

**2006-1003: THE PROGRAMMING OF A MICRO-CONTROLLER AS THE
LABORATORY COMPONENT IN PROCESS CONTROL FOR
UNDERGRADUATES IN CHEMICAL ENGINEERING**

Keith Lodge, University of Minnesota-Duluth

Associate Professor, Department of Chemical Engineering

The programming of a microcontroller as the laboratory component in process control for undergraduates in chemical engineering

Introduction

New funding generated by the College of Science and Engineering has financed enhancements of courses with computer technology. Here I describe briefly an enhancement, or perhaps a new direction is a more apt description, of Process Control (formerly ChE 4401, now ChE 4402); this is offered once a year in the spring semester. Typically, students take this course in their last semester; by this time, the novelty of life in the classroom is wearing a bit thin! For the most part, the course has roughly followed the first ten chapters of “Chemical Process Control”¹.

It is now easy to obtain inexpensive and reliable microcontrollers. In the spring of 2002, our students² first worked the Board of Education (BOE) from Parallax³; this is a convenient system with which to build circuits and to program the Basic Stamp, Parallax’s microcontroller. Of course, Parallax is not the sole source of microcontrollers; however, the depth and breadth of their educational material³ is particularly useful for those individuals who, like me, are new to the field. My research interests are not in process control and I took over the course after the departure of a specialist in the field. I have the single advantage of being an experimentalist. The preparation started in the summer of 2001; this exercise would not have been possible were it not for my single semester leave in the fall of 2001. During this time, I attended one of the regular two-day short courses organized for educators by Parallax⁴.

On the BOE, one assembles the desired circuit on a breadboard affixed to a printed circuit board; this contains the microcontroller, connections for power sources (wall transformer, or 9V battery) and input-output (I/O) connections to the microcontroller. The BOE provides a 5 VDC power source from which transducers may be run. For the purposes here, it is not necessary to design circuits; all the exercises were done with existing circuits with some minor modifications; these include adjustments to potentiometers and adding a switch. Our students meet the basics of electricity in freshman physics. Essentially, students have to be able to construct simple circuits from available diagrams; this requires the ability to locate pin 1, etc., on a chip, and recognize the polarity of an LED and an electrolytic capacitor. I encourage them to use a digital voltmeter (DVM) to check their interpretations of the color code on resistors! A separate switchable power strip was used to power the BOE through its wall transformer; it is necessary to switch off the power to the BOE before making any changes to the circuit.

Communication from the BOE to the outside world is made through a serial connection to a personal computer (PC). Programs are written in a special form of Basic, PBASIC; they are then compiled and downloaded to the microcontroller. Data can be transmitted back to the PC and viewed in debug window (*debug* command); they may be plotted and saved in a file using Stamp Plot Lite⁵, or captured directly into an Excel spreadsheet using an add-in program (StampDAQ

Real-Time Data Acquisition for Microsoft Excel) available from Parallax³. The latter mode of data collection is preferred, because our students are proficient in the use of Excel by this stage. The *debug* command is especially useful in developing programs. PBASIC contains special commands to handle data from an A-to-D converter (*shiftout*) and to run a servomotor (*pulsout*). Most of the work on the required of the students here is programming; our students will have taken a required programming course in their first two years (lower division). They may choose the programming language; however, they are encouraged to take Visual Basic. At the beginning of their third year they are required to take a course in computational methods (ChE 3031); here they are taught the use of Excel and MathCad. In the previous semester to Process Control, most will have taken a course in Separations (ChE 4111); here MathCad and its programming facilities are used extensively.

This exercise was started in our course Process Control (ChE 4401, a 3-credit course); this met for three 50-min lectures per week in a 15-week semester. The group of constituents and stakeholders, comprising students, faculty and members of our industrial advisory committee, (IAC) considered this exercise worthwhile and beneficial; this group encouraged us to incorporate it formally into our teaching of process control. A member of our IAC who works in an international controls company was especially enthusiastic. The course was modified so that it now has two 50-min lecture sessions and a 2-hour laboratory session per week (ChE 4402). A significant fraction of the course time was devoted to the work with the microcontroller; to fit this in some modifications of the original curriculum were necessary. In addition to developing a more economical presentation of the subject matter, the following steps were taken:

