Department of Electrical and Computer Engineering
Howard University

Cornell Cup Submission

Green Lighting
Final Report

Team members:
Ameer Baker
Chidi Ekeocha
Shamir Saddler
Ravindranath Jaglal
Isaac Collins

Instructor: Charles J. Kim

Table of Contents

Project Summary

The goal of this Green Lighting project is to implement a system that will enhance the work environment in a given room by regulating the intensity of the lights throughout the day, while minimizing energy costs. This is to address the energy costs associated with maintaining standard or optimal lighting conditions for a workspace. The system will consist of light sensors connected to low-end microcontrollers for each relevant sector of the room which link back to a Tunnel Creek board which will process the data, make decisions as to how the intensity of each light fixture should be varied to maintain lighting standards, or user preference, and command the instruction to the lighting control circuitry. In addition to a physical override circuit, the user will have access to a web interface via which the main system controls for each room can be manipulated, thus making it easier to manage the entire building's lighting from one terminal. The system will allow the users to experience appropriate lighting throughout the course of the day therefore creating a productive and enjoyable working environment while simultaneously saving energy cost for the respective institution. A prototype of this embedded system will be implemented and demonstrated to evaluate its effectiveness in workplace productive and energy cost saving.

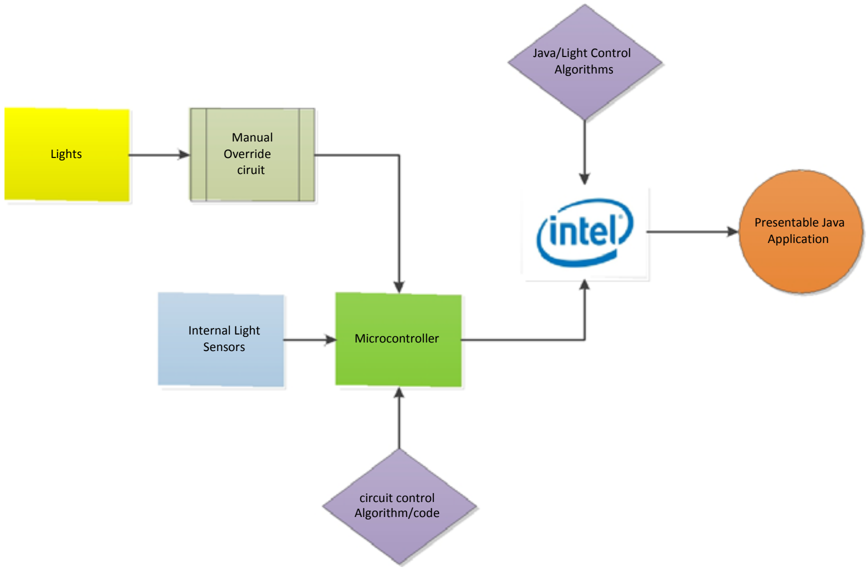
Challenge Definition

It is a well known fact that many major college campus buildings and office facilities use an abundance of energy throughout any given workday resulting in high energy cost. Ina standard office building or university classroom all the lights in a room tend to be on at maximum efficiency throughout the day when in reality, in a common situation where windows occupy the room, the intensity of the light doesn't need to be as high during the morning hours as it may be in the evening. As a complete solution to this challenge, we propose to solve this issue by implementing a system that can cut consumer power usage and promote a "green" environment by implementing a system that will regulate the light intensity inside the perimeter of a given room, to maintain lighting standards, or user preference in order to create a productive and enjoyable working environment while simultaneously saving energy cost for the respective institution. We feel this will be a worthwhile project to take up after we conducted research on the cost and power consumption of some of the offices on our school campus. We not only plan to demonstrate how we can save energy and reduce power consumption, but also we would like to display how much we can cut a University's operation cost annually by implementing this system in their school buildings. Furthermore we anticipate on this project being so effective that it will confidently set the standard on how university and business facilities regulate power and energy utilization. From a design standpoint, the system should be so discrete that the only people who are aware of its existence are the energy companies and the building maintenance personnel. From an implementation standpoint, the system will be able to read the intensity of light coming into a given room, compare the data to the light sensors inside the room, and fixate the amount of light being distributed throughout the room to maintain a constant level of light in the room without unnecessarily having all the light in the room on at maximum level. Finally from a consumer standpoint, a good solution will cut the respective institutions expense by providing a modest cut in energy bills every month which will add up overtime and within a few years of implementation, save the university thousands of dollars in energy costs. We strongly believe that when you can find ways to save money in operation costs, the advantages trickle down to the buildings frequent occupants. The benefits are endless when you are saving a significant amount of money in energy costs over a short period of time, giving the establishment more flexibility to invest in itself and its surrounding community.

Proposed Solution

The proposed solution to the light efficiency problem will comprise the application of all the available light energy control resources that available today, as mention in the current status of art above i.e. turning of lights, using efficient electric lights, and dimming the lights in the room space. This would be achieved using the sun's energy to complement the room's overall intensity at all times of day.

In addition, the overall system would be sensitive enough to respond to the differences in intensity experienced with daytime and nighttime; to area wise differences due to a passing cloud or obstacle; to closing the blinds in one part of the room. The consumers of this system will also be given the power to control the overall light intensity of the room, as well as area wise intensity via control switches locally, and remotely via computer programs on the internet or local area network. The major components and logic required to achieve this feat as mentioned above include;



- Light sensors across the perimeter of the entry points of sunlight to monitor and respond to the intensity of light outside at all times of day. Night time light sources may be strong enough to complement the light intensity of the room, but is not only dependent on the moon's positioning but also on other artificial light sources that may be present outside the building at the time.
- Light sensors across the vertices and edges of the room, or around the major areas of interest of the room/environment. This will yield the ability to maintain a uniform intensity across the room, as well as area wise control of light intensity within the room, by comparing the feeds from both sets of sensors and modifying the power feeds into the lights based on the algorithms we apply.
- Microcontroller kits to interface with the sensor packs (both inside and outside the room) and retrieve all the raw data from the sensors to be processed digitally by the Intel atom processor.
- The Intel processor, which will house a Linux or Windows operating system that will be the base upon which the applications will written to process the incoming data from the USB kits. This will also serve as the web server to host a web interface that can be used to control/model the entire room both locally and from remote locations.

