ENHANCING BIOREACTOR FUNCTIONALITY IN BIO-PROCESSING COURSES

Ali Givmanesh, Rupa Iyer
Engineering Technology Department
University of Houston

Abstract

This paper explores the technology behind the New Brunswick Scientific BioFlo 110, a modular fermentation device for microbial and cell cultures currently in use in biotechnology and bio-processing laboratories. The primary focus of this research is to perform a theoretical study of the bioreactor's primary control unit, and use the findings to develop intuitive and flexible software with data logging capabilities. The program features presented in this research allow lab instructors and students to enhance the bioreactor functionalities as an educational tool, reduce the time spent in the laboratory, and add more flexibility to the overall system.

Introduction

Biochemical engineering researchers have utilized bioreactors as the primary fermentation devices to grow and analyze various cells and tissues in the context of cell culture experiments. The University of Houston biotechnology program currently uses the New Brunswick Scientific BioFlo 110 in two of its laboratory courses to grow transformed bacterial cells; BTEC 4301: Principles of Bio-processing Laboratory and BTEC 4350: Capstone Experience in Biotechnology. The process of developing new cell products requires a highly controlled environment, which is provided by the Primary Control Unit (PCU) attached to the bioreactor vessel. Though the equipment within the University of Houston laboratory is fairly new, the technology available to improve their capabilities already supersedes them.

This paper explores the technology controlling the BioFlo 110 and proposes a technique to enhance the functionality of the bioreactors as an educational tool by introducing an onsite data logging method. Currently, students and instructors rely on live-feed information presented by the bioreactors unit itself. Unfortunately, though, the unit does not provide a method for the archiving of data related to the experiment, and is limited to a one-time viewing of this information. In situations where experiments are running overnight, any unexpected changes will remain undetected and unrecorded. The main goal of this study is to develop a system to log the experimental data onsite at variable time intervals in order to improve the quality and effectiveness of experiments and data analysis. The following sections explain the bioreactor device and introduce the new application developed in this research. At the end of this paper, several tests are setup to evaluate the logic and functionality of the proposed program.

Bioreactor Overview

The University of Houston Biotechnology laboratory consists of three complete sets of bioreactors used as modular fermentation devices to grow microbial and cell cultures within a laboratory setting. A typical configuration of the BioFlo 110 module currently in use is depicted in Figure 1. The Primary Control Unit (PCU) acts as the main user interface that controls various parameters in each bioreactor vessel, and its functionality is similar to the thermostat used to control the air conditioning units found in many homes. PCU's main purpose is to set the operating point for each available parameter and perform the functions necessary to maintain the current value close to the set operating point. Another critical module in the overall system is the power controller which functions to regulate temperature and agitation parameters of the

vessel. Additional parameters and sensors include gas mix, level control, dO_2, pH, and several additional pumps. The modularity of the bioreactor as a whole allows the instructor to add additional functionality to the system which can be used to perform more complex experiments. On the other hand, the lack of data logging scheme prevents the researchers to archive the process and validate the end products findings.
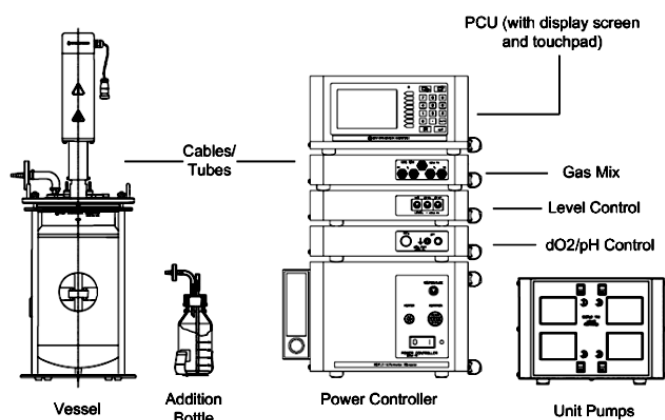

Figure 1: Typical Bioreactor Setup. [1]

Data Logging Application Overview

System Architecture

This section expands upon the idea and methodology used to solve the issue of data logging within the laboratory. Figure 2 represents the overall architecture of the proposed system. The onsite data-logging application is regulated by the desktop. The users can utilize this application to set the necessary settings in order to store the bioreactor parameters. They can also use the PCU module located on the bioreactor unit to set and adjust their experiment parameters. The students and lab instructor can view the archived data in two main environments:

- Text file, which presents the clients with unformatted data read from the bioreactor.

