



Teaching Advanced Touch Sensing Technologies Using ARM Cortex-M based Microcontroller

Dr. Nannan He, Minnesota State University, Mankato

Nannan He is an Assistant Professor of the ECET Department in Minnesota State University at Mankato. She received the Ph.D. in computer engineering from Virginia Tech. Her teaching and research interests are in safety-critical embedded software, real-time systems, and software verification.

Dr. Han-Way Huang, Minnesota State University, Mankato

Teaching Advanced Touch Sensing Technologies Using ARM Cortex-M based Microcontroller

Abstract

This paper presents our experiences of introducing in a senior level microprocessor course the latest touch sensing technologies, especially programming capacitive touch sensing devices and touchscreen. The emphasis is on the teaching practice details, including the enhanced course contents, outcomes and lecture and lab organization. By utilizing the software package provided by Atmel, students are taught to efficiently build MCU-based embedded applications which control various touch sensing devices. This work makes use of the 32-bit ARM Cortex-M4 microprocessor to control complex touch sensing devices (i.e., touch keys, touch slider and touchscreen). The Atmel SAM 4S-EK2 board is chosen as the main development board employed for practicing the touch devices programming. Multiple capstone projects have been developed, for example adaptive touch-based servo motor control, and calculator and games on the touchscreen. Our primary experiences indicate that the project-based learning approach with the utilization of the selected microcontroller board and software package is efficient and practical for teaching advanced touch sensing techniques. Students have shown the great interest and the capability in adopting touch devices into their senior design projects to improve human machine interface.

1. Introduction

Modern touch sensing devices allow the rapid, accurate and direct interaction with the user, which conventional keyboard and mouse systems cannot offer. Many touch sensing technologies are currently available. Among them, the analog resistive, surface capacitive, projected capacitive, infrared grid, optical imaging, and surface acoustic wave are the most important ones. The touchscreen, an electronic visual display controlled by touching the screen, is popular in many information appliances like tablet computers and smartphones. Semiconductor manufacturers have also recognized the trend of using touch device as a highly desirable user interface component, and started to integrate the touch-sensing technology into their products. To keep up with this industry technology change, we prepared a series of lecture and lab projects on touch sensing technologies for an advanced level microprocessor course provided to electrical and computer engineering students.

Different from the previous work on teaching simple capacitive touch device with the 8-bit Arduino board⁵, this new course extension covered the advanced touch sensing relevant topics, such as the history of touchscreen, the fundamental electronic principles underlying each of major touch sensing technologies, and the application issues for developing microcontroller-based embedded systems which utilize touch devices. Resistive touch and capacitive touch (i.e.

surface and projected) were taught in detail. To deepen students' understanding of the advantages and disadvantages of these touch technologies, the comparisons among these technologies and their applications in real-world electronic devices were discussed. Among these topics, this course focused on teaching students to program various touch sensing devices through a series of project examples, from the simple ones which use only one kind of touch device to more complex ones which involve multiple touch devices. The project-based learning was employed as the main teaching approach.

The Atmel SAM 4S-EK2 board was the main development board employed for practicing the touch device programming. The microprocessor employed on this board is the 32-bit ARM Cortex-M4. This board includes rich touch sensing devices, for instance five touch keys and a touch slider implemented which utilize the capacitive touch sensing technology, and a resistive touchscreen on a color LCD display. Furthermore, Atmel provides a royalty free and open-source software library called Atmel Software Framework which includes a complete set of library functions for developing various microcontroller applications in C. This framework includes the QTouch library which provides Application Programming Interfaces (APIs) for programming Touch elements in particular. Students learned to utilize this framework so as to provide touch sensing capability in their projects. Students have shown great interests in this new topic and are capable of incorporating touch devices to improve the human machine interface of their senior design projects.

This paper mainly presents our primary experience of teaching advanced touch sensing technologies to engineering and technology students, with the emphasis of how we designed the lab projects with the utilization of Atmel development board and software packages. The description of this course is first given, including learning outcomes, course contents and projects organization. Next, a survey of important touch sensing technologies is reported. Then, we present the lab projects and capstone projects developed by students using the selected Atmel development board integrated with touch devices and the software package to program these devices. Finally, the paper is ended with conclusions and future work.

