Improvements to a Mechatronic Systems Course for Mechanical Engineers

Clark T. Merkel and David S. Fisher, Mechanical Engineering
Rose-Hulman Institute of Technology

1. Introduction

This paper describes the content and recent changes to an existing course on mechatronics for mechanical engineering majors at Rose-Hulman Institute of Technology. The course focuses on the use of microcontrollers to provide smart device control. In addition to microcontrollers, the course has been changed to include topics in ladder logic, hands-on exercises with programmable logic controllers (PLCs), and PIC microcontrollers. These changes have provided a broader and more useful skill set for the students completing the course. This paper includes a description of their implementation as well as the student responses to these topics.

2. Mechatronic Course Content

Mechatronics is a synthesis of mechanical engineering, electronics, computer science, and system control. Most engineering students have been introduced to these topics but rarely are so many ideas combined into a single integrated course. Because a mechatronics course has such a broad topic base, it's not surprising that a course in mechatronics often takes on a different focus at different institutions. Depending upon where you take a course you might come away with a very different opinion about what it includes. Some university mechatronics courses focus heavily on system modeling aspects and the application of digital and analog electronics to implement them. Other universities offer the mechatronics course more with a focus on use and programming of microprocessors. Still other programs have a mechatronics course which is implemented primarily as a design course which includes integrated, hands-on projects. What seems to be the most common thread tying these different approaches together is that there is almost always a multidisciplinary, hands-on project part to the course. Being able to implement hardware combined with circuitry, microprocessors, and sensors is an important component of a typical mechatronics course. Providing theory and background are good, but working through the process of getting the details of a real device to work is even better.

3. Mechatronics at Rose-Hulman

At Rose-Hulman, the mechatronics course is the first and sometimes only exposure our mechanical engineering students get to digital systems, digital electronics, and microprocessors before graduation. The content of our course is dictated not only by the accepted definition of "mechatronics" but also by the strengths and weaknesses of other courses in the curriculum. Because our students receive considerable preparation in system modeling, analog electronics, controls systems, measurement, and computer programming in other courses, the mechatronics course attempts to focus primarily on topics that are still missing from their overall background. This deficit primarily includes topics of digital logic, digital electronics, and microprocessors.

For the last decade, this course has been taught using microcontrollers which have used Motorola microprocessor powered microcontroller boards that interface with different sensor inputs and a variety of physical outputs. The microcontroller boards include a number of peripheral circuits and chips which allow relatively easy access to external sensors, motors, and displays. A typical course has students use a portion of the quarter to get up to speed working with the microcontroller and then complete a project which requires smart control. Most recently, the microcontroller board that has been used in this class is the Handy Board microcontroller.

The Handy Board microcontroller was primarily created to be a useful hobbyist microcontroller board which is easily used for running and controlling simple robotic projects. It uses a Motorola 68HC11 microprocessor brain and has 32 kilobytes for programming and variable storage. The main board has six analog input ports, nine digital input ports, an on-board potentiometer knob control, four bidirectional, pulse width modulated 9 volt DC motor outputs, two momentary user switches, a piezoelectric buzzer, a modulated IR signal detector, and a 16x2 character LCD display. An add-on expansion board can be purchased which provides eight very useful low power digital output ports, additional analog ports, four Lego sensor ports, and six 3-pin connections for servo motors. Except for the 9 volt battery ports, the ports have a working voltage of 5V DC and are easily interfaced to standard TTL level circuitry. It can be programmed using either assembly language or with an interpreted language Interactive C. Figure 1 shows the Handy Board microcontroller.