- Excel sheet, which allows the clients to view the formatted and well-organized data.

The remainder of this section examines the communication between the desktop and the bioreactor and discusses the new user interface in the data logging application.
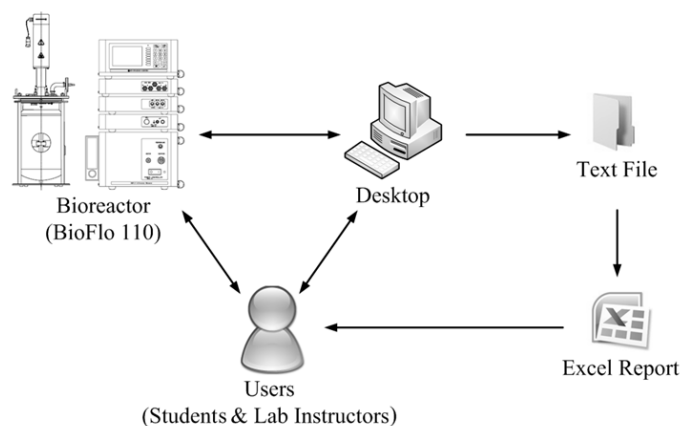

Figure 2: Overall System Architecture.

Desktop and Bioreactor Interface

The only hardware addition to the pre-existing bioreactor configuration is a desktop computer. It connects to the bioreactor via a RS-232 serial cable through conversion of the manufacturer's proprietary AFS/ModBus 25-pin connector. Communication between the PCU and desktop is examined by a serial port-monitoring program. The open source port monitoring program, 'Hercules 3.2.3', was chosen for this research because it provides an easy to use platform to test the different bioreactors commands separately for proper functionality. This program was used to ensure the command messages sent to the bioreactor are in the right format, and the messages received by the desktop are complete.

Table 1 displays the commands that are used on the serial port in order to communicate with bioreactors. The request format consists of four

hexadecimal values. The first value represents the unique ID number pre-assigned to each bioreactor. The following two values define the request commands. The last value contains a carriage return indicating the end of each request. The ID numbers for each bioreactor are pre-assigned to 00, 04, and 88 by the manufacturer. The responses from the six commands represent the bioreactor's information displayed on the unit itself. These information and parameter are viewed and set by the users through the PCU display screen and touchpad.

Table 1: BioFlo Serial Commands

Request Command	Request Format
Get Loop Names	(ID#) (R) (A) (CR)
Get Current Value	(ID#) (R) (C) (CR)
Get Set Point	(ID#) (R) (D) (CR)
Get Output %	(ID#) (R) (E) (CR)
Get Control Mode	(ID#) (R) (J) (CR)
Get Unit	(ID#) (R) (L) (CR)

User and Desktop Interface

Because the users interact with the desktop to provide the new application settings, an easy to use GUI using C# programming language has been developed to make this process possible. The C# language offers the power behind C++ programming schemes with the ease of the Visual Basic user interface [2]. The new application allows the students and lab instructors to capture and log the bioreactor parameters at various time intervals. This information is then stored onsite and can be presented in an Excel document for further data analysis.

New Data Logging Program

The application developed throughout this research is the most valuable deliverable of this project. This section will introduce and describe the data logging application in more detail.

One of the primary advantages of this program is its simplicity. The new application consists of only three forms:

1. Login Form,
2. Data Logging Form
3. Excel Report Form.

The login form interface is shown in Figure 3 part a, which prompts the user to input their username and password in order to continue to enter the main form. This method of authentication allows the software to distinguish between lab instructors and students. Currently, the lab instructors are allowed to make primary changes to the program, whereas students are limited to only view their experimental results. If the provided password and username are correct the program loads the data logging form shown in Figure 3 part b. This form requires the instructor to input the name of the experiment and total time of the bioreactor run. The user will also have to set the time interval that dictates the delay between each data capture. Because the bioreactors are connected, all three will be affected by the changes made in this data logging form.

The GUI is designed to interact with the users through the "Start" and "Excel Report" buttons. The start button initiates the data logging process, while the Excel report button initiates a new form. Figure 3 part c shows the Excel report form which uses the experiment name as input to produce the full report in three separate Excel documents. Instructors can quickly go through the following steps to start logging and monitor bioreactor data immediately.

