Integrating Concepts and Practice in Teaching Embedded Systems Design

Authors:

Isidoro Couvertier, Univ. of Puerto Rico at Mayagüez, Mayagüez, PR 00681, icouver@ece.uprm.edu
Manuel Jiménez, Univ. of Puerto Rico at Mayagüez, Mayagüez, PR 00681, mjimenez@ece.uprm.edu
Rogelio Palomera, Univ. of Puerto Rico at Mayagüez, Mayagüez, PR 00681, palomera@ece.uprm.edu

Abstract *— The use of a family of low -power microcontrollers to teach embedded systems design to EE and CE students is presented. A review of aspec ts in the processor family is presented to support their selection as part of the target devices used in the course. Course organization details provide insight into the teaching strategy and sample projects illustrate the students' learning experience. Th e impact of using the indicated family on the course is quantified through assessment metrics, project outcomes, surveys, and performance observations.*

Index Terms *— Embedded Systems Education, Design Experiences, Teaching Undergraduate Design*

INTRODUCTION

Effectively teaching Embedded Systems usually becomes an interesting challenge for instructors. In a typical course, students should be exposed to the right blend of hardware and software design concepts, successfully apply them to complete some sort of design project, and achieve some level of implementation of their designs [1]. The fact that implementation requires laboratory work, invariably involving debugging the hardware and software components of their designs on an off-the-shelf embedded microprocessor does not make the task any easier. Within all this process, the architecture and features of the target device as well as the capabilities and easiness of use of development tools play a central role in the overall teaching/learning experience.

This paper describes the use of Texas Instruments' MSP430 family of low-power microcontrollers to teach embedded systems design at the Electrical and Computer Engineering Department in the University of Puerto Rico at Mayagüez. The target audience in this teaching experience included undergraduate level students from within the department programs in electrical and computer engineering taking the Microprocessor Interfacing course. This course has been structured to provide a major design experience to students in both programs. Details on the course organization and covered topics provide insight into the teaching strategy followed throughout the semester.

A discussion is included about the general aspects of the MSP-430 devices and their development tools that support the decision of adopting them as part of the target devices used in the course. Descriptions of typical projects completed in the class are included, highlighting the student learning experience at the hardware and software levels and their system integration. The impact of using these devices and their tools is quantified throughout and at the end of each semester through formative and summative metrics, project outcomes, exit surveys, and student performance observations.

COURSE STRUCTURE

The target audience for the Microprocessor Interfacing (MI) course at UPRM includes both Computer Engineering (CpE) and Electrical Engineering (EE) seniors. Since the course is a requirement for the CpE program and a technical elective for the EE program, most of the students taking the course are enrolled in the CpE program. Students are usually within one year of finishing their degree. It has been noted that about fifty percent (50%) of the students enrolled in the course apply for graduation during the semester they take the MI course. The course is offered twice a year and typical enrollment is about 50 students.

The course begins with an introduction to embedded systems and microcontrollers, and a tour of the lab facilities. Class topics include the discussion of hardware and software aspects necessary to operate parallel input and output ports, display devices, event timers and counters, interrupt structures and servicing, asynchronous and synchronous serial communication, analog to digital conversion, and memories, among others. The class does not have an assigned textbook. Instead, a set of professor notes is provided emphasizing the topics relevant to the course. The introduction of the MSP430 to the pool of microcontrollers used in the class required updating these notes to include excerpts from the 430's user's manual. The class notes provide just the right amount of information needed for the students to follow through the topics covered in class and those aspects of the target microcontrollers necessary to grasp the fundamental concept of each topic. A set of reference books and user's manuals are available in the laboratory and in the campus library which allow the students to have access to the primary sources of information.

