

## Using Rapid Prototyping Tools for Automatic Control System Laboratory

Robert S. Cochran, Todd D. Batzel, and Peter J. Shull  
Penn State Altoona, Altoona, PA 16601  
[rsc148@psu.edu](mailto:rsc148@psu.edu), [tdb120@psu.edu](mailto:tdb120@psu.edu), [pjs18@psu.edu](mailto:pjs18@psu.edu)

### Abstract

In this paper, the development and implementation of laboratory tools for a control system laboratory is presented. Typical control system laboratory experiments are excessively time-consuming, require complex student instructions, and are inflexible and expensive. These drawbacks are addressed with the proposed system, which incorporates rapid prototyping software development tools, a user-friendly graphical user interface (GUI), a DC motor with feedback, a motor drive, and a low-cost digital signal processor (DSP) to carry out the control algorithm. This workstation provides a flexible, low-cost control system trainer that is capable of supporting a variety of demonstrations and experimental activities while encouraging student experimentation and emphasizing practical aspects of control systems design, simulation, and real-time implementation. An example of a laboratory exercise is included to illustrate the suitability of the system for student use and the effectiveness of the overall implementation. The prototype control system trainer is presently being integrated into the Engineering Technology curriculum at Penn State Altoona.

### 1. Introduction

Recent developments in the area of rapid prototyping tools for software have brought about a new paradigm in the development of control systems. In the industrial sector, these tools have enabled developers to quickly and easily design, simulate, and deploy complex real-time control systems [1], [2], [3]. This technology can be exploited to realize improvements in control system education as well. Typical control system courses focus on theory with little emphasis on implementation and practice [4]. Rapid prototyping tools provide a seamless transition from system design to implementation, and therefore can be employed to bridge the gap between *theory* and *practice* in our control system courses [5], [6].

In this paper, the apparatus and software for an automatic control system trainer that employs rapid prototyping tools is presented. The main goal of this work is to replace existing control system trainers, which have excessive response times, are difficult for students to operate, are inflexible and costly, and do not encourage student experimentation.

The proposed control system trainer consists of a permanent magnet DC motor and quadrature encoder, a mechanical load, a low-cost Digital Signal Processing (DSP) board with power electronics, and the MATLAB<sup>®</sup> Simulink rapid prototyping tools for software development. With this equipment, the current, velocity and position of the electric motor are controlled by the

DSP board, and the complete control system is programmed graphically. A GUI is used to provide real-time student interaction with the system, including gain selection, command input control, and data logging and display. The trainer is thoroughly tested to verify its suitability for laboratory use. With further development, the proposed control system trainer is expected to provide an excellent educational experience for students of automatic control systems.

## 2. Control Systems Laboratory Equipment Development

In this section, the use of rapid prototyping tools in the development of an educational workstation for automatic control systems is described. Software, hardware and the integration of those components to establish the control system workstation are discussed.

### 2.1 Software

Software tools such as Matlab<sup>®</sup> have long been used in the design, analysis, and simulation of control systems. Recently Matlab has introduced real time code generation tools that target common microcontrollers and DSPs. This tool translates a graphical control system description into executable code, therefore enabling real-time control systems to be quickly developed and tested. When integrated with appropriate hardware these tools form a visual modeling and simulation environment that is effective and easy-to-use for the entire control system development cycle.

To perform control system design and simulation, a graphical description of the system is entered into Matlab Simulink<sup>®</sup>. A high-level graphical control system description for DC Motor position control, for instance, is shown in Fig. 1. Each block in Fig. 1 can be examined by double-clicking on that block. In Fig. 2, the contents of the *A-to-D Block* of Fig. 1 are shown. This hierarchical design isolates students from implementation details embedded within low-level blocks, yet promotes student experimentation since modifications to the control system are performed in the graphical environment. Furthermore, since the graphical description shown in Fig. 1 looks much like the block diagram format commonly used in the study of control systems, students are comfortable working with the graphical control system description and can quickly configure and simulate a system.