1. Bode stability analysis was omitted. Stability analysis is done using the poles of a transfer function (Chapter 4 of “Chemical Process Control”¹).
2. The treatment of second-order systems was reduced to a bare minimum; enough was taught to explain stability, instability and quarter amplitude damping (QAD). (The discussion of first-order systems remained unchanged, as did the discussion of the first-order-plus-dead-time, or FOPDT, model.)
3. Various examples of Laplace transforms were omitted. The basics of Laplace transforms were kept; the course follows Chapter 4 of “Chemical Process Control”¹ in this regard. (Students need to have a mastery of Laplace transforms for the Fundamentals-of-Engineering exam in order to become registered professional engineers in Minnesota.) Teaching Laplace transforms is made easier because of our students’ familiarity with MathCad; I am able
 - a. to assign exercises involving partial fraction expansions, inverse Laplace transformations, initial and final value theorems; the students can check their answers for themselves by using the symbolic operators in MathCad.
 - b. to demonstrate how properties, such as offset in proportional-only control (P control) and the removal of offset with the inclusion of integral control (PI) control, carry over to processes higher than first order. (With MathCad, I also demonstrate the solution of 3 simultaneous first-order differential equations as in the treatment of an actuator, process and sensor model applied to a CST thermal mixing tank –see example 3.1 of “Chemical Process Control”¹.)
4. Modeling of dynamical systems was restricted to lumped systems.
5. The number of demonstrations that were held in the classroom was reduced; demonstrations are held at the start of a lab session.

Laboratory sessions

The work in the laboratory sessions fell naturally in two parts. The first part was devoted to formal training in the programming of the microcontroller and the assembly of the required circuitry; the syllabus for this is shown in the Table (following the References). The aim of the first part was to give the students all the tools necessary to do the work on a liquid-level control loop that was basis of the second part; this was couched in terms of a project.

The circuitry, with which to do the project (the second part), was constructed in the first part; the first part takes requires 6 – 8 lab periods typically. In the project, the students had to replace the temperature sensor with the pressure sensor; adjustments of the span and offset (potentiometer adjustments) and the calibration code were then necessary. The essential tasks for the project are summarized in the Table. Most of the project comprised the writing of programs to demonstrate the various phenomena. Students came to realize that combinations of various phenomena could be demonstrated within a single program. A project report (20% of their final grade) was required; this comprised fully documented programs and graphs showing the various phenomena. These reports also are examined at the end of the year by the faculty as a whole; they are measurement tools for our assessment of students' outcomes associated with ABET criteria A, B, C and K.

Experimental

In addition to the BOEs, kits obtained from Parallax relating to the various student work books⁶⁻⁸ were sufficient to provide the required electronic components. The liquid level apparatus is shown in Figure 1 (following the References); it comprises:

1. A sump - a plastic pan (Consolidated Plastics 67003LL) serves this function well.
2. An aquarium pump (Aquarium System Mini-Jet MN 606) run off the mains; the pump is positioned in the sump.
3. A constant head reservoir constructed by cutting the bottom off a 12-oz plastic pop bottle.
4. The "tank" within which the level of water is to be controlled. This is a 60-mL disposable plastic syringe (Becton Dickinson 309663); its scale is useful and connections are conveniently made to it with Luer Lok fittings. (One of the problems with teaching experiments and demonstrations in process control is ensuring that response times are not too long; the students' attention span is always shorter than a response time! So apparatus should be reasonably small.)
5. The level sensor. This comprises a glass tube fixed into the body of the syringe with aluminum wire. The top of the glass tube is connected with Tygon tubing to a pressure sensor (Schaevitz GA100-005WD, available from DigiKey). The sensor is mounted on a PCB board within a small plastic box; it should be kept dry so tubing to the pressure sensor is best run so it arches well above the tank. It will measure pressures in the range 0-5 in. of water and provide a voltage output of 0.5-4.5 VDC. It requires a supply of 5 VDC and so is conveniently powered from the BOE. (Now we also using a pressure sensor from Dwyer Instruments, model number 646-5.)
6. The control valve. Rubber tubing (5/16 in. OD, 1/16 in. wall) is connected to the outlet of the syringe and it passes round a horizontally-mounted metal rod (diameter 0.5 in) to the

bottom of an overflow device. The position of the overflow serves to set the zero level of the liquid in the tank. A section of aluminum channel is mounted horizontally on a hinge mounted vertically. The channel's position is such that horizontal motion towards the cylindrical piece of wood will squeeze the rubber tubing against the wood, thereby reducing the flow from the tank. The end of the channel is connected by a rubber band to the rotor arm of the servomotor, or servo (Parallax Standard or Hobbico CS-61 from FerretTronics). The servo is driven by a series of high-low voltage pulses; the duration of the high pulse fixes the position of the rotor arm. The PBASIC command, *pulsout*, creates these pulses. Power to the servo is supplied by a battery pack with an on-off switch (Radio Shack 270-409, 4 AA batteries); for diagnostic purposes, it is convenient to be able to switch off the servo while a program is running.

