Teaching Touch Sensing Technologies through Project-Based Learning

Nannan He[1], Han-way Huang[1] and Ying Qian[2]
[1]Dept. of Electrical and Computer Engineering Technology, Minnesota State University at Mankato
[2]School of Computer Engineering, Nanjing Institute of Technology, China
nannan.he@mnsu.edu and han-way.huang@mnsu.edu

***Abstract*— Compared with conventional keyboard and mouse systems, touch sensing input devices provide more accurate and direct interaction with the user. More and more touch devices appear in various domains. To keep up with this technology change, we have introduced touch sensing technologies to electrical and computer engineering technology students by using the project-based learning (PBL) approach. Our experiences indicate that the PBL approach is efficient and practical for teaching touch sensing techniques. Two different sets of courseware including hardware kits and software packages have been utilized in design projects to teach touch in two existing courses respectively. Students have shown the great interest and the capability in adopting touch devices into their senior embedded systems design projects to improve user interactions with the computing systems.**

Keywords—touch sensing; project-based learning; Microcontroller

I. Introduction

Touch sensing is one of the most exciting fields of technological advances that have the significant effects in our daily life. For instance, the touchscreen which is an electronic visual display panel with a touch responsive surface, has become pervasive in information appliances such as tablets and smartphones. There are a variety of touch sensing technologies for various applications. Analog resistive and projected capacitive (i.e. self-capacitive and mutual capacitive) are the most popular ones in the market [1]. Semiconductor manufacturers have also recognized the trend of using touch device as a direct and immediate human machine interface component, and started to integrate the touch-sensing devices into their products. To keep up with this industry technology change, there is a growing need to equip technology students with the knowledge of touch sensing and prepare them for the professional world. Thus, the main objective of this new teaching practice is to educate students not only the fundamentals of touch technologies but also the practical skills of integrating touch devices into the computing systems design.

To achieve the educational goal, innovative teaching and learning methods are required to make the connection between touch sensing theories and practical application of touch devices. A touch sensing system is typically complex and relates to both software and hardware components. The traditional teaching method consisting of only lecturing, homework assignments and test is no longer a sufficient instructional pedagogy. Project-based learning (PBL) has demonstrated a powerful learning strategy, capable of engaging students and creating an association between theory and practice by teaching concepts through real world problems [2]. Therefore, PBL was chosen as the learning paradigm and implemented to teach touch technologies in two existing undergraduate courses (*Programming Tools* and *Advanced Microprocessor*) offered to both computer engineering technology (CET) and electronic technology (EET) students.

In both courses, students are required to complete a series of design projects by utilizing touch sensing technologies. Each project presents a realistic, open-ended problem which is derived from actual engineering practice. It requires students to relate the problem to the central concepts of the subject, generate alternative solutions and make decisions on the best solution by the perception of problem constraints [2]. The courseware which can encompass hardware development kits, software tools and library packages are all from industry. However, since each course has different student outcomes, different sets of courseware have been adopted for students to design and develop projects. The *Programming Tools* course is required for junior year CET students, but optional for EET students. Students focus on studying the software programming aspect for controlling basic touch components and touch screen but with little knowledge of underlying hardware connection details. Thus, after completing the series of projects, students are expected to demonstrate the capability of utilizing existing software development tools and libraries (i.e., Atmel QTouch Library [3]) to efficiently program touch

