managed to find jobs much quicker than other engineering programs. This fact had been used to increase awareness of the importance of Computer Engineering, but Computer Architecture is too fundamental to sustain the increase in the interest of students.

The result of the analysis of the achievement of students should allow the lecturer to fine tune the marking scheme further without increasing the workload of students. It also shows that early intervention should reduce the number of students who cannot complete sessions less than 6. The danger is that students do not learn the fundamentals. Maybe a more practical and fun way of showing the results should be developed.

6.0 CONCLUSION

The quantitative data supports the conclusion that it is possible for students to achieve various stages in the design of a microprocessor by themselves as shown by the considerable number of students who managed to complete session 9, the register to register instructions. Being a student centred learning; the level of competence is determined by each student within the one credit hour that is allocated for the laboratory sessions. The sessions are not too easy based on the absence of any student completing session 10, Input-Output instruction.



Parameter	Value	Type
INDATA_ACLR	ON	String
INDATA_REG	INCLOCK	String
NUMWORDS	2^WIDTHAD	
USE_EAB		
WIDTH	16	
WIDTHAD	2	
RDADDRESS_REG_A	OUTCLOCK	String
RDADDRESS_ACLR_A	ON	String
RDADDRESS_ACLR_B	ON	String
RDADDRESS_REG_B	OUTCLOCK	String
RDCONTROL_ACLR_A	ON	String
RDCONTROL_ACLR_B	ON	String
RDCONTROL_REG_A	OUTCLOCK	String
RDCONTROL_REG_B	OUTCLOCK	String
WRITE_REG	INCLOCK	String
WRITE_ACLR	ON	String
OUTDATA_ACLR_A	ON	String
OUTDATA_ACLR	ON	String
OUTDATA_REG_B	UNREGISTERED	String
LPM_FILE		

