2006-2172: A STUDENT OWNED MICROCONTROLLER BOARD

Hugh Jack, Grand Valley State University

HUGH JACK earned his bachelors degree in electrical engineering, and masters and Ph.D. degrees in mechanical engineering at the University of Western Ontario. He is currently a professor at Grand Valley State University and chairs the product design and manufacturing program. His research interests include controls and automation, including the use open source software for industrial control.

Nael Barakat, Grand Valley State University

NAEL BARAKAT has a Ph.D. in mechanical engineering from McMaster University. He is currently teaching controls, automation and ethics at Grand Valley State University.

A Student Owned Microcontroller Board

Abstract

Laboratories provide valuable opportunities for students to work with professional equipment. This equipment is often costly, potentially unsafe, easily damaged, or difficult to operate. For these and many other valid reasons students are often provided limited access to equipment. Unfortunately when a laboratory is closed a student does not have access to the equipment. Moreover, when the course is complete the student loses access altogether. Clearly this goes against our desire as educators to maximize student learning opportunities.

At Grand Valley State University (GVSU) junior level students majoring in Mechanical and Product Design and Manufacturing are required to take EGR 345 - Dynamic System Modeling and Control. This course contains a significant laboratory component that involves data acquisition and control. In the past this course used expensive equipment that posed the problems mentioned previously. To overcome these issues the course has moved to a model where the students purchase their own controller for $30. The boards contain an Atmel Mega32 microcontroller, USB connector and interface circuit, LEDs, connector, and a fuse. The boards measure 1 by 4 inches and can be used in any PC with a USB port. Students can use free software to write programs in C. In the lab the boards are connected to supplementary circuit boards with circuits for voltage regulation, motor drivers, and protection circuitry.

The paper describes the boards and how they support the course. Anecdotes and details are included for those planning similar implementations.

The Course Content

EGR 345 - Dynamic System Modeling and Control is required for all Mechanical and Product Design and Manufacturing students in their junior year. Prior to this course, students have taken courses such as Differential Equations, Statics, Circuits, C Programming, Writing, CAD/CAM, and Introduction to Digital Systems. The goal for the course is to prepare students to use modern methods to model and design electrical and mechanical control systems. Mechanical and electrical systems are modeled with differential equations and analyzed using techniques such as the explicit solution of ODEs, numerical integration, standard forms, phasor analysis and Laplace transforms. Consequently, control system are designed and analyzed with techniques such as block diagrams, Bode plots and root-locus techniques. Topics covered during the hour long lectures, three times per week, are listed below.

1. Translational systems
2. Differential equations
3. Numerical analysis
4. Rotational systems
5. Input-output equations

6. Electrical systems
7. Feedback control
8. Phasor analysis
9. Bode plots
10. Root locus analysis
11. Laplace transforms
12. Analog IO
13. Continuous sensors/actuators
14. Motion control

The lecture material is enhanced with a weekly three hour laboratory experience. Each laboratory section typically contains 12 students. The laboratories are structured to require students to do prelab design, simulation, and programming. During the laboratory, students are expected to construct systems, take readings to verify theoretical predictions, and then characterize the systems. By the midpoint of the semester, students are able to build basic negative feedback control systems. Upon concluding the laboratory sequence, students have had a mature control design experience as illustrated in the following list of laboratory experiments.

Lab 1 - Programming the Atmega32 Thumb Board - A basic programming tutorial on using the ATMega 32 for various I/O and interrupt tasks.
Lab 2 - Numerical Methods - Scilab and C programs were used for numerical solutions.
Tutorial - Creating Web Pages.
Lab 3 - A Feedback Controller - A proportional feedback control system using a PWM controlled transistor and tachometer for velocity feedback.
Lab 4 - Deadband Compensation for Bidirectional Motion - Stiction values were measured and then used in compensation subroutines.
Lab 5 - Position Control with an Encoder - An H-bridge was used to allow bidirectional motion with a potentiometer for position feedback.
Lab 6 - Motion Control - Subroutines were used to generate setpoints for motions that would start and stop smoothly.
Lab 7 - Modelling Brushed DC Motors - Parameters for an ideal DC motor were measured/calculated.
Lab 8 - System Modeling - The motor model was used to compare theoretical and actual responses of a feedback control system.
Lab 9 - Variable Frequency Drives - Industrial Variable Frequency (VFD) drives were used with AC motors.
Tutorial - Allen Bradley 161 Variable Frequency Drives.
Lab 10 - I/O Using LabVIEW - Industrial software and hardware was used for data input and output.
Lab 11 - Torsion - The students build a torsional pendulum and then measure the response to an input offset using LabVIEW.
Lab 12 - System Simulation with Simulink - The block diagram simulation tool in Matlab was used for system modeling.