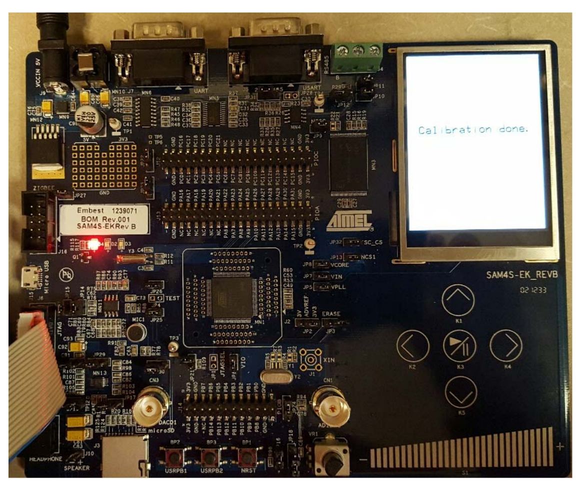
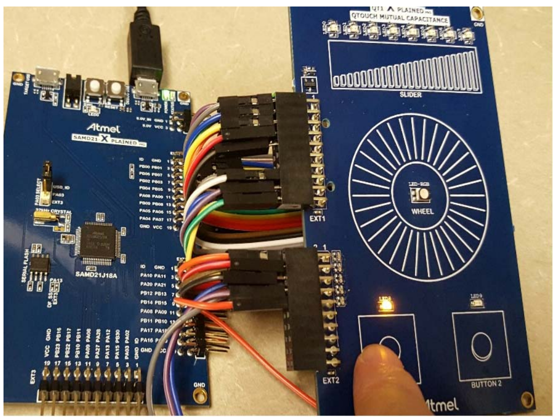


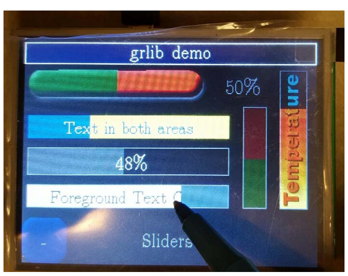
Figure 1. Atmel SAM 4S – EK

devices. Thus, a development board SAM4S-EK from Atmel as shown in Figure 1 has been employed in this course. Multiple touch sensing devices are integrated in this board, including five touch keys, a touch slider and a *resistive* touch screen above a color LCD display.

In comparison, the *Advanced Microprocessor* course is required for both EET and CET students in the senior year. The student outcomes emphasize in studying the electronic fundamentals of touch sensing devices and developing microprocessor-based embedded applications to control various touch devices to solve engineering problems in practice. Thus, two individual touch sensing kits: Atmel QT1 Xplained Pro and an Kentec 3.5inch LCD touchscreen boosterpack as shown in Figure 2 (a) and (b), have been used in this course.



(a) Atmel SAM D21 and QT1 Xplained Pro



(b) Kentec Touchscreen controlled by TI LaunchPad (underneath)

Figure 2. Separate Development boards

A. Related work

In [4], students were taught to construct a functional touch keypad by using household supplies, such as cardboard, aluminum foil, and tape from scratch and program this device work is different as the off-the-shelf touch devices, software packages and tools from industry have been used. Furthermore, the 32-bit ARM-based microcontrollers are used to program these touch devices for complex embedded systems development. This work is based on our previous teaching experience reported in [5], but with two significant enhancements. Firstly, some latest hardware extension boards such as QT1 Xplained Pro and separate touch screen with associated software packages have been added into teaching materials. A comparison of using integrated development board with these separate extension boards in teaching is reported. Secondly, projects have been re-designed by following the principles of a module-based PBL approach for computer engineering curriculum [6, 9]. The main purpose is to facilitate the courseware to be easily integrated by different related courses in CET and EET curriculum such as *embedded systems design*, and even computer engineering and electrical engineering curriculum. And this PBL approach can also guide the design of new teaching materials in the future.

This paper first presents our experiences of teaching touch sensing technologies in two courses offered to technology students, especially the design and development of projects by students who were instructed to utilize touch sensing devices, microcontrollers and associated software tools and library packages from industry. The comparison of two sets of courseware used in two courses is also discussed. The assessment and students' feedback are then reported. Finally, the paper is ended with conclusions and future work.

II. Proposed Approach

The course materials on the subject of touch sensing technologies taught in two courses are all for the 5 weeks period. The new student outcomes added to the existing *Programming Tools (PT)* course and the *Advanced Microprocessor (Micro)* course are summarized in Table 1. O1 is the common student outcome shared in two courses. But for this outcome, the Performance Indicator (PI), which represents the concrete actions the students should be capable of performing as the result of attending the course [10], is different. Comparatively speaking, students in the *Micro* course should demonstrate the deeper comprehension of the electronics theories of touch sensing technologies and stronger justification analysis capabilities of their strengths and weaknesses in the practical applications than the student who complete the *PL* course. The outcome O2 is for the *PT* course only. Students should be able to utilize the existing software tools to design touch sensing devices so as to provide users an alternative manner to interact with the computing system. But they do not need to grasp the connection and interfacing details between touch devices and control hardware components, such as microcontroller as the students in the *Micro* course. The outcome O3 in Table 1 is for the Micro course only. Besides programming touch devices, the student

