A COURSE SEQUENCE FOR INSTRUMENTATION AND CONTROL

CARLOS A. SMITH, RICHARD A. GILBERT
University of South Florida
Tampa, FL 33620

At the University of South Florida we believe that a controls and instrumentation background is vital for new chemical engineers. Our curriculum specifies a required three-semester course sequence in the area, with an additional elective course also available. The three required courses are

- Mechanical Engineering Laboratory I (MELab), 3 credits
- Instrument Systems I (ISys), 4 credits
- Automatic Process Control I (APC-I), 3 credits

and the elective course is

- Automatic Process Control II (APC-II), 3 credits

The first course is taught by the mechanical engineering department, and all the other courses are taught by the chemical engineering department. The first two courses are required for all mechanical engineering and chemical engineering students.

This paper will describe the subject matter and the goals the authors have set for each course. The principal goals for the complete sequence are to provide the students with the technical background:

- to immediately begin productive careers in any industrial controls group
- to understand the fundamental principles involved in data acquisition for experimentation and process control
- to be able to keep up with future developments in this very dynamic field

Because of the nature of the goals, all of the courses are practice-oriented and heavily dependent upon laboratory and project work.

Mechanical Engineering Laboratory I (MELab)

This one-semester course consists of one two-hour lecture and one three-hour laboratory period per week. The goals of the course are to introduce the student to various physical variable measurement devices and techniques and to develop proper report-writing skills (Table 1 lists the contents of the course).

Initial lecture times are devoted to laboratory safety issues, required elements in laboratory reports and presentations, and the properties of measurement and experimental procedure errors.

In the remaining weeks of lecture, students are introduced to measurements of fundamental interest to the process industry (i.e., temperature, pressure, and flow) as well as measurements that relate to the monitoring and control of mechanical systems (i.e., strain, torque, and displacement). Laboratory work during this phase of the course involves setting up and taking measurements with sensors similar to the ones discussed in class. The students provide written reports for each of the ten experiments and must include a summary of any required engineering calculations. They must also contain descriptions of the experimental arrangement and any special analog signal conditioning required to complete the measurement, i.e., any temperature compensation technique used for a thermocouple measurement, the type of measurement bridge used with the strain gauge, and the flow element analog output signal manipulation employed to obtain measurement signals proportional to engineering flow units.

Instrument Systems I (ISys)

This one-semester course consists of three one-hour lectures and one three-hour laboratory period per week. Since MELab is a co-requisite and/or a pre-requisite for this course, knowledge of various types of sensors and their analog output signal char-
characteristics is expected of the students. The course introduces the various ways digital technology incorporates a sensor into an instrument system used for process monitoring and control. Table 2 summarizes the topics presented in the course.

Although the digital skills learned in ISys are applicable in any research-oriented application, it is convenient to present the course by using process control examples. At this stage in their education, the students are quick to appreciate how the sensor and its digital interface can be used in an industrial control situation. They also understand the usefulness of a computer or digital controller as the center element of a control loop. By contrast, it is difficult to find examples in their chosen fields of study that are within the experience level of both the chemical and mechanical engineering students in the course. Finally, they are able to relate the digital technology to monitor and control projects of immediate interest to themselves, *i.e.*, digital display of wind speed on a sailboard, a variety of home comfort and security interlocks, etc.

Three aspects of control and data acquisition system interfacing

- discrete component technology
- software drivers
- dedicated computer interface boards

are covered. Lecture and laboratory time are divided equally among them. Component technology is discussed first while the selection/operation of interface boards is covered in the last four weeks.

The discrete component technology section of the course covers TTL devices and their use as interfaces between a process sensor, a controller, and the final control element. Presenting the discrete digital technology concepts first has the effect of delaying discussions about analog-to-digital conversion and other aspects of interfacing an analog sensor output to a digital system until those students taking MELab as a co-requisite with ISys have been exposed to a few examples of analog sensors. In this first part of the course, students explore simple control schemes that can be handled by an arrangement of discrete components. Examples include elementary on/off interlock logic, event counting, and time-delay situations. The function of open collector and three-state devices as interfaces to traditional industrial control voltages and computer circuits is also examined.

At this point in the course, our intent is to make students feel comfortable working with control TTL circuits. Example A (next page) summarizes a project given in APC-I—the same assignment is also given as an ISys lab. The problem is scaled down to facilitate the time base when discrete counters are used, but the logic portion of the assignment is the same. The full effect of the learning experience is not appreciated until the controls course is taken, but the lesson is not lost with time. Once the students solve the problem by both methods, they can appreciate the value and place for each.