The class normally meets three hours per week, where discussions cover class topics and other relevant information such as a seminars on how to prepare poster and oral presentations, discussions on considerations needed for the project such as economic, manufacturing, environmental, social and political, health and safety, etc. Students are assigned to groups consisting mostly of three or four members each. The Microprocessor Interfacing Laboratory is available to the students 24 hours a day, 7 days per week. The lab is equipped with Dell workstations, internet access, PC-based and handheld scopes, logic darts, digital labs, power supplies, components, and the IAR Workbench limited version supplied with the MSP430 FET tool. Other tools available in the lab are specific to other microcontrollers and microprocessors such as the Motorola HC12, MCore, HC11, Intel processors, and PowerPC; used by some groups in their projects.

Grading is based on exams and quizzes weighting 50% of the total points, and a design project and laboratory exercises accounting for the other half. From this second half, the project accounts for 80% of the grade. The project by itself has several stages which include: proposal, progress reports, final report, demonstration, and a final presentation. Although team working is requirement for completing the project, several mechanisms are implemented which allow tracking the individual performance of each student. The project evaluation weights both the individual contribution of each student as well as their ability to efficiently interact with their teammates.

THE MSP430 DEVELOPMENT INFRASTRUCTURE

The three major factors that favored the adoption of the MSP430 as part of the target devices of the course can be enumerated as: 1) The simple and yet powerful architecture of the 430, 2) Its compact and complete hardware development tools; and 3) The intuitive and usable software development environment. The following sub-sections provide some details on these factors. More detailed information can be obtained from the respective manuals and manufacturer's documentation.

MSP430 Overview

The MSP430 series is an ultra-low-power microcontroller family consisting of several devices featuring different sets of embedded components that target specific application areas. The microcontrollers are optimized for battery operation, featuring up to five different low-power modes that allow minimizing unnecessary power consumption. The particular family members used for class projects belong to the series MSP430x1xx, in particular the 120 and the 149. Members in this series consume less than 500μW when fully active, and can hold memory information in a RAM retention mode with as little as 0.1μA [2] The core CPU in an MSP430 is based on a 16-bit RISC architecture, incorporating 16 internal registers and a constant generator for improved code efficiency. Peripherals include an analog comparator for inexpensive analog-to-digital conversion and a serial communications interface that can be programmed for asynchronous or synchronous operation. The chip also has a built-in 16-bit timer, 8kB of flash memory, and twenty-two independently-programmable I/O pins that can be accessed for reading or writing in the same way than memory.

Hardware Development Tools

Students used a Texas Instrument's Flash Emulation Tool (FET) kit and IAR's Kickstart software environment to develop their projects. The hardware components of a MSP-FET430P120 FET kit, illustrated in Figure 1, show the typical components needed to program the '430 [3]. The kit includes the following:

- A Target Socket Module (TSM), consisting of an MSP430 mounted on a (28-pin) ZIF socket, on a 2.4 by 2.6 inches PCB. A board mounted 2x7-pin JTAG header allows connection to an interface module via a 14-conductor ribbon cable.
- An Interface Module (IM) that bridges the TSM with the computer parallel port via a 25-conductor ribbon cable.
- Four 1x14 pin headers and two spare microcontrollers, part numbers MSP430F123IDW in this particular case.

Sofware Development Tools

The Kickstart software development kit is a Windows-based development environment that enables students to edit, simulate, download, run, and debug their code from a single interface [4]. Figure 2 shows the interface window of the software development environment, indicating the different window components.

Applications can be developed in assembler and/or C using the integrated tools. Using C limits the use of memory to 4kB. The tool debugger, C-SPY, can be configured to operate with the actual hardware or as a simulator. When in hardware debugging mode, students can operate the microcontroller under JTAG and use of internal breakpoints. They also have access to the microcontroller registers (including special function registers) and memory. Reprogramming the device does not require disconnecting the MSP430 from the PC since the JTAG interface provides for in-socket programming.