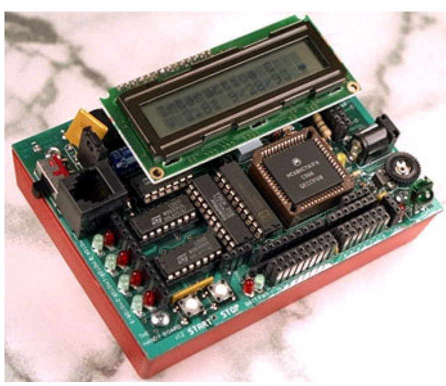


Figure 1: Basic Handy Board Microcontroller

While the Handy Board has proven to be a good instructional and fairly dependable tool for learning microcontrollers, it has also been found to have some important weaknesses. Specifically, the Handy Board was not a tool that students would often find used in industry. They were being taught use of a tool that really would not be used very much once they left the course, let alone the school. Secondly, the relatively high cost and inability of the Handy Board to be fully integrated into projects outside the class meant that it was not all that useful as a platform for implementing senior design projects. Students would use the Handy Board as a tool in the class, and pretty much that was it.

To overcome these two perceived weaknesses, the emphasis on the Handy Board microcontroller, while still a component of the course, was reduced and two additional topics were introduced to the mechatronics course. The first topic was to include a module on programmable logic controllers. This module, in addition to providing a hands-on experience with a simple PLC, provided background and experience with a tool that students are much more likely to encounter and work with in industry. It also provided a very good platform for the introduction and teaching of ladder logic in the context of the course.

The second addition was assembly language programming working with PIC microcontrollers. These cheap, all-in-one controller chips exposed the students to assembly language which helps explain the mechanisms of lower level of computer architecture. They are also a more useable controller that some students can and have been able to embed with their senior design projects. This module was not intended to provide extensive exposure to the many features of the PICs, but to provide enough of an introduction so that the students would understand the capabilities of these versatile controllers. The goal was to give students enough hands-on experience to make them confident programming the basic features of a PIC controller. This gives students the option to implement PIC usage in courses or work beyond Mechatronics.

The other major component of this class involves the completion of a 3 to 4 week design project. The project is used both to reinforce the use of smart controllers and reinforce the design process steps. The current project is posed as a very vague design objective. Students work through a formalized brainstorming session to generate project ideas and narrow down the concepts in a written project proposal. The guidelines for the project are very broad. The objective is stated as “sense something, control something, and activate something”. Students complete the project in teams of two. Most teams use the Handy Board as the smart controller for this project. A short oral presentation and demonstration of their working project is expected to be given during the lab period in the final week of class. Generally there is a wide variety of among the 20 or 30 projects groups. This variety helps expose students to many different ways the controller can be used in mechanical design.

In addition to the Handy Board, PLC, and PIC assignments, the other individual mechatronics’ topics are listed on a week by week basis in Table 1.

Table 1. Course content and general lecture topics

	Lecture/workday topics and activities
Week 1	Introduction to mechatronics, number systems, integer mathematics
Week 2	Boolean logic and Karnaugh mapping
Week 3	Programming in Interactive C
Week 4	Switches, relays, and transistors
Week 5	Sequential logic, flip flops, and integrated chip characteristics
Week 6	Ladder logic and programmable logic controllers
Week 7	Introduction to project and PLC workdays
Week 8	Introduction to PIC assembly language programming
Week 9	Project and PIC workdays
Week 10	Project workdays and topic review

4. Laboratory Topics

Students are expected to attend a three hour lab each week and complete tutorials that teach them how to use Handy Board, PIC microcontrollers, traditional digital IC's, and various I/O devices. They are also given access to the lab workspace 24 hours a day, 7 days a week. This open access is given with the condition that students are not to remove the controllers or other lab equipment from the lab area. Students are assigned work stations where they may store the equipment and have room to use for storing project work. While the security is not very stringent, loss and misuse of equipment has been almost nonexistent.

The laboratory activities are summarized on Table 2. There are five formalized laboratory tutorials on the Handy Board and three formalized laboratory tutorials on working with digital electronics and PIC microcontrollers. The other two weeks are provided as time to work on and present projects. The lab tutorials' primary purpose is to quickly introduce the students to a wide variety of features that are available with the microprocessors and to familiarize the students with many common sensors that will be available for their use when completing the project. These tutorials are available on-line at http://www.rose-hulman.edu/~merkel/

5. Laboratory and Workday Equipment

With three different types of control device used in this class and 20 to 30 different projects groups per quarter, there is a considerable amount of equipment required to supply the needs of this class.