Programming

Rapidly, it became apparent to students that programs should have a loop structure in which you measure the water level and then take some action there from. However, a difficulty arises; the servo requires a continuous series of pulses to maintain a given position; this is achieved by the repeated execution of the *pulsout* command. The microcontroller only executes a single command at a time. Therefore, to measure the level, the operation of the control valve has to be temporarily suspended. Students then came to realize that the measurement of the water level is not necessary for every cycle through the program; a counter could be established to measure the level on every 100th pass, say. In this way, the microcontroller could be programmed to spend most its time operating the valve. This experience here leads naturally to the idea of a control interval.

Fortunately, the Basic Stamp attracts an accomplished entourage of enthusiasts with the result there are good sources of code, ideas and devices to be found, especially on the web. The majority of its applications are in robotics, and the application here may represent a new departure. The problem described in the previous paragraph is of pedagogical value; however, it may be circumvented by use of a servo motor controller (Scott Edwards Electronics Min SSC II Serial Servo Controller) that provides a continuous series of pulses until a new instruction is received. The microcontroller is then free to do other things, and is not constrained to spend most of its time running the servo. This would give the students greater flexibility in writing programs. These servo controllers are available to those students who recognize the problem, and wish to circumvent it.

In order to program integral control in the fashion that is commonly described in chemical engineering texts, the time is required. Unfortunately, it is not possible to get a time from the Basic Stamp. Fortunately, a small clock board is available (Solutions Cubed Pocket-Watch B); this can be connected in easily to the BOE. Rather than supply an extra clock board for each BOE, two master clock systems have been set up to distribute the time through telephone wires around to each BOE in the laboratory.

PBASIC is easy to use and has good help system; examples of code can be cut and pasted in the active programming window, and tried out. The PBASIC software and Excel add-in program are free and can be downloaded from the Parallax website³.

Results and Discussion

Initial experiments were done to understand the behavior of the valve. In fluid mechanics, students will have learnt about flow through constrictions when dealing with flow meters. Figure 2 shows the measured relationship between flow rate and pressure drop with the valve fully open for a particular valve. Over most of the flow regime, the flow is laminar and so the relationship may be represented by

$$Q = C_v(x) \cdot \Delta P$$

Here Q is the volumetric flow rate, C_v is the valve coefficient, x is measure of the degree to which the valve is open (0 is fully closed, 1 is fully open) and ΔP is the pressure drop. This is to be contrasted with a “real” valve in which the flow is generally turbulent and so flow is described by $Q = C_v(x) \cdot \Delta P^{1/2}$. Other valves run by students did exhibit this more realistic behavior. A valve is characterized by the shape of the curve $f(x)$ vs. x , where $f(x) = C_v(x)/C_v(1)$. Data for the valve used here is shown in Figure 3, and this is typical of a quick-to-open valve⁹. Some students found approximately linear behavior. The reason as to why different valves constructed in nominally the same way exhibit different flow characteristics is presently unresolved.

Figure 4 shows the results of running a step test; the data were collected directly into an Excel spreadsheet (and in all the subsequent cases). The liquid level, as the digitized signal, is shown in the lower trace, represented by the circular markers. The x-axis represents the time in seconds and the y-axis represents the liquid level in units of one 10th of a millimeter. The use of the latter units is a convenience arising from the fact that arithmetic operations are integer only. Students had difficulties in appreciating that computations are done with integer arithmetic in small devices such as the Basic Stamp. A switch was used to move the valve from position to another; this is implemented by changing the signal sent, the pulse width, to the servomotor running the valve. By plotting the pulse width it is possible to set the time zero at which the step change was implemented, a necessary piece of information for fitting the data to a FOPDT model.