Another reason for presenting discrete component information is to show students the interrelationship between operation of the components and the function of a complete interface subsystem. This presentation is not an abstract academic exercise—it is an opportunity for the student to develop skills that can be employed in real instrumentation trouble-shooting scenarios.

In a research environment these trouble-shooting skills might be used to determine and maintain a specific arrangement of triggering and then multiplexing the signals from several analog experiemen-
tal measurements into a single A/D-based data logger. In an industrial setting, trouble-shooting situations inevitably involve problem diagnosis of a sensor, a controller, and a final control element in a loop with no initial certainty of which loop element is at fault. Understanding a loop's digital components facilitates the distinction between the analog and digital aspects of the control scheme. This helps the engineer isolate the portion of the loop that is not performing properly. Because control loops are so intimately related to the process under control, the person responsible for that part of the process usually directs the trouble-shooting "mission." More often than not, that person is a chemical engineer.

The course's software driver section focuses on data transfer to and from a digital system. For this purpose a software driver is defined as the subroutine responsible for the data transfer operation. Students learn to write code that directs interactions between a CPU and its I/O ports. The intent is to make them aware of the I/O port structure, how those ports transfer sensor information to the CPU, and the immediate operations required of a CPU to accomplish that task. Once these operations are understood, an engineer can alter the computer interface of an existing control system to meet the new or modified needs of the project.

Laboratory experiences for the software driver portion of the course include exercises in writing code to operate digital I/O ports as well as A/D and D/A converters. Students learn to write 8080 and then Z-80 code. They may cross-assemble their programs on available IBM-type computers. Completed programs are run on single-board computers that the student has connected to the ports of interest.

Single-board computers are used in the laboratory because they are simple to operate, they support a variety of INTEL "smart chips," and they are easy for the department to maintain. Because of the simplicity of these computers, students are forced to learn what the role of the I/O driver is and what must be done to complete the interface. They also develop valuable software debugging techniques that can be used with any higher-level language.

The last section of the course covers the function and operation of dedicated computer peripheral interface boards. By this time students are expected to understand the technical information provided by the manufacturer of such boards. Lecture and laboratory material cover A/D boards, thermocouple and RTD interface boards, and how to use them.

Although some of the interface boards used in the lab come with manufacturer-supplied menu-driven software packages, the students are directed to develop small routines that control the board. This approach shows them that they understand what the technical manual is trying to say, that they can write code that actually takes a measurement, and how to get maximum flexibility out of the board. Students develop their programs in Quick BASIC and use the program debugging skills they developed during the second phase of the course.

One laboratory assignment at this point in the course is based on an RS-232 serial port card and a serial printer. Requesting that the student create a program that alters the font of the printer is identical in practice to setting up a RS-232 supporting interface. The correct command string has to be sent to the instrument's control register, and

---

### EXAMPLE A

Consider a tank where a dilution process takes place. The tank and accessories are shown in the figure. A concentrated solution enters the tank where water is sprayed at a significant rate to promote mixing. There are four water valves feeding the tank. Because of process constraints not all valves can be open at the same time; maximum of two are permitted at the same time. It has been proposed to open the valves in the following sequence:

<table>
<thead>
<tr>
<th>Valves</th>
<th>Time Duration, sec.</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 ON 2 OFF 3 ON 4 OFF</td>
<td>5</td>
</tr>
<tr>
<td>1 OFF 2 ON 3 ON 4 OFF</td>
<td>10</td>
</tr>
<tr>
<td>1 ON 2 OFF 3 OFF 4 ON</td>
<td>10</td>
</tr>
<tr>
<td>1 OFF 2 ON 3 OFF 4 ON</td>
<td>5</td>
</tr>
</tbody>
</table>

Start/stop PBs are available to start/stop the sequence. In addition, a dilution sensor (conductivity sensor/transmitter) is available to measure the amount of dilution. If the solution becomes too diluted (conductivity switch goes high) the valves must close until the sensor indicates (conductivity switch goes low) to resume the valves sequence.

Design the logic for this process, develop the logic diagram, and program the PLC to accomplish this control strategy.

---

*Chemical Engineering Education*
proper delimiters must be used to frame the messages. Finally, status information about the instrument and the process must be imported through the RS-232 interface back to the computer.

Using a printer instead of another instrument in this experiment has several advantages: printers are cheap, and even low-priced ones give several protocol and delimiter options; students enjoy learning how to write their own software to drive a printer; and finally, students have no trouble transferring what they have learned to the more sophisticated, more expensive, and more difficult to repair RS-232 driven measurement instruments.