Handy Board and related equipment:

Each team of two students is provided with a Handy Board microcontroller box kit for use during the quarter. This box kit includes the expansion board as well as power supply, charger board, and communication cables needed to interface with the students' laptop computers. Additional resources in the kit include a multimeter and some other simple hand tools. Students are also provided with a variety of inexpensive sensors and actuators.

These include items such as photocells, thermisters, Hall effect sensors, touch switches, IR emitter-detectors, phototransistors, relays, potentiometers, DC motors, laser pointers, LEDs, and TV remote controls. For the tutorials, a large supply of Lego blocks was made available to provide structural components for simple machines and devices. The Lego assortment includes a generous supply of Lego Techniques construction pieces. The Techniques pieces include a wide variety of gears, pulleys, wheels, axles, bushings, and other machine parts that can be quickly and easily incorporated in the tutorial exercises. As the students put it, this is the senior level course where they are finally allowed to "play" with the Legos. The use of Legos help the students construct an integrated physical product while giving them time to focus on the electrical and microcontroller parts of the project. Some students choose to use traditional manufacturing techniques such as woodworking, milling machines, or lathe work. However many students feel the microcontroller portion of the project takes more than enough time without the more time consuming manufacturing processes.

Table 2: Lab topics and activities

Lab	**Topic**
Week 1	Handy Board Lab 1: Introduction to the Handy Board, simple commands, interactive mode, and writing simple programs. Activity: Write and run a program that uses the Handy Board digital I/O ports, LCD, buzzer, and knob potentiometer
Week 2	Handy Board Lab 2: Working with analog inputs, motors, potentiometers, and more IC commands Activity: Write a program to play a song
Week 3	Handy Board Lab 3: Using the expansion board, digital I/O, Lego sensors. Activity: Create a "Neat Program" to perform some task of your choosing based on previous lab skills
Week 4	Handy Board Lab 4: Motor control, sensor calibration, multitasking. Activity: Create a simple multitasking DC motor speed control program
Week 5	Handy Board Lab 5: Light/dark sensing, IR sensors, IR remote control, and proportional control Activity: Create a proportional motor position control program
Week 6	Digital Electronics Lab 1: Bread-boarding, digital circuits, and transistor networks Activity: Build a stepper motor controller using digital ICs and transistors
Week 7	PLC and Project Workday
Week 8	Digital Electronics Lab 2: Programming and simulating PIC microprocessor programs, using MPLAB IDE and PicStartPlus Activity: Program a PIC as a stepper motor speed control
Week 9	Digital Electronics Lab 3: A PIC microprocessor programming project. Activity: Program a PIC as a DC motor speed control with transistor H bridge and pulse width modulation
Week 10	Project demonstrations

A variety of common sensors and actuators are available in smaller quantities for use with the quarter projects. These can include servo motors, stepper motors, relays, transistors, 7 segment displays, TTL digital ICs, electrically operated valves, sonar sensors, fans, small pumps, heating units, small solenoids, and other assorted electrical components.

PLC and demonstration stations:

With the newly introduced module on PLCs, additional equipment has been added to the class to provide hands-on activities. The students are taught to use a ladder logic programming and simulation tool supplied by Rockwell Software called PicoSoft. This is a freely available software application which can be found at the Allen-Bradley web page at http://www.ab.com/plclogic/pico/picosoft.html. The software not only can be used to create and simulate ladder logic control systems (as shown in Figure 3), but the application also provides a communication interface for downloading and interacting with the Allen-Bradley family of Pico PLCs. These are relatively simple and inexpensive PLCs that are very appropriate to use with an introductory PLC course. The low end PLC in the Pico family offers eight digital input ports and four output relay ports. The units we have chosen for our implementation use a working voltage of 120/240 VAC although there are other models which work on 24 VDC power. Depending upon the PLC model, these units may include an internal clock, an LCD display, and 4 pushbuttons on the front of the unit. We have mounted and electrically connected the PLCs to make a number of portable PLC work units that can be used in the classroom, in lab, or for projects anywhere you can 120 VAC is available. One of these units is shown in Figure 4.

