

ARM Cortex M4F-based, Microcontroller-based, and Laboratory-oriented Course Development in Higher Education

Dr. Byul Hur, Texas A&M University

Dr. B. Hur received his B.S. degree in Electronics Engineering from Yonsei University, in Seoul, Korea, in 2000, and his M.S. and Ph.D. degrees in Electrical and Computer Engineering from the University of Florida, Gainesville, FL, USA, in 2007 and 2011, respectively. In 2017, he joined the faculty of Texas A&M University, College Station, TX, USA, where he is currently an Assistant Professor. He worked as a postdoctoral associate from 2011 to 2016 at the University Florida previously. His research interests include Mixed-signal/RF circuit design and testing, measurement automation, environmental & biomedical data measurement, and educational robotics development.

ARM Cortex M4F based microcontroller based and laboratory oriented course development in higher education

Abstract

ARM (Advanced RISC Machines) processors are frequently found in modern consumer electronics products such as smartphones and tablets. The author has been teaching two courses on embedded system software and microcontroller architecture for each Fall and Spring semester since Fall of 2016 and 2017, respectively. These two courses have been revised for more concentration on an ARM architecture and its applications to equip students to create various ARM applications. The microcontroller architecture course covers Verilog HDL, MIPS architecture, and ARM architecture. Next, students take the following embedded system software course. In this course, a MSP430 architecture is covered, and the ARM Cortex M4F processor and its high level C programming techniques using Tivaware and TI RTOS (Real-time Operating System) are covered. For the term project, students have a chance to build their own embedded system applications using an ARM Cortex M4F processor or MSP430x5 series microcontrollers. In order to enhance the laboratory components, a custom education board, BH EDU board, was designed and fabricated. This auxiliary education board has switches, a Bar LED, and a buzzer on the boards as well as headers for external components such as a LCD module and a keypad. Example robots using these educational boards are presented. The enrollment data as well as embedded system course assessment data from 2016 to 2018 is presented in the paper.

I. Introduction

The ARM (Advanced RISC Machines) architecture is one of the most popular RISC (Reduced Instruction Set Computer) architectures. ARM processors are frequently found in modern consumer electronics products such as smartphones and tablets. In engineering education, it is desirable to educate students about the ARM architecture and its immediate applicability to society and industry upon graduation.

The Electronic Systems Engineering Technology (ESET) program at Texas A&M University provides a strong educational program with an emphasis on embedded systems. The author has been teaching embedded system software (ESET 369) and microcontroller architecture (ESET 349) courses for each Fall and Spring semester since Fall of 2016 and 2017, respectively. Prior to this course revision, these two courses were organized to focus on teaching about a Texas Instruments (TI) MSP430 microcontroller and its in-house MSP430 application board. Now, these two courses have been revised for greater concentration on the ARM Cortex M4F architecture and its applications in order to equip students to create various practical applications using ARM Cortex M4F processors as well.

The microcontroller architecture course (ESET 349) covers Verilog Hardware Description Language (HDL), MIPS (Microprocessor without Interlocked Pipelined Stages), and ARM architectures. Students learn about these Instruction set architectures (ISAs) through assembly languages [1-3]. This course is a prerequisite for the next microcontroller application course, Embedded System Software (ESET 369). In this course, a TI MSP430 architecture is covered first, and students learn the ISA through a low-level assembly language. Next, the ARM Cortex

M4F processor and its high level C programming techniques using TI Tivaware and TI RTOS (Real-time Operating System) are covered in class [4][5].

Embedded Market Survey results of 2017 show the practical current statistics and references for embedded systems developers. One questionnaire is about the usage of RTOS and kernel, stating the following: “Does your current embedded project use an operating system, RTOS, kernel, software executive, or scheduler of any kind?” [6, page 54]. The results show that 68% of the responders use RTOS, kernels, execs, and schedulers [6]. In the author’s engineering technology program, it is highly desirable to prepare students to be immediately impactful to the industry as well as society. This is the reason why students are learning about the software framework as well as RTOS based programming since it is more practical and industrial oriented. As an educator, I provide program templates for laboratories and detailed instructions so that students can perform their laboratories successfully.