2. Course description

The overall goal of this new teaching practice is to equip students with the knowledge of advanced touch sensing technologies and developing microcontroller-based applications involving various touch sensing devices to solve engineering problems in practice. We taught the added course materials in 5 weeks, two hours of lecture time and three hours of laboratory per week. It has three major objectives.

- To improve students' awareness of common and different features among major touch sensing technologies.

- To introduce students existing popular touch sensing devices including touchscreen and relevant software development packages.
- To enable students to develop microcontroller-based systems to program touch sensing devices.

Course learning outcomes

In the design of this course, we identify course learning outcomes that stem from the overall teaching goal discussed before and extend to learning activities. We developed seven learning outcomes, grouped into three top outcomes.

1. To demonstrate the basic knowledge of advanced touch sensing technologies
 - 1.1 To understand the working mechanism of important touch technologies (i.e. resistive and capacitive touch)
 - 1.2 To articulate different features of various touch technologies and their main application fields.
2. To demonstrate the capability of designing touch sensing application
 - 2.1 To comprehend the composition of basic touch sensing devices (i.e., QTouch slider and keys from Atmel)
 - 2.2 To get familiar with the software tools and packages (i.e., Atmel QTouch Library) for programming touch devices
3. To demonstrate the capability of developing microcontroller-based touch applications
 - 3.1 To utilize software packages for developing simple touch sensing applications on ARM microcontrollers
 - 3.2 To utilize software packages for developing touchscreen applications on ARM microcontrollers
 - 3.3 To utilize touch devices to improve human-machine interface experiences of existing microcontroller-based embedded systems

Course contents

The topics covered in this course extension include origin of touchscreen, fundamental electronic principles, construction, features and benefits of touch sensing technologies such as analog resistive touch, surface capacitive touch and projected capacitive touch, comparison among popular touch sensing technologies, Atmel QTouch keys and slider, resistive touchscreen on a color LCD display, Atmel QTouch library, and the design and development of touch applications on ARM Cortex-M microcontrollers. Course materials were drawn from white papers, user manuals, example projects, technical reports and presentations¹⁻⁴ from major touch device vendors (i.e. Elo TouchSystems and 3M) and semiconductor companies which provided touch solutions. To the best of our knowledge, there is no textbook which introduces Touch sensing technologies and microcontroller-based development. Table 1 shows the classification of these

topics. In this table, the time schedule of each topic is given within the five weeks timeframe. This enhanced course is currently offered to senior students in the fall semester. Students are expected to have the basic knowledge of MCUs and programming in C.

Table 1. Course topics

1. Fundamentals (week 1)
Evolution of touchscreen
Analog resistive touch sensing (Four/five/eight wire resistive touchscreen)
Surface capacitive technology
Projected capacitive technology (Self-capacitance and mutual-capacitance)
Infrared grid & optical imaging touch sensing
Comparison of touch technologies
2. Touch device programming (Week 2)
Atmel capacitive touch acquisition methods (QTouch and QMatrix)
Atmel Touch devices (QTouch keys, sliders, wheels)
Resistive touchscreen
Atmel Software Framework
Atmel QTouch library
3. Microcontroller-based touch application development (week 3, 4)
Touch devices (i.e., key, slider) combined with UART development
Touch devices (i.e., key, slider) combined with GPIO development
Touch devices (i.e., key, slider) combined with PWM development
Basic touchscreen and graphical LCD development
Touchscreen combined with touch devices development
4. Final projects development (3, 4, 5)

Students started preparing their final project from week 3, including designing the project and completing the project proposal report. During week 4, five basic touch application lab projects were introduced to students, from the project creation, library modules integration, code editing and debugging, project analysis. These projects played the important role for our project-based learning. They inspired and demonstrated students the capability and applications of touch devices. Furthermore, these examples laid the programming foundations for students to develop their final projects which are more complex projects targeted at solving practical engineering problems.

3. Touch Sensing Technologies

In 1967, E.A. Johnson at the Royal Radar Establishment, UK published an article on the capacitive touchscreen. It is believed by most the world's first touchscreen. In 1974, Dr. Hurst with his company Elographics produced the first touchscreen featuring a transparent surface. And the following year, they patented a resistive touchscreen technology. In 1983, Hewlett-

Packard introduced the HP-150, a home computer with touchscreen technology. It had a built-in grid of infrared beams across the front of the monitor which detected finger movements. In 1990s, smart phones and handhelds with touch technology were introduced. In 1993, Apple released the Newton PDA, equipped with handwriting recognition; and IBM released the first smart phone called Simon, which featured fax function, and a touchscreen interface that allowed users to dial phone numbers. In 2002, Microsoft introduced the Windows XP Tablet edition and started its entry into touch technology. In 2007, Apple introduced iPhone with touch technology. Since then, the touchscreen became wildly popular on handhelds.