Collectively, the device, the driver, and the dedicated board portions of this course provide a sound background for all phases of digital interfacing. A detailed note set, a study guide, and a collection of over twenty-five different subject-related books on library reserve provide additional independent learning resources for the students.

The practical nature of the course materials, together with the hands-on concept of reinforcing laboratory exercises, make the course immediately rewarding for most of the participants. Those who do not enjoy it usually complain about the lack of specific guidance during the laboratory periods. However, it is a conscious decision on our part not to provide step-by-step laboratory projects. Several lab periods are provided so that each student can work with the isolated concepts. During this time, the problems students have in working with example circuits, programs, or subsystems are explored. They learn alternative routes to diagnose the problems, but little time is spent on telling them exactly how to fix their problem. This approach of suggesting more things to try instead of offering corrective instruction is just not what some students want.

**Automatic Process Control I (APC-I)**

This is a one-semester course consisting of two hours of lecture and three hours of laboratory per week. It is directed at the process industries and includes several examples of environmental and material handling processes. The textbook is *Principles and Practice of Automatic Process Control*.[1]

Table 3 presents the lecture content. Note that the subject of Laplace transforms is not discussed—it is assumed that the student has "learned" this subject in differential equations. To test this assumption and to help the students review the material, a test is given during the second week of class. Questions about temperature, pressure, level, and flow sensors are also asked during the test as an aid in helping the students review their MELab course.

The lecture begins with a presentation on system dynamics, block diagrams, and transfer functions. This material is usually perceived as very theoretical, mathematical, and in general boring. Therefore, a deliberate, complete discussion on why the material is necessary and how it is useful is presented on the first day of lecture. We stress mathematical modeling, the physical significance of gain, time constant, dead time, how the response of systems in series provides dead time, and the importance and significance of nonlinearities. We also provide a physical explanation of what the mathematics indicate.

Once the foregoing material is learned, we present a discussion on controllers: the action of controllers, the mathematics, the physical significance, and the different types of controllers are discussed in detail. Process identification, by low-order models, and controller tuning are discussed in the lecture and practiced in the laboratory. The identification method discussed is the step, or bump, testing procedure. Students are organized into groups of two in the laboratory, and each group is asked to tune the controllers for two different simulated processes. Each simulated process is connected to real controllers. The laboratory equipment consists of a Honeywell TDC 2000[2] distributed-control system, two CLC-002 Bailey,[3] and two Yokogawa[4] stand-alone controllers.

The topic of stability starts with a presentation of the development of the mathematical model, block diagram, and the closed-loop transfer functions of a closed-loop control system. We continue with a definition of the characteristic equation and its relation-
ship to stability. Using the Routh Test and direct substitution to study stability is then presented.

Finally, we demonstrate the effect on the stability of the loop if any of its parameters change, i.e., if a transmitter with a different span is used, if a faster or slower sensor is used, or if the dead time changes, etc. We also discuss the effect of the reset and derivative actions on the loop stability. It is important to notice that only the simple techniques of the Routh Test and direct substitution are used in this presentation (Pade approximations are used for dead times). The idea is to show in simple, but effective, language the effect of the different terms on the stability of the loop and that these techniques do the job.

In addition to the lecture material and the homework, we ask the students to complete about three simulation projects during the course. The simulation software package used is TUTSIM.\(^5\) (If a student wants to work with another simulation package, or with straight FORTRAN or BASIC, he/she is allowed to do so.) The TUTSIM simulation package is a PC-based system and is considered to be one of the easiest packages for this purpose. The TUTSIM package is highlighted in the recently published second edition of the textbook *Process Systems Analysis and Control*.\(^6\) The projects usually involve the development of the model for a process system, the simulation, and the tuning of the control system.

Toward the end of the semester two design projects are assigned. They provide an opportunity for the students to design control systems for complete processes. The students usually like these projects.

The laboratory presents another interesting part of the course—Table 4 shows the weekly schedule. The first week is used to stress the importance of the course. We use this period to motivate the student to learn the subject. We then present the TUTSIM software package in the second week. This presentation usually takes about an hour and a half, and then a project for the following week is assigned.

In the first week PLC relay logic (AND, OR, latches, etc.) and instructions on how to develop logic and ladder diagrams are presented. This is usually accomplished in about one and one-half hours, and then a project for the following week is assigned. In week two, timers are covered and a project is assigned. Counters are presented in the third week, along with another project assignment. Finally, in the fourth week, students read about sequencers, and a new project is assigned.