- Base Intensity controller/switch that will provide the ability to control the default intensity of individual sectors in the room, as well as the default intensity of the entire room as a whole.

We are able to demonstrate the efficiency of our system by comparing a test environment outfitted with our light efficiency system to an identical test environment without the system in place. With this experiment we can see in real time how our system performs under controlled conditions and compare and analyze those findings using known power and efficiency calculations to prove the effectiveness of our project. In addition to meeting efficiency requirements, we have also been able to create a system that can be implemented in almost any office space and is so user friendly that anyone can learn to run the system within minutes. Because of its practicality, the user command interface is outfitted with security protocols to limit access to the system. The production of the system was done almost entirely with just the funds given by Intel making the system not only an environmental success, but also a profitable investment within its first year.

Some changes we made from the original design include the dimensions and construction of the test environment, the switch from the BS2 microcontroller to the Audrino board, and the use of LED bulbs instead of luminescent bulbs just to name a few. As far as the test environment goes we decided we would get better test results if we had two identical rooms one with the system implemented and another without the system in which to base our test results. Thus we found the dimensions of our electrical engineering lab and scaled it down in order to construct our two test boxes.

After getting deeper into coding the algorithm for the microcontroller we became aware that the BS2 did not have sufficient ports and that we would need to purchase further peripherals such as a multiplexer in order to expand the few ports the board had. While waiting for the multiplexers which we were at the time slowing production we turned to other alternatives and came across the Arduino board which turned out to be a relatively cheaper, more compact, and more compatible with circuit elements we had in place. There was plenty of sample code provided as well for the Arduino board cutting hours out of production and algorithm preparation. Originally we were prepared to use luminescent bulbs seeing that they are the most prominently used in offices and classrooms, however, we decided in an attempt to increase the efficiency of our system, that we would use a lower power consumption alternative which we found in the UV LED bulbs. In similar systems, the use of LED bulbs is common for this particular reason.

Finally, changes made in the category of user interface include deciding against the web-based command to an application control system. This decision was based on the premise that applets are easier to comprehend and are less prone to error, providing the user with a practical experience. The applet gives a linear output of system data in categories that are as in depth as consumption per sensor-light unit. We have still included the original manual override function, providing the consumer with the luxury of full system control.

Scenario Considerations

In addition to the manual control of the area wise light intensity in the room, some of the scenarios that were considered for the passive response of this light energy/efficiency control system include;

- **Response to Clouds passing by the air space of the building at every giving point in time:**

  Clouds are natural sources of shades that could affect the influence of outside light from the sun in a room. The system will respond by increasing and/or decreasing the light intensity based on the dynamics of the clouds as picked up by the sensors at the time.

Depending the on the size of the room, the cloud may only cast a shadow on some part of the room, or may be large enough to cast a shadow that covers the area of the room.

- **Response to Daytime/Nighttime transition:**

  Obviously, we get most of our sunlight during the day, and the least during the night. This transition between day and night is also not instantaneous. As noon approaches, the environment gets lit up more by the sun, and dims down as nighttime approaches. The exact times of day when the intensity of light outside is at its maximum, rate of change of light and darkness also varies all year round based on the earth's position in it revolutionary orbit. Across the progression of a day, the room will respond accordingly to the slow change (increase, and decrease) of the exterior light intensity at all times.

- **Response to miscellaneous obstacles interfering with the exterior light intensity:**

  Anything can cast a shadow on any given area at any point in time. The system should respond by increasing the affected areas accordingly.

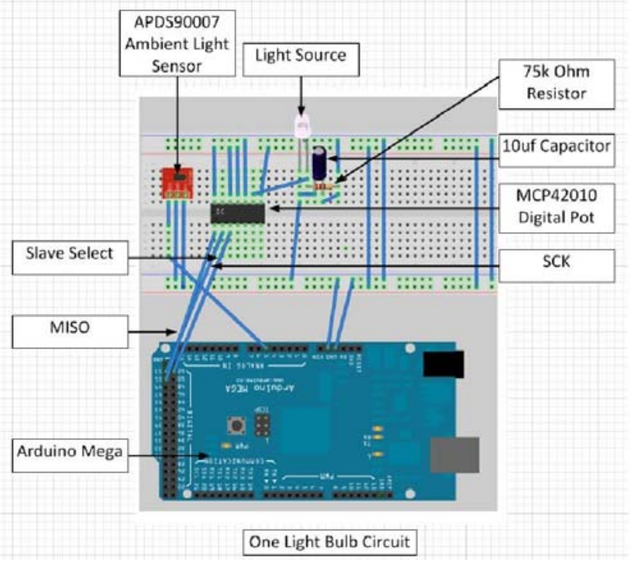
- **Response to closing windows/blinds;**

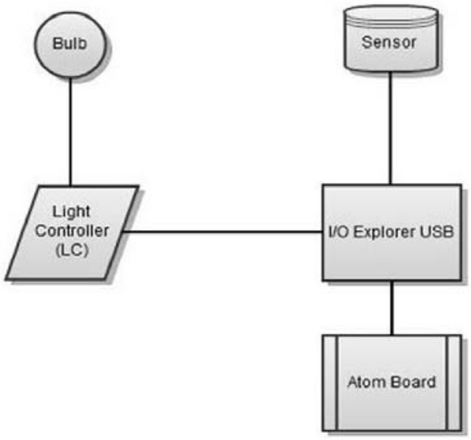
  The consumer will still have the ability of restricting vision into the room, or whatever reason one will have for closing the windows, without affecting the intensity of light in the room or the affected area at the time.