Touch Sensing Technologies

A touch device typically consists of three components:

- The **touch sensor** is a panel with a touch responsive surface. Systems are built based on different types of sensors: resistive (most common among touchscreen), surface acoustic wave, and capacitive (most smart phones). In general, sensors have an electrical current running through them and touching the screen causes a voltage change. The voltage change signals the location of the touching.
- The **controller** is the hardware that converts the voltage changes on the sensor into digital signals the computing device can recognize.
- The **software** interprets sensor signals and the information coming from the controller, and allows the computer or smart phone to react accordingly.

Analog resistive touch sensing

An analog resistive touchscreen typically consists of two layers as shown in Figure 1, a coversheet layer made of Polyethylene (PET) and a rigid back layer made of glass. Both layers are coated with an electrically conductive compound called **Indium Tin Oxide (ITO)**. The coversheet has a flexible, durable coating on the outer side. These two layers are separated with tiny and transparent insulating spacer dots.

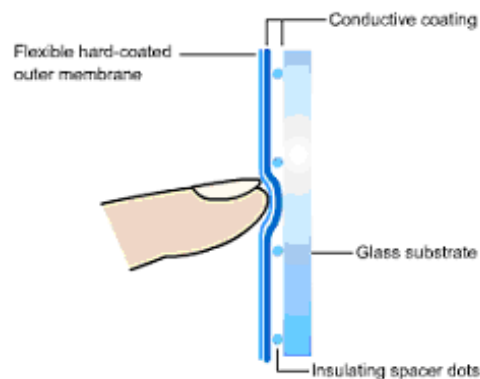


Figure 1. Illustration of Resistive Touch (Courtesy of Elo TouchSystems)

When the screen is touched, it pushes the conductive coating on the coversheet against the coating on the back glass, making electrical contact. The voltages produced are the analog representation of the position touched. An electronic controller converts these voltages into digital X and Y coordinates which are transmitted to the host computer for the software to find the relative position of the touch activation. The resistive touch technology has several implementation variations, such as the 4-, 5-, 6-, 7- and 8-wired touchscreen. While these variants are constructed similarly, they are different in how they determine the coordinates of the touch¹. 4-, 5- and 8-wired resistive touchscreen including their common application in various touch devices were introduced in this course.

In summary, resistive touch sensors are cost effective to manufacture because they have simple structures and offer a high degree of accuracy. They are also the most widely used touch sensing technology. A disadvantage of resistive touch technology is shorter lifetimes due to mechanical wear and tear, which makes it not very suitable for harsh environments. And this technology is susceptible to temperature and moisture extremes.

Capacitive touch sensing

A capacitive touchscreen panel generally consists of an insulator such as glass, coated with a transparent conductive material, such as ITO to hold an electrical charge. The human body is an electrical conductor, so touching the screen with a finger distorts the screen's electrostatic field, which can be measured as a capacitance change at the location of the touch. The location is then sent to the controller for processing.

Surface-Capacitive Touch

This technology works on the basis of a person or conductive stylus creating a path for an electrical current to flow from the surface of a touchscreen to ground. Specifically, the human body has inherent impedance ranging from about 20k to 300k Ω , which is much smaller than the inherent leakage impedance from the touchscreen to ground. Electrodes around the panel's edge evenly distribute a low voltage across the conductive layer, resulting in a uniform electrostatic field over the conductive layer. Whenever the screen is touched, the current is drawn from each corner and flows through the finger, through the person's body impedance, then through a body-to-ground impedance and back to the ground as shown in Figure 2. The touch current flow is very small—in the range of 20 to 500 μA . This current exceeds the touch