1.   Start the program
2.   Enter the name of the experiment

3. Set total duration (min): enter the total time of the experiment
4. Set time intervals (min): enter time interval for data captures to be recorded
5. Click the start button to start the data logging process
6. Progress Bar: shows the progress of the overall report based on the total duration set by the user
7. When progress bar is completed use the Excel button to go to the next form
8. Enter the name of the experiment to view the Excel report
9. Additionally use the "File" tab to restart, save, or close the program
10. Use the "Help" tab to view the instructions and program acknowledgment

Excel Report

Once the bioreactor run is completed, the user has the option of reviewing the collected data in a Microsoft Excel sheet. Microsoft Excel was chosen to view the stored data because of its sorting and graphing capabilities. Table 2 is an example of the data entry created by the new application. As indicated in the top row, the data was received from BioFlo A at 3:30 AM on November 17, 2010. The next entries are displayed sequentially based on the time interval defined by the user. For example if the user chooses 30 min intervals, the next data capture will be collected at 4:00 AM. This format was chosen because it represents the same format on the PCU's display screen. In order to perform further data analysis, various functions within Excel can produce different charts and graphs to illustrate the changes of each parameter over time.
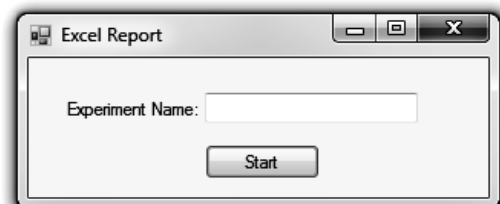

(a)
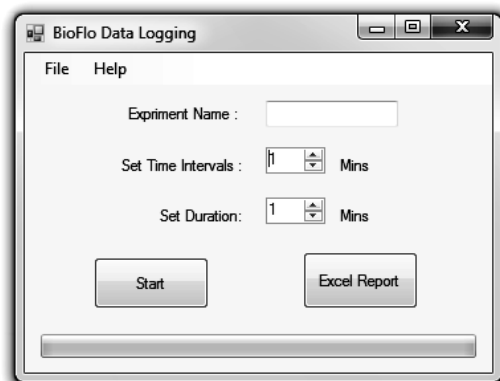
(b)

(c)

Figure 3: New Application Forms.

Table 2: Excel Data Entry Example.

BioFlo #	A					Date / Time			11/17/2010 3:30
	Temp	Agit	Pump A	Pump B	Pump C	pH	dO2	Gas	O2
Value	30.00	600.72	0.00	0.00	0.00	7.39	64.63	0.27	0.00
Unit	DegC	rpm	%	%	%	pH	%	SLMP	%
OP Points	30.00	600.00	100.00	20.00	50.00	7.00	100.00	5.00	0.00
Output%	0.52	49.96	0.00	0.00	0.00	-25.00	100.00	100.00	0.00
Control	Auto	dO2	Off	Off	Off	Auto	Auto	dO2	Auto

C# Programming

A simple and reliable algorithm is deployed in order to provide the necessary functions to communicate with the bioreactors, send and receive data on the serial port, and store the captured data in the text files. Developing a well-planned algorithm in the designing phase of the program is crucial to guarantee reliability of the application's logic. The algorithm and steps required to write the program in C# are listed below.

Form 1: Login

1. Start the Program
2. Wait for username and password
3. Check the username and password
   a. If correct, go to "Form 2"
   b. If incorrect, display the proper error message and go back to step 2

Form 2: Data Logging

1. Wait for three main inputs: experiment name, total duration, and time interval
2. Is total duration > time interval?
   a. If no, display proper error message and go back to step 1
   b. If yes, calculate number of reports = duration / interval
3. Create a folder with the same name as experiment name
4. Initialize serial port settings
5. Enter the main loop
   a. Write the proper commands to the bioreactors
   b. Read the responses from the serial port buffer
   c. Write the data to a text file stored in the created directory
   d. Time stamp the text file
   e. Number of reports = number of reports – 1
   f. Is number of reports = 0?
      i. If yes, exit main loop
      ii. If no, go back and repeat step 5.a to 5.f
6. Wait for "Excel Report" button
   a. If clicked, go to "Form 3"