Technical Analysis of solution

The One Light Model*:*

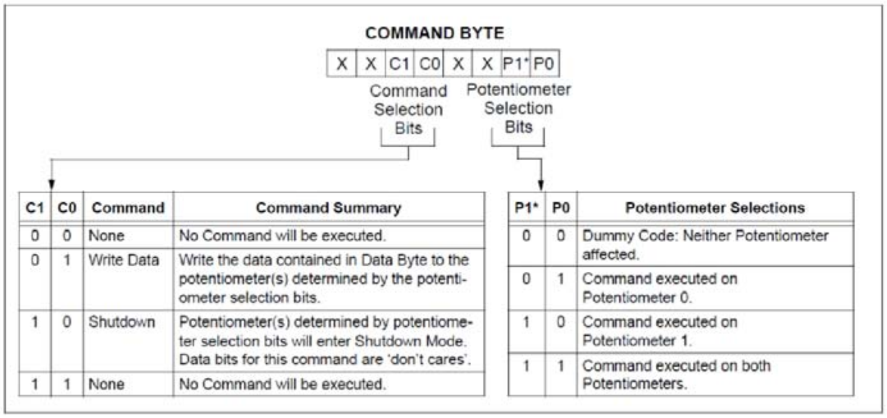
This model consists of one light bulb, one sensor, one resistor, and a corresponding capacitor and is the basis of our solution design. In actuality, the full circuit comprises repetitions of this circuit where the number of sensors and lights the circuit will have corresponds to the amount needed in the room. This miniature model also allows us to test individual components of the design, calibrate the voltage level of the sensors, and calibrate the response of the lights to the voltage readings we get from the sensors.





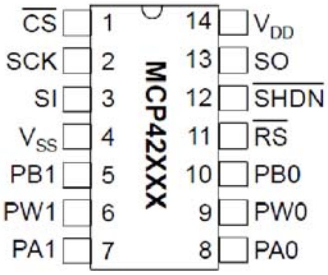
The MCP42XXX digital Potentiometer

The MCP42XXX devices are 256-position, digital potentiometers available in 10 kΩ,50 kΩ and 100 kΩ resistance versions. The MCP42XXX contains two independent channels in a 14-pin PDIP, SOIC or TSSOP package. The wiper position of the MCP42XXX varies linearly and is controlled via an industry-standard SPI interface. The device consume <1 µA during static operation. A software shutdown feature is provided that disconnects the "A" terminal from the resistor stack and simultaneously connects the wiper to the "B" terminal. In addition, the dual MCP42XXX has a SHDN pin that performs the same function in hardware. During shutdown mode, the contents of the wiper register can be changed and the potentiometer returns from shutdown to the new value. The wiper is reset to the mid-scale position (80h) upon power-up. The RS (reset) pin implements a hardware reset and also returns the wiper to mid-scale. The MCP42XXX SPI interface includes both the SI and SO pins, allowing daisy-chaining of multiple devices. Channel- to-channel resistance matching on the MCP42XXX varies by less than 1%. These devices operate from a single 2.7 - 5.5V supply and are specified over the extended and industrial temperature ranges.

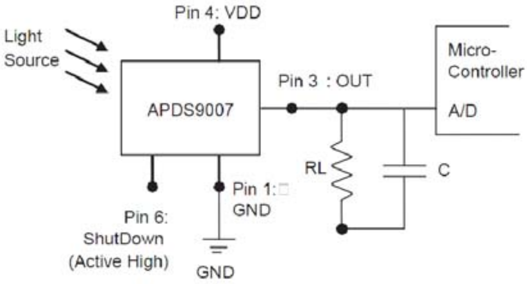


C1	C0	Command	Command Summary
0	0	None	No Command will be executed.
0	1	Write Data	Write the data contained in Data Byte to the potentiometer(s) determined by the potentiometer selection bits.
1	0	Shutdown	Potentiometer(s) determined by potentiometer selection bits will enter Shutdown Mode. Data bits for this command are 'don't cares'.
1	1	None	No Command will be executed.

P1*	P0	Potentiometer Selections
0	0	Dummy Code: Neither Potentiometer affected.
0	1	Command executed on Potentiometer 0.
1	0	Command executed on Potentiometer 1.
1	1	Command executed on both Potentiometers.



APDS90007 Light Sensors



The APDS-9007 is an analog current output ambient light photo sensor whose spectral response is close to the CIE standard photopic observer. APDS-9007 consists of a photodiode and an IC that performs amplification of the photodiode output signal and conversion to a logarithmic output current. APDS-9007 is able to produce a high gain photo current that can be converted to an output voltage via a standard value external load resistor. The magnitude of the output voltage, Vout is directly proportional to the photo current which is generated by the brightness of the light source shone on the photo sensor and the value of the load resistor used, RL. Increasing the brightness of the light source and/or the value of the load resistor will increase the magnitude of the output voltage. Selection of the load resistor RL will determine the amount of current-to-voltage conversion in the circuit. APDS-9007 allows a maximum saturation output voltage of (Vcc – 0.5V). Artificial light sources such as fluorescent lamps or in-candescent lamps produced ac noise with a frequency of 50/60Hz and 100Hz respectively. A capacitor of 10uF, which acts as a low-pass filter, is recommended to be added in parallel with the load resistor to filter out these interferences.

