Final Senior Design Project Report
Senior Design Project (EML 4551C) – Spring 2014 Deliverable
Team 07 - Microalgae Photobioreactor

Prepared By:
Markus Dillman
Stephen Kassing
Matthew Vedrin

Project Advisors:
Dr. Juan Ordonez
Department of Mechanical Engineering, FSU
Dr. José Vargas
Department of Mechanical Engineering, UFPR

Course Professors:
Dr. Chiang Shih
Department of Mechanical Engineering
Dr. Kamal Amin
Department of Mechanical Engineering

April 17, 2014
Department of Mechanical Engineering, Florida State University, Tallahassee

Table of Contents

1.0 Abstract

This year the customer needed a way to transform an airlift photobioreactor's current "batch" growth system into a "semi-continuous" growth system. Therefore, team 7 first cultivated microalgae for the project experiments using growth methods from the Federal University of Parana, in Brazil. Second, we designed and constructed an automated addition/extraction unit controlled by an Arduino Uno microcontroller. Thirdly, we improved the concentration sensors data-logging capabilities via serial communication with the addition/extraction. Finally, the team designed and assembled a working airlift photobioreactor based on the design from Brazil. A successful prototype was built, proving this semi-continuous system is possible.

2.0 Acknowledgement

3.0 Background

Coal, petroleum, and natural gas are all nonrenewable resources commonly used today. As consumption continues to increase, these resources are becoming more and more unsustainable. If changes are not made or alternatives are not found, these resources will eventually run out, crippling infrastructures and industries on a global scale. In addition, greenhouse gasses will continue to increase and negatively affect the environment. New methods of energy production and consumption must be implemented before significant consequences occur. Alternative fuels, such as biofuels derived from oils in crops, present one solution to this energy problem. Some examples of biofuel crops are corn, soybean, sugar

Figure 1 shows the Trigenerator System with Photobioreactor. This was completed by the 2011 Senior Design team. (Senior Design Project)

cane, canola, and microalgae. Microalgae are the only biofuel that can completely replace our

3

existing fuel sources. Microalgae take little room to grow compared to crop biofuels and produce biomass much quicker than other crops. According to Yusuf Chisti, within twenty-four hours, microalgae can double its biomass (*Biodiesel from Microalgae*).

The Florida Agricultural and Mechanical University – Florida State University College of Engineering (FAMU-FSU COE) in Tallahassee, FL, USA and the Federal University of Paraná (UFPR) in Curitiba, Paraná, Brazil started a partnership on senior design projects in 2005. The projects are unique in that they require international collaboration between universities. In this section, brief reviews of the past projects are explained. We will focus on the projects from 2010-present because the 2010 group was the first team to implement the microalgae in their objectives. The designs and prototypes mentioned below are currently at the FAMU-FSU COE. These were created based on existing photobioreactor systems and research at UFPR.

Starting in 2010, the combined team, consisting of a group at UFPR and a group at FAMU-FSU COE, developed a new way to monitor the effects of carbon dioxide on microalgae. The FAMU-FSU COE team built a bench-top airlift photobioreactor. The CO_2 was supplied though the bottom of the airlift tubes rose through the algae, and exited out the top. The microalgae concentration was monitored by counting the cells under a microscope and the optimal amount of CO_2 to get the most cell growth were determined (*Senior Design Proposal*).

In 2011, the team implemented this knowledge of the effect of CO_2 on microalgae into a trigeneration system. The trigeneration system was built by the 2005 senior design team. It is a system that created hot water, electricity, and refrigeration from fuel in an internal combustion engine. The final project is shown in fig. 1 (*Senior Design Proposal*).

Figure 2 shows the Photobioreactor created last year. It incorporated sensors to monitor the microalgae growth. (Senior Design Project)

The 2012 team went in a different direction. They built a mini-photobioreactor at the FAMU-FSU COE, designed as a small-scale prototype of the one in Brazil. In addition, they designed and implemented concentration and mass flow rate sensors to determine the optimal time to extract the microalgae in order to obtain the most biomass before it starts to die. The use of these sensors, when calibrated correctly, can save significant time compared to the previous method of manually counting microalgae cells. The mini-photobioreactor built in 2012 is shown in fig. 2 (*Senior Design Proposal*).

This year's project will include several objectives that build on the efforts of previous groups and a few objectives that require the design of new devices, as well. The project is intended to reach a milestone of continuous

photobioreactor operation, as well as to build more resources for research at UFPR and FAMU-FSU COE to strengthen this collaborative international project.

4.0 Project Overview

Problem Statement

The customer defined the problem statement in the project proposal as:

> "The UFPR-FSU senior design teams have worked in the past with photobioreactors that work in batches. Growth media is added initially to microalgae to be grown in the photobioreactor for a period of approximately 15 days. At the end of that period the biomass is extracted and a new batch starts
>
> We anticipate enhance biomass productivity by using a continuous growth system (as opposed to batch) in which biomass is continuously extracted as new cells grow attempting to maintain a nearly constant ideal cell concentration within the photobioreactor. Media (nutrients and water) will need to be added in the necessary amounts to make up for the extractions.
>
> The implementation of a continuous system, required at least: (i) a concentration sensor, (ii) an automatic unit of media supply, and (iii) a biomass extraction unit. The concentration sensor was partially developed by the 2012-2013 team, and it will be further enhanced as part of a master thesis. What you are asked to design is the units (ii) and (iii), and amend the previously developed concentration sensor (i)."

Thus, the customer needs a way to transform the photobioreactor's' current "batch" growth systems into "semi-continuous" growth systems. In order to achieve this, the customer is requesting the design and development of an "automatic unit of media supply" and "biomass extraction unit", as well as an improvement to the "previously developed concentration sensor."

Project Objectives

In order to accomplish the above task, team 7 must:

- ✓ Grow and maintain a microalgae culture for design project experimentation
- ✓ Improve and modify existing concentration sensor
- ✓ Design and build a prototype-scale airlift photobioreactor
- ✓ Design and develop a fully automated addition/extraction unit

5.0 Growing Microalgae

Growing microalgae is a very time consuming and delicate process. You cannot just start growing the microalgae in a photobioreactor. It must first be purchased online in a test tube, and then gradually increased in volume and concentration until you have enough algae at the right concentration to fill the photobioreactor. The following section details three important microalgae topics: cell counting, microalgae growth, maintenance, and cleaning. Finally, a list of materials is detailed at the end of this section in table 1. It will be referenced throughout this section.

Cell Counting

Cell counting is a time consuming process that is used to calculate the concentration of the microalgae. To start you must first set up the Biological Binocular Compound Microscope which should be in the CAPS lab. The steps to setting up the camera can be read below.

Figure 1 AmScope Microscope

1. Take the protective cloth off of the microscope.
2. Place the camera lens into one of the eye slots as seen in figure 1 below.
3. Plug in the USB from the lens into the CAPS lab computer (Login Password can be obtained from Dr. Ordonez).
4. Open the Amscope program on the computer desktop and turn on the microscope power.
5. The digital image should appear in the program. If not, follow the troubleshooting section in the microscope manual.
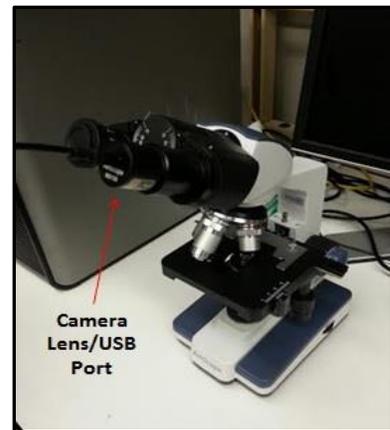
Now that the microscope is set up and turned on you can proceed to the next step. The counting of algae is done using a Neubauer Chamber with the clear glass cover seen below in figure 2. The glass cover and chamber are first cleaned with a Kim Wipe and placed on the lab counter. A plastic pipette is then used to take a couple sample drops of microalgae from the containers. Be sure to first stir the container of algae to make sure there is an even distribution of microalgae throughout. Next, carefully place a few drops of algae below the glass until you have an even distribution on the Neubauer grid as seen in figure 2. There are two open slits on the side of the Neubauer chamber, place the drops on the slits and you should see the algae spread over the grid. The Neubauer chamber can then be placed under the microscope. Make sure the correct magnification is set then proceed to the next paragraph.
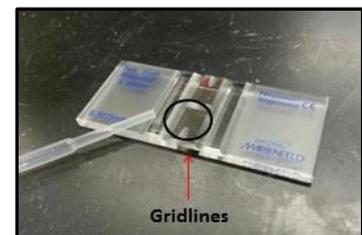