Table 1. Student Outcomes and PIs

Student Outcomes	PIs
O1 (both *PT and Micro*). Can have an appreciation for the need of popular touch sensing technologies.	*PL*: Identify basic features of resistive and capacitive touch and their respective application fields. *Micro*: State the electronic principles of resistive and capacitive touch sensing technologies and analyze their strengths and weaknesses in the application.
O2 (*PT*) Can design touch sensing components to meet the desired user input needs in the context of electrical and computer engineering.	(*PT*) Apply software tools and library packages to program common touch sensing devices, such as touch buttons, touch sliders and touch screen.
O3. (*Micro*) Can design and develop microcontroller-based embedded systems by using touch devices as user inputs in the context of electrical and computer engineering.	(*Micro*) Use ARM-based microcontrollers to connect via serial or parallel communication channels (i.e. SPI) and program touch devices such as touch buttons, touch sliders and touchscreen.

should be able to manipulate the communication interface between a touch device and a microcontroller for sending controlling data to and reading sensing status from the device.

A. Touch sensing technologies

A typical analogue resistive touch device has layers as shown in Figure 3: 1) a coversheet layer, 2) tiny and transparent *insulating spacer dots*, 3) a rigid back layer made of glass, and 4) an electrically conductive compound coated with both layer 1 and 3. When the surface is touched, it pushes the conductive coating on the coversheet against that on the back glass, making electrical contact. A controller converts the voltages produced to digital coordinates.

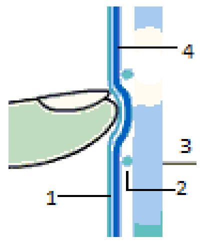


Fig 3. Resistive touch [7]

Projected Capacitive Technology (PCT) is presently the most prevalent touch technology. It detects touch by measuring the capacitance at each *addressable electrode.* PCT has two types of sensing modes: the self-capacitance mode and the mutual capacitance mode [8] as shown in Figure 4. In the self-capacitance mode, when a touch approaches the surface, the increased current due to another path from the electrode to ground is measured. The problem of ghost points with this mode was also discussed in the class [2]. In the mutual capacitance mode, system electronics measure the intersection between the electrodes individually to detect multiple touches during one touch scan. When a finger touches near an intersection, some of the mutual capacitance is coupled to the finger which reduces the capacitance at the intersection as measured by the system electronics.

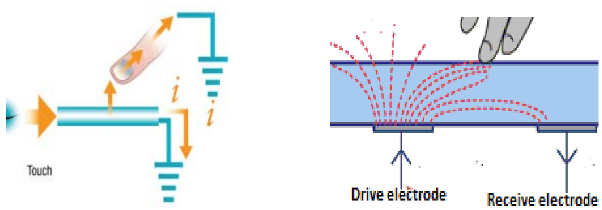


Figure 4 Self Capacitance (left) and Mutual Capacitance (right) (Courtesy of 3M)

In comparison, PCT is more robust to weather extremes and has higher durability than resistive touch. And PCT supports multi-touch. But resistive touch is more cost-effective due to its simple structure.

B. Projects design

Many education researchers argue that practical components and hands-on experiences should be brought into the engineering curriculum to educate students who can develop out-of-the-box yet practical solutions to engineering challenges [11]. Thus, project-based learning models have been increasingly integrated into engineering courses in recently years around the world [12-14]. PBL reflects the theory of constructivism. The central theme of the theory is that learning should be an active process in which learners construct new ideas by getting involved in the process of learning and experience [15]. Applying this principle, PBL offers authentic learning experience and requires students to go through an extended, "hands-on" process of inquiry to a design challenge [11, 15]. When incorporated in the course, PBL can help students develop engineering thinking and intuition and increased their motivation. It also allows students to control their own learning so that students can develop more responsibility and ownership of their study [16]. Additionally, after completing the projects based on the courseware from industry, students can gain real-life experiences, the skills and interpersonal abilities so as to make the graduates more marketable.