To support the course a custom textbook[1] and a laboratory guide[2] have been developed and are freely available on the Internet. The course makes extensive use of computational software such as Scilab[3], a free clone of Matlab, and C/C++ programming. The laboratories are supported by custom designed hardware based upon a standard embedded controller, the Atmel ATMega32.

Previous Boards

The EGR 345 laboratories were originally designed to use computers with Data AcQuisition (DAQ) cards and LabVIEW[4] for programming. LabVIEW is an industrial software package that allows data collection and control systems to be constructed very quickly for industrial applications. Unfortunately the high cost of the system (> $2000) made it unsuitable for laboratory environments where students commonly damage the DAQ boards. Moreover, the software hides many of the implementation details which is very convenient for professionals, but makes it very difficult to teach the fundamentals. Based upon these observations the use of LabVIEW was reduced (not eliminated).

In 2003 we began to use Axiom development boards[5] based upon the Motorola 68HC11. These boards have a price of $89 and allowed students to program systems in C. This shift was very successful and allowed students to implement control systems that could be used in consumer products using industry standard hardware. However these boards were also susceptible to damage and maintenance became a major issue. Typically, these boards were left in open laboratories. A few times during the semester we would find many of the boards inoperative. Although easy to repair, this was inconvenient. This was most likely because individual students would damage a board and ‘swap boards’ to debug their systems. During this process they would damage multiple boards.

In 2004 the Atmel ATMega32 was selected as a replacement microcontroller[6]. In support of this, we designed custom circuit boards that contained $37 worth of parts including the printed circuit boards (PCB). These were assigned to teams of students. The number of maintenance issues dropped substantially. The PCB was 3x4 inches, the largest allowed by the free version of Eagle design software[7]. The ATMega32 package selected was the 40 pin DIP so that students could assemble the boards themselves. The board also contained a voltage regulator, RS-232c driver, L293 H-Bridge, and numerous screw terminals. The outcomes from using these boards was very positive. However, we decided that it would be more convenient if the boards were lower cost, would use a USB port, and be able to operate without a power supply.

In 2005 the boards were redesigned to a two board set. The smaller board contained the microcontroller, a USB bridge/hardware, LEDs, a reset switch, and a connector. By itself the board can be plugged into, and draw power from a USB port for programming and simple testing. These boards contained $20 worth of parts including the boards, however the students were charged $30 to cover assembly costs. Students were required to buy these boards. A second larger board (owned by the department) was available for labs and projects. This board contained motor drivers, screw terminals, voltage regulators, and prototyping space. The two board arrangement allowed students to easily buy and carry the smaller board but take advantage of more mature features in the lab.

The Design of the Boards

The Atmel ATMega32 has 32K of flash memory, 2K of RAM, 8 analog inputs, 4 PWM outputs, and up to four 8-bit ports for general I/O. It is easily interfaced to a serial port, however to add USB connectivity an FTDI USB-Serial bridge[8] was used. Under Windows and Linux this IC appears as a serial port. The processor (thumb) board includes a power LED and four LEDs on port B for simple diagnostics. A reset switch is used to restart the processor. The design of the board allows it to be directly inserted into a USB port, eliminating the need for any extra cables. The board was also configured with two diodes to allow it to draw power from the USB port (5V, 0.5A) or from an external power source. In this case there was a voltage drop, but the ATMega32 works with a supply voltage under 4V. The board uses a 0.1" spaced connector. Although this consumes space it does allow students to 'poke' wires into holes and use the boards for simple tests without a full breakout board. The schematics for the processor board are shown in Figure 1. The boards were assembled and the bootloader software was programmed using AVRStudio and an AVR programmer through a JTAG port with an In Circuit Programming unit. To avoid adding another pin connector, the JTAG port was integrated so that it was on the pin connector. After initially programming the bootloader, a temporary pin connector is removed, eliminating the need for another connector. After the bootloader is installed, students can program the board using a USB port and Megaload[9]. The typical programming cycle for students is outlined below.