The response to the step test demonstrates the self-regulatory nature of the system; it moves from one steady state to another after a change. The clarity of the result shown here (almost textbook perfect!) belies the difficulties that were encountered by many students. Steady-state operation was difficult to achieve; this generated many complaints, which were met with the retort: “You now know a reason why process control is necessary!”. From the step test, the time constant (here about 16 sec), the dead time (here about 3 sec) and the steady state process gain can be obtained; these data can then used to obtain tuning parameters using the Cohen and Coon settings¹⁰ for instance.

On-off control is the first form of control that the students are encouraged to program; this has the simplest algorithm. Usually, students find that on-off control works well (data not shown); this leads them to inquire whether other forms of control are necessary. I ask them to consider how long a valve may last if it is required to move from a fully open to a fully closed position in continuous operation. Perhaps there are other methods of control in which the valve does not have to move so much. The execution of on-off control provides a natural entry into the PI method.

Figure 5 shows data from experiments with proportional control demonstrating offset after a change in set point. The offset is essentially eliminated by the inclusion of integral control as is shown by the data in Figure 6. Regulatory control (disturbance rejection) is important for chemical engineering; a disturbance may be mimicked here by dumping a small volume of water into the tank. An example of disturbance rejection is shown in Figure 7.

Concluding Remarks

This modification of the process control course reached its fourth year in the spring of 2005, when there were three lab sections with about 10 students in each section. For the last two years and for the spring of 2005 a junior or senior in Electrical and Computer Engineering has been employed as a teaching assistant; the student had taken a course in control systems. This is a great benefit for setting up each lab session, for assisting members of the class, and for development. Originally, students worked in groups of two. There are always various sources of pressure to increase the size of groups in laboratory work in general. However, this is a case in which it is feasible for students work alone, and this they now do. Upon hearing that they were expected to work alone, two students responded separately with the comment to the effect: "Well ...that means I will have to do some thinking to get through this!"

The students' rate of progress was directly related to their programming skills. The important intellectual discipline that programming brings is the ability to imagine how events, or processes, should occur with time; this is fundamental to the understanding of process control. In addition to developing their programming abilities, students learn about other facets of computing, viz. bits, bytes, words, integer arithmetic, serial communication, synchronous and asynchronous communication, and A/D conversion. In addition, they get some practical experience with fluids; many opportunities arise to remind them of material covered in Fluid Mechanics (ChE 3111), our gateway course into the final two years of the program (upper division). The liquid-level problem, and its linearization, is covered in the formal class periods and in homework in this process control course. However, an important side benefit for students is practical experience in fault detection. Most faults were attributable to incorrect wiring; the natural instinct was to conclude that a component had failed. The failure of components is rare; usually it is a potentiometer. The Basic Stamp itself is very robust. Incorrect connection of the sensors to the BOE can lead to failure. This can be avoided by using polarized connectors between the sensor and the BOE, so once the temperature sensor circuit is working the pressure sensor can easily replace it without redoing the connections to the BOE.

The costs involved are reasonable amounting to about \$300-350 per system (the BOE, components and liquid-level apparatus) at this stage. The costs of items that may fail or get destroyed inadvertently are small; for example, the servo costs about \$15. This is a cost effective way of exposing students to the practical aspects of process control, when you consider that an air-controlled actuator costs about \$1000. Another cost-effective practical approach is described by Moor and coworkers^{11, 12}; Lego®RCX kits with ROBOLAB™ software for LabVIEW™ are used to give students opportunities to construct various control loops that are relevant to chemical engineering practice.

A particular benefit was the direct interaction with students. Soon it was possible to tell what stage they had reached from the questions they asked. Even, the shy students recognized the need to ask questions if they were to make progress. The weaker students were able to feel a sense of accomplishment with their project work, although they did not get as far as the more able ones. It is pleasing to watch students with various abilities assist each other.

Acknowledgements

I am grateful to UMD and the Department of Chemical Engineering for a single semester leave in the fall of 2001 during which time this work was started. The short course, organized by Professor W. Schlick of Schoolcraft College, Livonia, Michigan and taught by A. Lindsay of Parallax in November 2001, was most useful. For unrelenting technical assistance, I am indebted to D. Long. The work of C. Patullo, J. Krause and E. Gieske (undergraduate assistants) was essential in running these exercises.