Form 3: Excel Report

1. Wait for experiment name
2. Create three sub directories for each bioreactor
3. Separate and store the data for each bioreactor
4. Initialize the Excel sheets
5. Create three Excel documents for each bioreactor
6. End the program

The remainder of this section highlights several pieces of code executed while running the bioreactor application. The presented codes mainly focus on sending, receiving, and storing the appropriate parameters. One of the main functions required in this program is the interface between the desktop and the bioreactor. There are several ways to achieve code-level communication between the desktop and the PCU. This application uses a controller called SerialPort, which was originally launched by Microsoft Visual Studio 2005, and is based on the method of P/Invoke call [3]. Upon ensuring that the communication port is not in use by any other application on the desktop, the program applies the following code in order to initialize the serial port settings.

```
serialPort1.PortName = "COM1";
serialPort1.BaudRate = 9600;
serialPort1.Parity = Parity.None;
serialPort1.ReadTimeout = 10;
serialPort1.StopBits = StopBits.One;
serialPort1.Open();
```

These standard RS-232 settings are defined in the Biolo 110 manual. After initializing the

serial port settings, main request commands are formatted to be sent to the three bioreactors.

The serial port function initialized earlier is used to communicate the four bytes of information included in each command. The example shown below sends the RC command to ID# 88 and returns all the current values of parameters from the third bioreactor to the serial port buffer. Similarly other messages in Table 1 are communicated through the serial port.

```
byte[ ] command = new byte[ ]
{ 0x88, 0x52, 0x45, 0x0D };
serialPort1.Write(c, 0, command.Length);
```

Another function included in the serial port library named, "Data Received Event Handler", will read all the existing bytes arriving at the serial port. Upon reading the response from each bioreactor, the information is saved using a string variable called "data". The following piece of code shows how this function is setup within the program.

```
mySerialPort.DataReceived += new
SerialDataReceivedEventHandler
(DataReceviedHandler);
string data = mySerialPort.ReadExisting();
```

Each time the data is received, it is appended to a simple text file with its time stamp saved on a separate file used later for Excel sheet formatting shown in Table 2.

The next section explains the project initial test setup and the result of the system performance evaluation. Because this application was implemented within the biotechnology courses during the spring of 2011, the student laboratory results are also presented.

Implementation and Performance Evaluation

This chapter is dedicated to test and evaluate the performance of various components utilized within the new bioreactor application. The first test performed is used to assess the data logging functionality by capturing the bioreactor's information using different time intervals. The second test evaluates the same program by implementing it within the biotechnology courses during spring 2011 semester.

Initial Beta Test

Because the design and implementation phase are normally followed by the performance evaluation phase, the initial test is setup to test the new application and evaluate reliability and functionality of the C# codes. Due to the modularity of the architecture, the application can be tested using various input values to ensure the consistency and accuracy of collected data.

Escherichia coli is used as the main bacteria in this test. It is grown in the BioFlo 110 growth chamber by controlling two independent parameters of temperature and pH as well as three dependent parameters of agitation, gas, and dO_2. The pH is a measure of the acidity/basicity of a solution, and will remain in the range of neutral to slightly basic during the experiment. Agitation is the mixing of the media by a motor controlled impeller to increase aeration. Gas is the measure of atmospheric air being added to the chamber. The amount of dissolved oxygen present in the liquid growth media is known as dO_2. The PCU parameters of operating point and control option were set using the values presented in the Table 3.

Table 3: Beta Test Parameters

Parameter	Setpoint	Control	Unit
Temperature	30	Auto	^oC
pH	7.5	Auto	pH
Agitation	600	dO_2	Rpm
Gas	5.0	dO_2	SLPM
dO_2	100	Auto	%

The operating point for temperature is set to 30 degree Celsius, and pH is set to 7.5. Both control options are set to auto, allowing the PCU to control and maintain the current value at set operating points. A cascade system is setup to make sure the dO_2 level is stable at a set point of 100 percent. Because gas level and agitation control options are set to dO_2, their current values will increase or decrease accordingly in order to keep dO_2 as close as possible to its operating point. The range of cascade parameters cannot be unlimited; therefore the maximum and minimum values of both gas and agitation are defined by the lab instructor using the PCU interface.