1. Write a C/C++ program with GVIM, Notepad, Wordpad or some other text editor.
2. Compile the program with gcc[10] to a .hex file.
3. Insert the board in the USB port; it is recognized as a serial port.
4. Use Megaload to download the .hex file.
5. Use a terminal program to monitor or interact with the running program.

The second (breakout) board was designed to include power components, as shown in Figure 2. This board includes prototyping space, screw terminal connectors, a 7805 5V regulator, a TIP 122 Darlington coupled NPN transistor, and an L298 H-Bridge driver. To be used, the breakout board is plugged into the thumb board.

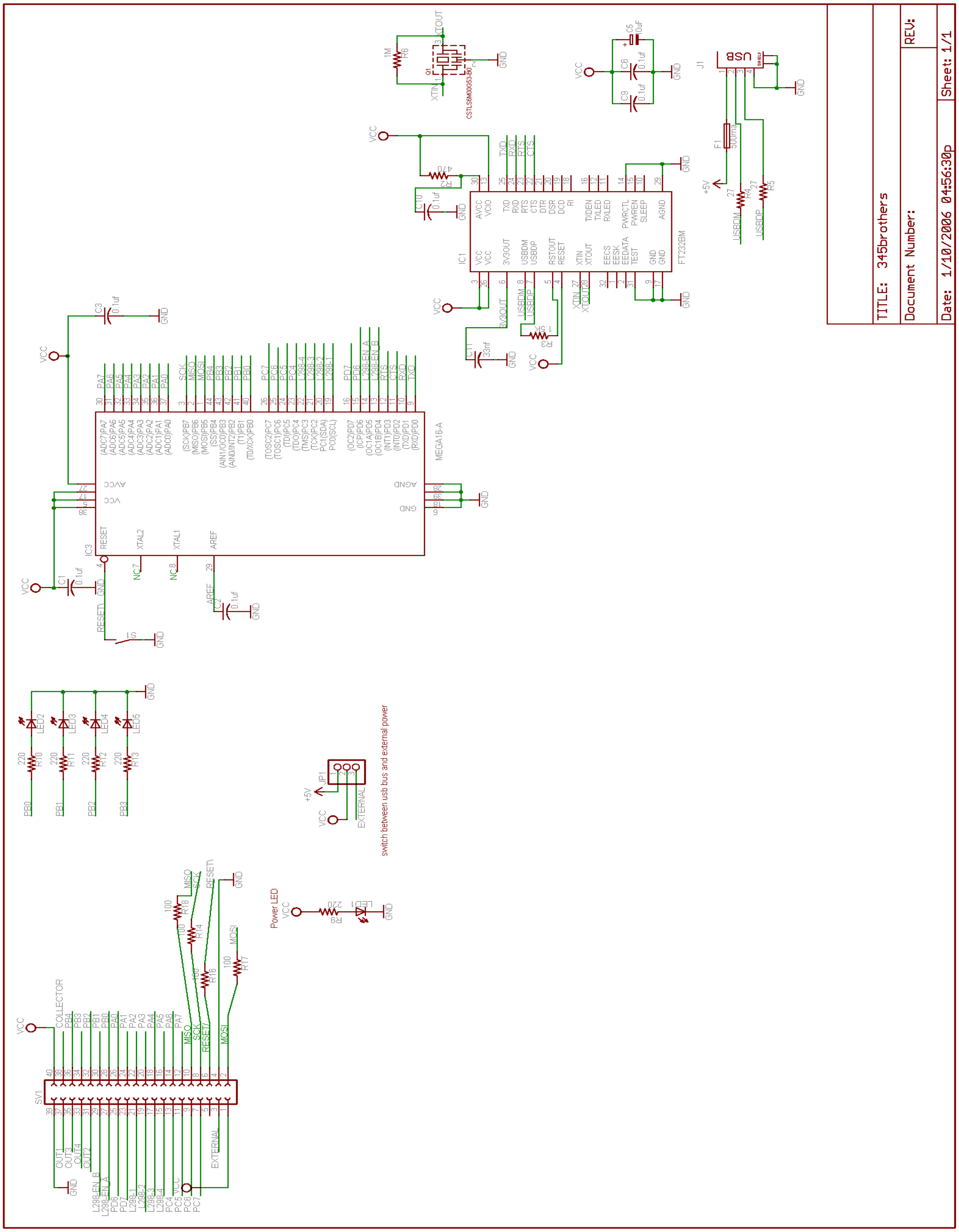

Figure 1 - ATMega32 Processor Board Schematic

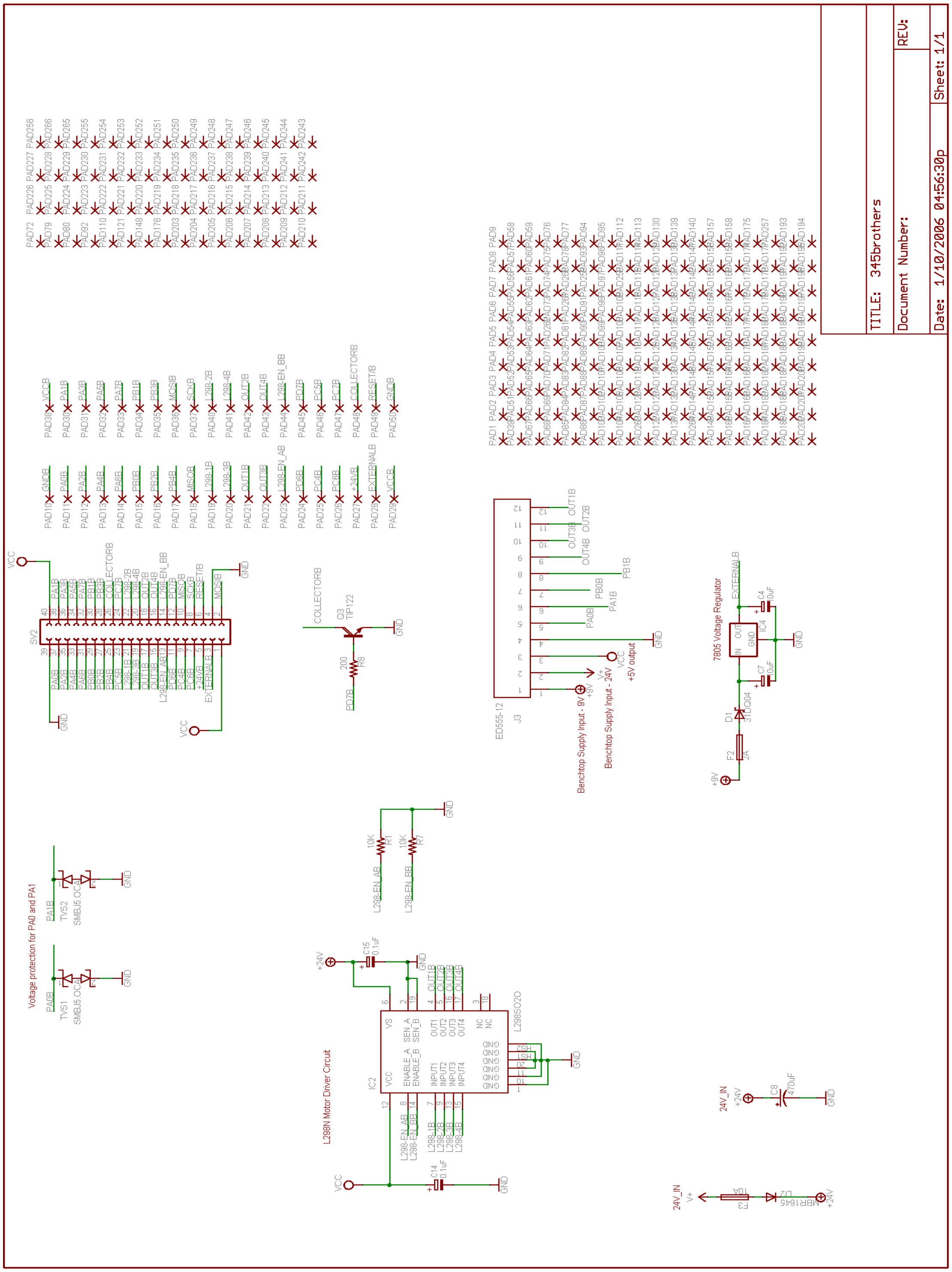
TITLE: 345brothers
Document Number:
REV:
Date: 1/10/2006 04:56:30p
Sheet: 1/1
Voltage protection for PA0 and PA1
L298N Motor Driver Circuit
7805 Voltage Regulator
Benchtop Supply Input - 9V
Benchtop Supply Input - 24V
+5V output
TIP122
ED555-12
L298SO20