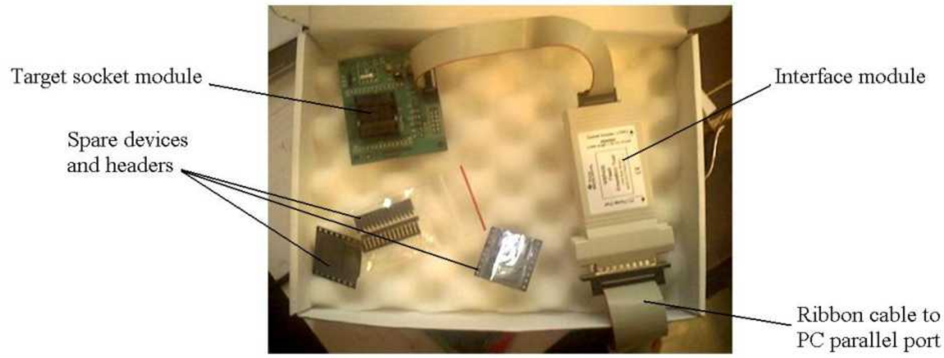


FIGURE 1
THE MSP430 FLASH EMULATION TOOL.

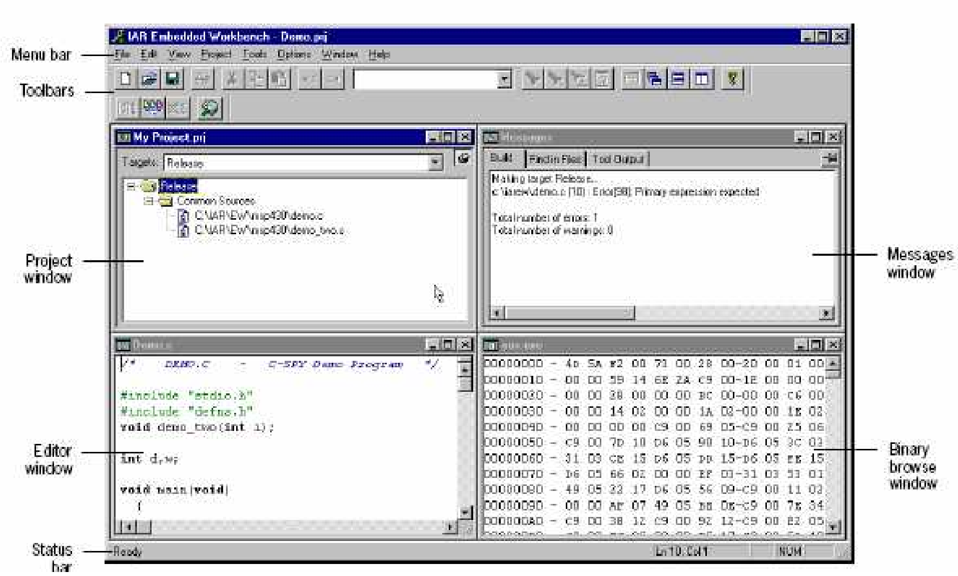


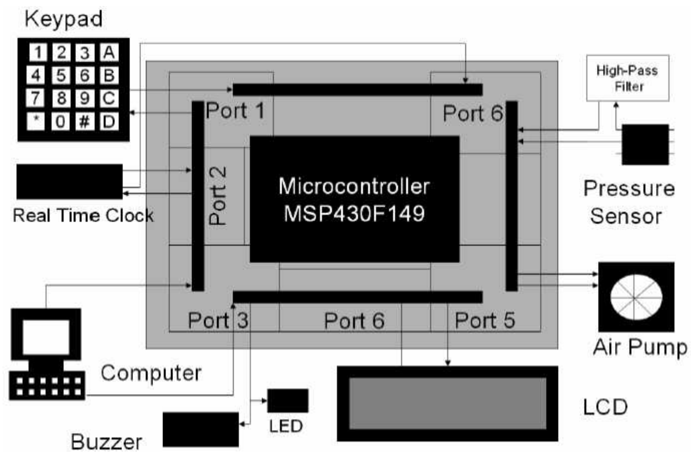
FIGURE 2
COMPONENTS IN THE KICKSTART SOFTWARE DEVELOPMENT ENVIRONMENT.