Our projects design in both *PL* and *Micro* courses reflects the principles of PBL from following specific aspects.

- Firstly, students are required to complete five projects in sequence in both courses. The complexity of the projects gradually increases by integrating more peripheral functions of microcontroller with touch elements. In particular, the first two or three projects involve programming, analyzing and debugging a single touch component only, such as touch button slider, wheel or touchscreen. The rest of the projects are all about using the touch component as input device to control the functions of other system applications. For example, the capacitive touch slider is used to smoothly control the duty cycle and frequency of the pulse width modulation waveform generation for motor control.
- Secondly, as students are gradually empowered to solve the tasks, we designed the last capstone project, which presents a more complicated and challenging task. It encourages students to apply the knowledge and skills they gained from the previous lab projects to develop solutions to new problems, such as developing a touchable calculator.
- Thirdly, according to the research on motivating the PBL [17], throughout the whole PBL process, students are instructed to run the code they developed to test the results of their design solution. Therefore, students received instant feedback that motivated them to use learning and metacognitive skills to diagnose problems and evaluate overall outcomes.

The teaching of touch technologies in both courses is structured such that part of the classes are used to cover basic concepts and design principles, but most class time is used to discuss and work on the projects either individually or in the 2 to 3 students group. Details of the projects and learning outcomes are presented in Table 2 and 3. All project requirements are general and open-ended. Students can apply any touch element(s) implemented with the same touch technology and any common microcontroller (MCU) function depending on the engineering problem being solved. At the same time, students are still given some example applications as references, and some guidelines of approaching the problem, and the necessary knowledge that is essential to identifying solutions.

Project 1 in the *PT* course corresponds to both outcome O1 and Q2. It not only provides students the direct experience of basic touch devices in practice, but also helps them start with the *capacitive* touch elements programming. The rest projects serve to achieve the outcome O2 from different technical aspects. Project 2 is developing a pure *resistive* touchscreen project. In project 3 and 4, students are required to integrate *capacitive* touch element (s) and the *resistive* touchscreen respectively with one basic microcontroller application, such as pulse width waveform generation for motor control. Finally, students complete a capstone project.

Table 2. Project Design and Learning Outcomes in *PT*

Project description	Learning outcomes
1. Develop a software project to detect the status (i.e. pressed-1 or released-0) of touch buttons and linear position (0~255) of the touch slider on the board, and display these values at the user terminal.	- Identify the data structures that represent the touch keys status, and the position of the slider. - Apply library functions to monitor the status of touch elements.
2. Design a software project to calibrate the touchscreen controller and configure the graphical color LCD controller underneath and draw shapes, lines or texts on the board.	- Use event callback mechanism to sense the touch position on the touchscreen. - Apply library functions for basic graphic LCD operations.
3. Design an MCU project to use capacitive touch elements onboard as inputs to control MCU functions, such as using buttons to initiate and stop the signal generation, and sliders to control the frequency of the signal.	- Apply software packages to program capacitive touch devices to smoothly control the waveform generation, motor control or other basic MCU applications.
4. Design an MCU project to apply the resistive touchscreen onboard to provide a graphic user interface (GUI) to control the signal generation or other basic MCU applications.	- Apply software tools to program the resistive touchscreen as the touchable GUI for the motor control or other basic MCU applications.
5. Design a capstone project to apply more than one touch elements or touchscreen with graphic LCD as user interfaces to develop practical applications.	- Apply at least one touch elements or touchscreen with the graphical LCD to solve a realistic engineering or computing problems.

As shown in Table 3, project 1 in the *Micro* course corresponds to the outcome O1. The rest projects are designed to serve the outcome O2 from main two aspects. Students are required to develop two different modes of capacitance touch in the project 2 and 3 respectively. In project 4 and 5, students need to do design and development on the touchscreen extension board.

C. Selected hardware and software courseware

Nowadays, major microcontroller vendors in industry have started providing their solutions to develop touch applications. Such technical solutions typically consist of following three components.