The teaching method of this microcontroller architecture course from low-level assembly to high level software framework or kernel based programming is a right fit for our Texas A&M University engineering students. However, it may need adjustments depending on the institutional preferences and educational goals. For instance, teaching Arduino based C/C++ programming is not a primary objective for our junior level Engineering Technology students since most of them were already exposed and well familiar with it through their introductory level engineering courses. However, Arduino platforms are discussed briefly during the ESET 349 and ESET 369 courses because Energia is an Arduino implementation on TI processors. In this course, the Energia kernel C++ source code package is briefly studied and analyzed. Arduino is an open platform and is currently highly impactful in education and industry; however, it may not be a suitable choice for many commercial industrial applications that require stable consumer products. Based on the readers’ educational goals, the topic lists about Arduino materials may be revised or amended.

II. Microcontroller architecture and embedded system software course organization

Prior to enrolling in these courses, students should pass two courses on digital logic and C programming. Assuming the students have mastered these prerequisites, the microcontroller architecture courses can build upon this foundation.

In the microcontroller architecture course, students will continue to learn more about Verilog HDL using FPGA boards. Students will learn from the Computer Organization and Design MIPS edition Textbook [1]. This book gives a good foundation and understating of computer architecture. Learning MIPS architecture is not the ultimate goal; however, this book and its lessons will give good understanding of intermediate level computer architecture, which helps the transition to ARM. After learning the fundamentals and basic MIPS architectures up to Chapter 2, the course switches to another book which focuses on learning ARM architectures through the study of instructions and the assembly languages [2]. This book covers ARM Cortex M4 and Thumb-2 instructions. This ARM textbook was used through Chapter 8. Then, the students will spend the rest of the semester with the MIPS textbook, starting at Chapter 3.

For the next microcontroller application course, embedded system software course, instead of following a textbook, several important topics are covered with the author using ample and up-



Figure 1. Laboratory room for embedded system education

to-date open resources directly provided by the microprocessor/microcontroller/IC module manufacturers with the instructor's design/code templates.

This embedded system software course covers the basic use of a real-time operating system (RTOS). In order to build the students' understating of RTOS, from time to time the author uses and teaches RTOS contents from an old textbook, *An embedded software primer* [7]. As it is mentioned, in this course, an MSP430 architecture is covered first. Then, the ARM Cortex M4F processor and its high level C programming techniques using Tivaware and TI RTOS (Real-time Operating System) are covered.

As their term project, students have a chance to build their own embedded system applications using an ARM Cortex M4F processor or MSP430x5 series microcontrollers of their choice. This is similar to a maker project. This term project experience for students has been well received. For the instructor, it is an important process to evaluate their overall understanding of the concepts and practical aspects of what they have learned throughout the embedded system education.

To assist this process, all the laboratory assignments are carefully designed to be synchronized with lecture materials, and they were designed to prepare students to be capable of carrying out their own term project idea. The laboratory assignment topics include LEDs, Switches, 4 x 4 Keypads, Buzzers, LCD (Liquid Crystal Display) modules, ADCs, accelerometers, H-bridge DC motor control, Servo motor control, UART (Universal Asynchronous Receiver-Transmitter), and I2C (Inter-integrated Circuit). These are the list of core concepts that students need to understand and use for their term project. They could obtain the knowledge through hands-on laboratory and the term project experience.

The author proposed and helped renovate the laboratory room. This is because the educational environment can affect students' creative thinking and their learning. Previously, it was a room

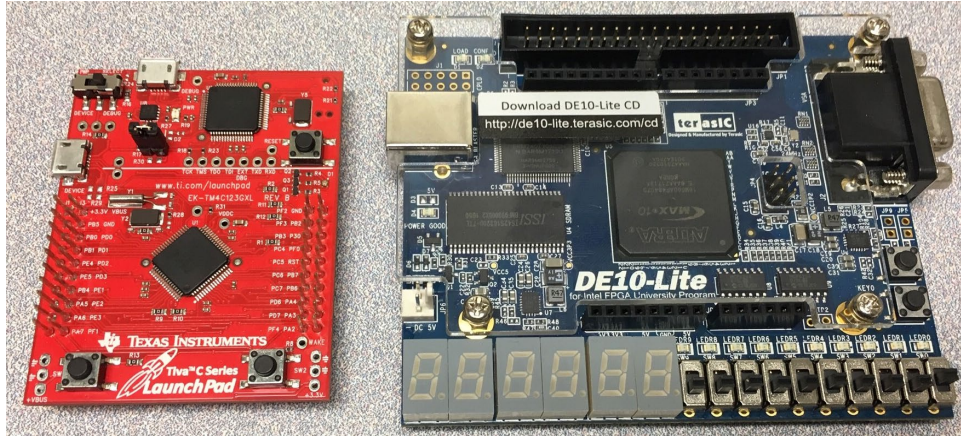


Figure 2. TM4C123G Launchpad (left) and Intel DE10-Lite board (right)