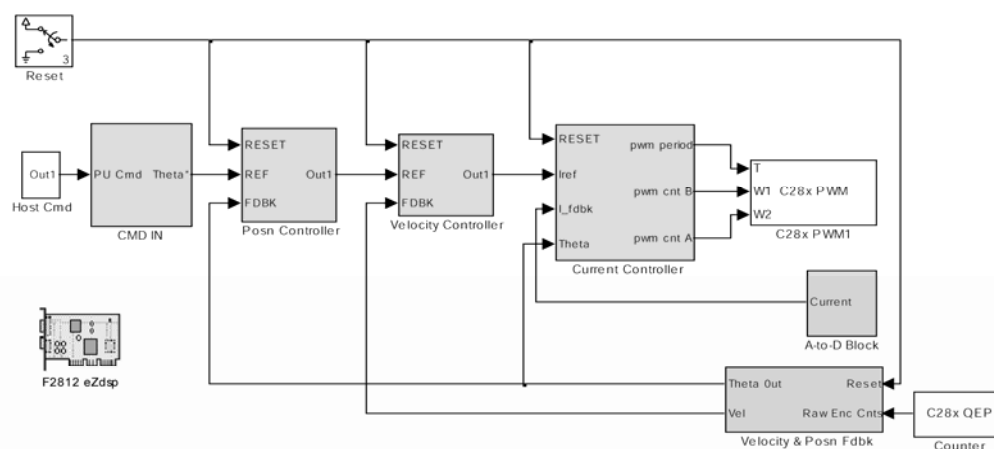


Fig. 1. Graphical description of permanent magnet DC motor position controller.

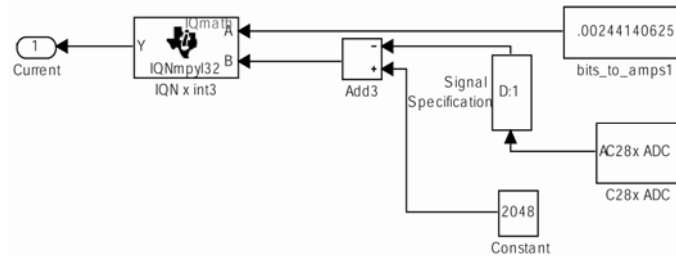


Fig. 2. Contents of the *A-to-D* block of figure 1.

When simulations yield satisfactory performance, the graphical description of the controller is converted by the Matlab<sup>®</sup> Real-Time Workshop into C code, cross-compiled, and downloaded to the DSP board. No programming is required. The DSP executes the auto-generated code and communicates information with the host computer via Real Time Data Exchange (RTDX). The GUI, which is shown in Fig. 3, provides the opportunity to optimize control system gain settings in real-time, and provides access to operating conditions associated with the system.

## 2.2 Hardware

Following satisfactory simulation of the graphical description of the DC motor control system, the design is ready to be deployed using actual hardware. For the workstation described in this paper, the hardware consists of a permanent magnet DC motor with encoder, a DSP development board, and power electronics interface to drive the motor. The rapid prototyping tool is a key element in implementing the control system, since it quickly translates the graphical description of the system into executable code capable of operating the hardware in a real-time experiment.

The automatically generated code is downloaded and executed by a TI 2812 development board (Spectrum Digital Inc. ezDSP2812). This board contains a 32-bit fixed-point DSP operating at 150 MHz and with 128 kwords of on-chip memory. This DSP is optimized for motor and motion control applications, and carries 16 channels of 12-bit analog-to-digital converters, integrated encoder interface, and 6 programmable pulse width modulation (PWM) output channels. The 32-bit fixed point mathematics are easily managed with the fixed-point blockset that is integrated with Matlab Simulink.

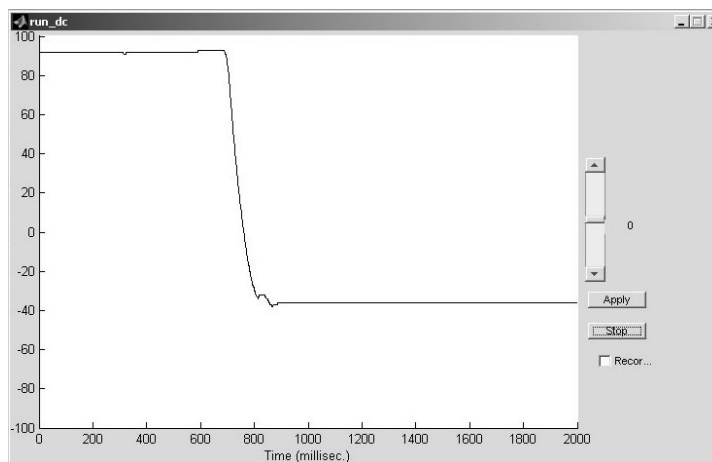


Fig. 3. Graphical user interface for the DC motor control workstation.