Figure 2 Neubauer Chamber

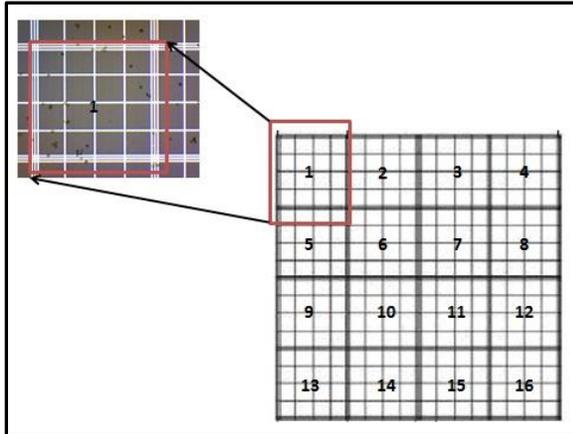The actual cell counting can begin once the Neubauer chamber is placed under the microscope. Use the microscope manual to adjust the focus and light. Once done, move the longitudinal and latitudinal wheels, moving the chamber, until the grid lines appear on the screen. You may need to make minor adjustments to the focus in order to clearly see the grid lines and microalgae cells. The Neubauer chamber contains a set of 4x4 major (16 squares total) and 5x5 minor gridlines seen to the right in figure 3. Because of the microscopic zoom, only one grid can be counted at a time (seen to the left in figure 3). The goal is to count all the algae on the 4x4 grid, using a counter. Next, multiply that number by 10^4 to get the concentration of microalgae in cells/mL. Note: we will often refer to a cell concentration by the number not multiplied by 10^4. Counting should be done on a daily basis in order to get a good algae growth curve. It is a good way to monitor the algae to make sure it is continuing to grow. An ideal growth curve is seen below in figure 4. The different phases are labeled. A continuous system is plotted in red and a semi-continuous system is plotted in black dash lines.
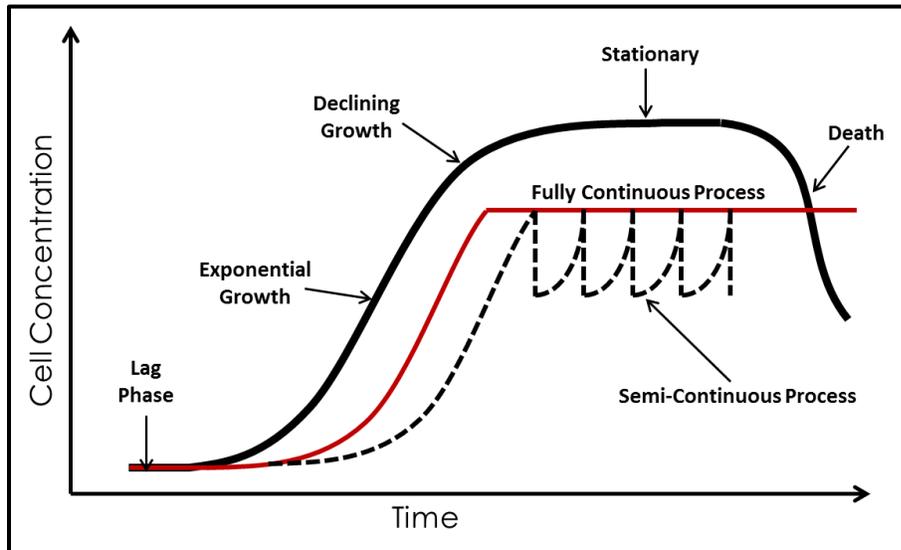
Figure 3 Neubauer Counting Grid

Figure 4 shows several growth curves of microalgae

Microalgae Growth

The growth of microalgae is a very important phase of the project. You cannot simply start the algae growth in the photobioreactor. It must be cultivated from a test tube and gradually increased in volume until you have enough to fill the photobioreactor. The microalgae will need 2-4 months of growth before it can be placed in the photobioreactor. The growth is performed in the CAPS wet lab. Each growth phase is

Figure 5 Vile of Microalgae and AlgaGro

outlined below and a list of materials is given at the end of this section.

PHASE 1: Purchasing Microalgae, Growth Media, Preparing Containers, Lighting

This year and previous year's projects have used the species Chlorella Vulgaris due to its resistant properties, optimal growth temperatures, and low or high light growth levels. The algae are bought online for $8.00 from Carolina Biological and are shipped in a test tube.

In order for the algae to grow you must prepare a food medium for it to eat. We purchase Alga-Gro concentrated medium tubes (Pack of 12) online at Carolina Biological. They are $39.95 for each pack of 12 tubes. Each test tube of media is mixed with 1L of distilled spring water. Make sure the container has been cleaned and sterilized before mixing in order to avoid contamination. The microalgae and growth media are shown in figure 4 above. The amount of Alga-Gro will be mentioned specifically in each section. If needed, more information can be found in the resource pamphlet that comes with the Alga-Grow.

Figure 6 Lab scale algae growth

Finally, the growth containers must be prepared. There are four major containers used thought the growth process: 200mL and 500mL Erlenmeyer flasks, gallon jugs, and finally the glass tank. The flask and gallon jugs are prepared in a similar manor. You will need: glass tubing, black rubber plugs with two holes, air compressor, and the air compressor lines. All of these materials are in the wet lab at CAPS. The steps for setting up the flask and gallon jugs can be seen below and the complete setup is seen in figure 5.

1. Cut the glass tubing to the exact height of the container being used.
2. Place the correct size plug into the flask or jug with the glass tube going through one of the openings. The tube should be an inch off the bottom of the containers.
3. Run rubber air line from the compressor to the end of the glass tube.

The tank is set up with minor difference as seen in figure 8 below. It will be explained later in the section when that growth phase is presented. The algae must be given air in order to process the CO_2 and turn that into Oxygen through photosynthesis. It is important to rinse all the glass

and materials with distilled spring water before they are exposed to the algae. The proper lighting equipment for growth is in the wet lab under the ventilation hood. There are two florescent lights on the bottom, facing the algae from two sides, and one growth LED light that is hung from above. All the lights are set on a timer to come on the same time of day the sun is shining. Now that the equipment is prepared, proceed to the next section to begin the growth of the microalgae.

PHASE 2: Beginning Growth 200/500mL Flasks

Now that the microalgae and growth media has been purchased and prepared, it is time to start the initial growth phase. The steps for this phase are seen below. Our team started with two test tubes of Chlorella Vulgaris (perform each step twice for each batch). Keep in mind, this process is not perfect and may change from year to year. Each bath of microalgae can grow differently. Cell counting for this stage should be performed right after each cultivation then 1-2 times per week after that to make sure the algae is maintaining steady growth.

1. Mix 10mL of each stock culture (test tube) with 115mL of fresh Alga-Grow food medium into a 200mL Erlenmeyer flask.
2. Place new culture under cool-white florescent light for 7-10 days to allow the algae to grow.
   - ➢ NOTE: Make sure to agitate the algae daily by shaking the flask. This keeps the algae from growing on the bottom of the flask.
3. Add air after 7-10 days and wait another week until the algae starts to become a darker green color.
   - ➢ NOTE: To add air, follow the instructions in the previous section. The light should be on a timer, but the air should be on all the time. After 7-10 days you should notice the color change from a lime green to a much darker green (Then is time to subculture the algae- go to next step).
4. When you believe the color has turned dark enough it is time to perform the first sub-culture. Poor the 125mL of algae into a 500mL Erlenmeyer flask. Add food media until the volume reaches 400mL and continue to input air for 1-3 weeks. Phase 4 is now complete.

Figure 7 shows the microalgae in the gallon jars

PHASE 3: Growth in Gallon Jugs

After the algae, in the 500mL Erlenmeyer flasks with air, looks green and high in concentration, it is time to subculture into the gallon jugs. Because we started with two test tubes, we have a total of 800mL of algae, 400mL in each flask. Once the algae are moved into

the jugs it is best to start counting the cells 4-5 times a week in order to establish a good growth curve. The steps for phase 3 are posted below.

1. Add 1L (1000mL) of distilled spring water/Alga-Gro mix into each gallon jug.
2. Add the 400mL of algae from each flask into the jugs with air input. Mix well and take an initial cell count. A photo of the gallon jug set up is seen in figure 5 above.
3. Let the algae grow for 1-4 weeks until the cell concentration reaches 650-750 ($x10^4$) cells/mL.
4. After 1-4 weeks it is time to perform the major expansion. Prepare 5L of Alga-Gro for each jug (each batch so a total of 10L).
5. Evenly distribute the 10L of Alga-Gro and 2.8L of algae into 5 jugs. This should give you just over 2.5L in each gallon jug. You can see this in figure 6.
    ➢ NOTE: each original test tube will produce 5L+1.4L of algae at this stage.
6. Now you must wait until the average concentration of each batch reaches 800 ($x10^4$) cells/mL. Once it does, proceed to phase 4 below.
    ➢ NOTE: you do not have to count the cells for all five jugs daily. You can rotate and only count 2 per day.

PHASE 4: Growth in Tank

Now it is time for the final phase of micro-algae growth outside of the photobioreactor. When the concentration has reached the correct level, mentioned above, it can be combined into one large batch. The tank is needed for this phase to allow the algae to get adjusted to living together before it is placed in the photobioreactor. At this point, you should have 5 different jugs, each with 2.6L, with a total volume of 12.8L. The steps for this final sub-culture are written below and can be seen in figure 7.

1. Prepare 24L of Alga-Gro growth media. This required two packs of Alga-Gro because each pack only contains 12 test tubes (one test tube per liter).
2. Stir each jug to make sure all the algae are no longer sticking to the glass.
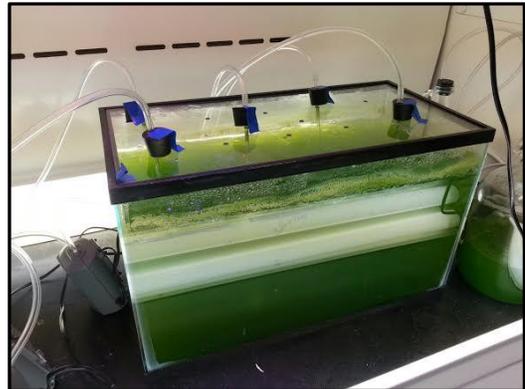
Figure 8 shows the microalgae in the tank

3. Poor all five jugs into the tank and mix with the 24L of Alga-Gro.
    ➢ NOTE: Make sure to have someone constantly stirring the tank when mixing all the algae together.
    ➢ NOTE: It is also a good idea to have 0.5L of reserve algae set aside for reserve.
4. A clear glass cover was made and has holes for the air to fit into the tank as seen in figure 7. Normal air and light is required for this phase.