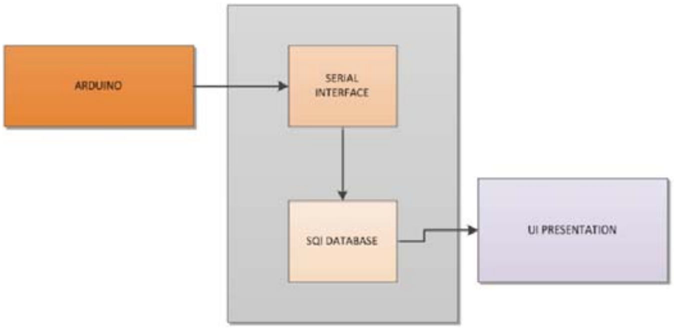
Arduino Mega 2560

The Arduino Mega 2560 is a microcontroller board based on the ATmega2560. It has 54 digital input/output pins (of which 14 can be used as PWM outputs), 16 analog inputs, 4 UARTs (hardware serial ports), a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; it is simply connected to a computer with a USB cable or powered by an AC-to-DC adapter or battery.

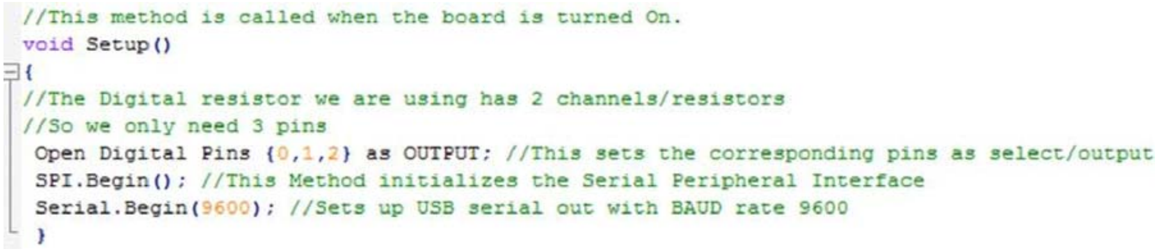
Arduino Mega 2560

Green Lighting Code Base:

This project code base consists of 3 distinct layers of code that are used to govern the behavior of the hardware as well as display the results on a UI level using JAVA. The Arduino layer picks up the readings from the hardware, and stores them on a database on the Atom Board which can be accessed remotely. The presentation program picks up these readings, and can also write back to the database. These changes to the database are picked up by the Serial controller interface, and written back to the digital resistors.

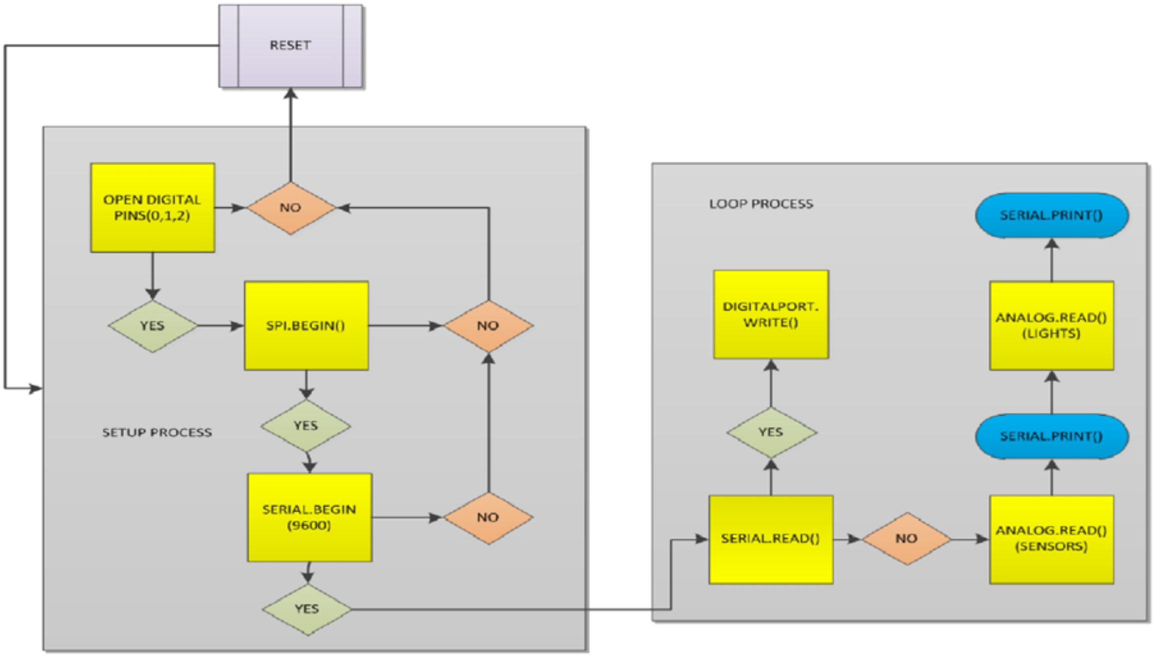


The first (Lowest) layer is the Arduino code, which is written using the Arduino IDE, in a language similar to JAVA. Its primary function is to take the Light and Sensor analog readings, and output them over Serial USB, as well as write back data to the Digital resistor through the SPI interface. Although not specified explicitly in the flow chart above, the Arduino code allows for incoming data from the Serial interface and this data are used to change the value of the Lights by writing them directly to the Digital resistors. The program consists of two main function, the setup () function, and loop () function which is run repeatedly while the Arduino board is on. The Setup() function simply initializes the necessary interfaces required by the program i.e. the digital pins, the SPI interface for communicating with the resistors, and the Serial connection which is initialized with a baud rate of 9600 bits/s.