The permanent magnet DC motor is equipped with an encoder and analog tachometer for rotary position and angular velocity feedback. As shown in Fig. 4, the motor hardware arrangement provides a small and light test stand that is easily stored. A prony brake was included to provide a variable in the control system. The prony brake tension is easily adjusted by using the screw on the top of the test stand. The arrangement is such that a variable inertia disk could also be included in the system to provide another variable in the control system.

The power electronics provide the required interface between the DSP and the DC motor test stand. The selected power electronics interface is a commercially available solution (Spectrum Digital DMC 1500) that provides a 10 amp three-phase inverter with PWM signal conditioning, on-board power supplies, encoder interface circuitry, and signal conditioning for the analog-to-digital interface. The DSP board connects directly to the compatible power electronics board. The DSP board is shown mounted on the power electronics in Fig. 5.

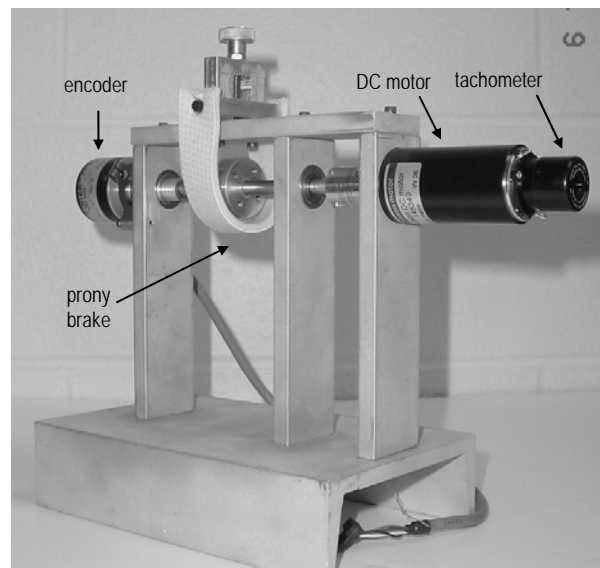


Fig. 4. Test stand with DC motor, encoder and prony brake.

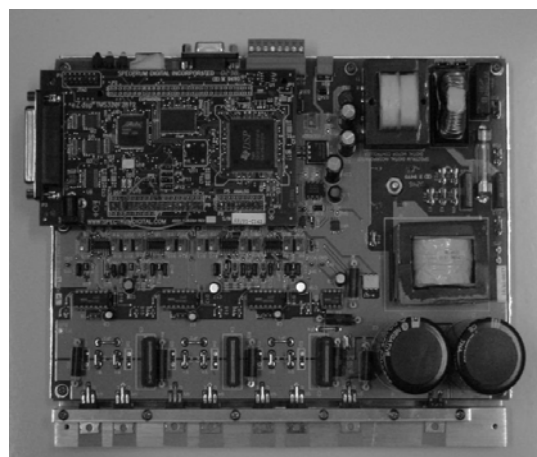


Fig. 5. Power electronics and DSP board.

## 2.3 System Integration

The control system design process is generally considered to consist of three iterative procedures: design, simulation, and implementation. Although many automatic control system courses do not emphasize the implementation aspect of the design cycle, the ideal workstation would support each step of the design cycle. That is, the system must support learning of control system theory, design, and implementation through hands-on laboratory experiences that do not present a steep learning curve to students. The integrated components of the proposed workstation are shown in block diagram form in Fig. 6.

The typical design cycle using the proposed workstation includes determination of the plant model, controller design and simulation, implementation, and final gain selection. The components of this design cycle are normally iterated until design goals are met.