5. Now the algae growth outside of the photobioreactor is complete with a total of 37L of Chlorella Vulgaris.
   ➢ NOTE: At this point the cell counting only needs to be performed 1-2 times a week to make sure the algae is continuing to grow and not declining.

6.0 Airlift Photobioreactor Prototype
System Overview

Team 7 designed and built a prototype-scale airlift photobioreactor. The airlift is a simple loop of clear PVC pipe that circulates the algae via airlift. The CAD drawing of the airlift is shown in figure 9 to the right. The primary goal of this unit is to test a smaller scale photobioreactor that will eventually maximize algae production with minimal land surface area. The system is composed of clear 1-1/2" schedule 40 PVC pipe. After the algae is placed in the photobioreactor, air is supplied to the bottom of the degassing column via air compressor. Currently, we are using a portable air compressor located at the OGZEB lab on main campus. The air passes through a diffuser mesh which prevents large pockets of air from forming in the degassing column. Large air pockets can create very high shear stress on the algae. This can kill the algae. The air bubbles rise in the degassing column causing circulation. The team built a stand for the photobioreactor. It is composed of weatherproof pressure treated lumber. The entire system is next to the solar cells just west of the OGZEB building. The specifications of the system are seen below.

Figure 9 shows the CAD of the Photobioreactor
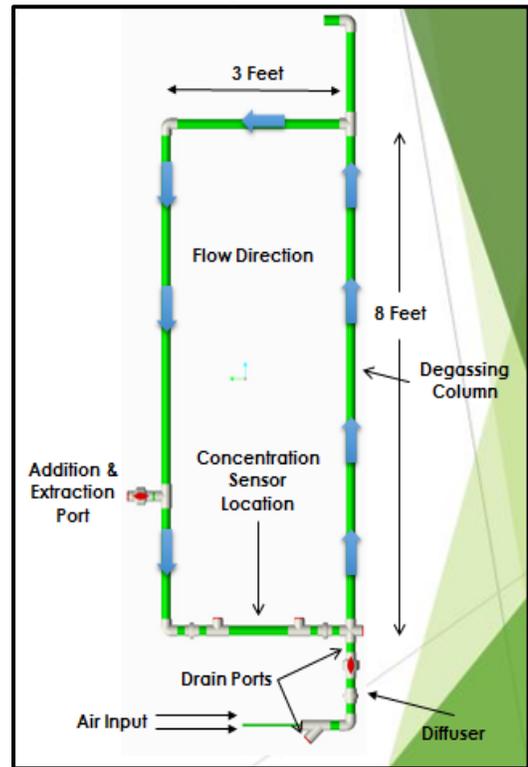
- ✓ Total Length of Pipe: 24 feet of clear 1-1/2" PVC
- ✓ System Volume: 10L
- ✓ Max Height of System: 10 feet above ground level
- ✓ Degassing Column Height: 8 feet
- ✓ Width: 3 feet
- ✓ Air Input: Portable Compressor (100PSI)
- ✓ Stand Material: Weather Pressure Treated Lumber

Drain Process

To clean the photobioreactor and/or to replace the diffuser mesh within the

photobioreactor, first the photobioreactor must be drained into a storage tank to keep the algae alive during the cleaning process. Before any draining, set the system to standby through the computer so that it will not make and concentration measurements or try to perform addition and extraction cycles while it is trying to be cleaned. Remove the concentration sensor and mass flow sensor and place out of the way of the photobioreactor. Shut off the regulator and disconnect the air supply with the quick disconnect fitting. Close the addition and extraction ball valve. There are two drain ports. The first is located just above the lower ball valve that controls the air input. A hose is attached to the drain valve to direct the flow into the storage container to prevent major losses of algae. After all the algae that are located above this valve has drained, the drain valve is then then closed. The lower drain valve, located on the bottom of the photobioreactor, is now used to drain the rest of the algae from the system using the same hose to also direct the flow into the storage container until empty.

Cleaning and Diffuser Replacement

Now that our system is empty, the diffuser union can be detached and the mesh can be removed. The union is cleaned by hand and so is the bottom part of the system that leads to the air regulator with a cleaning sponge attached to a wire. Next is to remove the mass flow sensor pipe. For the right side of where the mass flow sensor was, the cleaning sponge is inserted such that it is positioned to rise through the degassing column (sitting vertical). A secondary flange is then attached to seal up that part of the photobioreactor so that our air pressure line can be attached to the secondary flange to blow out the cleaning sponge through the top of the photobioreactor, this can be repeated until the photobioreactor is as clean as desired. Make sure to close the lower ball valve that is located just above the diffuser flange before trying to put

Figure 10 shows the completed photobioreactor.

compressed air into the system through the secondary ball valve, otherwise there will be no pressure build up behind the sponge and it will not work. For the side opposite to the degassing column, the left side of where the mass flow sensor used to be, the same input is used to force the cleaning sponge up and out through the top of the degassing column using the secondary flange. Before the air is shot into the photobioreactor for this part, the sponge cap will need to

be removed to allow the sponge to shoot out the right side of the photobioreactor and can be collected later for reuse.

Now the photobioreactor is cleaned and it's time to reassemble the photobioreactor. First, reattach the mass flow sensor after removing the secondary flanges. Then reattach the diffuser union with the new diffuser mesh, if it needed to be replaced. Then open the addition and extraction ball valve and the lower ball valve just above the diffuser union. Ensure all drain valves are closed.

7.0 Sensor and Addition/Extraction Unit Detail

The Arduino Uno microcontroller was the basis of the controls for this project. Two Arduino Uno's were used, one on the concentration sensor and one on the addition/extraction unit. More information can be found about the Arduino microcontroller and programming at http://www.arduino.cc/.

Overall Control

There are two main sub-sections of the overall control system for this project: 1. the "master" controller (or the addition/extraction unit) and 2. the "slave" device (or the concentration sensor). With the goal of the project being to semi-continuously operate a microalgae photobioreactor, the slave device provides cell concentration information to the master controller. The master controller then interprets this data to execute semi-continuous algae cultivation. The only slave device that exists in this project is the concentration sensor device, however any number of slave devices can be connected to the master controller. In order to do so, the user simply needs to create a parallel circuit from the second slave device to the Tx/Rx ports on the master controller. In other words, both slave devices will be connected in parallel to the same Tx/Rx ports on the master controller.

Below is a simple flow diagram to provide a brief visual representation of the physical setup and information flow for the master/slave control system.
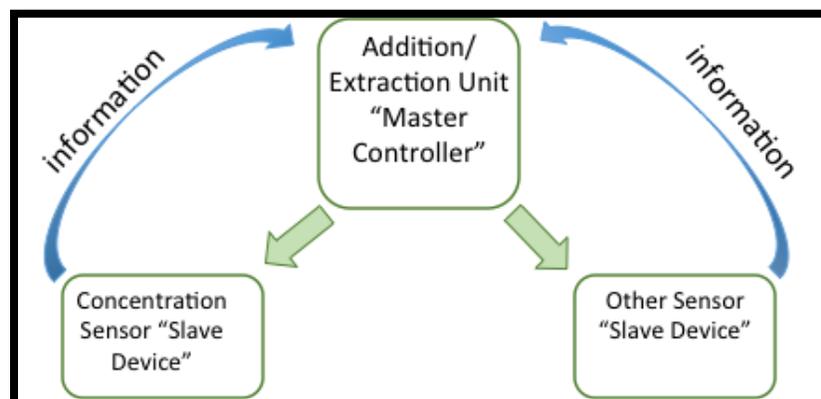

Figure 11 Communication overview

13

Outlined below are some general notes and precautions about the controls/sensors. The ordering of the bullet points in this general section is more arbitrary than the following sections and is intended to provide a rough overview of the control process across systems.

General Notes

- The idea is to send information from the concentration sensor (and any future sensors) to the master controller in order to operate the semi-continuous process.
- The concentration sensor is based on optical transmittance through a clear section of the tubular photobioreactor. A calibration curve needs to be created in order to determine the relationship between amount of light transmitted and concentration of algae in the photobioreactor.
- The master controller is part of the addition/extraction unit, which also controls two solenoid valves and a peristaltic pump to perform the semi-continuous extraction and addition processes.
- The microcontrollers used in this project are Arduino Uno's, however any microcontroller can be used for this system. Arduino was chosen because of its low cost, ease of use, and use in previous phases of this project.
- The system at large uses serial communication to send and receive data between microcontrollers.
- All data is sent to the master controller (addition/extraction) via serial communication and stored via onboard microSD flash storage memory card.
- The master controller is Ethernet "capable", however this year's project was not able to implement this into the programming code.
- No external interfaces exist for either of the microcontrollers, however it is suggested that future efforts incorporate some sort of an external display interface where commands may be received. The only "external" command that can be executed is pushing the "reset" button on any of the microcontrollers.
- All modifications to programmed code must be made via computer upload through the USB connection ports on each of the microcontrollers.
- It is advised to power the arduino microcontrollers with the power adapters when running these programs. The USB connection can also provide power, but it is safer to have the power connected at the same time because of the current being drawn by the additional devices.
- EXTREME CAUTION should be taken when dealing with these electrical circuits. There is not a significant amount of power flowing through the circuits because the components do not draw a lot of current, however it is still dangerous to be working around live wires – especially those from the 12V power supply.