1) *Touch devices or touchscreen* can be classified into two categories in industry. Some touch components are integrated with microcontroller development boards as one chip. Recently, Even low-cost development boards are equipped with multiple touch elements. For example, the STM32L-152C Discovery board (~$14) from ST Microelectronics integrates a linear touch slider and four touch keys at the bottom of the board. Other touch devices are on individual chips and used as the hardware extensions. For instance, the low power audio capacitive touch Boosterpack from Texas Instruments (~$10) includes touch button and touch wheel for audio control.
2) Software libraries or drivers are used by the application software to easily interface with touch devices.
3) *Facility tools* are applied to assist developing touch-based applications, such as simulation tools to measure and adjust the sensitivity of touch sensors.

Table 3. Project Design and Learning Outcomes in *Micro*

Project description	Learning outcomes
1. Use facility tools associated with touch devices to visualize the real-time touch sensor debugging data and perform the runtime tuning of major touch sensor parameters.	- Identify and tune important touch sensor parameters of a touch device to improve its sensitivity (i.e. detect threshold, detect hysteresis, touch delta etc.)
2. Design an MCU project to use *self-capacitance* touch elements onboard to control the signal generation for driving motors, or other common MCU applications.	- Apply software tools and libraries to program the self-capacitance touch elements (i.e. button, wheel, slider etc.) for common MCU applications.
3. Design an MCU project to use *mutual-capacitance* touch elements onboard to control the signal generation for driving motors, or other common MCU applications.	- Apply software tools and libraries to program mutual-capacitance touch elements (i.e. button, wheel, slider etc.) for basic MCU applications.
4. Design a project to setup the serial or parallel connection between a MCU and a touchscreen, and perform touchscreen calibration and some basic graphical display functions.	- Apply the MCU knowledge and software drivers to configure and program the connected touchscreen and underlying LCD.
5. Design an MCU project to program touchscreen to provide a GUI interface to implement some touchable graphical applications.	- Apply the software tools and libraries to control both resistive touch surface and underlying graphical LCD.

In the *PT* course, the Atmel SAM4S-EK evaluation kit equipped with a 32-bit ARM Cortex-M4 microprocessor, has been chosen as the hardware platform of projects for three reasons. 1). This kit includes rich touch sensing elements as shown in Figure 1: a 2.8" color graphical LCD display with *resistive* touchscreen, five touch keys (UP, DOWN, RIGHT, LEFT and VALID) and a touch slider implemented with the same type of Atmel patented capacitive touch acquisition method called QTouch which is based on the *self- capacitance* sensing mode. The interface details between touch devices and microcontroller are embedded in the kit without much user intervention. 2) Atmel offers both Atmel Software Framework (ASF) which is a microcontroller software library and a QTouch Library for simplifying the development of capacitive touch sensing applications on AVR and ARM microcontrollers. 3) ASF provides nearly one hundred practical example projects for developing various applications on this evaluation kit. Two examples are mostly related to programming the touch elements on board, which are useful to demonstrate students their capabilities in the application and inspire them to apply them in the projects.

In the *Micro* course, the SAM4S-EK kit was found to be not cost-effective and insufficient to develop general touch device projects. The projects are difficult to be migrated to other microcontrollers due to the integrated nature of the SAM4S-EK. And it is hard to support modular design. Therefore, two individual hardware extension boards Atmel QT1 Xplained Pro kit and Kentec touchscreen as shown in Figure 2 are used. The Atmel QT1 Xplained Pro includes both self- and mutual-capacitive two extension boards. In comparison, SAM4S-EK does not have touch elements implemented by using mutual capacitive technology. Atmel QTouch Composer is the software tool used to develop projects and conduct real-time visualization of the touch sensing data for the analysis and debugging purposes. Students apply it to design, develop and test projects from Atmel. This board and the software are used for first three projects. The Kentec touchscreen has 320 *240 pixels, 256K colors and touch. Its interface uses the 8-bit parallel connection. The software development tool is the eclipse-based code composer studio from Texas Instruments. This extension board is used for last two projects.

III. RESULTS AND DISCUSSIONS

We report the capstone projects developed by students in the following. Through these projects, students have shown the capabilities of programming touch devices and applying them into complex MCU projects.