The workstation can be used to perform hands-on experiments to determine the modeling parameters of the plant (DC motor), such as tachometer gain, armature resistance and inductance ( $R_a$ ,  $L_a$ ), the motor back-emf and torque constants ( $K_\omega$ ,  $K_t$ ) and mechanical time constant ( $\tau_m$ ). For example, to determine armature resistance and inductance, the DSP may be programmed (graphically) to excite the armature with a step voltage waveform, while the resulting current waveform is displayed and analyzed via the graphical user interface.

After the plant model is determined, the appropriate controller may be designed and simulated on the host computer. When the designer is satisfied, the resulting control system is downloaded to the DSP for real-time analysis. In the real-time experiments, the DSP performs the controller algorithm and sends selected data to the host computer for display and analysis. Parameters associated with the control system may be adjusted in real-time through the host computer interface. This interactive experimentation provides an excellent experiential learning opportunity for control system parameter tuning.

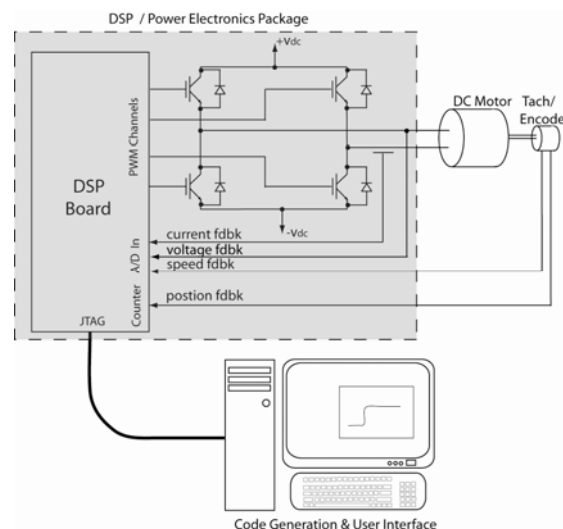


Fig 6. Block diagram of control system workstation.

The host PC shown in Fig. 6 is dual-purpose. In the design cycle, it is used to analyze and simulate the system. During the implementation phase, the PC host provides the user interface that communicates data through the JTAG interface of the DSP to the PC parallel (or USB) port. This communicated data is displayed on the host, and can also be recorded to a file on the host computer for further analysis with Matlab or Excel.

### 3. Control System Experiments Using Rapid Prototyping Tools

In this section, samples of the control system experiments developed using the rapid prototyping tools are presented. Results are included to demonstrate the effectiveness of the developed learning module.

#### 3.1 Experiment Descriptions

Although the workstation is capable of supporting a variety of plants, the focus will be on the DC motor plant described in Section 2. A typical series of experiments using this plant include the model determination, torque control, velocity control, and position control.

*Model Determination:* In an initial laboratory session, experiments may be conducted to determine the modeling parameters of the DC motor plant. The model determination experiments can be dual-purpose; it may act to acquaint students with the DSP and host computer GUI operation as well as the obvious objective of plant modeling.

*Torque, Speed, and Position Control:* After the motor model is determined, a current, velocity, and position control may be developed sequentially by students to meet a given specifications. Single loop current, speed, and position control design laboratories may be employed to demonstrate control systems design for first order and higher order plants. In latter laboratory experiments, the advantage of the cascaded control loop structure can be demonstrated.

Many motion control systems employ cascaded position, velocity, and current control loops [7], as shown in Fig. 7. The advantage of the cascaded structure is that the design and tuning of the loops are conducted step-by-step beginning with the innermost current loop. The tuning of the loops is performed independently, with the assumption that the bandwidth increases toward the inner loop.

As a first laboratory experience with control system design for a first-order system, students may design a current (torque) controller for the DC motor. The primary goal for the current control design is for high bandwidth. This is necessary so that in later experiments, the current

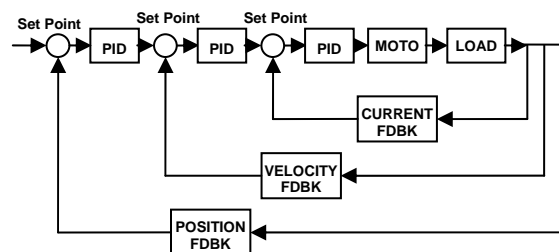


Fig. 7. Block diagram of the cascaded DC motor feedback system.

controller may be regarded as a current source to the slower outer loops. In the cascaded system, the current controller also plays a major role in stabilizing the position control [8]. This can be demonstrated in subsequent laboratory experiments.