Concentration Sensor (Slave Device)

The only operational instructions include making sure the concentration sensor (also known as the slave device) is connected to the power source and the lights on the Arduino are lit. The "reset" button may be pressed periodically to refresh the Arduino, however this is not necessary as the Arduino is programmed to remain in idle until it receives communication from the master controller.

In terms of general maintenance and upkeep, the concentration sensor device should be examined once every day to ensure there is no broken or malfunctioning hardware. In addition, it is advisable that before heavy rains or bad weather conditions either the concentration sensor is removed from the photobioreactor and stored safely indoors or the sensor is routinely checked for water leaks.

The electrical circuitry of the concentration sensor is attached in the appendix. The bill of materials for the concentration sensor is essentially all of the electrical components shown in the circuit diagram, as well as the casing, weatherproof, and cushions materials seen in the image of the sensor below. None of the materials are especially unique and so specific information about them will not be mentioned here. It can be noted, however, that the casing was made from a large pvc pipe that was then cut and modified with hinges, most of the foam was obtained from UFPR in Brazil (it was some sort of previously used insulation material), and most of the components have been hot glued to the inside of the case. This provides a relatively permanent placement for the internal components while making it at least feasible to detach them.
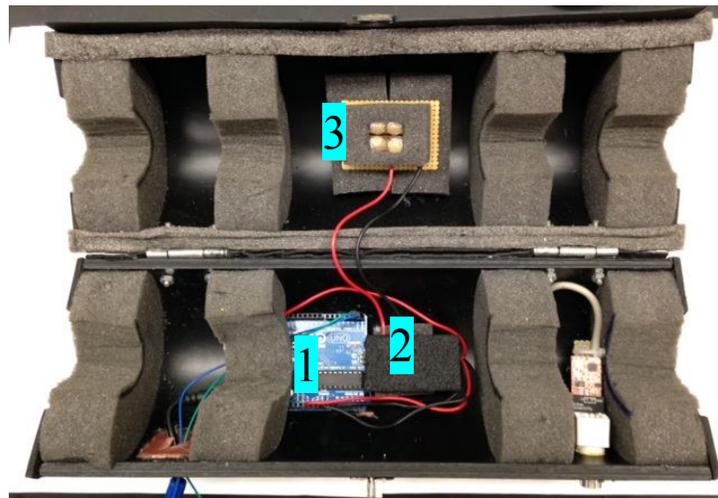

Figure 12 Concentration Sensor Hardware: 1. Arduino Microcontroller; 2. Wheatstone Bridge (LDR1, LDR2, R1, R3) & Control LED (LED5, R2); 3. Test LED's (LED1-4, R4-7)

The programming code for the concentration sensor microcontroller can be found in the appendix of this report. The best way to read the code is to download the Arduino Integrated Development Environment (IDE) at http://arduino.cc/en/main/software and copy the code from this report into the IDE. The code, itself, is relatively well commented and so no more explanation will be included here about specific details of the program.

The concentration sensor should be mounted on the airlift photobioreactor at the specified location (see the CAD depiction earlier in this report) so that the latch faces down at the ground and the hinges face the sky. This will prevent the most water from entering the device if it rains.

Addition/Extraction Unit Controller (Master Controller)

The addition/extraction unit consists of an Arduino microcontroller, a motor driver circuit board, two solenoid valves, and a peristaltic pump. A schematic of the control process is shown below.

The Arduino Uno is able to store data to the microSD memory storage card. Troubleshooting and program updates can be made through modification of the Arduino's program when it is connected to the computer via the USB serial connection. The Arduino controls two 12V DC solenoid valves and one 12V DC peristaltic pump via an L298 Dual Motor Driver module from RobotShop.com. The peristaltic pump can be operated both forward and backward so that only one pump needs to be used for both addition and extraction.
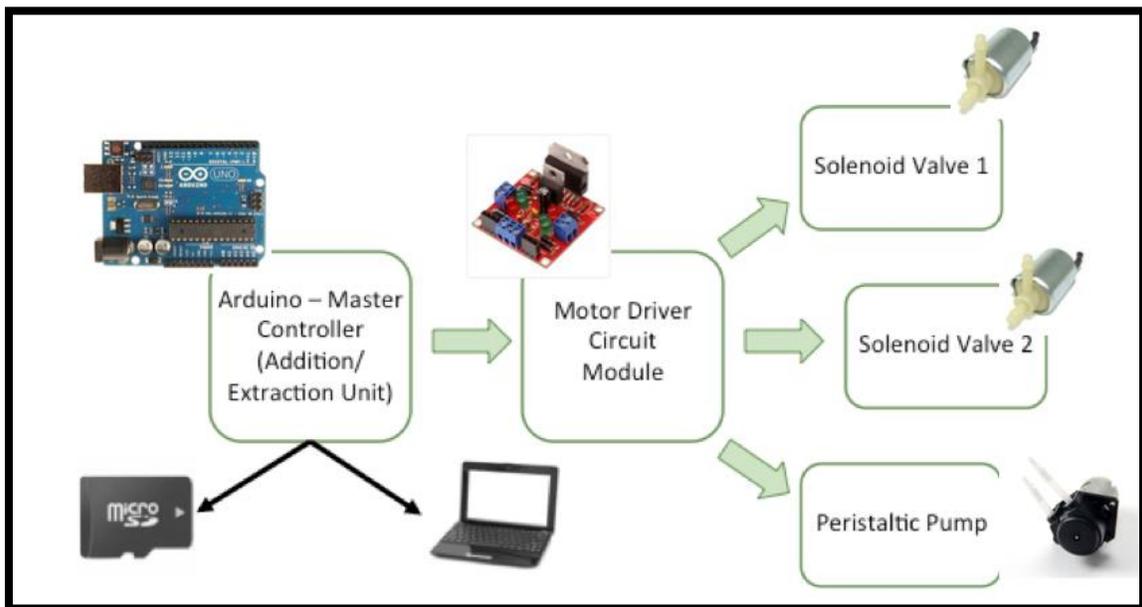
Figure 13 Addition/Extraction Component Schematic

Aside from performing waterproof, wear, and damage checks to all of the components

and ports the operation of this unit is similar to that of the concentration sensor device. The entire control of the unit is designed to be autonomous once the proper program is initially uploaded to the Arduino and therefore there are no physical operational tasks required while the unit is running.

An image depicting the temporary wiring setup for the addition/extraction unit housed inside the waterproof box is shown below. The blue device is the Arduino microcontroller. The red device is the L298 Dual Motor Driver. Also shown here is a small non-solder bread board used to make the electrical connections before the wires are more permanently soldered.
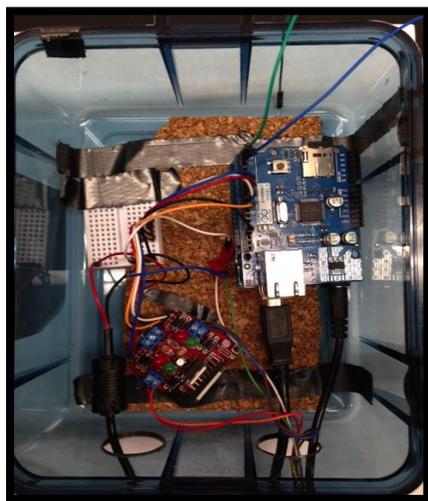
Figure 14 Addition/Extraction Circuit Box

An image depicting the addition/extraction unit connected to a tabletop test setup is shown below. All components are easy to disconnect and reorder. The setup should be tested to ensure that the Arduino program matches with the physical setup – i.e. the solenoid valve used for extraction is wired to the "extraction solenoid" pin on the program.
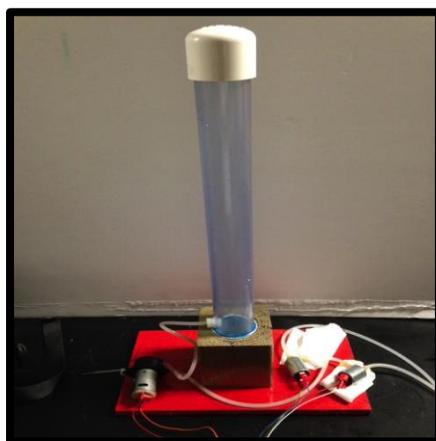
Figure 15 Test Unit with Addition/Extraction Components

17

Finally, one important note about connecting the addition/extraction unit to any system is to ensure that the electrical connections are arranged properly. The circuit diagrams outlined in the appendices should be a good guide; however a few steps will be mentioned here. The 9V power supply to the Arduino and the 12V power supply to the motor driver should be plugged into the power source (i.e. the wall outlet). The ground line must be shared between the 9V power source and the 12V power source on both ends – if not; there could be significant electrical repercussions on the Arduino board.

General Process Steps of Sensors & Controls

There are two Arduino Uno microcontrollers in the current system at large: one is part of the concentration sensor and will be referred to as the "slave" controller, the second is part of the addition/extraction unit and will be referred to as the "master" controller. The master and slave controllers communicate via serial ports located on each Arduino.