Figure 5: Laboratory Session 9:Register to Register, by a student

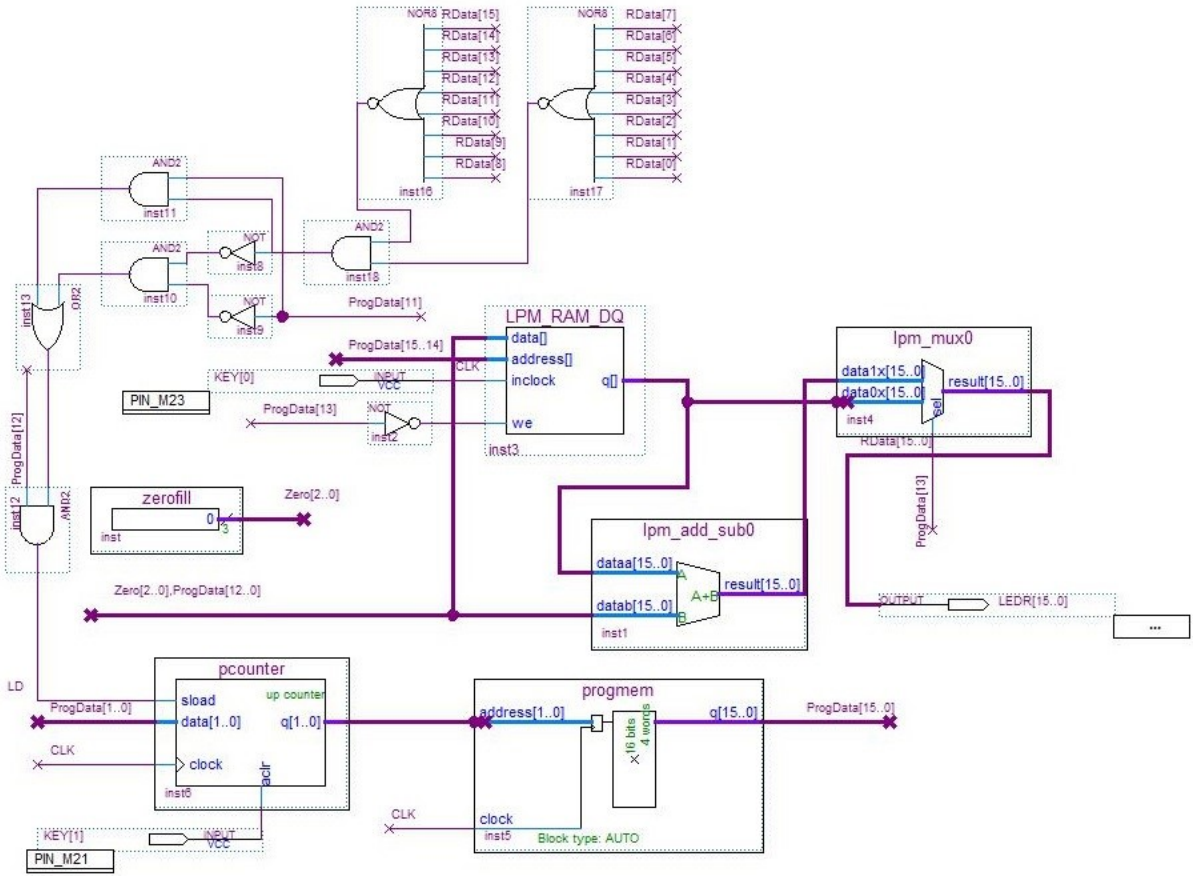


Figure 6: Laboratory Session 8: Conditional Jump, given in the laboratory manual

REFERENCES

ACM and IEEE Computer Society, 2005. *Computing Curricula 2005.* [Online] Available at: http://www.acm.org/education/curric_vols/CC2005-March06Final.pdf

Ahmad, O., 2014. *Implementing a Cpu using Fpga.* s.l.:Smashwords.

Association for Computing Machinery (ACM) and IEEE Computer Society, 2016. *Computer Engineering Curricula 2016: Curriculum Guidelines for Undergraduate Degree Programs in Computer Engineering.* [Online] Available at: https://www.computer.org/cms/Computer.org/professional-education/curricula/ComputerEngineeringCurricula2016.pdf

Department of Computing, Imperial College London, 2016. *CO332-Advanced Computer Architecture.* [Online] Available at: http://www.imperial.ac.uk/computing/current-students/courses/332/ [Accessed 10 Jan 2017].

Feldman, M., 2017. *Putting the Rise of Chinese Supercomputing in Perspective.* [Online] Available at: https://www.top500.org/news/putting-the-rise-of-chinese-supercomputing-in-perspective/

Institute of Computer Engineering, Austria, 2016. *Digital Design and Computer Architecture (WS 2016).* [Online] Available at: http://ti.tuwien.ac.at/ecs/teaching/courses/ddca16 [Accessed 10 Jan 2017].

International Engineering Alliance, 2013. *Graduate Attributes and Professional Competencies Version 3.* [Online] Available at: http://www.ieagreements.org/IEA-Grad-Attr-Prof-Competencies.pdf [Accessed 10 Jan 2017].

Lee, J. H., Lee, S., Yu, H. C. & Suh, T., 2012. Pipelined CPU design with FPGA in teaching computer architecture. *IEEE Transactions on Education,* August, 55(3), pp. 55(3):341-348.
Li, Y. & Chu, W., 1996. Using FPGA for Computer Architecture/Organization Education. s.l., ACM.

Mano, M. M. & R. Kime, C., 2008. Logic and Computer Design Fundamentals - 4th International Edition. Singapore: Pearson Prentice Hall.

Nelson, V. P. et al., 2004. The Computing Curriculum - Computer Engineering (CCCE) A MODEL FOR COMPUTER ENGINEERING CURRICULA IN THE NEXT DECADE. s.l., Amerian Society for Engineering Education, p. Session 2532.

Nikolic, B., Radivojevic, Z., Djordjevic, J. & Milutinovic, V., 2009. A Survey and Evaluation of Simulators Suitable for Teaching Courses in Computer Architecture and Organization. *IEEE TRANSACTIONS ON EDUCATION,* NOVEMBER, 52(4), pp. 449-458.

Schoeberl, M., 2016. *Computer Architecture Lab.* [Online] Available at: https://en.wikiversity.org/wiki/Computer_Architecture_Lab [Accessed 10 January 2017].