that was packed with many computers. Now, more than half of the computers have been removed from the room. The center area is now open and available to the students for discussion and the room has been equipped with more and newer instrumentations as shown in Figure 1. This laboratory room is now cable of accepting various teaching activities and styles such as active learning. A decent level of the laboratory environment is an important factor for the quality of this hands-on laboratory experience for the students.

III. Teaching Tools

A. Microcontroller architecture course

Students use Intel DE-10 Lite boards for learning Verilog HDL, shown in Figure 2. This is an inexpensive FPGA development board [8]. Students uses Quarts Prime Software to program this FPGA chip [9]. For MIPS education, students can use a MIPS simulator software, QtSPIM [10]. As an ARM Cortex M4F processor, TM4C123G Launchpad was used [11]. Students can write and debug their code using Code Composer Studio [12].

B. Embedded system software course

Students use two different kinds of MSP430F5529 and TM4C123G Launchpad boards. To expand the applications in class using these microcontrollers and to provide a better laboratory experience, custom “BH EDU” education boards were designed and fabricated[13] and are used in this embedded system software course. The BH EDU boards have switches, a Bar LED, and a buzzer on the boards as well as headers for external components such as a LCD module and a keypad. Students connect this educational board and the TM4C123G Launchpad board manually using jumpers. However, power and ground were already connected through the PCB. The task left for students is to complete data and control lines. This is a more flexible approach than using an educational booster pack, and it has more educational benefits in learning about hardware and software debugging skills for the students’ comprehensive understanding of the design and development process. While a fixed hardware approach could generate possible issues in reuse of codes from students who already completed the course, this flexible approach can help reduce these issues by changing the hardware connection settings each semester by the instructor. The

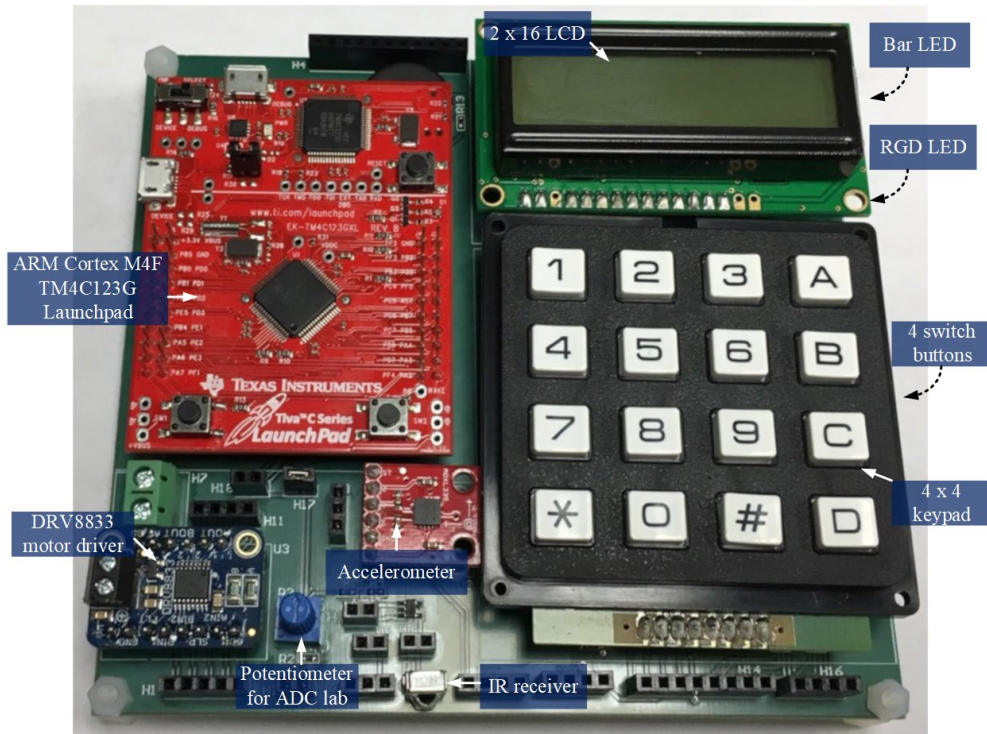


Figure 3. BH EDU board with TM4C123G Launchpad (ARM Cortex M4F)

code from the last semester is not applicable anymore. At the same time, it provides safer experiential learning because power and ground connections were already completed. If power and ground are not properly connected, it could be dangerous in some cases.