PROGRAM SECTION	MASTER CONTROLLER (ADDITION/EXTRACTION UNIT)	SLAVE CONTROLLER (CONCENTRATION SENSOR)
Setup (reset button or power-up)		
Step 1	Initialize variables and input/output pins	Initialize variables, input/output pins, and onboard hardware (microSD)
Data Acquisition Phase		
Step 1	Send command character, "b" for "begin", to the slave controller to request concentration data. After sending, enter while loop until serial reads the handshaking character. If no response within x seconds, send another character and wait again.	Enter while loop until serial reads buffer command character "b" from master.
Step 2		Once command is received, send handshaking character, "b" for "begin", back to master to indicate that it is ready to send data.
Step 3	Receive the handshaking character, "b", from the slave controller and enter while loop to continuously check serial buffer for concentration data. If waiting for more than x seconds, return to beginning of "*Data Acquisition Routine"*	
Step 4	Receive measurement and immediately save it into a data file on an onboard microSD card.	Enter data acquisition process: take concentration measurements. Send each measurement to master as it is taken.
Step 5	While in while loop waiting for data: if receive an "e" command from slave, leave the data acquisition while loop and calculate average of all measurements recently received.	When finished taking concentration measurements, send command character, "e" for "end", to master indicating data acquisition is finished.
Step 6	Input average value into concentration calibration equation. If average concentration is greater than or equal to target value, enter "Addition/Extraction Phase". If average concentration is less than target value, pause for a short duration and then return to the start of the "Data Acquisition Phase"	Return to while loop at beginning of "Data Acquisition Phase" to wait for next data request from master.
Addition/Extraction Phase		
Step 1	Sends a digital 5V signal to the motor driver, which holds open the extraction solenoid valve. Then pause for a short time to ensure the extraction valve opens before pump operation.	
Step 2	Send a second digital 5V signal to the motor driver, which corresponds to operating the peristaltic pump in the "extraction" direction of flow.	
Step 3	Return the digital signal that controls the pump back to 0V, pause for a short duration, then return the digital signal that controls the extraction solenoid valve to 0V to close the	

	valve.	
Step 4	Wait for a short duration and then repeats steps above for the addition valve and "addition" direction of the pump.	
Step 5	Exit "Addition/Extraction Phase", pause for a short duration, then return to the beginning of "Data Acquisition Phase"	

8.0 Designs for Manufacturing, Reliability, and Cost

Design for Manufacturing

The photobioreactor and all units are a small prototype-scale system compared to an actual photobioreactor. All parts for the smaller, prototype photobioreactor can be bought in bulk to make manufacturing economically affordable. The system can be scaled up very easily. More piping can be added to the system at an affordable cost. The only potential problem is finding a large scale peristaltic pump. Currently they are used for medical and biological purposes on a smaller scale. This is something that future projects should look into. A larger peristaltic pump that is economically reasonable is ideal for a larger scale photobioreactor.

Design for Reliability

The photobioreactor system is designed to be operated with very little human interaction and no motion, it is a stationary object. Therefore, there are very little safety risks. The frame has a potential risk of falling over outside when subjected to high winds. With this in mind we will place sand bags at the base of the photobioreactor for base support. Addition safety lines are also attached to the photobioreactor to hold it in place. All the electrical components will be properly placed inside waterproof electrical boxes and sealed. Finally, in the operation manual there is a method of cleaning the photobioreactor. If this is done each time a new batch or experiment is started the survivability of the photobioreactor will increase. Little or no other maintenance has been needed at this time.

Design for Cost

The list of materials and cost for the growth of micro-algae is seen below in Table 1. Most of the material was purchased from Carolina Biological, an online biological supply store. The algae food and spring water is not a very cost effective way to grow algae. Ideally, we would like to create our own food and use tap, lake, or salt water. Due to the lack of time, further research is needed to be performed in order to successfully grow algae using these methods. We are currently studying the effect of a sugar water food substitute to see if we could use it in the future. We are working with two chemical engineering students and are comparing the growth concentration of sugar water versus the online food Alga-Gro. Next year's group should continue this research because the current food we are using has cost us around $240 dollars. We did have to purchase some lab equipment to improve the quality of algae growth. Also our lab did not have a working microscope to count algae cells. Our team had to purchase one

online for $250 dollars. Our sponsor did reimburse our team for the microscope. The materials for the lab will be passed down to future projects.

The airlift photobioreactor and frame was built with cost effective materials that would still meet our performance standards. The frame was constructed using prime pressure treated lumber from Home Depot due to its weather resistant properties and easy machining. Our team could have used a metal steel material but decided the extra cost and challenges of machining was not worth it. Also, this is only a temporary frame. In the future, the photobioreactor will be permanently attached to a wall or building. The lumber, nails, pipe hooks, and other materials for the frame were all purchased online using Home Depot "in-store" pickup. This saved shipping cost and time and was very important economically for our project. The materials and cost for the frame are seen below in Table 2.

The airlift photobioreactor was constructed using Harvel Clear Rigid Schedule 40 PVC pipe from US Plastic Corp. The pipe is easy to handle and is recommended for consideration in food and pharmaceutical plants, laboratories, and chemical installations. Due to that fact that it had to be shipped in ten foot sections, the cost of shipping was $90 dollars. This was very high but after calling multiple companies, it was our best option. Our team decided to order an extra ten feet of pipe in case we made a mistake during the construction phase of the project. This would avoid another high cost of shipping for US Plastic Corp. Once received, the pipes were cut to exact dimensions at the MagLab machine shop. All the fitting and other material were purchased from Home Depot and picked up in store to avoid other shipping charges. The list of parts and cost for the photobioreactor is seen in Table 2 below. Testing and other parts of the project did not cost the team money.

The materials and cost for the addition and extraction unit, concentration sensor and controls are all seen below in Table 3. All of the materials for the addition and extraction unit were purchased online from McMaster-Carr because of its fast shipping and reliability. The peristaltic pump and solenoid valves were found on Amazon. We decided to use the least expensive pump and solenoids because this is just a prototype and we want to prove the system works before we order more expensive parts. At the end of the semester we should have enough data to prove whether or not our devise operates to our specifications. If our design meets the specification better and more reliable parts should be purchased. Many of the electronic and control products had very high shipping times. We purchased the products with the best combinations of shipping time and cost.

9.0 Communication

This project involves two universities, Florida State University and the Federal University of Parana. Communication has been a very important part of this project. In the fall Matt Vedrin was working in Brazil as an exchange student. We held weekly video meetings on Tuesday and

Thursday to inform exchange each other's progress and discuss future plans. This semester we have held weekly meeting with our sponsors Dr. Ordonez and Dr. Vargas. These meetings, held on Friday, were vital to the success and direction of our project. Each team member presented their part effectively throughout the semester. After each presentation the team sat down and reviewed the comments from our classmates. All our documents are saved in Google Drive. Everyone who is part of the project has access to each document. We have a team calendar that has each team member's class schedule, making it easy to schedule meetings. Our team website is now online and has all the important presentations and documents.

10.0 Recommendation for Future Work

This section proposes suggestions for future work to be carried out in the future semesters. Keep in mind, these are only suggestions and should not be carried out until given permission by Dr. Ordonez or another sponsor. To begin, the future team should research an alternative growth media for algae growth. This year the team spent well over two-hundred dollars in growth media. Other methods, if tested, could prove to be even better than Alga-Gro at a fraction of the cost. Sugar based spring water and pig manure are a few options of alternative growth media. Second, the future team should further develop other methods of non-invasive mass flow sensing. This is a very important part of the photobioreactor. The mass flow rate is needed to effectively grow microalgae. A non-constant flow to high or too low may agitate the algae and kill it. The algae must be held at a constant flow in order to have consistent growth. Thirdly, this year's team developed a semi-continuous prototype system. The future team should research into methods of developing a fully continuous addition and extraction system. Next, another suggestion for future work is to create a larger airlift system capable of increasing the production of microalgae. This will bring the team one step closer to creating a full scale photobioreactor growth system. Finally the next team should try to optimize the algae growth by using an automated sensor and addition/extraction system. A summary list of the future improvements can be seen below:
- ✓ Research alternative algae growth media
- ✓ Further develop non-invasive mass flow sensor
- ✓ Research and develop continuous addition/extraction system
- ✓ Create a large scale system capable of increasing the production of microalgae
- ✓ Optimization of algae growth using automated sensors and addition/extraction system

11.0 Budget and Expenses

A list of all the materials purchased for the project is seen in the tables below. Table 1 show all the microalgae materials. Table 2 shows all the materials for the photobioreactor. Finally, Table 3 shows the materials for the controls and sensors. The budget for our project was $1500 dollars. Our team spent a total of $1494. The total cost of the microalgae growth was $649.93, the photobioreactor was $620.98, and the controls and sensors was $216.56.