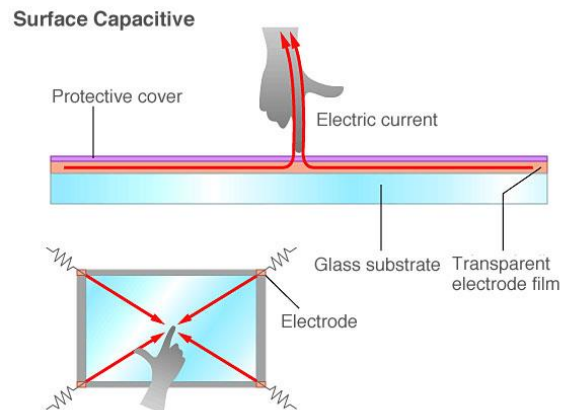


Figure 2 Surface Capacitive

threshold set by the controller and the position of the touchdown location is precisely calculated as being inversely proportional to the distance from the contact point to the corner.

Overall, the surface capacitive touchscreen is commonly used within the gaming industry. Its function is unaffected by moisture, temperature, and dust. Since there is no flexing of the surface, the actuations are very high, capable of reaching over 100 million times, which implies better durability. The drawback of this technology is that it is only a single touch. Moreover, it has low resolution, and is prone to false signals from parasitic capacitive coupling, and needs calibration during manufacture. The user will also have to use a finger or conductive stylus, the passive stylus cannot be used for surface capacitive touchscreen.

Projected-Capacitive Touch

Projected Capacitive Technology (PCT) is fast becoming one of the most prevalent touch technologies for an expanding variety of applications ranging from consumer devices to commercial applications. PCT detects touch by measuring the capacitance at each *addressable electrode*. When a finger or a conductive stylus approaches an electrode, it disturbs the electromagnetic field and alters the capacitance. This change in capacitance can be measured by the electronics and then converted into X, Y locations that the system can use to detect touch. There are two main types of sensing methods, self-capacitance and mutual capacitance². To sense touch in a self-capacitance implementation, the electronics scans through each electrode and measures the amount of current on each electrode to establish a steady-state current. When a touch approaches the screen, the electrons couple to the electrodes and increases the current as it creates a path to ground. As shown in Figure 3, the electronics measure the current on each electrode to ground in this implementation. This method produces a stronger signal than mutual capacitance, but it is unable to resolve accurately more than one finger, which results in misplaced location sensing. Two different touch detection manners – multi-pad and rows-and-columns were introduced to students. The problem of ghost points with this technology was also discussed in the class.

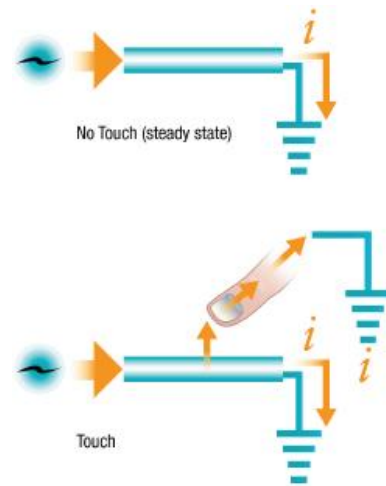


Figure 3 Self Capacitance
(Courtesy of 3M)

Mutual capacitance is the capacitance between two "charge holding objects." Projected capacitance touchscreens intentionally create mutual capacitance between elements of columns and rows in the vicinity where each intersect the other. This allows the system electronics to measure each node (intersection) individually to detect multiple touches on the screen during one screen scan.

When a finger touches near an intersection, as shown in Figure 4, some of the mutual capacitance between the row and column is coupled to the finger which reduces the capacitance at the intersection as measured by the system electronics. This reduced capacitance crosses the "touch threshold" set by the electronics indicating a touch has occurred.

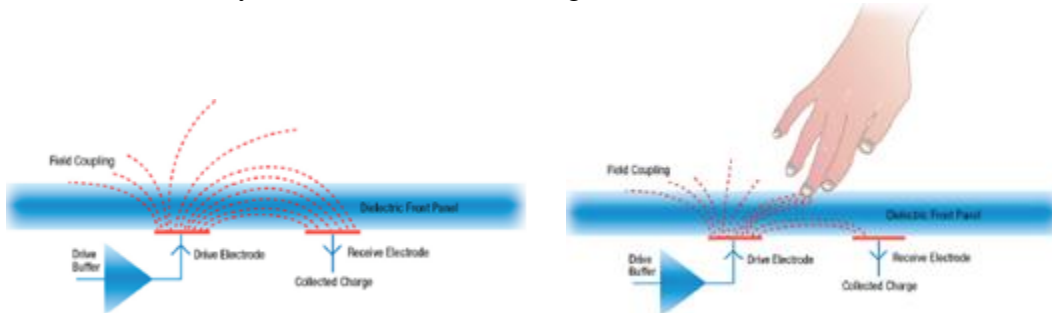


Figure 4 Mutual Capacitance

Touch technologies comparisons

We compared common touch sensing technologies, and rank them (5 – best, 1 – worst, 0 – not applicable) based on a set of features as shown Table 2³.