- Agitation can vary between 200 – 600 rmp
- Gas level can vary between 0 – 5.0 SLPM

The experiment is set to run approximately for 24 hours. During this time, the total duration and time interval inputs are varied to ensure the program is functioning properly. All the parameters were collected over three main time intervals, as presented in Table 4, on November 16th and 17th, 2010.

Table 4: Beta Test Time Intervals

Name	Time Period	Time Intervals
T 1	09:45 – 10:35	Every 2 min
T 2	11:00 – 16:20	Every 5 min
T 3	16:30 – 07:00	Every 30 min

After the three time durations expired, the collected data in text files were transferred into Excel sheets. There was no error in any of the data that were stored and each capture was performed on time based on the defined initial inputs. The filter and graph options were combined to produce graphical charts for each parameter in order to show the consistency of the program's performance. As expected, the temperature and pH values stayed stable at 30 and 7.5. Temperature values are shown in Figure 4, while pH values are presented in Figure 5.

In this experiment, agitation and gas value regulate the dO_2 level. Therefore, when the actual dO_2 value passes the set point, the agitation speed will automatically decrease until the dO_2 set point is reached. Conversely, when the actual dO_2 value drops below the set point, the cascade system response is to increase the value [1]. Figure 6 combines all three parameters results in logarithmic based graph. The changes in the dO_2 value over the course of this experiment are marked, and since agitation reached its maximum level of 600 rpm, the gas level was adjusted respectively to maintain the dO_2 level. Therefore adjusted gas values are directly proportional to dO_2 modification and values.

The main purpose of this test was to evaluate the serial communication between the PCU and the desktop. In actual bioreactor experiments data is typically captured every 15-30 minutes, but in this test the first interval was set to 2 minutes to check for a possible bottleneck value. The results confirm that the program can produce highly accurate results at different time intervals set by the user. The following section explains the implementation of all three bioreactors into the ongoing biotechnology courses.

Implementation Test

Bio-processing involves the use of organic substances and their subsequent utilization in various arenas such as commercial, medical, or scientific research. Different fields of study such as food processing, thermal design of building, biomedical devices, pollution control, and global warming all require knowledge of how energy and mass can be transported through materials.

The experiments setup in the biotechnology courses during the spring of 2011 semester include the bio-processing of *OPH (Organo-phosphorus Hydrolase) enzyme*, and the discovery of the optimal growth conditions of an unknown strain of *Escherichia coli*. The optimal growth is affected by all 9 parameters of the bioreactor.
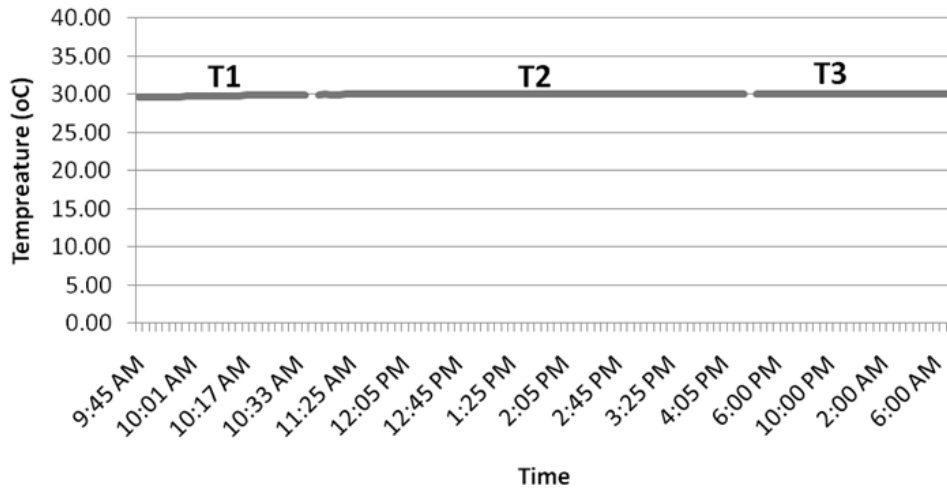
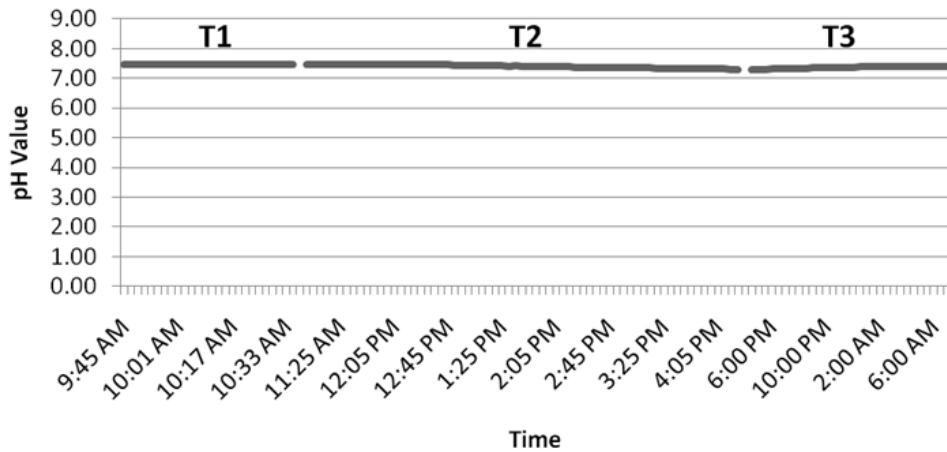Figure 4: Temperature (°C) Data Representation.