Table 1: Microalgae Materials

Part	Vendor	Purpose	Cost (Qty.)
Chlorella Vulgais Algae Vile	Carolina Biological	Starting amount of Algae	$10.95 (1)
Carolina Alga-Gro (Pack of 12)	Carolina Biological	Algae Growth Media	$239.70 (6)
Rubber Stopper (Size 6)	Carolina Biological	For small flasks	$12.95 (1)
Rubber Stopper (Size 10)	Carolina Biological	For Gallon Jugs/Tank	$12.95 (1)
Karter211Z3 Standard Microscope Slip	Amazon	Glass Cover	$5.95(1)
Pyrex Tubing (Hard Glass)	Carolina Biological	For air line into containers	$16.4(1)
Hemacytometer Neubaur Chamber	Amazon	For counting algae cells	$71.86 (1)
200mL and 500mL Erlenmeyer Flask	CAPS lab	Algae Growth Container	Donated
Gallon Jugs	CAPS lab	Algae Growth Container	Donated
Tank	CAPS lab	Algae Growth Container	Donated
Amscope Microscope	Amazon	Counting algae cells	$249.88 (1)
Growth Lights	CAPS lab	Lights for algae growth	Donated
Spring Water	Publix	Medium for algae	$20 (20)
Air Compressors	Amazon	Pump air to algae	Donated
Hand Counter	Amazon	Helps count algae cells	Donated
Kimtech Science Kimwipes	Amazon	Used to clean Neubaur Chamber	$4.99 (1)
		Total Cost	**645.93**

Table 2: Photobioreactor Materials

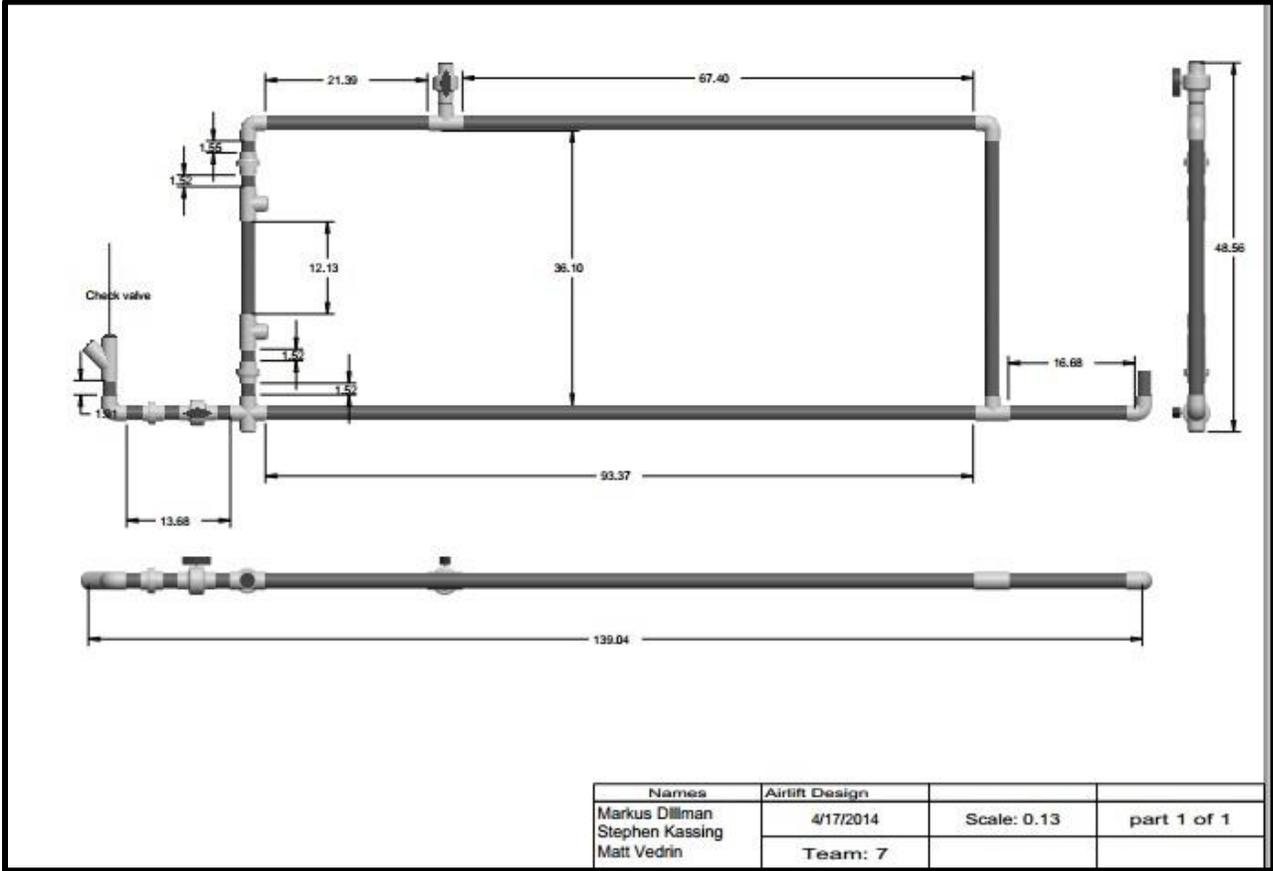| Part	Vendor	Purpose	Cost (Qty.)
1-1/2" Clear Sch 40 PVC Pipe	US Plastic	Pipe for photobioreactor	$328 (40')
Purple Primer and PVC Cement	Home Depot	Glue PVC	$7.47 (1)
PVC 1-1/2" End Cap	Home Depot	Photobioreactor material	$3.99(3)
90 Degree Elbow	Home Depot	Photobioreactor material	$5.12 (4)
1-1/2" Slip Tee	Home Depot	Photobioreactor material	$7.76 (4)
1-1/2" Wye	Home Depot	Photobioreactor material	$2.04 (1)
1/2" Hose Bibb Valve	Home Depot	Drain valve	$9.14 (2)
1-1/2" to 1/2" Bushing	Home Depot	Photobioreactor material	$9.36 (3)
Vair Pressure Regulator	Home Depot	Regulates air	$29.95 (1)
Legend 1-1/2" Ball Valve	Home Depot	Photobioreactor material	$13.26 (2)
1-1/2" PVC Union	Home Depot	Attach 2 PVC pipes	$16.8 (5)
Stainless Steal Wire Cloth	McMaster	Diffuser material	$15.62 (1)
1/4" Brass Check Valve, 2nd	Granger	Prevents air into regulator	14.99 (1)
1/4" Brass Check Valve	McMaster	Not Used	$22.03 (1)
Watertight Box	Walmart	Controls Box	$21 (2)
Brass Fittings	Home Depot	Attach air lines	$15 (1)
Outdoor Extension Cord 25'	Walmart	Power cord to PBR	$25 (1)
Air Line	Home Depot	Supply air to photobioreactor	$5.95 (1)
Pressure Treated Lumber	Home Depot	Frame Material	$43.5 (1)
1-1/2" Pipe Hook	Home Depot	Attaches pipe to frame	$7.50 (6)
Screws and Nails	Home Depot	Frame Material	$17.5 (1)
		Total Cost	**620.98**

Table 3: Controls and Sensor Materials

Part	Vendor	Purpose	Cost (Qty.)
Arduino Uno Starter Pack	adafruit	Control Communication	$65.99
Female Terminals	For Aker Electronics	Controls	$5.20
Multi-Use Sealant Tape	Home Depot	Waterproof Sealant	$3.95
8-Wire Electrical Cable	For Aker Electronics	Control Wire	$10.95 (10')
L298 Compact Dual Motor Driver Kit	Robot Shop	Controls	$18.95
Straight Nylon Barbed Tube Fitting	McMaster	Addition/Extraction	$2.76
Tee Nylon Tube Fitting (1/16" ID)	McMaster	Addition/Extraction	$3.69
Nylon Tube Reducer (1/4" to 1/16")	McMaster	Addition/Extraction	$3.15
Silicon Rubber Tube (1/16" ID)	McMaster	Addition/Extraction	$10.52
12V DC Dosing Peristaltic Pump	Amazon	Pump for Addition/Extraction	$14.99
Soleniod Valve	Amazon	Addition/Extraction	$14.76 (4)
5V Relay Module	Nexus Cyber	Controls	5.99 (1)
SD Card (Pack of 2)	Walmart	Did not use (needed microSD)	$4.99
Micro SD Card Breakout Module	Nexus Cyber	Controls	7.99(1)
9V DC Power Adapter	Nexus Cyber	Controls	5.99(1)
Ethernet Shield with MicroSD Slot	Nexus Cyber	Controls	13.99(1)
5V Electric Relay	Mouser	Controls	7.26 (4)
12 Volt DC - 2 Amp	12V Adapters	Controls	27.98(2)
		Total Cost	**216.56**

12.0 Appendix (programming code, etc....)

CAD Drawings

Names	Airlift Design		
Markus Dillman Stephen Kassing	4/17/2014	Scale: 0.13	part 1 of 1
Matt Vedrin	Team: 7		

Names	Airlift Design		
Markus Dillman	4/17/2014	Mesh Holder	part 1 of 2
Stephen Kassing			
Matt Vedrin	Team: 7		

Names	Airlift Design	Stand	
Markus Dillman	4/17/2014	Scale 0.1	Part 1 of 1
Stephen Kassing			
Matt Vedrin	Team: 7	Units: Inches	