The BH EDU board, shown in Figure 3, was designed to be compatible with a 40 pin TI Launchpad board header. Currently, TM4C123G and MSP430F5529 Launchpad boards are used in this course. It is worth mentioning that the BH EDU board is compatible with any other TI Launchpad boards with the 40-pin Booster Pack Pinout.

The inside of the BH EDU lab kit is shown in Figure 4. This lab kit includes the BH EDU board and other components such as a keypad and LCD. Students need to bring their own Launchpad board and the jumper cables to use this kit. The kit is stored in a plastic container with its own kit number on the top and side. It is stackable in the lab closet. In this semester, 20 lab kits were fabricated and used. They have been distributed to the students during their lab hours. This management of the lab kits helped in reducing the accidental loss of the kits.

For the term project, students are given a maker project with restrictions. Students can build their own embedded system or robot applications. To assist this process, I have created and shown a wide range of robot and embedded system examples through the classes. Two examples of robots that use this BH EDU board as shown in Figure 5. The two wheel drive (2WD) robot is shown on the left side, and the four wheel drive robot (4WD) is shown on the right side. These robots were built on a custom designed educational acrylic plate. This plate was used as a robot base and was designed in SolidWorks. It was fabricated in-house using a laser cutting machine. The material is a 1/4 inch acrylic sheet; however, it is possible to use a thicker acrylic sheet or