References

1. Riggs, J.R., (2001), "Chemical Process Control", 2nd edition, Ferret Publishing, Lubbock, Texas.
2. Lodge, K.B., (2002) "The Programming of a Microcontroller as an Integral Part of Process Control for Undergraduate Chemical Engineers", *2002 ASEE North Midwest Section Annual Conference*, Madison, Wisconsin, October 10-12, 2002.
3. Parallax, Inc., (2005) <<http://www.parallax.com/>> [accessed December 29, 2005]
4. Parallax, Inc., (2005) <http://www.parallax.com/html_pages/edu/index.asp> [accessed December 29, 2005]
5. Selmaware, Inc., (2005) <<http://www.selmaware.com/>> [accessed December 29, 2005]
6. Parallax, Inc., (2005) What's a MICROCONTROLLER?, Student Workbook
<http://www.parallax.com/html_pages/edu/index.asp> [accessed December 29, 2005]
7. Parallax, Inc., (2005) Basic ANALOG and DIGITAL, Student Workbook
<http://www.parallax.com/html_pages/edu/index.asp> [accessed December 29, 2005]
8. Parallax, Inc., (2005) INDUSTRIAL Control, Student Workbook
<http://www.parallax.com/html_pages/edu/index.asp> [accessed December 29, 2005]
9. page 49 of reference 1
10. page 277 of reference 1
11. Moor, S.S., and Piergiovanni, P.R., (2004) "Learning Process Control with Legos", *Proceedings of the 2004 American Society for Engineering Education Annual Conference and Exposition*, Session 1526, Salt Lake City, UT.
12. Moor, S.S., Piergiovanni, P.R., and Metxger, M., (2005) "Process Control Kits: A Hardware and Software Resource" *35th ASEE/IEEE Frontiers in Education Conference*, Session T2G, October 19-22, 2005, Indianapolis, Indiana.

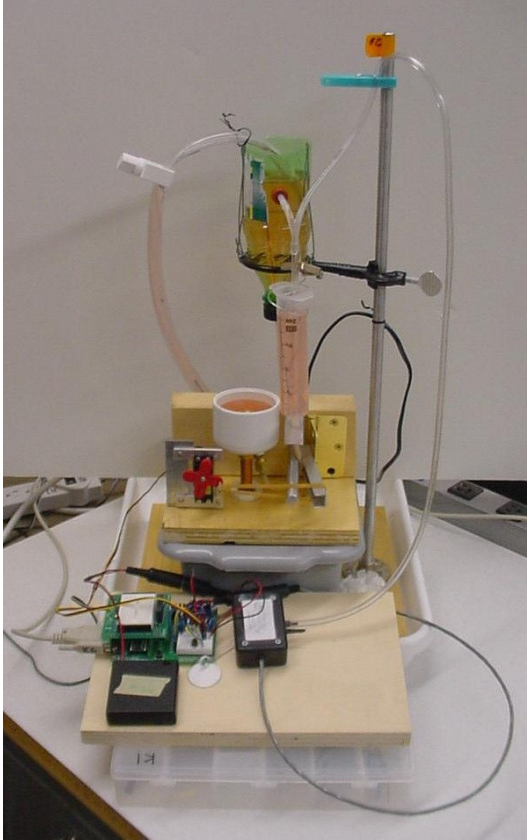
Table. Syllabus for the programming of a microcontroller in Process Control

<u>Formal training</u>		<u>Project</u>
Subject matter covered in laboratory sessions relating to the programming of the Basic Stamp		With the circuit constructed so far ** it should be possible to run experiments with the liquid-level system to demonstrate:
Topics	Source	
1. Input and output, Use of a switch for single input	Experiments 1 & 2 in “What’s a Microcontroller”*	1. the calibration of the sensor
2. Use of switches to input binary numbers. Serial and synchronous data	Experiment 2 in “Basic Analog and Digital”*	2. self-regulation
3. Basic analog to digital conversion (8-bit converter)	Experiment 3 in “Basic Analog and Digital”*	3. a step test to determine the parameters in a FOPDT model of the system
4. Example of analog-to-digital conversion with the LM53 temperature probe	Part of Experiment 4 in “Industrial Control”*	4. on-off control
5. Plotting and capture of analog and digital data from the Basic Stamp	Excel add-in program (StampDAQ Real-Time Data Acquisition for Microsoft Excel)	5. P-control
6. Driving a servo motor with the Basic Stamp	Experiment 3 in “What’s a Microcontroller”*	6. offset
		7. PI control
		8. signal aliasing (control interval selection)
		9. disturbance rejection with various modes of control
		10. tuning performance (e.g., QAD, if appropriate).
*Student workbooks published by Parallax ⁶⁻⁸ .		**The circuit after topic 4 in the formal training.