26
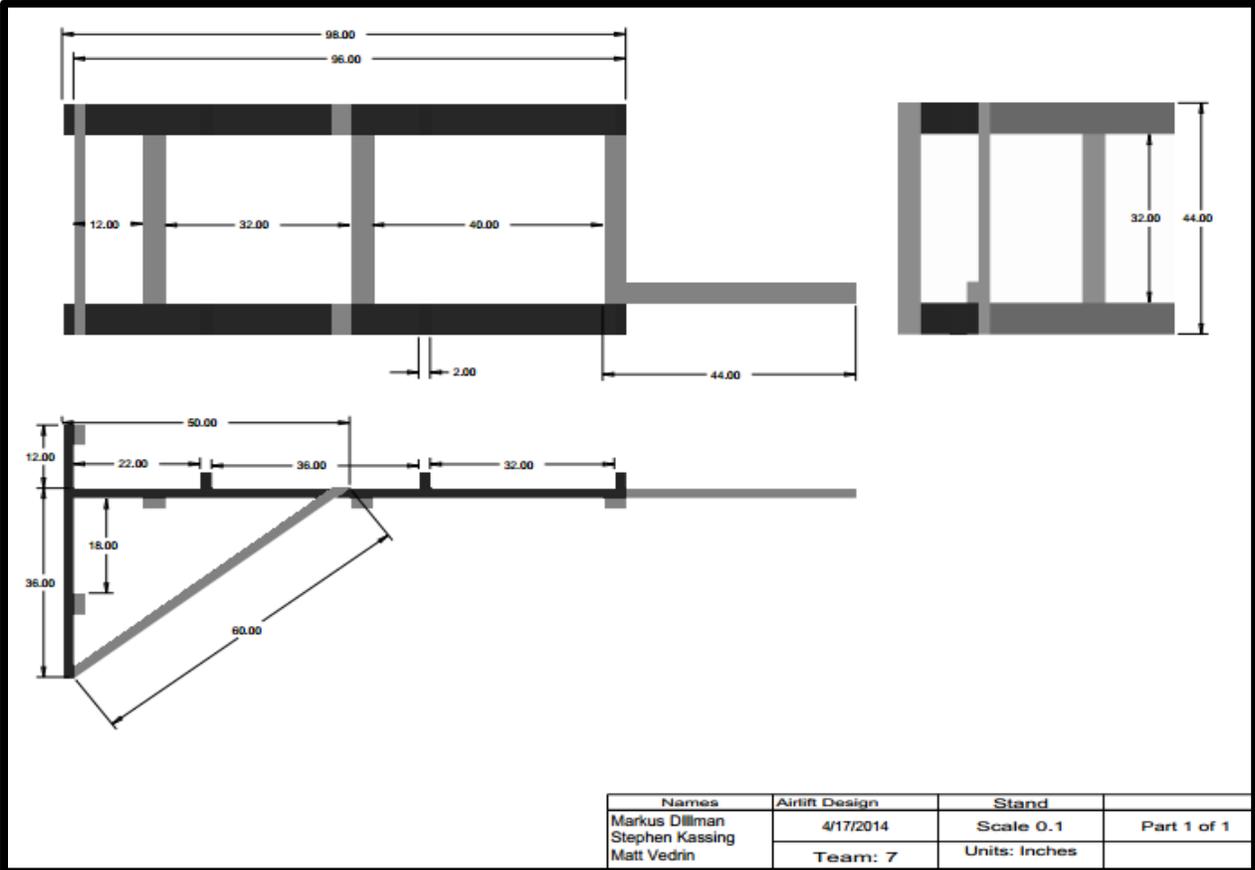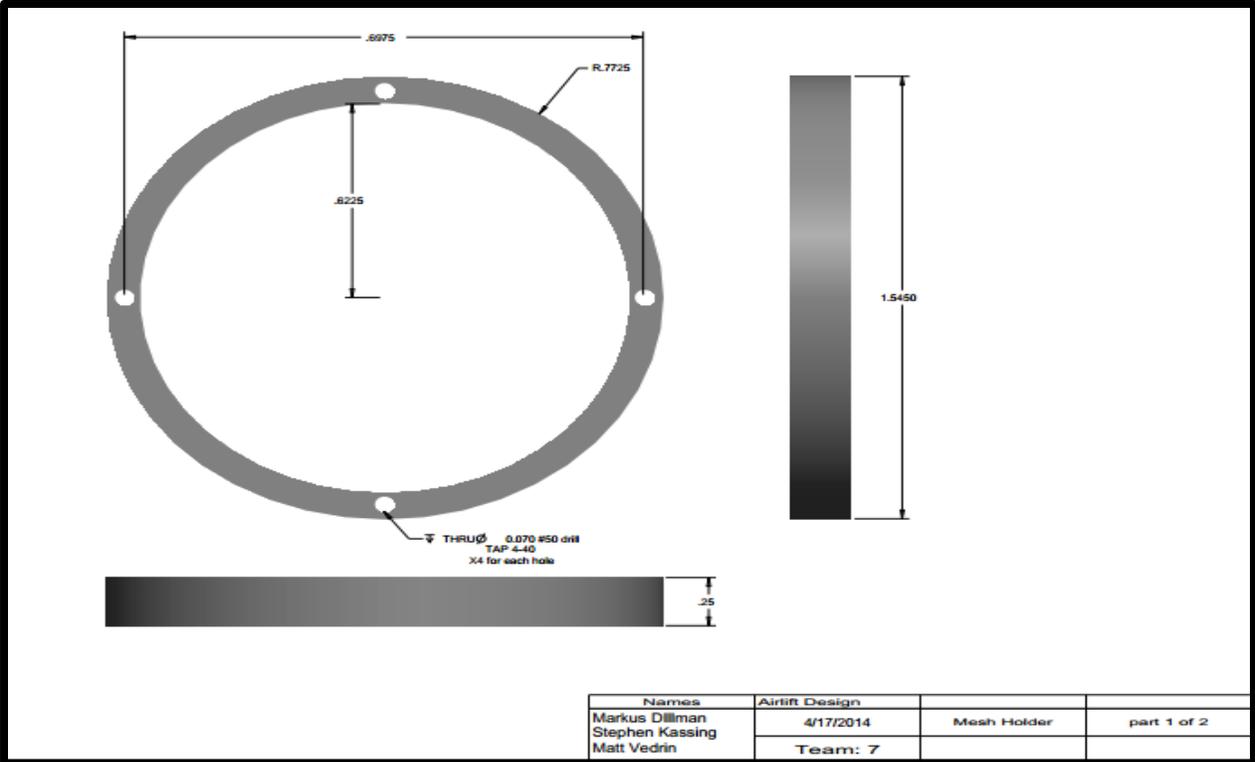
Addition/Extraction Arduino Code

```
/*
Addition/Extraction Controller (Master)
Created by: Matthew Vedrin (mcvedrin@gmail.com)
Date updated: April 17, 2014

*This program is a combination of code available in the public domain
and unique code written by the author.
*This code is not to be shared in the public domain.

-This program serves as a master controller for controlling a semi-continuous
microalgae cultivation process. It receives concentration data from a slave
controller, connected via serial, and interprets that data in order to
determine if addition/extraction should be performed, and if so, how much.
-It also uses an ethernet shield to store data to a microSD card on a file
created and edited by this program. The intention is for the microSD card to
be removed periodically and inserted into a computer in order to store the
data more permanantly.
 */

#include <SD.h> //this code needs the SD library

//const = constant (can't be changed by program)
const int chipSelect = 4; //define pin for SD chip communication (4 for e-shield)
const int SolA = 8; //define pin for add solenoid
const int SolE = 9; //define pin for ext solenoid
const int MPolF = 6; //define pin for "forward" motor pole (HIGH = forward, LOW =
backward)
const int MPolB = 7; //define pin for "backward" motor pole (HIGH = backward, LOW =
forward)
const int soldel = 750; //delay for the solenoids in ms
const int pumpdel = 750; //delay for the pump in ms
//use long b/c delays are in ms and will probably exceed 32,767
long pumpdur; //duration of pump operation in ms (later calculated)
unsigned long previousMillis; //initialize previous millis
//use string to store numbers from slave b/c serial only reads bit by bit
String readString; //initialize the string variable, "readString"
int conc; //instantaneous conc reading from slave
int mode = 0; //arduino mode, initialize to 0
int cnt = 0; //initialize counter to 0
//use long in case sum > 32767 (warning: long takes longer to execute)
long sum = 0; //initialize sum to 0 as long variable
int avg; //initialize average
//use char to read character and not ASCII number
char rd; //initialize "read" character variable for receiving char from slave
File dataFile; //initialize the File variable
```

```
//use floats to have decimals
float C1; //concentration (10^4cells/mL) before extraction (corresponds to avg)
float V1 = 9000.0; //operating volume (in mL) of PBR at desired level
float V2; //volume (in mL) in PBR after extraction
float Ct = 650.0; //target concentration
float N1; //current number of cells (before extraction)
float Nt; //target number of cells
float Vdif; //Volume to extract and add
float prate = 1.0; //pump rate in mL/s
float rconc; //real concentration value
long stallcnt = 30000; //set time that indicates potential stall in program



// the setup routine runs once when you press reset:
void setup()
{
  Serial.begin(9600);
  // initialize the digital pin as an output.
  pinMode(SolA, OUTPUT); //set solenoid pin to output
  pinMode(SolE, OUTPUT); //set solenoid pin to output
  pinMode(MPolF, OUTPUT); //set pump forward pole pin to output
  pinMode(MPolB, OUTPUT); //set pump backwawrd pole pin to output

  //Start all devices in off position
  digitalWrite(SolA, LOW); //start add solenoid off
  digitalWrite(SolE, LOW); //start ext solenoid off
  digitalWrite(MPolF, LOW); //start forward pump pole off
  digitalWrite(MPolB, LOW); //start backward pump pole off

  //*Serial.print("Initializing SD card..."); //for troubleshooting
  // make sure that the default chip select pin is set to
  // output, even if you don't use it:
  pinMode(10, OUTPUT);

  // see if the card is present and can be initialized:
  if (!SD.begin(chipSelect))
  {
    //*Serial.println("Card failed, or not present"); //for troubleshooting
    // don't do anything more:
    return;
  }
  //*Serial.println("card initialized."); //for troubleshooting
}

// this loop routine repeats until Arduino loses power or is reset
void loop()
```

```
{
 Serial.println('b'); //send "b" character (begin) to slave for handshake
 previousMillis = millis(); //create a time stamp for later comparison
 rd = Serial.read(); //read any characters sent back from slave

 //remain in this while loop until rd = 'b'
 while(rd != 'b') {
  rd = Serial.read(); //keep checking serial

  //if 30 seconds of inactivity, resend 'b' command
  if (millis() - previousMillis >= stallcnt) { //current time - reference time
    Serial.println('b');
   previousMillis = millis(); //reset time stamp for reference
   } //if (millis() - previousMillis >= 30000)
 }  //while(rd != 'b')

 if (rd == 'b') {
  rd = 'a'; //reset rd to != 'b'

  // open the file. note that only one file can be open at a time
  File dataFile = SD.open("datalog.txt", FILE_WRITE);

  //for microSD card troubleshooting:
  /*
   if (dataFile) {
     Serial.println("Data File Open Success!");
   }
   else {
     Serial.println("Data File Open FAIL :-(");
   }
  */

  //Enter Data Acquisition Routine (mode = 0)
  previousMillis = millis(); //create time stamp for later reference
  while (mode == 0) {
   char c = Serial.read();  //stores current byte from serial buffer as char

   //if data acquisition exceeds 30s, assume no activity and break loop
   if (millis() - previousMillis >= stallcnt) {
     cnt = 0; //reset count
     sum = 0; //reset sum
     dataFile.close(); //close data file
     mode = 2;
     break; //break the "while (mode == 0)" loop
   }
```

```
//data acquisition sequence
if (c == '<') { //check for command that preceeds one measurement
  delay(2); //short delay before reading next char
  previousMillis = millis(); //time stamp for later reference

  //gather data
  while(Serial.available()) {
    char c = Serial.read(); //read next byte sent from slave
    if (c == '>') { //check for command that follows one measurement
      break; //break the "while(Serial.available())" loop
    } //if (c == '>')
    else { //if no '>' then continue reading bits
      readString += c; //adds current bit value to the readString
      delay(2);  //slow looping to allow buffer to fill with next character
    } //else

    //if 30 s pass, assume glitch in program and start at void loop()
    if (millis() - previousMillis >= stallcnt){
      mode = 2; //change mode value to leave "while (mode == 0)" loop
      break; //leave "while(Serial.available())" loop
    } //if (millis() - previousMillis >= 20000)
  } //while(Serial.available())
} //if (c == '<')


//break data logging routine if end command received ('<e>')
if (readString == "e") {
  readString = ""; //reset readString
  mode = 1; //set new mode to enter add/ext section
  break; //breaks "while (mode == 0)"
} //if (readString == "e")

//log data information if measurement received
if (readString.length() > 0) {
  cnt = cnt + 1; //counter
  conc = readString.toInt(); //convert readString into a number
  dataFile.print("cnt,"); //print label to datalog.txt
  dataFile.print(cnt); //print info to datalog.txt
  dataFile.print(",d,"); //print "d" to indicate raw data
  dataFile.println(conc); //print raw data conc value to sd card

  /*for troubleshooting
    Serial.print("cnt,");
    Serial.print(cnt);
    Serial.print(",d,");
    Serial.println(conc);
```

```
    */

    sum = sum + conc; //add new raw data to sum for later averaging
    readString = ""; //reset readString to no information
  } //if (readString.length() > 0)
 } //while (mode == 0)
} //if (rd == 'b')

//if mode == 1 calculate avg & rconc, decide if add/ext needed
if (mode == 1) {
 //If cnt > 25 there will be calculation problems because sum cannot > 32767
 avg = sum/cnt; //calculate average of measurements taken
 rconc = -10.289*conc + 10789; //calibration equation to calculate real conc
 sum = 0; //reset sum to 0
 cnt = 0; //reset count to 0
 dataFile.print("cnt,");
 dataFile.print(cnt);
 dataFile.print(",a,"); //indicate that printed value is an average
 dataFile.print(avg); //print average value to sd card file
 dataFile.print(",rconc,");
 dataFile.println(rconc);

 /*for troubleshooting
   Serial.print("cnt,");
   Serial.print(cnt);
   Serial.print(",a,");
   Serial.print(avg);
   Serial.print(",rconc,");
   Serial.println(rconc);
 */

 dataFile.close();
 //*Serial.println("Data File Closed");

 //Check if add/ext is required
 if (rconc >= 700) {

  //calculate pump run time, "pumpdur", to achieve desired dilution
  C1 = rconc;
  N1 = C1*V1;
  Nt = Ct*V1;
  V2 = Nt/C1;
  Vdif = V1-V2; //volume to extract and add (in mL)
  pumpdur = 1000*(Vdif/prate); //duration of pump operation (in ms)

  /* for troubleshooting
```

```
  Serial.println(C1);
  Serial.println(N1);
  Serial.println(Nt);
  Serial.println(V2);
  Serial.println(Vdif);
  Serial.print("pumpdur,");
  Serial.println(pumpdur);
*/

avg = 0; //reset avg to 0
mode = 0; //reset mode to 0

//*Serial.println("Starting Extraction Routine");

//*Extraction Routine

//set solenoids in position for extraction (avoid delay in program)
digitalWrite(SolA, LOW); //leave addition solenoid closed
digitalWrite(SolE, HIGH); //open extraction solenoid
delay(soldel);

//operate pump for extraction for "pumpdur" seconds
digitalWrite(MPolF, HIGH); //activate forward pumping
digitalWrite(MPolB, LOW); //make back direction as "ground"
delay(pumpdur);

//turn off pump and allow for short time lapse (pumpdel)
digitalWrite(MPolF, LOW); //stop forward pumping
digitalWrite(MPolB, LOW); //leave backward at "ground"
delay(pumpdel);

//close extraction solenoid (both solenoids closed) w/ small time lapse
digitalWrite(SolA, LOW); //keep addition solenoid closed
digitalWrite(SolE, LOW); //close extraction solenoid
delay(soldel);

delay(1000); //delay one second between extraction/addition (arbitrary)

//*Addition Routine

//open addition solenoid only w/ small time lapse = soldel
digitalWrite(SolA, HIGH);
delay(soldel);

//operate pump for addition for "pumpdur" seconds
digitalWrite(MPolF, LOW);
```

```
            digitalWrite(MPolB, HIGH);
            delay(pumpdur);

            //stop pumping
            digitalWrite(MPolF, LOW);
            digitalWrite(MPolB, LOW);
            delay(pumpdel);

            //close addition solenoid
            digitalWrite(SolA, LOW);
            delay(soldel);

            //delay before next round of concentration measurements
            delay(10000); //delay before requesting for more data from slave ('b')
        } //if (rconc >= 700)
      else { //reset mode and delay if average routine not executed
        mode = 0;
        delay(10000); //delay in-between sending b's to slave
      } //else
   } //if (mode == 1)
} //void loop ()

/*
  // re-open the file for reading:
  File dataFile = SD.open("datalog.txt");
  if (dataFile)
   {
    Serial.println("datalog.txt:");

    // read from the file until there's nothing else in it:
    while (dataFile.available())
    {
        Serial.write(dataFile.read());
    }
    // close the file:
    dataFile.close();
  // if the file isn't open, pop up an error:
   }
*/
```