Table 2. Technologies Comparison

Features	Resistive (4-wire)	Surface Capacitive	Projected Capacitive	Surface acoustic Wave	Infrared
Ambient Light Sensitivity	5	5	5	5	3
Calibration Stability	2	2	5	4	5
Cost	5	3	2	4	1
Durability	1	4	5	4	5
Multi-Touch	0	0	5	0	5
Weather Resistance	2	3	5	2	2
Best applicable size	2''-26''	12''-26''	2''-10''	12''-26''	12''-24''
Mobile/Handheld	5	0	5	0	0

4. Use of Atmel touch devices and ARM Cortex-M microcontrollers on projects

In this section, we present our experience of teaching engineering and technology students the knowledge of touch technologies from two aspects: development board selection, and lab projects and capstone projects.

Microcontroller board selection

As the computing capability of MCUs (for example, the ARM Cortex-A and Cortex-M series) increases dramatically, major microcontroller vendors started providing their solutions to develop touch applications. Such technical solutions typically consist of following components: 1) touch devices or touchscreen either embedded in some microcontroller boards or as individual

chip (i.e., a Booster Pack from TI); 2) software library to program the touch devices; 3) facilities to assist developing touch-based applications, such as simulation tools to measure and adjust the sensitivity of touch sensors. Nowadays, many low-cost development boards are equipped with touch elements that can be used for the education purpose. For example, the STM32L1 discovery board from ST Microelectronics includes a linear touch slider and four touch keys. Freescale's FRDM-KL25Z development platform provides a capacitive touch slider; Texas Instrument offers low power touch devices for keys, sliders, wheels and proximity applications. Atmel offers two types of patented capacitive touch acquisition methods, QTouch and QMatrix. QTouch is based on the self-capacitance sensing technology. Placing a finger on the touch surface introduces external capacitance that affects the flow of electron charge at the touch point. It supports the detection of proximity of a finger rather than absolute touch and is easy to tune sensitivity. In comparison, QMatrix detects a touch by a scanned passive matrix of electrode sets, which belongs to the mutual-capacitance sensing technology category. It is more resilient to moisture & environment changes.

In this course, the Atmel SAM4S-EK2 evaluation kit, which is equipped with a 32-bit ARM Cortex-M4 microprocessor, was chosen as the development board for conducting lab projects due to three main reasons. Firstly, this kit includes rich touch sensing elements: a 2.8" color graphical LCD display with resistive touchscreen, five touch keys (UP, DOWN, RIGHT, LEFT and VALID) and a touch slider. Both touch keys and sliders utilize the QTouch acquisition method. The touch keys use five pairs of IO pins of MCU to detect their states (i.e., pressed or released); the slider uses three pairs of IO pins to detect a linear finger displacement on a sensitive area. Secondly, Atmel offers a QTouch Library as the free software library for simplifying the development of capacitive touch sensing applications on AVR and ARM microcontrollers⁴. It provides multiple library files for each touch device such as key, slider and wheel. Using the library files, the host application can easily make touch measurements and determine the status of touch sensing elements. Moreover, The Atmel Software Framework (ASF) provides a rich set of software modules classified as *boards*, *drivers*, *components* and *services* to program microcontrollers. In fact, QTouch library is included in ASF as a *service* module. And the software tool ASF wizard facilitates importing the QTouch Library and linking together other driver or service modules of an application project. Thirdly, Atmel provides around one hundred example projects for developing various applications on the SAM4S-EK2 kit. Two examples are mostly related to the Touch elements development. One is sensing the status of touch keys and slider on board; another is calibrating the resistive touchscreen, and displaying the touching position on the terminal after the successful calibration. In this course, we developed four more projects for this evaluation kit to demonstrate the application of QTouch elements and touchscreen in the ARM-based embedded systems. Students could not only quickly learn programming microcontrollers and touch devices from these example projects, but also use them as the basis to develop their own applications. More details of these projects and their usage in this course are presented in the following.