A. Touch controllable servo motor

This project presented a MCU application to drive a Hi-Tec HS-311 servomotor. Users could control the speed of the servo-motor by the touch slider on the SAM4S-EK board and the rotation direction by the touch buttons (RIGHT, LEFT and VALID) on board. The RIGHT key controls rotating counter-clockwise, the LEFT key controls the motor's rotation as clockwise, and the VALID key in middle starts or stops the rotation. The graphic LCD screen of the microcontroller is also applied to display the status of touch elements. The slider is represented as a rectangle at the bottom of the screen whose filled area corresponds to the position of slider and also the speed of the servomotor. Figure 5 displays the test setup and its GUI on the touchscreen. Students showed high interests in using touch to flexibly and smoothly control the operation of motor.

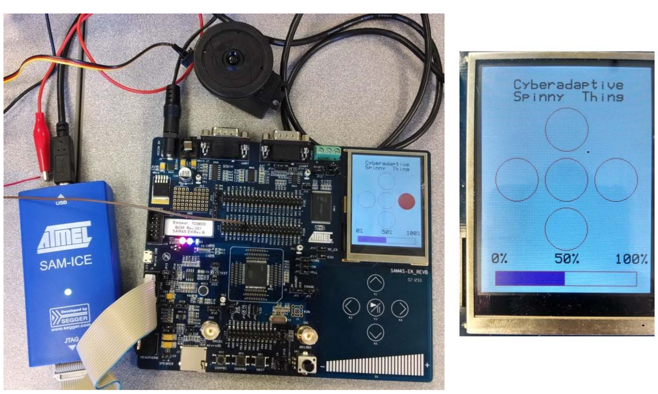


Figure 5. Capstone 1 lab setup and GUI

B. Touchable calculator

This project created a basic touchable calculator on the resistive touchscreen. Ten numbers 0-9, four basic arithmetic operators "/", "x", "-" and "+" and two control symbols "AC" and "=" are drawn within sixteen circles on the screen. To enter an operand, user touches on the corresponding circle by stylus, for example the circle labeled with "3". The entered number is then displayed on the top bar as shown in Figure 6. The program then reads the operator and the second operand from the user in the same manner. After the user touches the "=" symbol, the result of the operation is displayed on the top. The program cleans the memory and starts reading new entries from the users by pressing the circle labelled with "AC".