Figure 1.

The apparatus

Red dye has been added to the water.



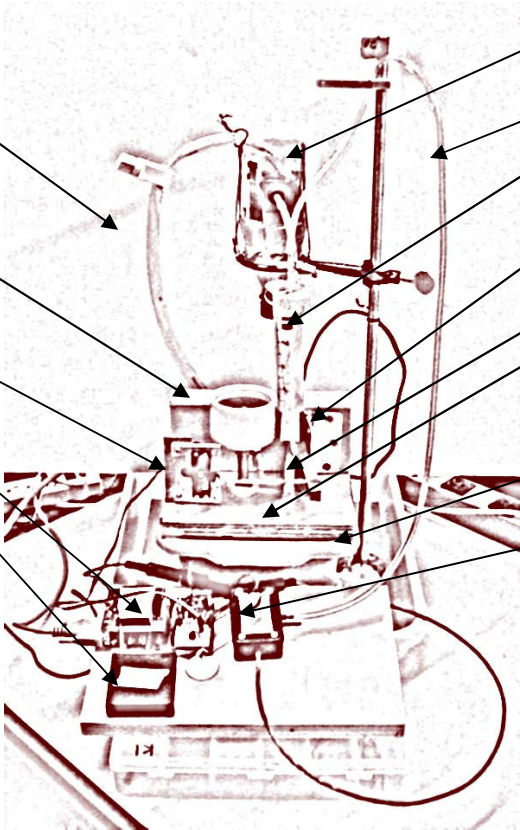
Water line from pump

Overflow device

Servo motor

Board of Education

Battery pack (servo power)



Constant head reservoir

Tubing to pressure transducer

“Tank” (60-mL syringe)