We have also created two demonstration stations, constructed by lab technician Ron Hofmann, which students use to test out their PLC ladder logic control programs. One is a mixing station which includes PLC control of a pump, a motor with mixer, and an electrically controlled drain valve. Sensors available to the PLC include two fluid level float valves and two momentary pushbuttons. The mixing control problem requires students to initialize fluid flow when the start switch is pressed. The PLC monitors the fluid level until the tank has filled and then the mixer motor runs for a prescribed amount of time. Following mixing, the PLC opens the drain value and monitors the fluid level until the tank has emptied. An emergency shut off switch is also monitored by the PLC and can implement system shutdown. The mixing tank demo station is shown in Figure 5.

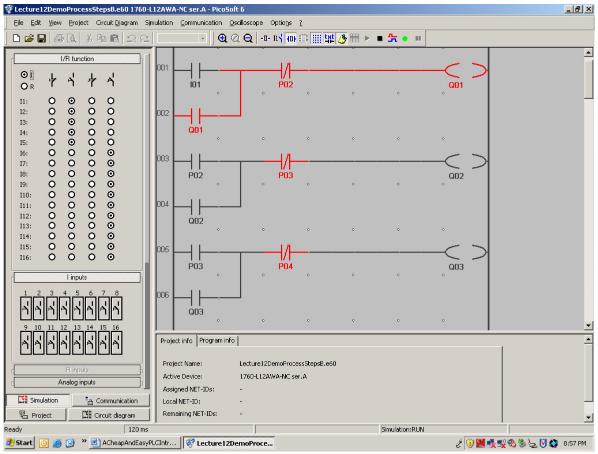
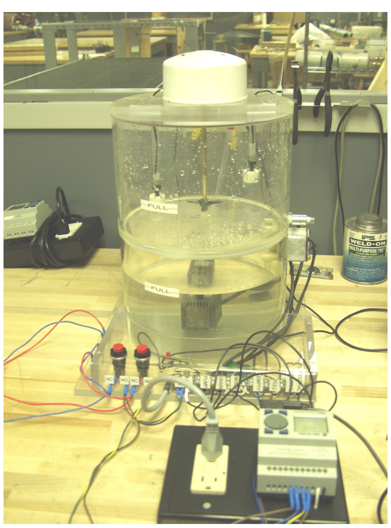


Figure 3: PicoSoft ladder logic program

Figure 4: Pico PLC work unit

The second demonstration station is a short conveyor system. It has an indexing supply rotor which delivers individual soda pop cans from a stacked supply to the conveyor. There are IR break beam sensors which detect pop cans of different height and a solenoid controlled diverter arm which can sort the cans into different exit chutes. Momentary switches are also mounted on the device to initiate PLC controlled startup and shut down commands. The typical control problem the students are expected to implement is to turn the conveyor off and on, initiate can indexing at regular intervals, count the number of cans that are processed, and direct the tall and short cans to appropriate exit chutes. The conveyor demo station is shown in Figure 6.