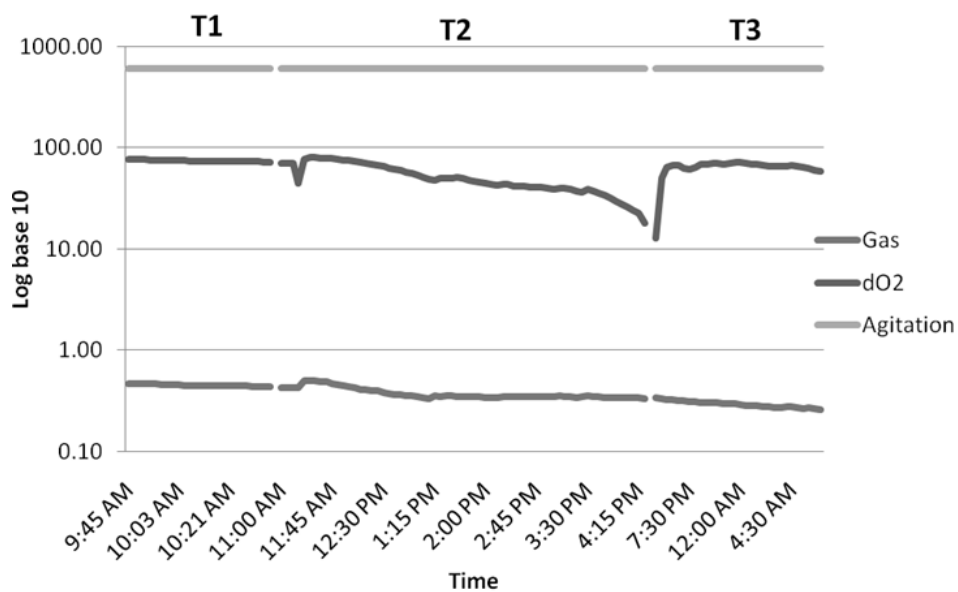Figure 5: pH Value Data Representation.

Figure 6: Cascade System Result (log 10).

The students in the course are divided into three groups of five members, and are given the task to understand the effects of three selected parameters on the end product. Each group performs three consecutive runs while changing their assigned parameters and keeping other parameters at a standard level. Below is the class setup and assigned variables for each group.

- Group A modifies the pH value and examines its effects on the end result.
- Group B manipulates the agitation speed to examine its role on end product.
- Group C examines the effect of pump speed on the bacteria's growth.

Each group will start their first bioreactor run with assigned values of their parameters set by the instructor. This value can be used as a baseline for future comparison. During the second and third runs, each group will modify the values of their parameters in order to understand and analyze the effect of their decision on the cell culture growth. Three operating points chosen by each group are listed in Table 5. By the end of the semester students provide their reasoning behind choosing the second and third values and use their final product to prove or disprove their theory.

Table 5: Bioreactors Experiments.