In summary, the tools included with the FET, both software and hardware, are easy to use, well documented, and permit the development and debugging of applications with relative ease.

SAMPLE PROJECTS

The students' projects begin with a proposal to the class instructor. Once the proposal is approved the students begin working on the design, its validation, and prototyping. Throughout the entire process, the students are required to take into consideration not only technical aspects of their designs, but also have to consider their manufacturability, cost, expert and user opinions, physical limitations, sustainability, environmental impact, ethics, safety, and health factors among others. From all these considerations it appears that designing for manufacturability is one of the most difficult to grasp. We have already sought the help of industry partners so that students can be exposed to this and other issues during field trips to manufacturing facilities. A long list of projects have been developed using the teaching style described in this document, many of them fully documented in the laboratory web site [5]. Brief descriptions of three representative projects developed with the MSP430 are provided below.

Blood Pressure Guard: This project aimed at developing an enhanced, computer-based, Sphygmomanometer, a medical instrument used to read the systolic and diastolic pressures in the human body. The device allowed to perform a diagnostic of the user's cardiovascular health by maintaining a history of blood pressure lectures and other personal data on a remote computer. The systems provided for transferring the readings to the computer, where a the analysis and diagnosis were performed. Detailed description of the diverse stages of development of this project is available through the laboratory site [6]. Figure 3 illustrates the system block diagram and a picture of the finished prototype.

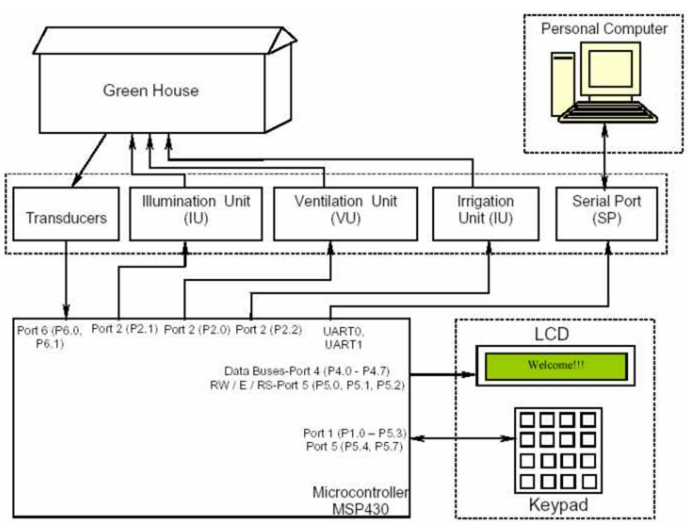


A) SYSTEM BLOCK DIAGRAM

B) FINISHED PROTOTYPE

FIGURE 3
BLOOD PRESSURE MONITOR

Intelligent Greenhouse: The objective of this project was to develop a system to provide autonomous control of water irrigation, illumination, and ventilation, in a closed environment, such as a greenhouse, based on the soil water pressure and a profile of the plant species being cultivated. An optional personal computer connected to the system allowed web monitoring of the greenhouse status as well as access to a historical database for diagnosis and forecasts. Figure 4 provides both, a block diagram of the system as well as a picture of the finished prototype. Additional details of this project can be found in the laboratory web site [7].