Figure 2 - Breakout Board Schematic

The circuit board was primarily designed using surface mount components, as shown in Figure 3. As shown, there are two boards that are cut apart. The upper board is the USB thumb board that the students purchase. The lower board is the breakout board that is supplied in the labs. These boards were assembled by a co-op student working for the School of Engineering. Solder paste and a reflow oven were used for mounting most of the surface mount parts.

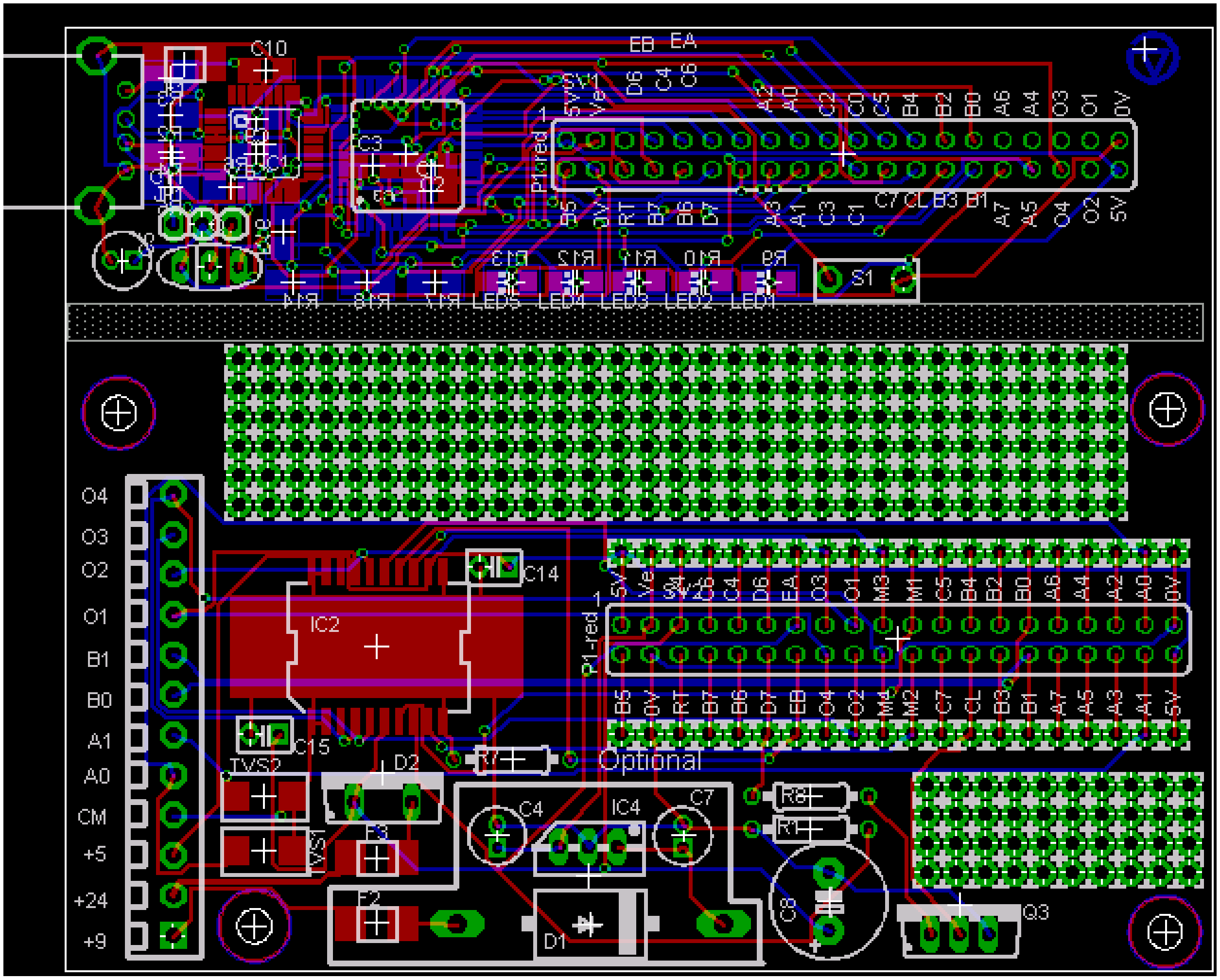
C10
EB EA
5V
Ve
D6 C4 C6 A2 A0 C2 C0 C5 B4 B2 B0 A6 A4 O3 O1 0V
B5 0V RT B7 B6 D7 A3 A1 C3 C1 C7 CL B3 B1 A7 A5 O4 O2 5V
C3
S1
LED5 LED4 LED3 LED2 LED1
R14 R18 R17
O4
O3
O2
O1
B1
B0
A1
A0
CM
+5
+24
+9
IC2
C14
C15
TVS2
D2
F3
F2
D1
C4
IC4
C7
R8
R1
C8
Q3
Optional
P1-red
5V Ve C4 D6 EA O3 O1 M3 M1 C5 B4 B2 B0 A6 A4 A2 A0 0V
B5 0V RT RT B7 B6 D7 EB O4 O2 M4 M2 C7 CL B3 B1 A7 A5 A3 A1 5V

Figure 3 - Board Layout (Note: the upper half is the thumb board, the bottom half is the breakout board, they are cut along the shaded area.)

In the fall of 2005 a total of 55 juniors purchased thumb boards. To support these students in labs and projects the department built and supplied 16 breakout boards. This move dramatically reduced the maintenance problems with only 5 students reporting damaged ICs. Students who did damage the boards were charged a nominal amount to replace the damaged parts.

Unfortunately, in the interest of being overly protective, the design originally used 0.5A fast blow fuses for the USB port 5V power supply. When using higher current loads these fuses would fail eventually. This problem was remedied by replacing the fuses with 1A slow blow fuses.

The final boards used in the course can be seen later in Figure 4. The students used various cases to carry the boards and some customized their cases. There was a definite ‘cool factor’ related to the thumb boards.

Experiences in the Lab

Laboratory experiments were published before the lab. Students were expected to do prelab work before arriving and record their results in a lab book with numbered pages. Also, they write and debug programs before the lab using the thumb boards. At the beginning of the lab, the students start by submitting their lab books to the lab instructor for prelab verification. After assembling the necessary external equipment to conduct the experiment, the evaluated lab books are returned and students modify their plan of action accordingly. Experimental analysis and post experiment analysis are also added to the lab book.

Some labs were run in teams and some were individually based. As the lab experiments list showed, each experiment included dealing with a different complex system, using a computer and an interface/control board, to achieve a new goal. Experiments that were not concluded during the lab period were completed by the students at their own convenience without challenges in equipment availability. A complete technical lab report was requested from a sub-group of students each lab period on a rotating basis to emphasize technical writing and communication. By the second half of the semester the major project started to surface and most of the students resorted to using their own boards as the project centerpiece. At that point they were so familiar with the microcontroller that no hesitation was expressed towards utilizing it on their own in their projects.