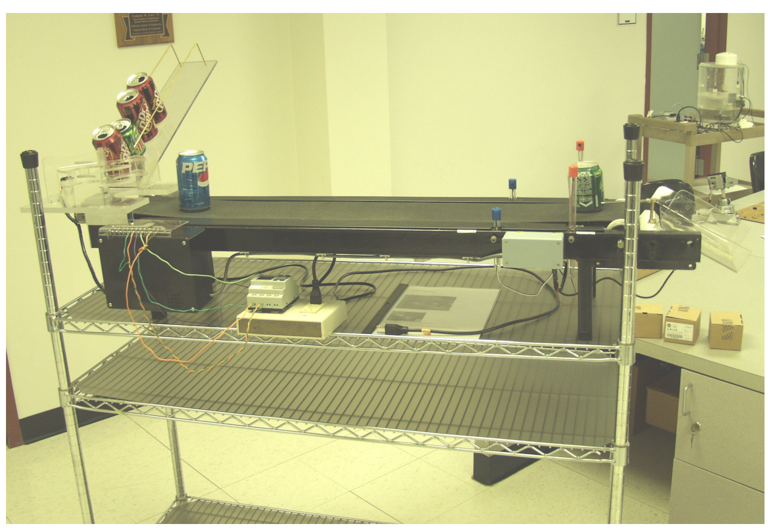


Figure 5: Mixing station

Figure 6. Conveyor station

Digital electronics and PIC microcontrollers:
The other topic added to the mechatronics course includes three laboratory sessions used to breadboard digital circuits. In the first of these three labs, students build a circuit to provide the control sequencer to a stepper motor driver. In this lab, student students breadboard the circuit in stages using traditional TTL digital ICs, NAND gates, D flip-flops, an H-bridge, and a 555 timing chip. This lab's primary purpose is to reacquaint students with the skill of bread-boarding and show them that the logic gates they have been studying in lecture are actual devices that can be used to build working digital control devices. This gives the students a different tool that they can use outside of microcontrollers for solving problems if a microcontroller is overkill. The sequencer they build is modeled by the four stages shown in Figure 7.

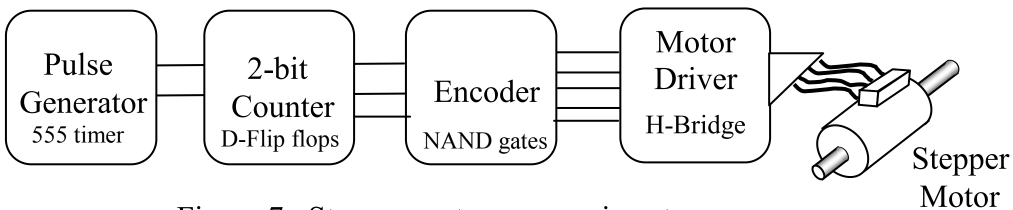


Figure 7. Stepper motor sequencing stages

The other two labs have the students program PIC microcontrollers and breadboard the transistor networks needed to drive stepper motors or DC motors using pulse width modulation. The first of the PIC labs walks the students through all the steps needed to create the code, simulate it, and burn it to the microcontroller chip. In this lab, the program code is provided to the students and they are to understand and modify it. They also complete the breadboard circuit to run the stepper motor. In the second PIC lab, they are expected to generate their own PIC assembly language code to run a bi directional DC motor using pulse width modulation. In both PIC labs students complete the simulation and downloading of the program to the microcontroller then proceed with the construction of the circuits needed to run the motor.

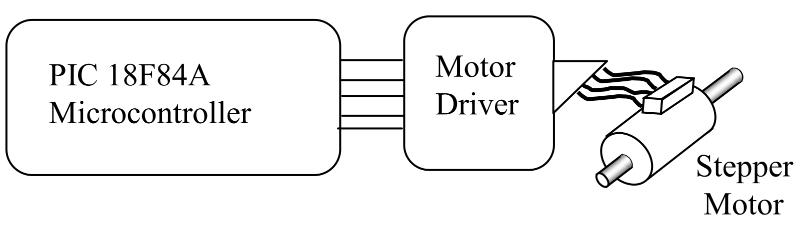


Figure 8: PIC circuit stages

While students work in pairs, each student completes their own breadboard circuit for each of these three labs. The programming of the microcontroller chips is done using PicStartPlus programmers. We currently are using PIC16F84A chips for both PIC labs. It provides up to 13 digital I/O ports which the user can freely allocate as inputs or outputs. We use the MPLAB IDE software which is an integrated development environment which

Microchip freely provides for use with their line of microcontrollers. In additional to a full screen editor, it provides debugging and simulation features. MPLAB is the only software tool needed to complete downloading and burning of the program code to a chip using the PicStartPlus programmer. While the PIC programs can be written in several languages, for these labs we have chosen to program the chips in assembly language. This choice was made because we felt it provided the truest view of how these controllers operate. Additionally, this seemed to be a good opportunity to broaden their program language skills. Instead of giving them another C like language, learning assembly would give students one additional skill to take with them from the class.