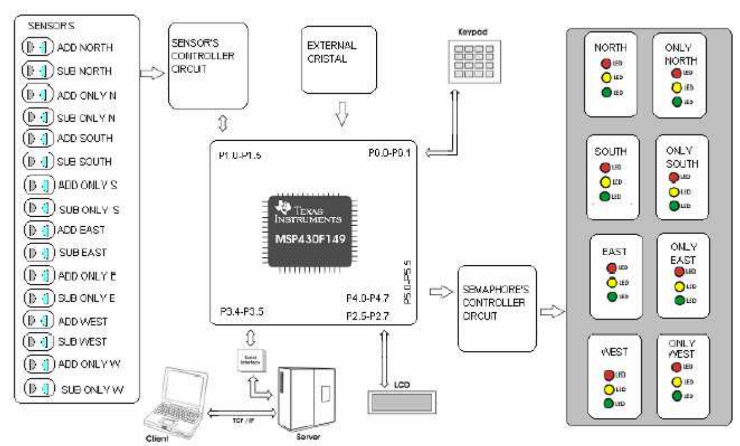
A) SYSTEM BLOCK DIAGRAM

B) FINISHED PROTOTYPE

FIGURE 4
INTELLIGENT GREENHOUSE PROJECT

Smart Remote Traffic Controller: This project consisted of an arrangement of sensors and timers to monitor and control traffic on a given street intersection. The system operated by monitoring the flow of cars in each lane of the intersecting streets and with the aid of a decision algorithm, the timing of the red and green lights in the semaphores were adjusted to expedite transit in the lanes with heavy traffic. In addition, the system had a TCP/IP interface to provide remote access to the semaphore controllers, facilitating system keeping, monitoring, and testing tasks. Figure 5 provides a block diagram of the system and a picture of a finished prototype. Additional details are available in the project web page [8].

A common denominator to all projects is the use of standard input/output devices such as keypads and displays, as well as remote access through a serially connected personal computer. Interrupt-driven multiprocessing is a requirement for event handling and low-power modes are used whenever possible. These components ensured that no mater what project the students ended-up developing, the learning objectives of the course were all satisfied.



A) SYSTEM BLOCK DIAGRAM

B) FINISHED PROTOTYPE

FIGURE 5
SMART REMOTE TRAFFIC CONTROLLER PROJECT

IMPACT OF MSP430 ON THE COURSE

The impact of introducing the MSP430 to the MI course has been multifaceted since both student learning and teaching practices have been improved. Formative and summative metrics applied throughout the semester allowed to track the impact on these areas.

Formative assessment metrics were applied to the students throughout the semester to keep track of their progress. These include a pretest administered at the beginning of the semester to assess their previous knowledge confidence, several one minute questions, short quizzes, partial examinations, written reports, and oral presentations. Summative tools were used at the end of the semester to assess the effectiveness the overall teaching/learning experience. Items in this category include a demonstration, a formal presentation, a final report of the completed project, a final exam, and an exit survey.

The pretest typically shows that most students do not feel comfortable with the knowledge they have acquired by the time they get to the MI course. Frequently, they are unaware of how much they have learned in the preceding courses. The one minute questions were extremely useful to track the student learning process and increasing their confidence. Six such questions are typically administered in a semester. In the first of such tests, a common complaint was that one minute was not enough time. However, as the semester progresses it is remarkable how much they are able to write in one minute. Quizzes were mostly used as a way to prepare students for the partial examinations. In the oral progress presentations students show their progress and talk about the problems faced with their designs.

Previous to the introduction of the MSP430 in the course, all projects were preformed on either 68HC11/12, PowerPC 605, or Intel 8051/52-like processors, where there are plenty of I/O ports for peripheral handling. Upon introduction of the MSP430, one of the aspects that troubled the students the most was the fact that they had only three ports to deal with from a 28-pin package. This seems to have come from talking to students from previous semesters who had four or more I/O ports available for their projects. However, the same students who were challenged with having to deal with such "lack of ports" were able to finish their projects on time and even reported some left over pins. Once the structure of the MSP430 was introduced and examples illustrated how to use control registers to command some device within the microcontroller was presented, students realized that they had all the tools for building their projects contained in a 28 pin package.

Formative test results point to improvements derived from the adoption. One minute questions started to show increased confidence earlier in the semester. The number of problems reported in oral progress presentations by those adopting the MSP430 was considerably less than those using other processors, and this also impacted their performance in quizzes and partial examinations. Summative results, although still showing high rate of completion in the number of projects (over 75%), indicated that those using the MSP430 were able to have working prototypes earlier in the semester, which allowed them to better fine tune their projects.