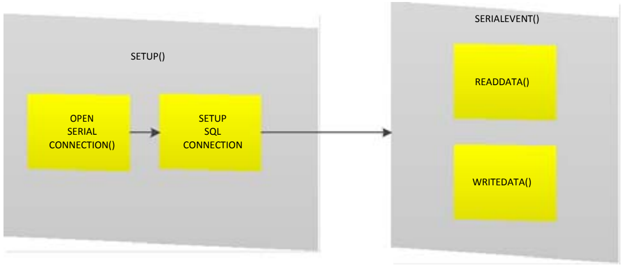
```
//This method is called when the board is turned On.
void Setup()
{
//The Digital resistor we are using has 2 channels/resistors
//So we only need 3 pins
 Open Digital Pins {0,1,2} as OUTPUT; //This sets the corresponding pins as select/output
 SPI.Begin(); //This Method initializes the Serial Peripheral Interface
 Serial.Begin(9600); //Sets up USB serial out with BAUD rate 9600
 }
```

The loop () function is designed to first read from the Serial for any incoming messages before writing the light and sensor values to it. The data read from serial is parsed and written directly to the digital pins which in turn control the resistors and the lights.

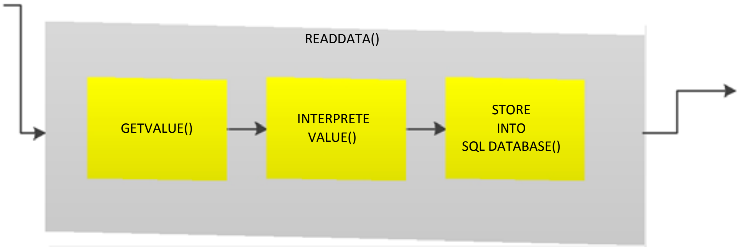


Flow Chart Showing the behavior of the Arduino program.

The Second layer of code is the Serial interface/Database controller which just like the Arduino code, comprises of two main functions; the Setup () and SerialEvent () functions. Just like the previous Setup () function, the Setup () function in this layer initializes the required interfaces. In this case they are USB serial interface and the MYSQL database connections. The SerialEvent () function is slightly similar to the loop () function. The main difference is that it is only called when there is a Serial transaction i.e. when there is data coming in through Serial, or when there is data to be written to Serial. The Serial/Database interface also contains code for the default behavior of the lights which is used by default, unless changed by the user.



Serial interface/Database controller Flow Chart



Sample of the helper functions in this interface

Product Performance Evaluation

Reliability:

The design method used to create our overall system and incorporate all necessary parts (layout, sensors, LEDs, microcontrollers, etc.) focused on how reliable we could make our final product. Specifically, if any sensor were to fail, the layout would ensure that no system failure would occur due to the presence and location of the remaining sensors. If any LED were to fail, the sensors would respond to this decrease in luminosity and properly notify the remaining LEDs of this decrease.

Potential Failure Modes:

The simplicity of the components being used ensures that any potential failure would not result in a human safety issue. That being said, if a failure were to occur within the system then the initial proposal to save money and be energy efficient would be undermined. In order to prevent this from occurring, we recognized which drastic failures could potentially occur, the likelihood of their occurrence, the likelihood of their causing a mishap, hazard, or system failure, and how to either properly mitigate them or minimize their effect on the system.

ATOM Failure: If the ATOM processor were to fail, there would be a complete system failure. To prevent this from occurring, regular maintenance would have to take place to ensure the correct outputs and values are being sent/stored/converted, as well as to prevent any physical damage. The highest priority would be given to this component due to its all-encompassing functionality and importance.

Arduino Failure: If the Arduino controller were to fail, the system will receive all of its information (input/output from the sensors and lighting) directly from the ATOM processor. This component's failure is not as high a priority as the main processor, but is still considered a priority in terms of things not to fail.

Sensor Failure: If any of the sensors in the system fail, the system will continue to run but with each remaining sensor requiring to cover a more broad area of room space. Since the layout requires seven (7) sensors, the design is such that if any one sensor were to fail, there would be no need to rearrange the remaining sensors to pick up this unaccounted for area.

Lighting Failure: If any of the lighting devices (LEDs) were to fail, the overall luminosity would likely decrease. In order to account for this change, the LUX of the remaining functioning lighting devices would have to be increased via the digital resistor and ATOM board's functionality.

Interface Failure: If the web interface were to fail there would have to be a manual override, meaning the device would not be controlled remotely, but would still function properly.

Alternative Designs/Future Modifications

As with every engineering design or problem there are always upgrades that can be made to the proposed solution. Some of these modifications will come as a result of the data acquired from the conducted test. In order to leverage the power of the Intel board, applications can be written to monitor other devices within the room using smart meters to check and report their power consumption. Since the consumer could control literally everything about his lights, he could as well have control over the other appliances in the room. Also, the design or arrangement of the light sensors may vary primarily due the size and shape of the room, as well as the positioning of the windows in the room. Finally the visual presentation of the data and control can also be extended to Mobile applications to give the consumer more flexibility with control. As we look back on the finished product, after tests have been run, adjustments have been made, and last minute preparations are underway, we allotted a decent amount of time to see where the system could use the most improvements and further, on what functionalities we could have included to make the end user experience even better.

One major element that we would have like to improve is our test environment. Ideally we would have liked to implement the system in our engineering lab where the room specifications met our requirements. However the problem we ran into was that we could not gain permission from university officials to tamper with the existing lighting arrangement. In order for us to achieve this feat we would need at least two months just to convince administration to let us conduct this experiment using campus equipment and then we would need another three months of trial and error to construct a feasible solution. We feel confident though, that with more time we could have successfully created a marketable product.