7. Student Survey Results

At the completion of our Winter quarter we gave our students a survey to see what they thought about the importance of various topics and asked them what their experience was like covering that topic. In order to determine which topics were most important to students, the topics in the course we broken up into three categories containing a total of 14 individual topics (Table 3). The topics were then ranked to determine the most important topics to the student (with 1 being the most important) and ranked to determine which topics provided the best experiences for the students (with 1 being the best experience). In addition to the rank the topics were evaluated in terms of how many standard deviations above or below the mean that topic scored. This is listed in Table 3 as the Z-Score, where a score of -0.5 would mean that topic was of below average importance to the students, where the mean for that topic was 0.5 standard deviations below the overall mean. There were 60 students out of 66 that completed this survey. The Z-Scores were then plotted on a bar graph in Figure 9. Bars above zero represent topics with above average scores and bars below zero represent topics with below average scores. The topic numbers listed on the x-axis of figure 9 correspond to the topic numbers in table 3. The categories are divided by the vertical bars into digital system topics, PLC topics, and PIC Microcontroller topics.

Table 3: Student survey responses to the importance for the various course topics and their evaluation of their experience learning that topic [in terms of Z-score and Rank]
(Note the figure numbers correspond to Figure 9)

Importance Z-Score	Experience Z-Score	Fig#		Importance Rank	Experience Rank
0.26	0.27	1	**I. Digital Systems**		
0.40	0.60	2	Number Systems	2	3
0.79	0.66	3	Boolean Logic	1	2
0.26	0.67	4	K-Maps	5	1
-0.06	-0.22	5	Seq Logic	9	9
-0.11	-0.40	6	Digital Electronics	10	11
0.30	0.33	7	Breadboarding	3	4
0.07	-0.06	8	**II. PLC Topics**		
0.29	0.08	9	Switches and relays	4	7
-0.14	-0.52	10	Transistors	11	12
0.04	0.10	11	Ladder Logic	7	6
-0.05	-0.12	12	Working with PicoSoft	8	8
0.19	0.15	13	PLC Demo stations	6	5

-0.51	-0.61	14	**III. PIC Microcontroller**		
-0.69	-0.87	15	PIC Assembly language	14	14
-0.51	-0.65	16	Working with MPLAB IDE	13	13
-0.33	-0.31	17	Implementing a PIC in circuit	12	10