The Project

EGR 345 contains a major semester project that reinforces the topics learned in the lectures and laboratory. The semester project uses a project management approach to guide students teams to solve a complicated design problem. In the fall of 2004 the project was a two wheeled self balancing robot that could follow a line. In the fall of 2005 the project was an automated system for aiming and shooting balls at four targets in a head to head competition. The targets were randomly activated and the students were expected to aim towards the target and shoot a ball using a pneumatic supply and valve. Points were awarded to the team that hit the target first. The competition was controlled using a PLC for timing and scoring.

During these design projects the students designed all aspects of the device following given objectives and constrains including cost, weight, and performance. They then built and tested their devices. A few of the devices built are shown in Figure 4.

In total there were 55 students divided into 11 teams of 5. The objective was to produce a device that could aim and then shoot a ball at a target. The targets were randomly activated and then the students boards were signalled with one of four 5V inputs. The devices mostly used a servomechanism to turn towards the targets and then shoot a ball with compressed air. The compressed air was controlled with a PLC that handled all of the competition management tasks. The competition is described in greater detail on the course web page and students reports are also available[2].

The design objectives and constraints of the project were selected so that teams can use the microcontroller boards and apply the knowledge obtained during the laboratories. For the 2005 design the students selected a variety of sensors for measuring the barrel position including encoders and potentiometers. The designs were encouraged to be low cost ($57 to $156) and low weight (510g to 1.56 kg). The results of these efforts can be seen on the course home page[2]. The teams use the ATMega32 boards to build their machines. The projects included the design and building of the systems including mechanical, electrical, software, strategy, and system modelling.