On the teaching side, the discussion of several key topics of the course resulted simpler and easier to grasp by the students. For example, the discussion of output compare features when discussed on the 68HC11 architecture took over three lectures and student frustration in the laboratory was fairly common. On the new processor the same topic was covered in a single lecture and most students able to get it working properly in their first attempt. Similar experience was obtained in the discussion of vectored interrupts and interrupt service routines.

It is clear that the affective dimension of the students was reached in many of them who reported that this was the best experience they had ever had in school and wished other courses could be like this.

Among the results of the exit surveys we have the following: more than 77% of the students where Computer Engineering majors, over 50% of the students had one more semester left to finish school, 85% answered the course helped them understand the difference between a microprocessor and a microcontroller, 91% said the course helped them understand microprocessors and embedded systems, and 72% answered they had to deal with physical design limitations to be able to develop the project. Regarding the assessment metrics, over 70% answered the required oral presentations were worthwhile, over 60% said the same about required written reports, and more than 90% said the lab work was important to understanding the concepts. All groups using the MSP430 agreed that this microcontroller helped them to better understand embedded systems.

CONCLUSION

The MSP430 tools have been in use in the laboratory for over three semesters now. In this period of time, the simple to use and yet powerful features of this family of microcontrollers have impacted teaching and learning practices in the microprocessor interfacing laboratory of the ECE Department of the UPRM. This has been evidenced by multiple metrics used in the course and laboratory to assess the impact.

ACKNOWLEDGEMENT

The authors would like to thank the kind support of the Mixed Signal Products group at Texas Instruments, Dallas, TX, for their kind support in integrating the MSP430 tools in the microprocessor interfacing course and laboratory.

REFERENCES

[1] W. Wolf and J. Madsen, "Embedded Systems Education for the Future", In Proceedings of the IEEE, Vol. 88, No. 1, pp. 23-30, Jan. 2000.

[2] Mixed Signal Products, "MSP430x1xx Family, User's Guide, SLAU049D", Published by Texas Instruments, Inc., 2004, [Posted on the World Wide Web], http://focus.ti.com/lit/ug/slau049d/slau049d.pdf, Last retrieved May 2004

[3] Mixed Signal Products, "Texas Instruments MSP-FET430 Flash Emulation Tool (FET) User's Guide" Texas Instruments, Inc. 2002, [Posted on the World Wide Web], http://www-s.ti.com/sc/psheets/slau138/slau138.pdf, Last retrieved June 2004

[4] IAR Systems, "IAR Embedded Workbench Kickstart Ver. 3", Available through Texas Instruments, Inc. 2004, [Downloadable through the World Wide Web], http://www-s.ti.com/sc/psheets/slac050a/slac050a.zip, Last retrieved Aug. 2004

[5] Microprocessor Interfacing Laboratory, ECE Department, University of Puerto Rico at Mayaguez, [Accessible through the World Wide], http://www.ece.uprm.edu/~micro2, Last updated May 2004

[6] L. Sanchez, F. Cintron, and J. Borges, "Blood Pressure Guard", Microprocessor Interfacing Laboratory Project Report, [Accessible through the World Wide], http://www.ece.uprm.edu/~micro2/Projects/spring2004/group7, Last downloaded June 2004

[7] A. Villalain, M. Gonzalez, N. Perez, and J. Valera, "Intelligent Greenhouse", Microprocessor Interfacing Laboratory Project Report, [Accessible through the World Wide], http://www.ece.uprm.edu/~micro2/Projects/spring2004/group9, Last downloaded June 2004

[8] M. Cabrera, F. Medina, C. del Rosario, and K. Rosario, "Smart Remote Traffic Controller", Microprocessor Interfacing Laboratory Project Report, [Accessible through the World Wide], http://www.ece.uprm.edu/~micro2/Projects/spring2004/group6, Last downloaded June 2004