	Group A	Group B	Group C
Parameter Under Test	pH	Agitation	Pump Rate
Ex 1 (02-14-11)	6.5	550 rpm	1%
Ex 2 (02-21-11)	7.5	400 rpm	0.5%
Ex 3 (02-28-11)	6.0	250 rpm	5%

The implementation test is conducted every Monday for three consecutive weeks. Each bioreactor run takes about 24 hours to be completed. The students utilize the rest of each week to process the cellular biomass generated by their experiment. The main activities include harvesting, spray or freeze drying of the harvested biomass, or purification steps for the recovery and enzymatic characterization of a protein product [4]. After processing all three samples, each group compares their own protein products side by side to determine which selected value produced the most desirable outcome. The operating point of each parameter that optimized the growth of *Escherichia coli* and their corresponding ID numbers are shown in Table 6. The experiment ID number indicates the group letter and the week the experiment was performed on. For example B1 indicated the first run of group B.

Table 6: Optimal Growth Results

	Group A	Group B	Group C
Parameter Under Test	pH	Agitation	Pump Rate
Optimal Growth Value	7.5	550 rpm	1%
Experiment ID	A2	B1	C1

In order to validate the findings shown in table 6, the students have to make sure the set parameters stayed within an acceptable range during each run. The data logging program allows students to compare and view their specific variables side by side for all three experiments. The results collected from three experiments for each group are presented in Figures 7, 8, and 9. As shown in the following figures, all experiments can be successfully validated because each variable was maintained at the operating points defined in Table 5. At this point, the student can also verify that their end results and optimal growth findings are directly proportional to the changes made to their variable. These same results can also be shared with other students and organizations in

order to improve the yield and quality of a particular cellular biomass being cultured.

Conclusion

Bio-processing and growing cell cultures require both sensitive and accurate instruments, as well as tightly controlled environments. An essential component of achieving both viable and reproducible results requires the tracking and archiving of all the parameters and procedures necessary to complete the experiment in order to duplicate the results accurately and increase the yield of cell product. The program introduced in this paper allows students and instructors to log and store necessary data from the BioFlo 110, offering a better understanding of the fermentation process, increasing productivity, efficacy, and allowing for more efficient troubleshooting. This software also enhances the New Brunswick Scientific BioFlo 110 as an educational tool by providing the graphs and tables needed to perform detailed data analysis. The flexibility introduced by the system will increase productivity of lab managers, and the design architecture of the data logging system, using readily available communication technologies, greatly improves the data analysis and study of bioreactor experiments.
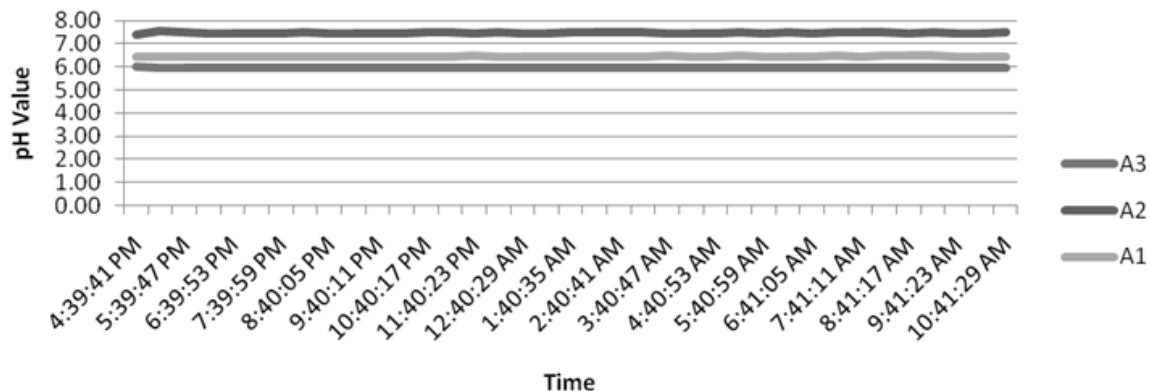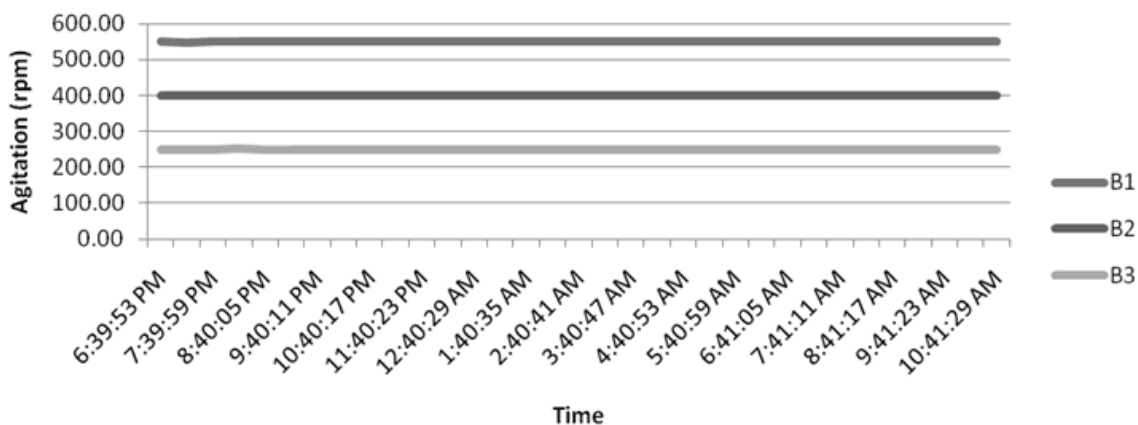