Lab projects and capstone projects

The lab projects that we prepared for this course serve different teaching purposes. We organized them into two groups according to the complexity of the projects applied them to the course in two phases. Projects are graded based on the project report and presentation. Table 3 gives an overview of four basic lab projects that were introduced to students in the first phase.

Table 4 Phase I lab projects

Project Name	Function description	Learning purposes
Getting_started	<ul style="list-style-type: none"> - Blink on-board LED at fixed rate. - Display messages on a terminal application via serial UART function 	<ul style="list-style-type: none"> - Create, build and debug projects for the 4S-EK2 board in Atmel Studio; - Get familiar with ASF and ASF wizard - Program core peripherals of ARM Cortex-M4 microprocessor.
QTouch	<ul style="list-style-type: none"> - Detect the status (i.e., pressed or released) of five Qtouch keys and slider on board. - Output Qtouch values when touching via serial UART IO. 	<ul style="list-style-type: none"> Study QTouch library. - Use the C structure that holds the status of touch keys, and linear positions of slider. - Use API functions for configuring and monitoring the status of QTouch elements.
Graphic_LCD	<ul style="list-style-type: none"> - Configure the graphical LCD controller on-board. - Draw text, image, basic shapes (i.e., line, rectangle, circle) on LCD. 	<ul style="list-style-type: none"> Study library for graphical LCD - Use API functions to initialize LCD and turn on the backlight. - Use API functions to choose foreground color. - Use API functions to draw basic shapes on LCD.
Touchscreen	<ul style="list-style-type: none"> - Configure and initialize the touchscreen controller as well as the underlying graphical LCD. - Calibrate touchscreen, then output the pen position on touchscreen at the terminal. 	<ul style="list-style-type: none"> Study library for resistive touchscreen - Use APIs for configuring touchscreen. - Use APIs for the touchscreen calibration. - Use the event_handler() as the callback when the pen is pressed on touchscreen.

We first introduced student four basic lab projects which utilize only one kind of touch element. The lab manual for these projects provides the step-by-step instructions about creating an ASF board project for the SAM 4S-EK board, adding into the project the ASF software library modules, building and debugging the projects. The major ASF software library modules we used includes standard serial I/O driver, Atmel QTouch services, and components for touchscreen programming (e.g., backlight controller, LCD controller, resistive touch service). Through the practices in these phase I lab projects, students could have a better understanding of the usage of common API functions in these library modules.

Phase II projects are summarized in Table 5. These projects were used to as examples to show students how to develop common MCU projects with the integration of touch elements. Students could also use them as the basis to design and implement their final course projects.

Table 5 Phase II lab projects

Project Name	Function description	Learning purposes
QTouch_PWM	<ul style="list-style-type: none"> - Use one QTouch key to start and stop the PWM generation. - Use QTouch slider to control the duty cycle of generated PWM waveform. (Left and right poles of the slider represents 0 and 100% duty cycle respectively.) 	<ul style="list-style-type: none"> - Study how to use the status of Touch sensing elements to configure the application of timer and PMW functions. - Prepare to develop complex projects, such as using touch device to smoothly control servo motor speed.
touchscreen_GPIO	<ul style="list-style-type: none"> - Draw two circles on the touchscreen with different blue/green colors. - Detect the touch on two circles: Touching on the blue/green circle toggles the on-board blue/green RGB led respectively. 	<ul style="list-style-type: none"> - Study how to integrate touchscreen into the conventional GPIO applications of MCU.
touchscreen_LCD	<ul style="list-style-type: none"> - Adjust the four corner positions of the calibration area. - After calibration, draw rectangles /circles from the center of touching position with fixed size (width and height) / radius. 	<ul style="list-style-type: none"> - Study how to use touchscreen in the common GPIO applications of MCU.
touchscreen with QTouch	<ul style="list-style-type: none"> - Draw a circle at the center of LCD. - Use four QTouch keys to move the circle towards up, down, left and right at a fixed distance. 	<ul style="list-style-type: none"> - Study how to develop applications which combine both QTouch features with touchscreen features.

We briefly introduce three 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 in practice.