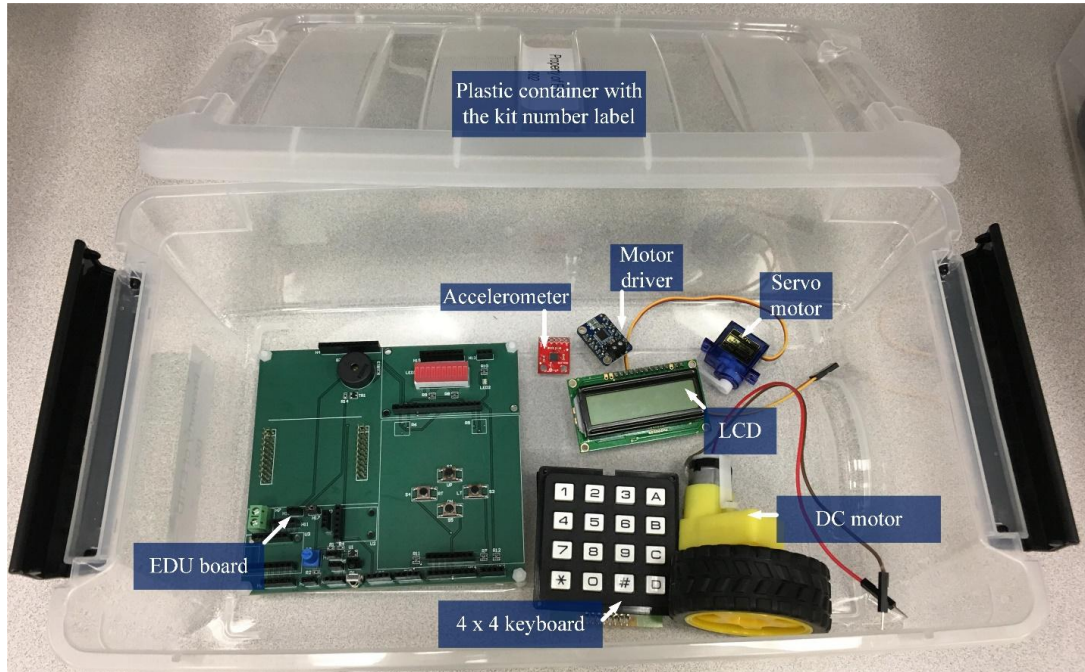


Figure 4. Inside of the BH EDU board kit

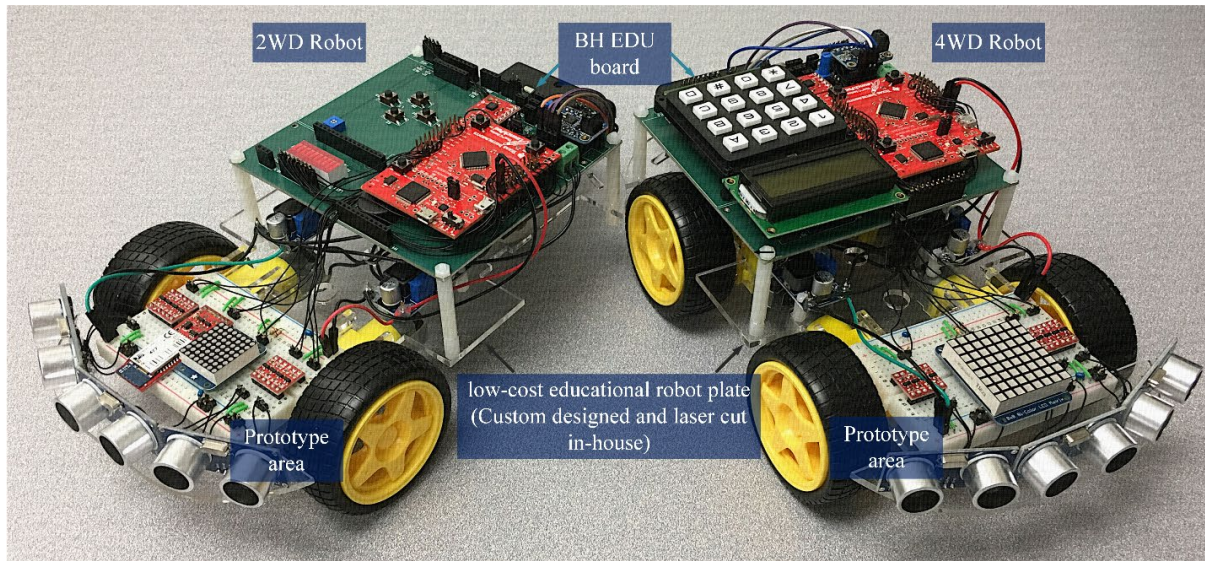


Figure 5. Educational robot applications using BH EDU boards. 2WD wireless Bluetooth controller robot on the left side and 4WD autonomous object avoidance robot on the right.

other materials as needed. 3D printing of this base plate is possible; however, it may be challenging in some 3D printers due to warping. The 2WD robot is an example of a wireless Bluetooth controlled robot with object detection functions. The 4WD robot is an autonomous robot example capable of avoiding objects, running similar to a Roomba.



Figure 6. Term project pictures (2017 on the left and 2018 on the right)

IV. Lecture materials, Laboratory Assignments and Term Projects.

The group activities such as group projects are a collaborative learning opportunity to prepare students to be successful collaborative members in industry and research organizations. Lab assignments and term projects are organized to be carried out in pairs. Students will work with their lab partners, and they will learn how to collaborate for the goal of the lab assignments. Lab groups are assigned in Lab 1 through a fishbowl method, and after the midterm exam, new lab groups are assigned. This will give them an opportunity to work with other students.

For the embedded system software course, for their term project, students are encouraged to find their own partner. Students tend to consult with me reading the term project. Interestingly, it has been observed that students tend to exchange their ideas actively with other student groups to seek possible solutions to their problems based on their varied experience in hardware and programming.

A. Microcontroller architecture course

The laboratories for the microcontroller architecture course are structured as nine laboratory assignments and one term project. One extra lab can be given to all students as an opportunity to make up their lowest lab score.

The outline of the laboratory assignments and term project was listed in the microcontroller architecture (ESET 349) course as follows:

Laboratory #0: The purpose is to install software on their PC, and to get familiar with the tools that will be used in the laboratory sessions.

Laboratory #1: The purpose is to introduce simple FPGA task implementation.

Laboratory #2: The purpose is to introduce the D-flip flop HDL implementation and to design and implement a 4-bit ripple counter.

Laboratory #3: The purpose is to introduce QtSPIM and a basic MIPS assembly code.

Laboratory #4: The purpose is to introduce the implementation of simple mathematical calculations using QtSPIM

Laboratory #5: The purpose is to introduce an assembly program that uses Memory Mapped I/O.