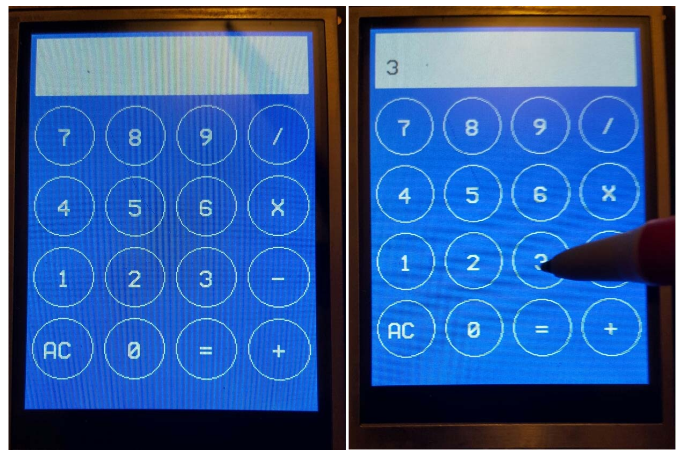


Figure 6. Touchable Calculator

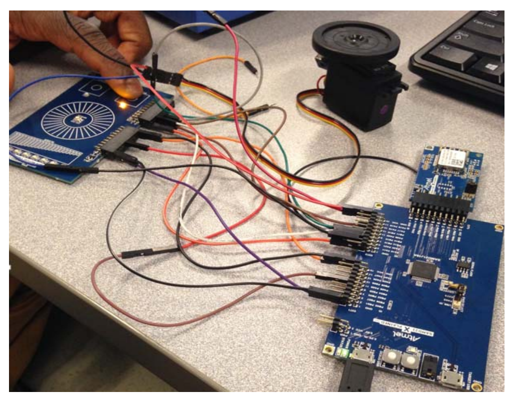


Figure 7. Hardware setup for capstone project C

C. Cyber *adaptive rotating thing for Internet of Things (IoT)*

This project is a basic IoT with some touch application by the use of SAM D21 MCU board (right), QT1 mutual capacitance touch (left) and a WINC1500 extension for the WiFi connection (right top). Users could control the speed of the servo-motor by the touch slider on the QT1 board and its start (i.e. button pressed) and stop (i.e., button released) by two touch buttons on the board. The status of touch elements can be monitored in locally and remotely in two ways. The first is to turn on or off the corresponding LEDs. For example, the LED besides a touch button lights when the button is pressed as shown in Figure 7. Another is to use an APP installed on the Android smart phone to show the status of all touch elements on the QT1 board. The status data are first sent to a wireless access point via the WiFi extension boarded connected with the SAM D21 MCU, and then from the access point to the connected smart phone.

IV. Evaluations

Multiple methods have been used to formally assess the teaching effectiveness of touch technologies integrated in two existing courses, including the graded projects and assignments, and course evaluation surveys conducted at the end of the semester. Students commented that the project design format improves their learning in both courses. Results indicating that the level of understanding and applying touch technologies and instructor's help in two courses are presented in Figure 8 and 9 respectively.

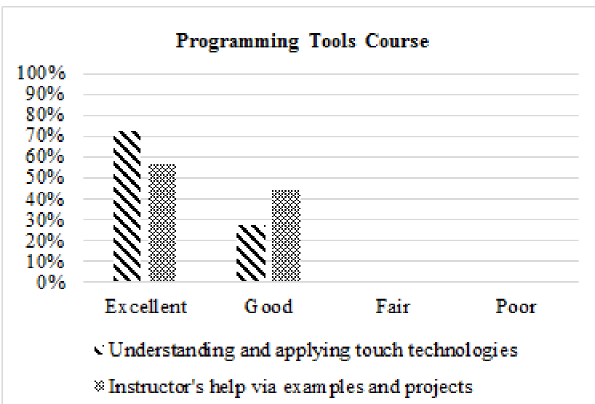


Figure 8. Students' assessment in the *PT* course

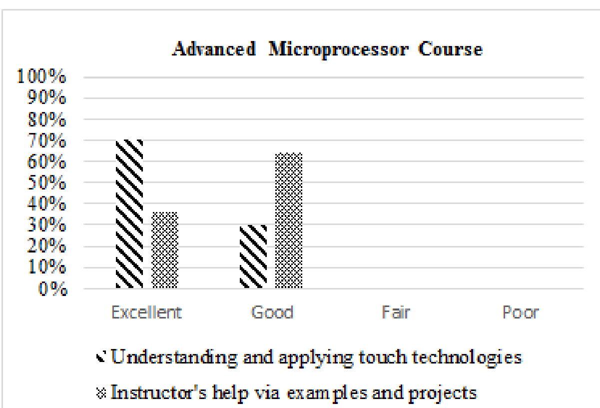


Figure 9. Students' assessment in the *Micro* course

The survey questions details are given in Table 4. The two percentage numbers in each cell represent the results from the *PT* and *Micro* course in the form of *PT/Micro*. An additional survey question is given to students who took both *PT* and *Micro* courses. "*Which set of hardware boards is more helpful in learning touch technologies and why?*" Students feedbacks are summarized as the following: The SAM4S-EK with its software tools is easier and quick to develop touch applications prototype, but much harder to integrated with other embedded applications. The projects with the individual touch components require students to do some research on the interface between MCU and touch elements. Such experiences equip students more practical knowledge of applying touch elements with different MCUs. From the instructor point of view, the individual touch boards are independent and flexible in teaching. We could choose a different touchscreen board without modifying the teaching materials for the QT1 board. Furthermore, they are more cost-effective. The SAM4S-EK is about $100. The total price for the second set is around $65.