Cyber adaptive spinny thing

This project presented a MCU application to drive a Hi-Tec HS-311 servomotor. Users could control the speed of the servomotor by the QTouch slider on the SAM4S-EK2 board and the rotation direction by the QTouch keys (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 QTouch 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(a) shows the wiring diagram, (b) displays the test setup and (c) GUI. Students showed high interests in using QTouch to flexibly and smoothly control the operation of motor.

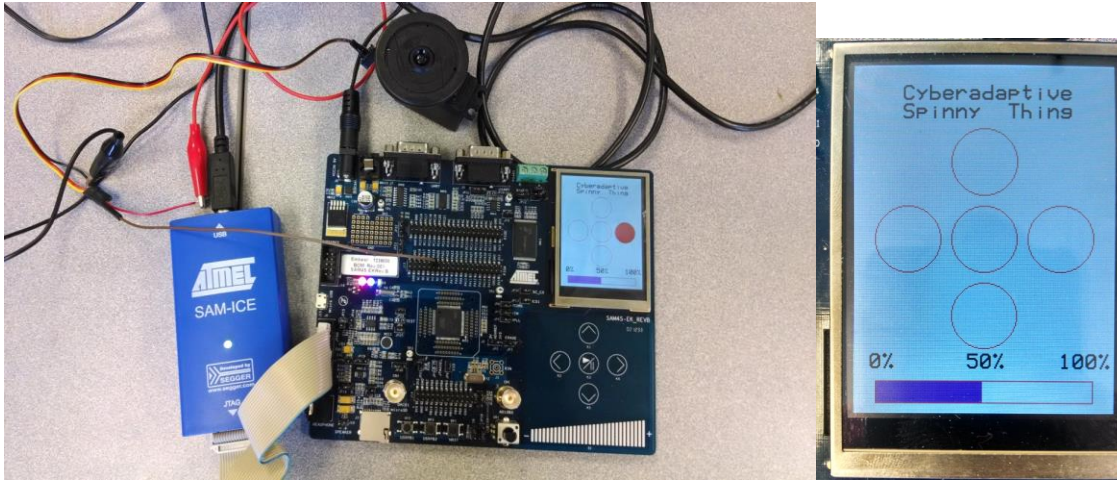
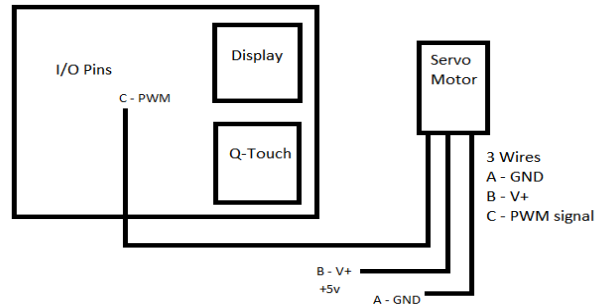


Figure 5 Capstone project 1 (a) Design diagram, (b) Test setup, (c) GUI

Touchable calculator

This project is about creating a basic touchable calculator on the resistive touchscreen on the SAM 4S-EK2 board. Ten numbers 0-9, four basic arithmetic operators “/”, “x”, “-” and “+” and two control symbols “AC” and “=” are drawn within sixteen squares on the screen. To enter an operand, user touches on the corresponding square by stylus, for example the square labeled with “5”. 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 square labelled with “AC”.



Figure 6 Capstone project 2

Miniature TIC-TAC-TOE game

This project aims at creating a simple version of the Tic Tac Toe game with two players using the touchscreen and QTouch keys. Once the board is powered on, a user menu is shown on the screen as in Figure 7 (a). Then, the user calibrates the touchscreen by precisely pressing red dots at each corner using stylus. After the successful calibration, the game starts off with a grid consisting of two vertical lines and two horizontal lines as shown in Figure 7 (b). User 1 could first press the QTouch key K1 (TOP), afterwards, when a cell of touchscreen is touched, the program draws a circle in blue within that cell. Then, User 2 could start his/he move by pressing the QTouch key K3 (MIDDLE), a circle in black is drawn within the area of the cell that has been touched. In this project, Qtouch keys are used for switching between two players, displaying game rule and clearing touchscreen. Figure 6 (b) shows the results after two movements from players.