Concentration Sensor Arduino Code

```
/*
Concentration Sensor Program
Created by: Matthew Vedrin
Date updated: April 17, 2014

*This program is a combination of code available in the public domain
as well unique code written by the author.
*This code is not intended to be shared in the public domain.

-This program waits for a starting command from the master controller, 'b'.
-It then takes voltage readings from the "test LDR", in reference to the
voltage from the "control LDR", and sends the information (8-bit leads to
readings of 0-1023) directly to the master controller.

-Currently the arduino does not actually use the output power from pin 9.
-A low-pass filter is required if using pin 9 to ensure that the PWM signal
is smoothed out so that a constant 3.3V is being supplied to the circuit.
-For now, the positive line of the circuit is simply connected to the 3.3V
pin on the Arduino board so all components are constantly "on"
*/

//***Max arduino int value is 32,767!!!!

#define CNT 20 //number of sample counts taken at a time
int analogOutPin = 9; //defines the power output pin - needs to be a PWM pin
int i; //initialize i for summation loop
int intensity = (3.3/5)*255; //set light intensity at 3.3V for PWM output 0-255
int samplerate = 5; //measurements per second
int conc; //initialize concentration variable
int del = 1000/samplerate; //delay needed for desired sample rate
char rd; //character variable for reading command from master ('b')

// the setup routine runs once when you press reset or upon startup:
void setup() {
  // initialize serial communication at x bits per second (baud):
  Serial.begin(9600); //if using PLX DAQ --> need to set to 128000 baud, otherwise 9600
works
  analogReference(EXTERNAL); //sets AREF pin as reference voltage for all A0-A5 pins
  pinMode(analogOutPin,OUTPUT); //set the analog pin to output
}

// the loop routine runs over and over again forever:
void loop()
{
  while(Serial.available()) {
```

```
rd = Serial.read();
if (rd == 'b') //check if begin command was sent from master
{
Serial.println('b'); //send begin handshake back to master
rd = 'a'; //reset rd so loop does not continue
// Create sum of 20 values for averaging the readings
analogWrite(analogOutPin, intensity); //turn on 3.3V power to analog out
delay(2000); //2s delay for light stability and to ensure master is ready
for (i=0; i<CNT; i++) //takes CNT # of concentration readings
{

  //Send each reading to master
  conc = analogRead(A0); //store current value to variable "conc"
  Serial.print('<'); //tells master that measurement data will follow
  Serial.print(conc); //print concentration to serial (to add/ext unit)
  Serial.println('>'); //tells master that the entire value has been sent
  delay(del); // delay to match samplerate --> 1/delay = Hz 1/(samplerate/1000)
}

Serial.println("<e>"); //tell master that all data has been sent (e for "end")

analogWrite(analogOutPin, 0); //turn off LED after sampling
} //end the sampling until add/ext requires more data
else //if the serial is not available (i.e. not > 0), then delay
{
delay(200); //short wait before rechecking for commands from master
}
}
}
```