Another improvement that could be made is to increase efficiency of the existing systems power consumption. As it stands, the system must draw power to run the ATOM board and the Arduino board while simultaneously powering the digital resistors and sensors. Also, because the system is constantly checking for changes in light distribution and the sensors are always gathering information, more research needed to be done on whether we wasted energy at some milestone in the design process thus defeating the purpose of our proposed system to save the user energy. Furthermore, more research needs to go into finding out the average period of time it takes before the system becomes a profit to the consumer by means of saving in energy costs, the equivalent value of the system itself.

Demonstration at Intel-Cornell Cup

We plan to make a prototype office ranging between 4 - 5 feet long, 3 to 4 feet wide, and 4 to 5 feet high. Through this model we will demonstrate the impact outside light have on the light intensity present inside the prototype office. We will make sure that the light varies accordingly and that it can be controlled by a web interface and system settings changed based on whatever needed.

Performance measures

We begin the evaluation of this project by first creating a test room containing ‘n’ amounts of light fixtures which produce a fixed amount of light intensity based on the power being fed through the sources. Once the simulation criteria have been established we will factor in the estimated amount of time the lights are left on per day, including the power spikes associated with turning the lights off and on. In addition to this information we will also take into consideration the degradation factor of the light bulbs being used and conceive a method of calculating the estimated overall power consumption for a years’ time. Moving forward with these calculations, we can implement this knowledge in conjunction with power simulation software, in addition to surveys about the effectiveness of the lights, in order to attain a final conclusion on how much energy our system will save the user. In order to deem the green lighting project successful; in comparison to simulation data and surveys, the system should save at least 5% in energy costs as well as produce positive survey results. Comparisons will also be made to similar preexisting studies such as a field study conducted by the Institute of Research in Construction (IRC) in Canada which drew results from the long-term data of a single building using a particular lighting system. This lighting system was a workstation-specific, 3-lamp system with integral occupancy, light sensors and individual dimming controls. The three controls affected the light output of the two lamps directed downwards, while the single lamp directed upwards was always on during scheduled occupancy. (Galasiu, Newsham, Suvagau, & Sander, 2007)

Feasibility and Resources Available

Due to the proposed design of the solution, the selection of devices to be used, and the technical background represented from the members of the group, it is felt that implementing the Green Lighting project successfully is feasible. The timeline that was developed to maintain reasonable progress takes into account what objectives must be met, when they must be met by, and the amount of effort placed on each task. These criteria are essential to the management of the Green Lighting project while moving forward with the implementation, not only to keep structure, but also to evaluate how effective the group’s usage of time is. If any task is given low effort (on a scale of 1-7), in the future, any similar tasks will be reevaluated and either negated or assigned higher priority as a means to put forth more productivity. However, it was taken into account that there may be setbacks as far as arrival of materials, or any other uncontrollable factors, thus all proposed “Completion Dates” are tentative and may potentially be adjusted. There are several current technologies which use different methods of conserving energy, whether it is a system which automatically turns off unused devices after a certain allotment of time, light bulbs which are more energy efficient, or ability to control the intensity of a light in a room. These current technologies show that our proposed solution of not only controlling but also maintaining optimal lighting conditions as a means to drastically save energy is possible. The difference between the current technologies available and our proposed solution are the combination of different factors such as what the optimal level of lighting is for comfortable visibility, and the ability to constantly maintain this comfortable and efficient level.

- Special Resources

A scaled test environment would be the only special resource needed. This would ideally include a full-scale room where the final product may be tested and used. This environment can be a lab, a classroom, or any usable room that is available upon request.

References

Roberts, Gregory. "Natural Lighting Goes to the Head of Class in Today's Schools - Seattlepi.com." *Seattle News, Sports, Events, Entertainment | Seattlepi.com - Seattlepi.com*. Web. 12 Oct. 2011. <http://www.seattlepi.com/news/article/Natural-lighting-goes-to-the-head-of-class-in-1113875.php>.

"Student Success Includes Natural Lighting: Fluorescent Lights Can Negatively Impact the Learning Environment | Suite101.com." *Karen Stephenson | Suite101.com*. Web. 12 Oct. 2011. <http://karen-stephenson.suite101.com/student-success-includes-natural-lighting-a138865>.

"All About Microcontrollers | Code, Circuits, & Construction." *Tom Igoe*. Web. 17 Oct. 2011. <http://www.tigoe.net/pcomp/code/controllers/all-about-microcontrollers>.

"Building Maintenance - Lighting and Occupancy Sensors." *Welcome to Green California!*Web. 17 Oct. 2011. <http://www.green.ca.gov/EPP/building/sensors.htm>.

"Electric Lighting Controls | Whole Building Design Guide." *WBDG - The Whole Building Design Guide*. Web. 17 Oct. 2011. <http://www.wbdg.org/resources/electriclighting.php>.

Nilsson, James W., Susan A. Riedel. Electric Circuits. 8th ed. Upper Saddle River, NJ: Pearson Education, Inc., 2008.

Sedra, Adel S., Kennth C. Smith. Microelectronic Circuits. 6th ed. New York, NY: Oxford University Press, Inc., 2010.

Galasiu, A., Newsham, G., Suvagau, C., & Sander, D. (2007). *Energy saving lighting control systems for open-plan offices: a field study*. Leukos: National Research Council.