During the fifth week of discrete logic lab, the final project (exam) is presented. During the past

### Table 4

<table>
<thead>
<tr>
<th>WEEK(S)</th>
<th>CONTENT</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Introduction to process control</td>
</tr>
<tr>
<td>2</td>
<td>Introduction to TUTSIM</td>
</tr>
<tr>
<td>3-9</td>
<td>Discrete and sequential logic</td>
</tr>
<tr>
<td>10</td>
<td>Valves</td>
</tr>
<tr>
<td>11-14</td>
<td>Tuning of feedback controllers</td>
</tr>
<tr>
<td>15</td>
<td>Wrap-up</td>
</tr>
</tbody>
</table>

### Table 5

#### Automatic Process Control II: Lecture

<table>
<thead>
<tr>
<th>TOPICS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Control loop stability</td>
</tr>
<tr>
<td>a. Root locus</td>
</tr>
<tr>
<td>b. Frequency response</td>
</tr>
<tr>
<td>c. Pulse testing</td>
</tr>
<tr>
<td>2. Cascade control</td>
</tr>
<tr>
<td>3. Ratio control</td>
</tr>
<tr>
<td>4. Feedforward control</td>
</tr>
<tr>
<td>5. Selective, override, and constraint control</td>
</tr>
<tr>
<td>6. Multivariable control</td>
</tr>
<tr>
<td>7. Digital control</td>
</tr>
<tr>
<td>a. Z-transforms, sampling, and stability</td>
</tr>
<tr>
<td>b. PID discrete controllers</td>
</tr>
<tr>
<td>c. Dead-time compensation</td>
</tr>
<tr>
<td>• Smith-prediction</td>
</tr>
<tr>
<td>• Dahlin's controller</td>
</tr>
<tr>
<td>d. Filtering</td>
</tr>
</tbody>
</table>
five semesters this project has been prepared and presented by Dow Chemical, U.S.A., Louisiana Division. Dow's personnel present the project to the students and then return two to three weeks later for the students' presentation. The students are asked to deliver an oral presentation and a written report.

Once the material on discrete and sequential logic is presented, the lab continues with a one-period discussion on control valves and subsequent periods on controller tuning. Through these exercises the students realize the effect of nonlinearities on the controller tunings and loop stability.

We have described how APC-I has been taught for the past five years. It is quite heavy in laboratory practice and includes topics such as PLC logic which are not usually covered in classical courses. We continually question whether such a departure from the norm provides a correct education. Often, we try new things, or modifications, such as a modification in the spring 1992 semester that added a bit more time to the subject of stability.

**Automatic Process Control II (APC-II)**

This is an elective one-semester course with a two-hour lecture and a three-hour lab per week. Usually twenty to thirty percent of the undergraduates take APC-II, and it also serves as the first graduate course. We use the APC-I textbook with additional notes on digital controls provided.

Table 5 shows the material presented in the lecture. Three modeling and simulation projects are usually also assigned. Table 6 shows the weekly assignment for laboratory practices. The equipment is the same as that used in the undergraduate course. Some exercises take a great deal of time because we show the benefits of the techniques in detail. For example, during the feedforward control, we first ask the student to control a process with simple feedback control. Then steady-state feedforward is implemented and its performance is compared to the performance of feedback control. We then add lead/lag and go through a similar comparison. Finally, dead time compensation is added to the feedforward and the results are compared. We ask the students to do the same with two different processes.

In the lab, one to one and one-half periods are used to explain about distributed control systems and stand-alone controllers. The students actually learn to configure one system. The student also learns in more detail about the available computing power of these controllers, helping them to design and implement different control strategies.

In the spring 1992 semester a design project was also assigned to each group of two students. Two instrument and control engineers from the local Badger Engineers office provided the design projects and acted as "leaders." Each group was asked to design the control strategies for a process. In addition, each individual student was asked to completely specify all the instrumentation for a control loop.

**CONCLUSION**

This sequence of required courses provides extensive "hands-on" laboratory work and lectures that are focused on the practical aspects of understanding the elements, functions, and properties of a control loop. Although process control is used as the unifying theme for the sequence, the students have also been able to relate the material to their reaction engineering course and their capstone project design course. In addition, the department has several active undergraduate research projects that provide an opportunity for the students to use their knowledge of sensors and computer interfacing in a more traditional university laboratory environment.

The course descriptions refer to the courses as they are taught today. As new sensors, computer hardware, and control software become available, the number of interfacing options increases and the implementation of more advanced control strategies becomes possible. It is our intent to make sure such developments are reflected in our course sequence.

**REFERENCES**

2. Honeywell Inc., Phoenix, AZ
3. Bailey Controls, Wickliffe, OH
4. Yokogawa Corp. of America, Newnan, GA
5. TUTSIM Products, Palo Alto, CA
8. Allen-Bradley, a Rockwell International Co., Milwaukee, WI 53204