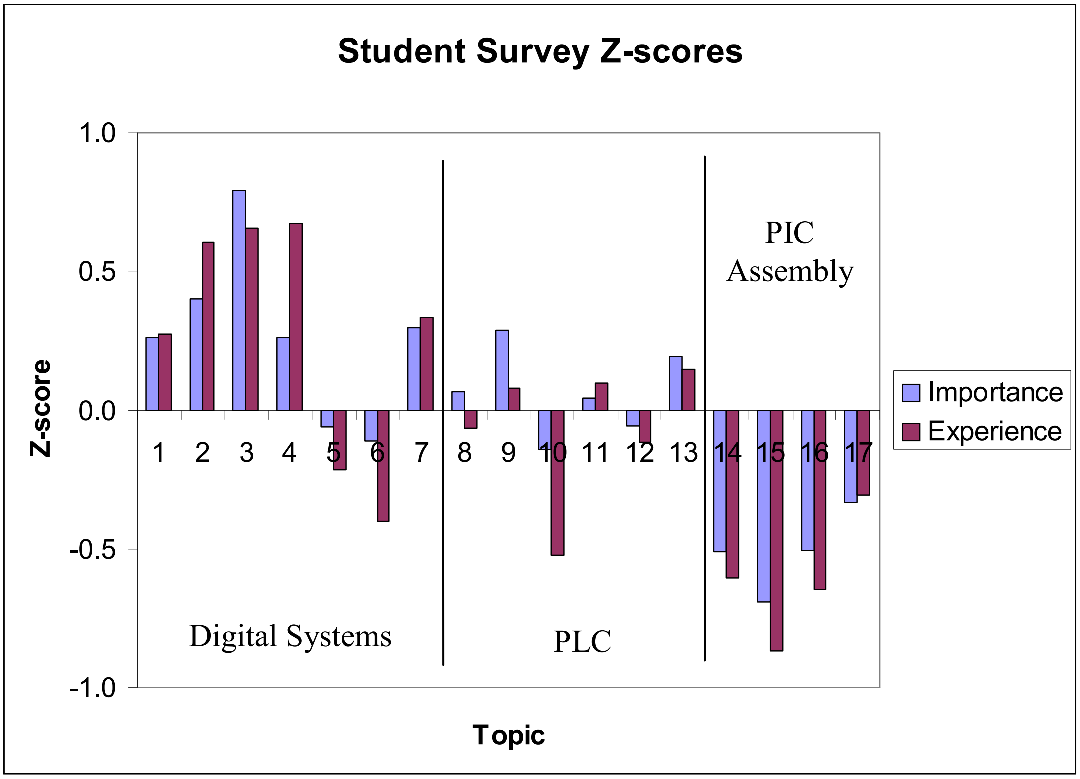


Figure 9: Student responses to the importance of various topics and their experience learning that topic (x-axis numbers defined in Table 3)

In addition to asking the students about which topics were important, we also asked them which topics we should spend more or less time covering. The results of the Student Time Recommendation are summarized in Figure 10. Bars above the x-axis represent topics that the students felt we should spend more time covering and bars below the x-axis represent topics that the students felt we should spend less time covering. The percentages correspond to the percentage of time, more or less, we should spend covering that topic. For example a 0% rating would mean that on average we are currently spending the perfect amount of time covering that topic in the students' opinion.