Rotor velocity and position control loop designs follow logically, since the innermost current controller has already been designed. The velocity controller is first designed and implemented, followed by position control. Both of these controllers may be designed using a single loop, and using the cascaded structure. Comparison of these implementations will allow students to determine which strategy is more stable, and easy to tune.

### 3.2 Experimental Results

In this section, the experimental results of a DC motor position control system using the cascaded loop structure shown are shown. The goal of the experiment is to obtain fast and accurate response to changes in the commanded rotor position with low steady-state error. The input from the user in this experiment is the tension of the brake, the rotor position command, and the PID variables for each cascaded loop of the feedback control system. The motor response can be viewed in real time through the GUI, which will display the current, velocity, and position of the motor. This data also can be saved and then later analyzed in MATLAB.

The results of this experiment are shown in Figs. 8 and 9. Fig. 8 shows the motor position and controller output with well-tuned PID variables for each controller (current, velocity and position)

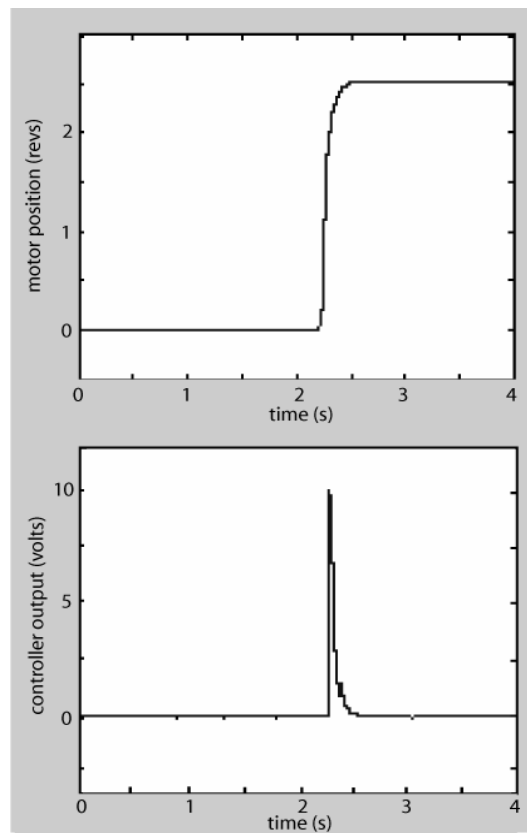


Fig. 8. Motor position (top) and controller output (bottom) for a tuned system.

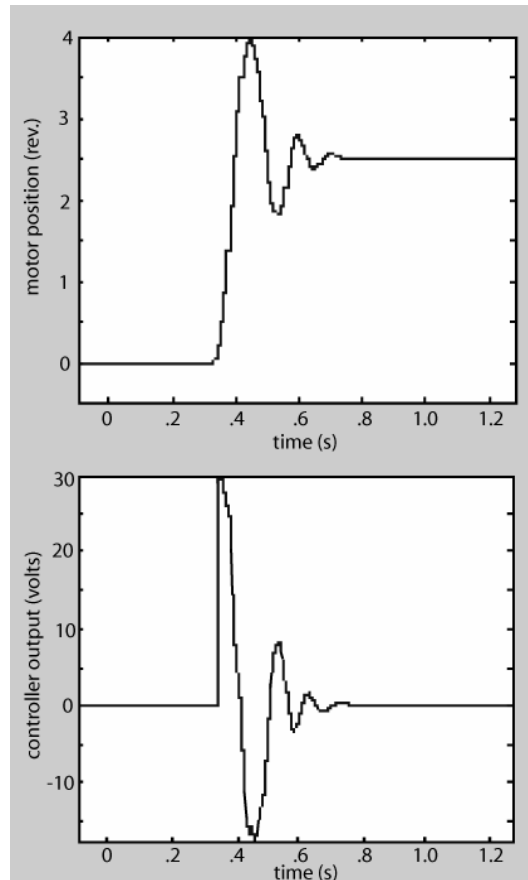


Fig. 9. Motor position (top) and controller output (bottom) for an untuned system.

loops). As seen in Fig. 8, the motor responds immediately, has no overshoot and an excellent settling time. This is an example of what the students will want to achieve at the end of their lab experiments.