Laboratory #6: The purpose is to introduce an assembly program that can do a simple task using an ARM Cortex M4F board.

Laboratory #7: The purpose is to introduce an assembly program to perform simple S/W button tasks.

Laboratory #8: The purpose is to introduce an assembly program to control a bar LED and S/W buttons.

Laboratory #9: The purpose is to introduce simple parallel communication between the FPGA board and TM4C123G Launchpad.

Laboratory (Extra): The purpose is to write an assembly program and FPGA implementation that can generate a hamming code.

Term project: This term project is an extended lab of two weeks. For the term project, students will create a simple 8-bit microcontroller core system using the FPGA board and TM4C123G Launchpad.

B. Embedded system application course

The embedded systems software course is also structured as nine laboratory assignments and one term project. Similar to the microcontroller architecture course case, one extra lab can be given as an opportunity to make up their lowest lab score.

The outline of the laboratory assignments and a term project was listed in the embedded system software (ESET 369) course as follows:

Laboratory #0: The purpose is to install software on their PC, and to get familiar with the tools that will be used in the laboratory sessions.

Laboratory #1: The purpose is to introduce the basic MSP430 GPIO programming for switches and LEDs in C language as well as the use of an oscilloscope.

Laboratory #2: The purpose is to introduce the assembly language of MSP430 GPIO for switches and LEDs as well as the simple use of the EDU board.

Laboratory #3: The purpose is to introduce the keypad.

Laboratory #4: The purpose is to introduce the use of the timer and buzzer, and to program it to play a simple song.

Laboratory #5: The purpose is to introduce the LCD as well as to display characters on LCD.

Laboratory #6: The purpose is to introduce the ADC which can measure the voltage and display the results on the LCD module

Laboratory #7: The purpose is to introduce an accelerometer, and to write a program that can read accelerometer output signals.

Laboratory #8: The purpose is to introduce the H-bridge motor driver control for a DC motor and to write a program that can control the speed of a DC motor using PWM signals.

Laboratory #9: The purpose is to introduce the UART and Servo motor control as well as to write a program that uses both UART and controls a servo motor.

Laboratory (Extra): The purpose is to introduce I2C and EEPROM and to write a program that can communicate with an EEPROM IC. This is an optional lab.

Term project: This term project is an opportunity for students to demonstrate their skills learned throughout the course. This is like a maker project in a group or in a pair. Students were given more than the minimum of two weeks for the term project, in which they create their own embedded system application of their choice but with the restriction of the use of the specific microcontroller.

The pictures at the day of the term project presentation are shown in Figure 6. There are typically around 20 ~ 30 groups at the term project presentation day. The presentation date and time typically has been announced to other colleagues and staff. Some of them visit and have a chance to talk with students about their projects. This is a good opportunity for students because some have shown interesting projects. And, faculty members and visitors can use it as a recruiting opportunity for their research activities. Also, some of the students have been invited to be part of an institutional annual campus-wide open house for prospective students and their families. Also, some of them participated a campus wide institutional engineering showcase as a class project outcome [14].

V. Assessment

The number of enrolled students in these two courses is shown in Figure 7. It shows the trend of the increasing enrollment in these two courses. This is due to the growth of the Engineering Technology department at Texas A&M University. The number of laboratory sections is shown in Figure 8. Each section has about 15 to 19 students. Since each section can accommodate a limited number of students, the number of the laboratory sections has been increased. Currently, three graduate students have been teaching and managing a total of 10 laboratory portions of these courses.

The institutional standard course feedback questions are listed as follows:

1. Class Preparation: The class activities are well-prepared and organized

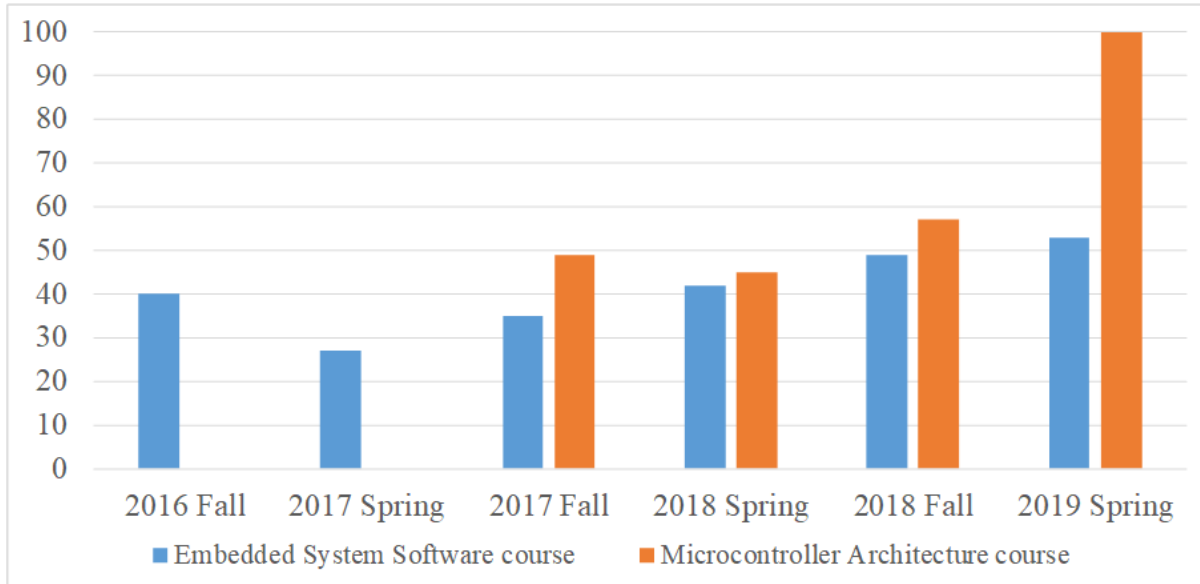


Figure 7. Enrollment to the embedded system software course and microcontroller architecture course.

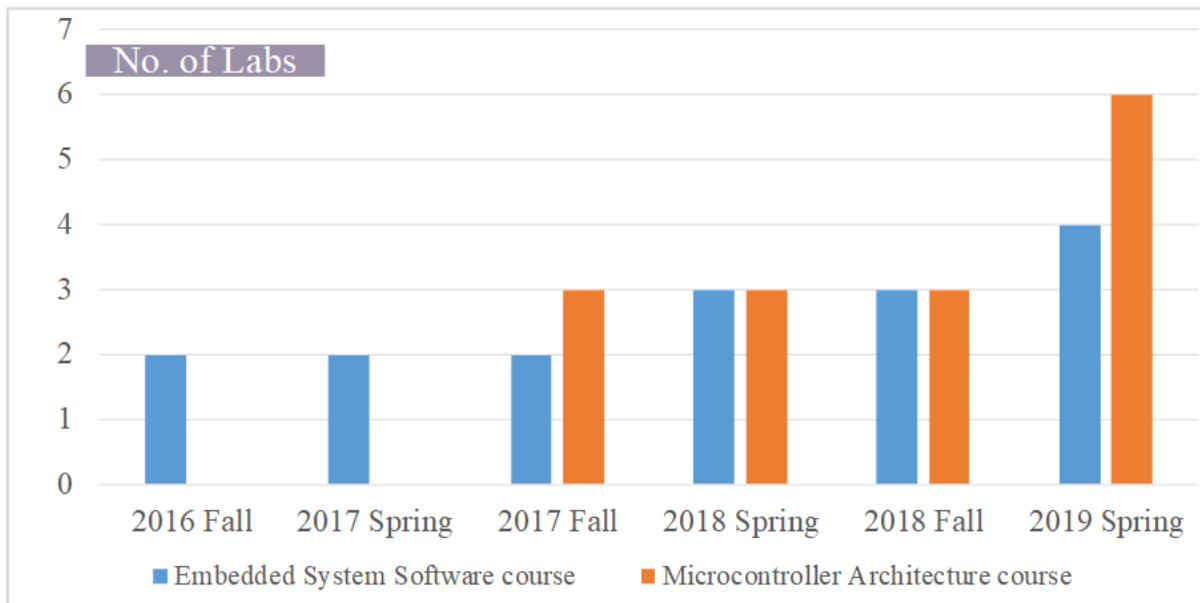


Figure 8. The number of lab sections of the embedded system software course and microcontroller architecture course.