Table 4. Summary of student feedbacks

Survey Questions	**Excellent [%]**	**Good [%]**	**Fair [%]**	**Poor [%]**
Understanding and applying touch technologies	*(PT/Micro)* 75/70	25/30	0/0	0/0
Instructor's help via examples and projects	60/40	40/60	0/0	0/0
Appreciation for popular touch sensing technologies and their applications	90/85	10/15	0/0	0/0
Willingness to use touch elements in senior or other course projects	85/80	10/10	5/10	0/0
Confidence in applying touch elements in senior or other course projects	70/75	20/20	10/5	0/0
Software tools and hardware boards in practicing touch elements development	75/70	25/15	0/15	0/0
Use of projects in helping study touch technologies	90/90	10/10	0/0	0/0

V. Conclusions and Future Work

This paper presents our experiences of applying the PBL approach to teaching technology students the touch sensing technologies in two existing courses. The increasing number of low-cost MCU development kits embedded with touch elements has become available for teaching touch technologies. The two sets of hardware boards and the associated software tools and libraries have been employed to teach students to configure, program, debug and test touch devices in the industry. In the future, we will encourage students to develop more complex MCU projects with touch devices.

References

[1] Geoff Walker, "Fundamentals of Touch Technologies", 2013. www.walkermobile.com/Touch_Technologies_Tutorial_Latest_Version.pdf

[2] J. W. Thomas. "A review of research on project-based learning". http://www.bie.org/index.php/site/RE/pbl_research/29, 2000.

[3] Atmel QTouch Library: http://www.atmel.com/products/TouchSolutions/touchsoftware/default.aspx

[4] M.B Pertle, C.E. Davis, and J.E. Ruyle, "Innovative Engineering Outreach: Capacitive Touch Sensor Workshop", *Proceedings of ASEE Annual Conference, Indianapolis, IN*, June, 2014.

[5] N. He, H.W Huang, "Teaching Advanced Touch Sensing Technologies Using ARM Cortex-M based Microcontroller", Proceedings of the ASEE Annual Conference, Seattle, 2015.

[6] T. Yakhno, E. Ekin and T. Aktuglu, "Work in progress - Module-Based Active Learning approach for introductory level of Computer

Engineering curriculum," *2009 39th IEEE Frontiers in Education Conference*, San Antonio, TX, 2009, pp. 1-2.

[7] White paper – “Comparison All Resistive Touch Technologies”, 2014. http://www.elotouch.com/Technologies/compare_resist.asp

[8] Touch Technology Brief from 3M Touch Systems, “Projected Capacitive Technology”, 2013.

[9] K. Qian, X. Hu and C. T. Dan Lo, "Work in progress — Integrate embedded systems into CS curriculum with labs-in-a-box courseware," *Frontiers in Education Conference (FIE), 2010 IEEE*, Washington, DC, 2010, pp. T2F-1-T2F-2.

[10] G. Roger, “Student Outcomes and Performance indicators”, 2010. http://www.abet.org/wp-content/uploads/2015/04/program-outcomes-and-performance-indicators.pdf.

[11] K. J. Chua, W. M. Yang, and H. L. Leo. “Enhanced and conventional project-based learning in an engineering design module.” *International Journal of Technology and Design Education, 24*, 437-458, 2014.

[12] S. E. de Lucena and O. Mogames, "Unveiling fieldbus network technology through project-based learning," *Frontiers in Education Conference (FIE), 2012*, Seattle, WA, 2012, pp. 1-5.

[13] J. Dong, X. Qin and P. Chen, "Enhancing collaborative project-based learning using participatory design approach," *Frontiers in Education Conference (FIE), 2014 IEEE*, Madrid, pp. 282-285.

[14] Z. Avery *et al*., "Implementing Collaborative Project-Based Learning using the Tablet PC to enhance student learning in engineering and computer science courses," *Frontiers in Education Conference (FIE), 2010 IEEE*, Washington, DC, pp. F1E-1-F1E-7.

[15] H. Jackson, K. Tarhini, B. Maggi and N. Rumsey, "Improving students understanding of engineering concepts through project based learning," *Frontiers in Education Conference (FIE)*, Seattle, 2012, pp. 1-6.

[16] M. Frank, I. Lavy, and D. Elata. “Implementing the project-based learning approach in an academic engineering course”. *Intl. Journal of Technology and Design Education, 13*(3), 273–288, 2003.

[17] P.C. Blummenfeld, E. Soloway, R.W.Marx, J. S. Krajcik, M. Guzdial, and A. Palincsar. “Motivating project-based learning: Sustaining the doing, supporting the learning”. *Educational Psychologist, 26*(3/4), 369-398, 1991.