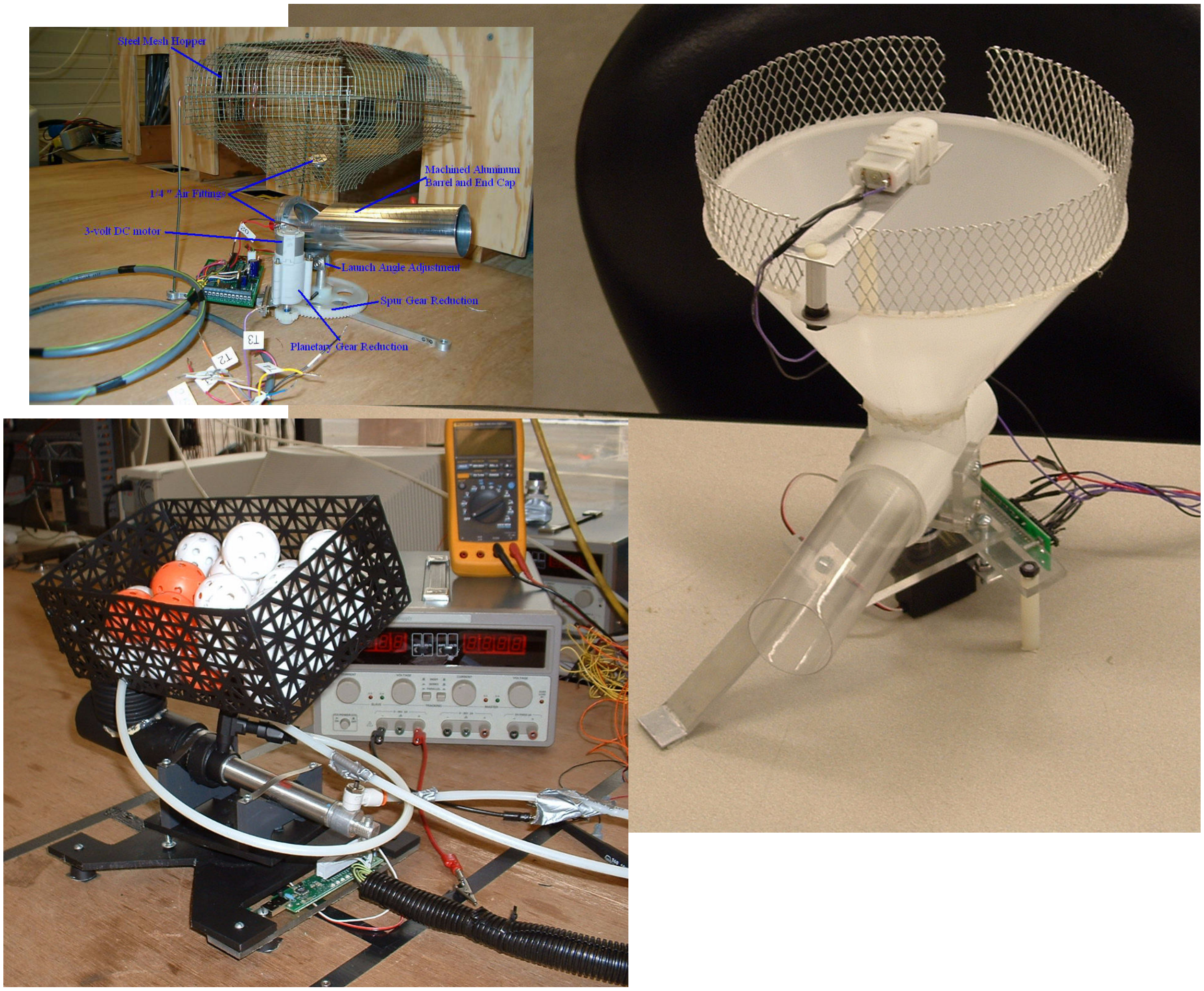

Figure 4 - Student Projects Using the Microcontroller Boards

The microcontrollers used in EGR 345 have also been used by students in other courses. Most notably students used this hardware in EGR 301 - Analytical Product Design. In that course, they are expected to design and build a consumer product. In the fall of 2004 the product was a pill dispenser and in the fall of 2005 the project was a self contained coffee maker[11].

Conclusions

Aside from the problem with the fuses (that has now been overcome) these boards worked very well. In future course offerings, the cost per board will be reduced. These boards, or another generation will be used in coming years. The size of the boards meant that many students carried them in their backpacks. Many students used carrying cases (such as eyeglass cases) for protection. A project in another course began making injection molded prototype cases for the boards that will be expanded further in the coming year.

With the new focus on microcontrollers the students were able to implement a greater number of more advanced control systems during the course. Moreover, when they did use LabVIEW for the first time in laboratory 11, they wrote more advanced programs than students in previous years who had taken the LabVIEW only version of the course. The knowledge students obtained in the laboratory was applied in the semester project.

The results of using microcontrollers continues to be highly successful. On a weekly basis in the laboratory, students implement complex control systems using C programs. The low cost of the boards enables student ownership of hardware used in the design of common consumer products.

Acknowledgements

This work was made possible with the help of others including Jeffry Roberts and Kurt Hammons who developed the final board. Dr. Andrew Sterian provided valuable assistance with the Atmel architecture.

References

1. Jack, H. “Dynamic System Modelling and Control”, Grand Valley State University, Version 2.6, 2004.
2. Jack, H., “EGR 345 - Dynamic System Modelling and Control”, course homepage, http://claymore.engineer.gvsu.edu/~jackh/eod/egr345.html, 2006.
3. Scilab, http://www.scilab.org.
4. National Instruments, http://www.ni.com.
5. Axiom Manufacturing, http://www.axman.com/.
6. Jack, H., “A Modeling and Controls Course using Microcontrollers”, ASEE Annual Meeting, Portland, OR, 2005.
7. Eagle Software, CadSoft, http://www.cadsoft.de/.
8. Future Technology Device International, Ltd., http://www.ftdichip.com/.
9. Microsyl (Megaload), http://www.microsyl.com/.
10. Gnu C Compiler project, http://www.gnu.org/software/gcc/.
11. Farris, J., “A Product Design Sequence“, ASEE Annual Meeting, Chicago, June 2006.