Fig. 9 shows a DC motor response that exhibits overshoot and slow settling time. Comparison with Fig. 8 shows the increased control effort required of the poorly-tuned system. The workstation allows students to manipulate PID gains until a response similar to Fig. 8 is achieved – thus demonstrating the effect of each PID variable. This exercise provides students an excellent educational experience related to the tuning of automatic control systems.

#### 4. Conclusion

A control system workstation that employs rapid prototyping tools has been developed and evaluated for use in an automatic control system laboratory. The system has been found to be flexible and low-cost, and encourages student experimentation. The rapid-prototyping aspect of the workstation allows students to implement the control system quickly, permitting a hands-on learning experience that bridges the gap between control systems design and implementation.

Although a DC motor is used as the plant in this experiment, it should be mentioned the DSP is capable of supporting a variety of control plants, such as a simple RC circuit, buck converter or



other student-developed plants. In the future, more plants will be developed for laboratory experimentation.

## 5. Bibliography

- [1] Anakwa, W.K.N., E. Cohen, A. Naik, D. Carlton, D. Glen, and J. Lopez (2001). Tools for rapid prototyping of embedded control systems. *The 27<sup>th</sup> Annual Conference of the IEEE Industrial Electronics Society*, vol. 1, pp. 90-94.
- [2] Yan, Y. and R. Zane (2004). Digital controller design for electronic ballasts with phase control. *IEEE 35<sup>th</sup> Annual Power Electronics Specialists Conference*, vol. 3, pp. 1855-1860.
- [3] Hawkins, C.E., and I.J. Berry (1997). Rapid prototyping in the Automotive Industry. *IEEE Colloquium on System Control Integration and Rapid Prototyping in the Automotive Industry*. Vol. 1, pp. 1-3.
- [4] Shiakolas, P.S., and D. Pitabongkarn (2003). Development of real-time digital control system with HIL magnetic levitation device. *IEEE Transactions on Education*, vol. 46, no. 1, pp. 79-87.
- [5] Uran, S., D. Hercog, and K. Jezernik (2004). Experimental control learning based on DSP2 learning module. *2004 IEEE International Conference on Industrial Technology*, vol. 1, pp. 310-315.
- [6] Van deMolengraft, R., M. Steinbuch, and B. DeKraaker (2005). Integrating experimentation into control courses. *IEEE Control Systems Magazine*, vol. 25, no. 1, pp. 40-44.
- [7] Leonhard, W. (1985). *Control of Electrical Drives*. Chapter 7. Springer-Verlag, Berlin.
- [8] Ohm, D.Y., and R.J. Oleksuk (1998). On practical digital current regulator design for PM synchronous motor drives. *Thirteenth Annual Conference Proceedings of the Applied Power Electronics Conference and Exposition*. vol. 1, pp. 56-63.

## 6. Biographies

ROBERT S. COCHRAN received the B.S. in electro-mechanical engineering technology in 2005 from Penn State Altoona, Altoona, PA. He is presently the engineering laboratory coordinator at Penn State Altoona.

TODD D. BATZEL received the B.S. and Ph.D. degrees in electrical engineering from the Pennsylvania State University, University Park, in 1984 and 2000, respectively, and the M.S. degree in electrical engineering from the University of Pittsburgh, Pittsburgh, PA, in 1989. Currently, he is an Assistant Professor of Electrical Engineering at Penn State Altoona.

PETER J. SHULL received the B.S. in mechanical engineering from Bucknell University, Lewisburg, PA, in 1982. He earned the M.S. in materials science and engineering in 1992, and Ph.D. in 1995 from Johns Hopkins University, Baltimore, MD. Dr. Shull is presently an Associate Professor of Engineering at Penn State Altoona.