Hinge

Aluminum channel

End of metal rod

Rubber band

Sump

Pressure Transducer

Figure 2. Flow data obtained with a fully opened valve

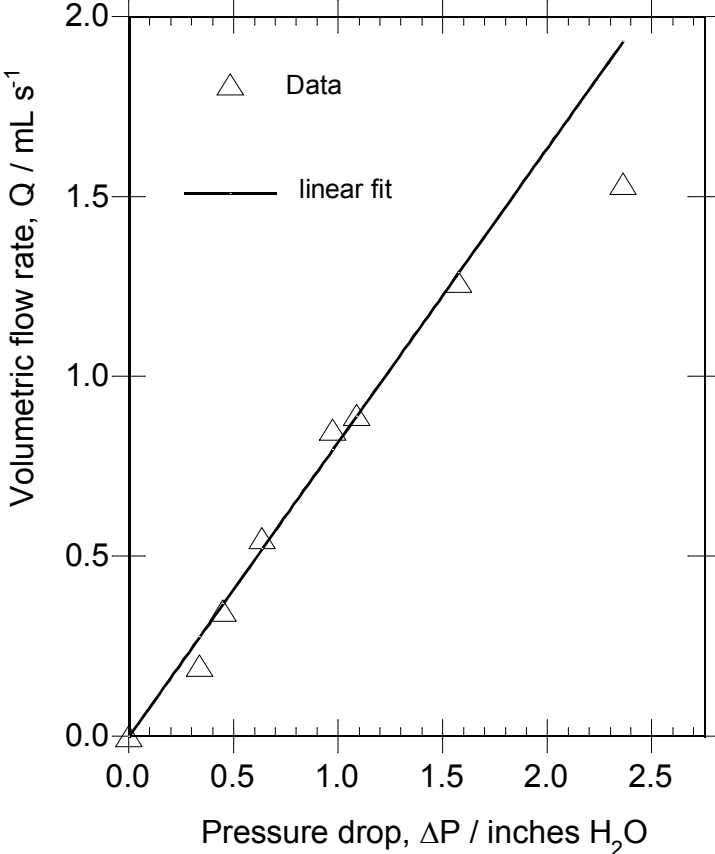


Figure 3. Inherent characteristics of the valve

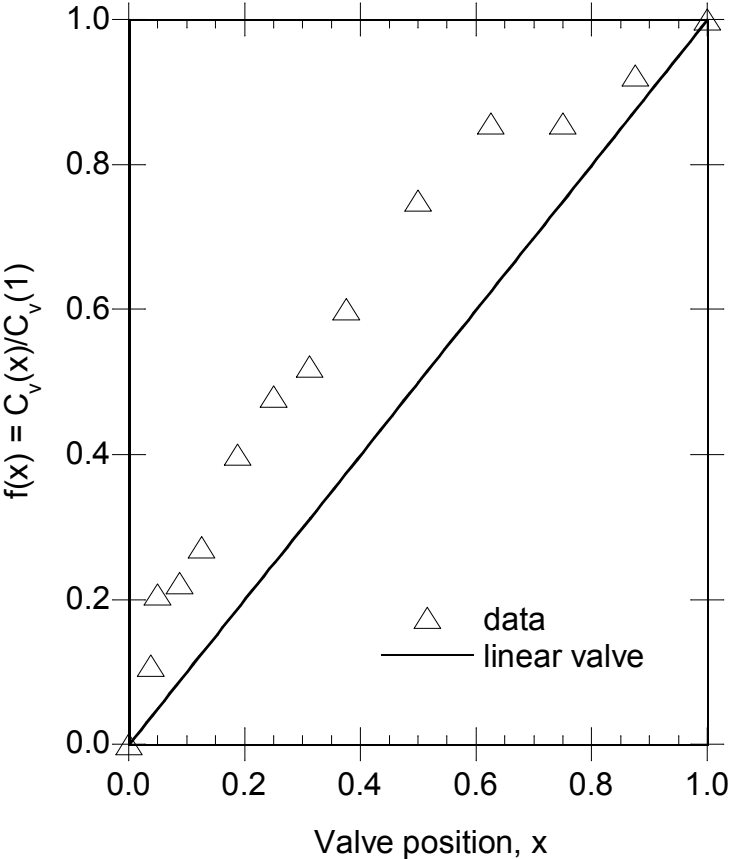


Figure 4. The results of step test with which the FOPDT model of the system is determined.