Appendix:

TimeLine			
Start Date	**Completed Date**	**Task**	**Deliverables**
			All team members have read all the rules and discussed them
			Faculty member commits to acting as an advisor for the team for the upcoming academic year
14-Sep	15-Sep	Review Intel-Cornell Cup Rules	Advisors have read all the rules and discussed them
14-Sep	12-Oct	Brainstorm Ideas	List of potential ideas for the Challenge, prioritized in terms of both exciting and risk
23-Sep	30-Sep	Perform initial search for already existent versions of ideas & whether ideas are feasible	Refined list of potential ideas for the Challenge, prioritized in terms of both exciting and risk
28-Sep	12-Oct	Review with Advisor for Idea Selection	Determine which idea to pursue further, and what would be a back-up idea should something come
14-Oct	14-Oct	Register Team	Register team on the Intel-Cornell Cup website
1-Oct	5-Oct	Research Other Solutions to Similar Challenges	Create a List of solutions to similar challenges with references where more information can be found; List pros and cons of each solution
1-Oct	11-Oct	Determine Key Use Cases	Written description of the way we intend the solution device to be used; Create a Context Diagram for the device; List of use case behavioral diagrams (matrices) for the most important ones
1-Oct	12-Oct	Determine Key Requirements	List of originating requirements for the solution device
28-Sep	12-Oct	Review challenge definition with Advisor	Approval of Challenge Solution research, Context Diagram, Behavior Diagrams, Solution Description, and Originating Requirements

26-Sep	2-Oct	Identify potential Faculty Reviewers	List of other faculty who have expertise in this area and might be willing to act as a reviewer for the team
26-Sep	13-Oct	Complete Application Write-Up for Challenge Definition	Challenge Definition write up is complete and reviewed by all team members
18-Sep	12-Oct	Determine solution concept	
18-Sep	12-Oct	Determine main subsystems for solution concept	List of main subsystems for the proposed solution
30-Sep	11-Oct	List initial feasibility issues & resource needs	List of potential feasibility issues and concern level in terms of at least uncertainty, time, and resource usage; Potential start of a FMEA matrix
9-Oct	11-Oct	Address feasibility issues & resource needs	Written description of solution, Updated Use Cases & Requirements
15-Oct	19-Oct	Functional Flow Diagram & Subsystems	
19-Oct	21-Oct	Create Functional Flow Diagram for Subsystem Connections	Functional Flow Diagram focusing on the inputs and outputs from each subsystem. New requirements for subsystem interfaces
25-Sep	15-Oct	Revisit initial feasibility issues & resource needs	Update list of potential feasibility issues and concern level in terms of at least uncertainty, time, and resource usage
15-Oct	21-Oct	Address new feasibility issues & resource needs	Update Functional Flow Diagram, Use Cases, & Requirements.
15-Oct	21-Oct	Determine Potential Equipment Needed	List of at least Expensive / Hard to Obtain Equipment Anticipated
15-Oct	21-Oct	Check for & Reserve Equipment Availability	Compare List with known lab equipment, Prepare questions for meeting with Advisor
12-Oct	14-Oct	Contact Potential Reviewing Faculty	All identified Potential Reviewers contacted with role expectations and to see if there is initial interest Meetings set up with interested reviewers for just after main Advisor meeting

8-Oct	11-Oct	Preparation for Review with Advisor	Updated Written description of Solution sent to advisor 48 hrs prior to meeting time, Update Functional Flow Diagram, Use Cases, & Requirements and be ready to discuss
12-Oct	12-Oct	Review concept with Advisor	
12-Oct	12-Oct	Ensure solution meets the challenge's needs	Approval or Noted Modifications for Functional Flow Diagrams, Subsystem Breakdowns, Written Descriptions, and all Uses Cases & Requirements
12-Oct	12-Oct	Ensure identified feasibility issues and resource needs are addressed	Approval or Noted Modifications for Functional Flow Diagrams, Subsystem Breakdowns, Written Descriptions, and all Uses Cases & Requirements
12-Oct	12-Oct	Identify any new feasibility issues and resource needs are addressed	Update list of potential feasibility issues and concern level in terms of at least uncertainty, time, and resource usage
12-Oct	14-Oct	Ensure team has / will be able to gain skills needed to create this solution	Approval from Advisor
12-Oct	14-Oct	Discuss potential Reviewers responses	Approval from Advisor
5-Oct	11-Oct	Improve ideas and create Final Proposed Solution	All Noted Modifications from Advisor Meeting are addressed
5-Oct	5-Oct	Final Advisor Check	Approval from Advisor
2-Oct	5-Oct	First Draft of Application Write-Up for Challenge Definition	
12-Oct	14-Oct	Confirm Faculty Reviewers for Project	Commitment from faculty that they will act as reviewers, List of tentative dates that may work good for reviewer(s) throughout the semester
16-Oct	19-Oct	Determine what functions your proposed solution performs that directly address the challenge needs	Matrix of Challenge's Needs (Rows) and the Key Functionality of Solution (Columns), where the cells show how strongly the functionality and the need are related

16-Oct	19-Oct	Determine questions you want answered that would show your solution is meeting (or not meeting) these needs well	List of questions that could answer whether a need is met
16-Oct	19-Oct	Determine ways to could obtain / measure answers to those questions (i.e. to form metrics)	List of potential tests that could be used to measuring the key functionalities' performance with respect to various challenge needs ; For each test, a list of output(s) (metrics) should be created
16-Oct	19-Oct	Determine what are the critical components of the proposed solution for meeting the challenge's needs	List of components that are critical to the key solution functionality identified earlier as being important to meeting the challenge need
16-Oct	19-Oct	Determine metrics/tests to show these components are operating well for our solution's important functionality	List potential tests for at least the core components as they relate to the key functionalities
16-Oct	19-Oct	Estimate what kind of test equipment, data, time, etc would be required	Test lists are updated to include potential special or harder to obtain materials or test equipment, or require a significant amount of time to perform
16-Oct	19-Oct	Review this in terms of feasibility	Refine test lists to make sure they are do-able within the time and resources available
16-Oct	19-Oct	Review the Performance Measures and the Resources needed to collect them with your Advisor	Approval or Noted Modifications for Performance Metrics, Tests and Test Equipment/Resources
16-Oct	19-Oct	Make any changes necessary	All Noted Modifications from Advisor Meeting are addressed
19-Oct	19-Oct	Final Advisor Check	Approval from Advisor
		First Draft of Application Write-Up for Performance Measures	