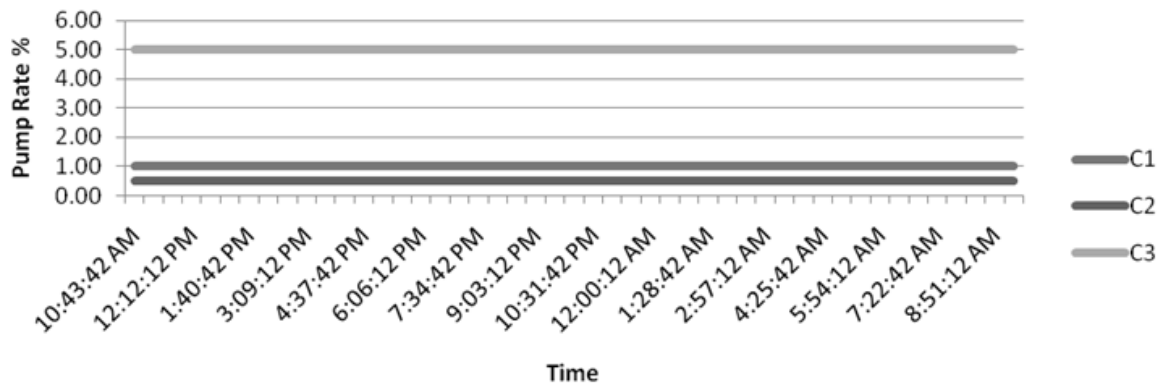Figure 7: Group A pH Results.

Figure 8: Group B Agitation Results.

Figure 9: Group C Pump Rate Results.

References

1. "BioFlo 110 Modular Benchtop Fermentor: Guide to Operations," New Jersey: New Brunswick Scientific Co., Inc., 2007, pp. 1,139-141.

2. M. H. Lutz and P. A. Laplante, "C# and the .NET framework: ready for real time?," *Software, IEEE,* vol. 20, pp. 74-80, 2003.

3. Z. Yue-Qin and G. Wen, "Design and Implementation for Communicating between Computer Serial Port and Mobile Phone," in *Information Engineering and Electronic Commerce, 2009. IEEC '09. International Symposium on*, 2009, pp. 182-186.

4. R. Iyer and M. E. Wales, "Integrating Interdisciplinary Research-based Experiences in Biotechnology Laboratories," *Advances in Engineering Education,* 2011.

Biographical Information

Ali Givmanesh is a graduate student at University of Houston. He is pursuing his degree in Engineering Technology at the college of technology. He was involved in the IEEE student organization in his undergraduate program of computer engineering and graduated in 2004 with summa cum laude honors. His graduate thesis objective has been to integrate academic study and network communication solutions to enhance bio-processing equipments.

Rupa Iyer, Ph.D., is an Associate Professor in the department of Engineering Technology in the College of Technology at the University of Houston. She is the founding director of Biotechnology programs and also directs the Center for Life Sciences Technology. In this capacity she has been responsible for developing the Biotechnology degree program and the core initiatives of the center that include education, research, workforce development, and outreach. Her research interests are in environmental biotechnology and interdisciplinary research- based education.