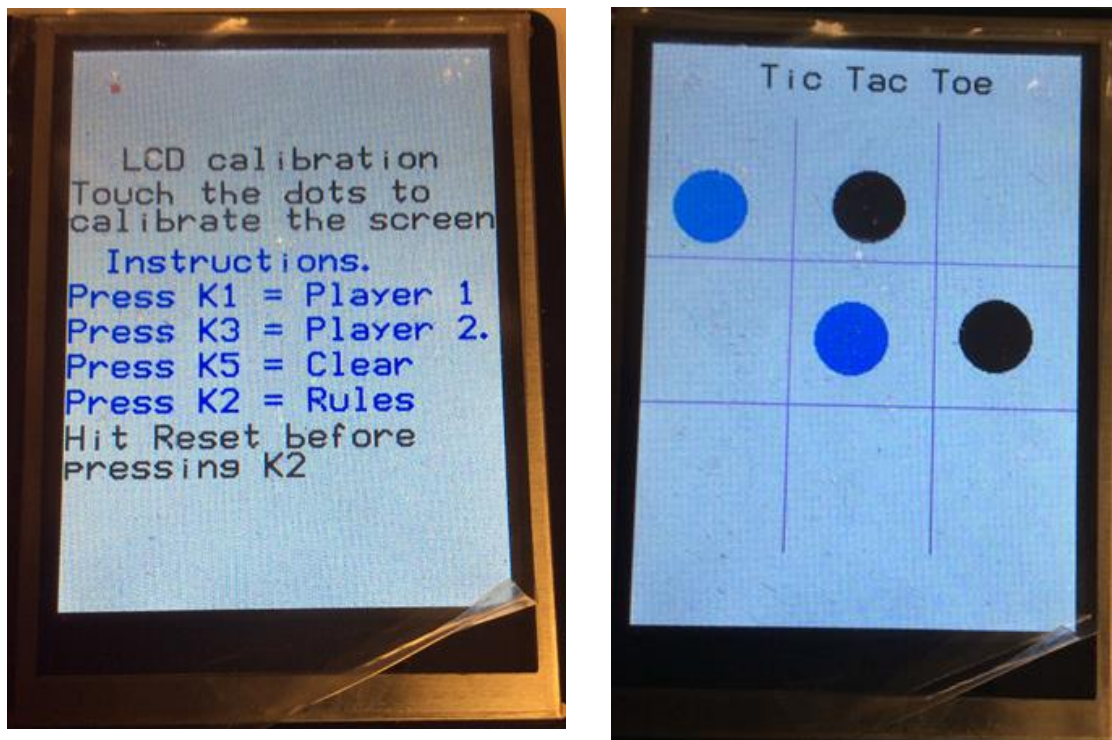


Figure 7 Capstone project 3 (a) User menu, (b) Game panel

The course objectives are assessed using examinations and lab projects and capstone projects. Student work consists of laboratory reports and project presentations. The student course evaluation form includes both survey questions and numerical score. 90% percent of students showed their interests in this new course material and would consider incorporating touch sensing devices in their senior projects to improve human-machine interface and provide more delicate input control compared with conventional knobs and joysticks. Students also give suggestions of doing more projects with touch screen.

5. Conclusions

This paper presents our experiences of teaching engineering and technology students advanced touch sensing technologies. The increasing number of low-cost MCU development kits embedded with Touch elements has become available for teaching Touch technologies. The Atmel SAM4S-EK2 board was employed in the advanced Microprocessor course for students to teach Touch devices programming. This evaluation board is equipped with rich resistive and capacitive touch sensing elements. By utilizing the Atmel QTouch library and Atmel Software Framework, students are able to develop MCU-based capstone projects using touch sensing devices. Some enhanced the user interface of traditional motor control project; some are touchscreen centered projects for multi-media applications. In the future, we will encourage students to develop more complex MCU projects with Touch devices.

References

1. White paper – “Comparison All Resistive Touch Technologies”, 2014.
http://www.elotouch.com/Technologies/compare_resist.asp
2. Touch Technology Brief from 3M Touch Systems, “Projected Capacitive Technology”, 2013.
3. Geoff Walker, “Fundamentals of Touch Technologies”, 2013.
www.walkermobile.com/Touch_Technologies_Tutorial_Latest_Version.pdf
4. Atmel QTouch Library:
<http://www.atmel.com/products/TouchSolutions/touchsoftware/default.aspx>
5. 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.