2. *Assignments: The examinations, assignments, projects, etc. aid me in achieving the class objectives.*
3. *Communications: The instructor clearly explains material so that I can understand it.*
4. *Responsiveness: The instructor is open to my questions, and effectively answers them.*

5. *Academic concern: The instructor seems to care that I learn this material.*
6. *Availability: The instructor willingly makes time to help other students and me.*
7. *Fairness in Grading: The instructor is fair and consistent in evaluating my performance in the course.*
8. *Environment: The instructor maintains a good learning environment for me.*

Total number of students enrolled	Total number of Responses	Average numerical grade earned by students (4.0 scale)	Average Course grade (5.0 scale)
193	131	2.95	3.89

Table 1. Embedded systems course evaluation for the last five semesters of 2016 Fall ~ 2019 Spring.

The numeric course evaluation of the embedded system application course for five semesters was shown in Table 1. The total number of the enrolled students was 193, with 131 students responding. The average course grade was 3.89 out of 5.0 scale, and the average numerical grade earned by the students was 2.95 out of 4.0 scale. From the students' feedback, some of them have shown positive appreciation of lab organization that helped their learning saying the lab material was very well thought out and interesting. After they take these courses, they continue with their education with other courses such as Capstone project courses. From their verbal communication, I have heard positive outcomes from the term project experience.

IV. Discussion & Conclusions

The author had been constantly striving to shift the emphasis from the MSP430 to ARM processors starting Fall of 2016. ARM processors have been widely adopted by industry and have been used in a wide range of applications based on the current market. In this paper, the transitional effort in the course development and outcome was described. It generated microcontroller architecture and embedded system software courses providing a balanced approach between theory and hands-on learning skills through relevant laboratory assignments and term projects. The author plans to continue the effort in delivering practical and hands-on lectures and laboratories. In the Engineering Technology Department at Texas A&M University, a resident master's program will start from this Fall of 2019. The author plans to teach a new graduate course focused on the advanced embedded systems with the emphasis on the middleware and artificial intelligence.

Acknowledgements

The development of this work was supported by the ESET (Electronic Systems Engineering Technology) program at Texas A&M University, and the portion of the development was supported through Dr. Hur's Texas A&M start-up research fund. Thanks to Dr. John Bosshard for revision of this manuscript and the encouragements toward this task.

References

- [1] “IEEE Standard for Verilog Hardware Description Language,” IEEE Std 1364-2005 (Revision of IEEE Std 1364-2001), 2005.
- [2] D. A. Patterson and J. L. Hennessy, “Computer Organization and Design MIPS Edition: The Hardware/Software Interface,” Newnes, 2013.
- [3] W. Hohl and C. Hinds, “ARM Assembly Language: Fundamentals and Techniques,” Crc Press, 2016.
- [4] Texas Instruments, “TivaWare for C Series (Complete),” Available: <http://www.ti.com/tool/SW-TM4C>
- [5] Texas Instruments, “TI-RTOS: Real-Time Operating System (RTOS) for Microcontrollers (MCU),” Available: <http://www.ti.com/tool/TI-RTOS-MCU>
- [6] eetimes/embedded.com, “2017 Embedded Markets Study,” Available <https://m.eet.com/media/1246048/2017-embedded-market-study.pdf>
- [7] D. Simon, “An embedded software primer,” Vol. 1. Addison-Wesley Professional, 1999.
- [8] Terasic, “DE10 Lite Board,” Available: <https://www.terasic.com.tw/cgi-bin/page/archive.pl?Language=English&No=1021>
- [9] Intel, “Quartus Prime Lite Edition,” Available: <http://fpgasoftware.intel.com/?edition=lite>
- [10] James Larus, “SPIM: A MIPS32 Simulator,” Available: <http://spimsimulator.sourceforge.net>
- [11] Texas Instruments, “ARM Cortex-M4F Based MCU TM4C123G Launchpad Evaluation Kit,” Available: <http://www.ti.com/tool/EK-TM4C123GXL>
- [12] Texas Instruments, “Code Composer Studio (CCS) Integrated Development Environment (IDE),” Available: <http://www.ti.com/tool/CCSTUDIO>
- [13] B. Hur, “TI-BH-EDU-board-kit,” doi: 10.5281/zenodo.2538993, Available: <https://github.com/bh-projects/TI-BH-EDU-board-kit>
- [14] Texas A&M University, “Engineering Project Showcase,” Available: <https://engineering.tamu.edu/student-life/project-showcase/index.html>