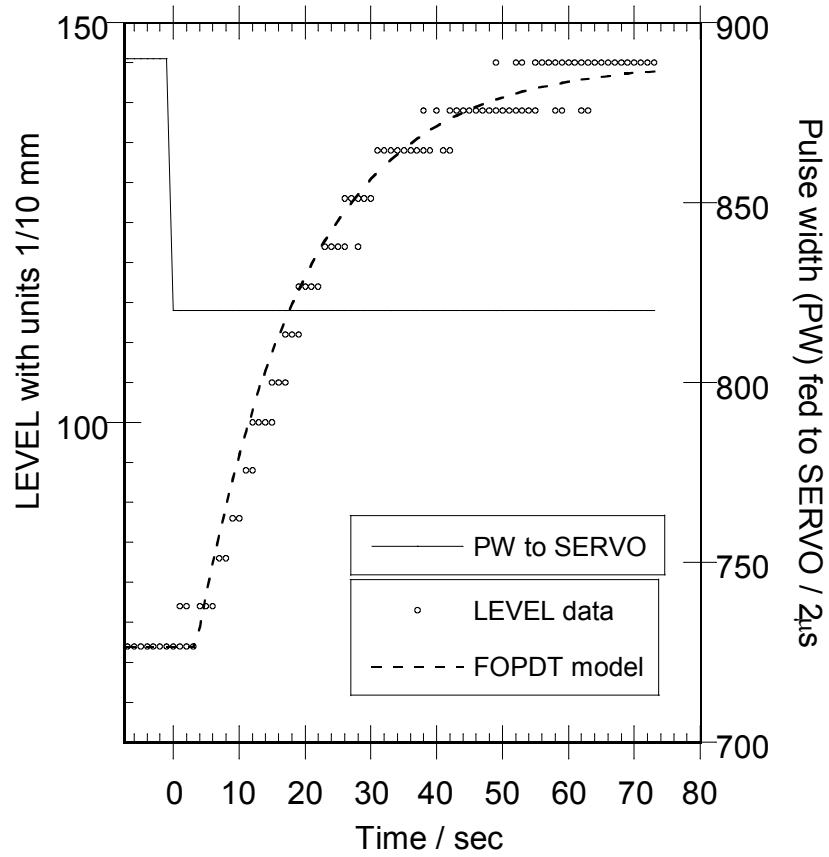


Figure 5. Proportional control showing offset

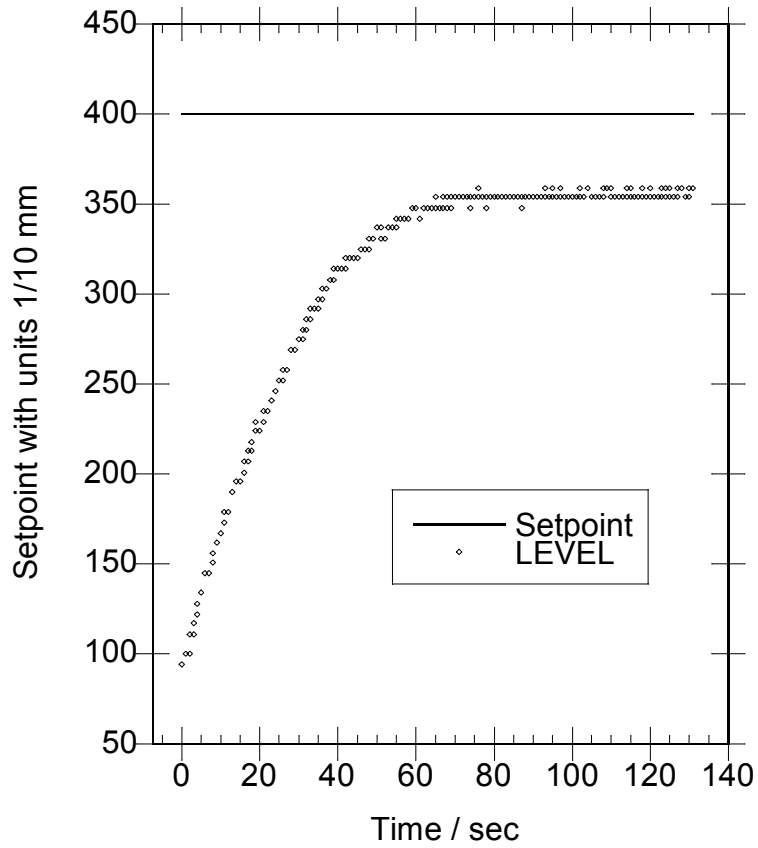


Figure 6. Proportional and integral control (PI) showing elimination of offset

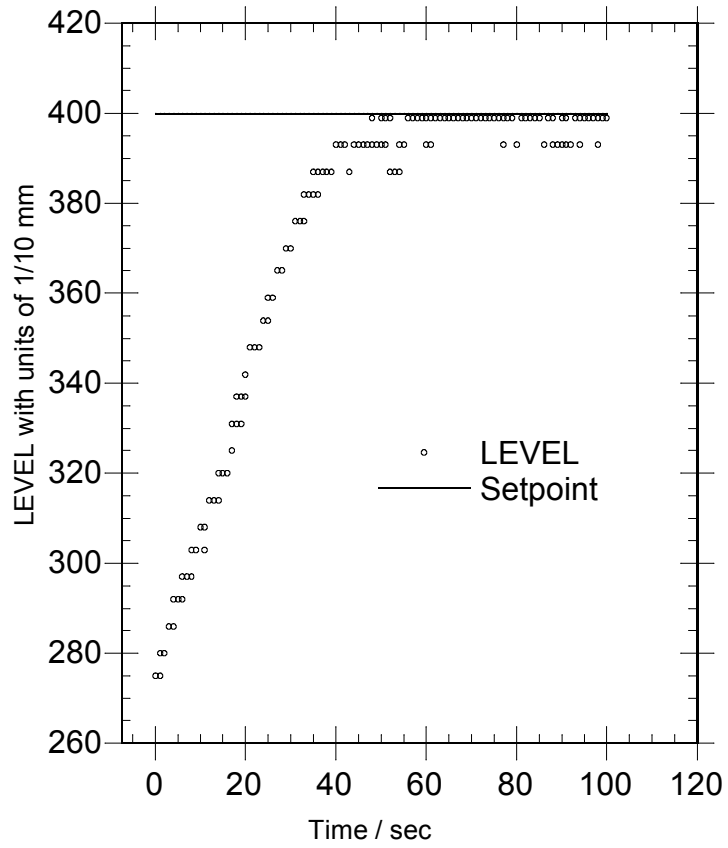


Figure 7. Disturbance rejection under PI Control
Thirty milliliters of water were quickly dumped into the tank at the time of 25 sec.