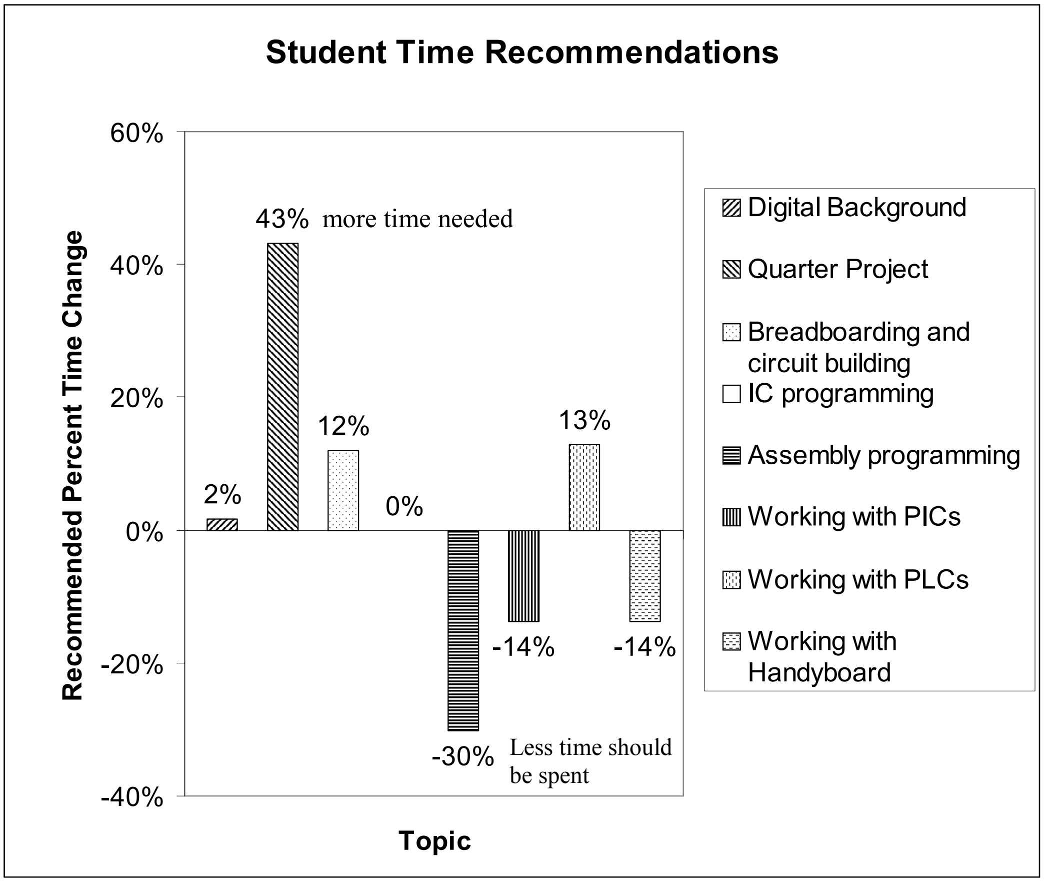


Figure 10: Student recommendations for where to spend more time or less time

Looking at the student survey responses there are a many conclusions we could draw. The most obvious and general observation is that students didn't seem to care for the PIC's and/or assembly language programming. There was also a real interest in getting more time to work on the quarter project. The one piece of data that might have been interesting to add to this data set is how difficult students found the topics. It's common that if a topic is very hard for students they'll rate it very low even if the topic has real value. Conversely if a topic is fairly easy, it will get rated very high. For example students said that number systems (i.e. binary, octal, hexadecimal) were really important and they had a great experience learning those topics. Those high scores may be due to the fact that these topics are very easy to learn and understand. I'm not sure how "easy" corresponds to "important" but there seemed to be a correlation in the student surveys. Whereas learning to use PIC microcontrollers was extremely *un*important. This may have resulted from assembly language programming being very challenging and frustrating for student that have no experience programming at this low level. This being said I think it is very important to get student feedback to see how they perceive the course. It forces an instructor to really think hard about what learning experiences best benefit students and what needs revision.

6. Conclusion

These two primary modifications, PLC projects and PIC microcontrollers, are have been in place for classes starting with the fall and winter of 2005-06. The initial response to the PLC module has been very positive with students asking for more time there. The hardware demonstration stations have made the topic much more meaningful. The students found that running a simulation of a program and running the same program on an actual system don't always produce the same response. Assumptions they make in simulation are not always the same as the actual device setup and can end up producing different run results. As we continue to work out the bugs with our current units we hope to expand the number of PLC units and stations. Overall, the PLC modules look to be very successful.

The feedback on the PIC module was more mixed. While a number of students had indicated this was a skill they very much wanted to acquire and were looking forward to the PIC module, quite a number of other students were less than enthused about these two labs. They did not particularly seem to enjoy or appreciate programming in assembly language, which was really quite different from their previous programming experiences. We will be approaching the introduction to assembly language programming in the second iteration of this new module differently. It may be that the students need a better introduction to working with PICs to feel more comfortable with these microcontrollers. It might also be that our students just don't like assembly language programming. One of the options we will be looking at for future implementation will be the possible use of one of the higher programming languages for programming the PIC such as PicC, PicCLite, or PicBasic.

The end result of these changes is that students in the current version of the mechatronics course are now getting hands-on experience with three different controllers that may used to provide smart control for mechanical designs. While one of these controllers is a hobbyist controller, the other two have very real use in industrial settings. Introducing the students to relevant and smart control devices was the main reason for implementing these changes to the mechatronics course. We feel that these changes have successfully helped student learning and preparation for future mechatronics activities.

7. Bibliography

1. Handy Board Web site: at http://www.handyboard.com/
2. Merkel, C, Web page at http://www.rose-hulman.edu/~merkel/
3. Microchip Web site: http://www.microchip.com/
4. Petruzella, Frank D., "Programmable Logic Controllers", 3rd Ed. published by McGraw-Hill.
5. Rockwell Automation, Web page: "Pico Controllers" at http://www.ab.com/plclogic/pico/