23-Oct	26-Oct	Determine Key Milestones by examining, subsystems, key functionality, & testing for the proposed solution & performance metrics	List of Potential Milestones
23-Oct	26-Oct	Determine what the main deliverables for each milestone will be	List of Expected outputs (deliverables) from each milestone
23-Oct	26-Oct	Determine main tasks & subtasks to achieve each milestone's deliverables	List of tasks and subtasks for each milestone
23-Oct	26-Oct	Split up Milestone's deliverables amongst tasks & subtasks where appropriate	Refer to parent task
23-Oct	26-Oct	Add new deliverables to tasks & subtasks as needed	Refer to parent task
23-Oct	26-Oct	Include testing tasks	Testing tasks added to milestones
		Estimate the correspondence between effort and time	Assigned an estimated hours to complete for each task; Leave as an equation so dates can be easily adjusted as the team gains experience in estimating their ability to perform various tasks
28-Oct	28-Oct	Determine Dependencies between Tasks	List of tasks (if any) that need to be performed before each task
28-Oct	28-Oct	Determine good times for Review Meetings	List of tentative times from advisor and review faculty that milestones
28-Oct	28-Oct	Create Schedule of Tasks	Schedule is determined; Spreadsheet will work but Gantt or PERT charts might be preferable; Note which steps (most likely those later in the process) that have the most uncertainty in their time estimates
28-Oct	28-Oct	Using dependencies, identify all potential starting tasks	List of potential starting tasks

28-Oct	29-Oct	Identify critical path	Identify which sequence of tasks has to happen as soon as possible in order to have entire project end as soon as possible
28-Oct	30-Oct	Using dependencies, determine which starting points or other tasks have significant slack in them	Recorded the slack for each task
Ongoing	Ongoing	Check against course, exams, & advisor schedules	Adjusted schedule to be realistic to "outside of project" time demands
2-Nov	ongoing	Determine which parts of the solution would have the greatest impact and/or have the most uncertainty surrounding them	Assigned an impact/importance rating & an uncertainty rating to at least the tasks with the greatest effort, most dependencies, and/or are most critical to key solution functionality
2-Nov	Ongoing	Modify solution to mitigate risks	Adjusted Functional Flow Diagram, Uses Cases, & Requirements to take into account considered risks
28-Oct	Ongoing	Add requirements to mitigate risks	Added Requirements needed to address the identified uncertainties and reduce risk
28-Oct	Ongoing	Determine Back-up plans for areas of greatest concern/uncertainty	List options for what could confidently be done instead should risks be realized
28-Oct		Update Importance & Uncertainty ratings to account for the existence of back-up plans	Updated impact/importance ratings & an uncertainty ratings to at least the tasks with the greatest effort, most dependencies, and/or are most critical to key solution functionality
28-Oct		Re-Examine the proposed schedule for the effect of these potential uncertainties	Identified points in schedule where uncertainty would be answered or risk would be lessened once passed; Identified tasks where uncertainty / risk would have the greatest effect
Ongoing	Ongoing	Examine schedule for worst case scenarios	Version of schedule where every activity takes the maximum expected time
Ongoing	Ongoing	Re-Examine the proposed schedule for potential bottlenecks	Adjusted schedule that minimizes bottlenecks and offers suggested alternative work that can be done at potential bottleneck points

30-Oct	31-Oct	Modify Proposed Solution that will be promised in the application to ensure that you can deliver what you promise	Adjusted Functional Flow Diagram, Uses Cases, Requirements, & Written solution description
30-Oct	30-Oct	Sketch out back-up plan schedule for most critical/uncertain elements	Estimate on effort for all Back-up ideas
31-Oct	31-Oct	Rank ideas that would like to be added to solution if time / resources allow	List of potential ideas ranked by estimates on benefit toward the challenge needs and effort to complete
31-Oct	ongoing	Re-check for reasonable options to mitigate the risk	Adjusted deliverables from tasks
	ongoing	Ensure there are back-up plans should the concerns be realized	Adjusted deliverables from tasks
15-Oct	17-Oct	Challenge Definition	Written by at least one team member; Reviewed by at least one other
15-Oct	17-Oct	Proposed Solution	Written by at least one team member; Reviewed by at least one other
15-Oct	17-Oct	Performance Measures	Written by at least one team member; Reviewed by at least one other
15-Oct	17-Oct	Timeline & Milestones	Written by at least one team member; Reviewed by at least one other
15-Oct	17-Oct	Feasibility & Resources Available	Written by at least one team member; Reviewed by at least one other
15-Oct	17-Oct	Potential Concerns & Alternative Plans	Written by at least one team member; Reviewed by at least one other
16-Oct	16-Oct	Present Application Materials	Meeting scheduled & conducted; Review sheets completed by all attendees; List of changes needed recorded
16-Oct	16-Oct	Make any needed changes	All changed sections also reviewed by another team member

17-Oct	24-Oct	Submit all materials on-line	Confirmation Email Received
		Reconfirm use of BS2 boards with ATOM chip	Confirmation in writing with tentative available dates and time
		Order ATOM Board and Arduino Board	Purchase order placed, Receipts turned into Advisors
unconfirmed	unconfirmed	Reconfirm use of EE lab for demonstrtion	Confirmation in writing with tentative available dates and time
		Reconfirm feasability of app creation for mobile use and interface	Confirmation in writing with tentative available dates and time
February	March	Determine current power consumption	Mapped route of streets and elevations for the test trip
February	March	Determine what new minimum consuption should be	Log of the demand along the city test trip, capable of being adjusted for various rider & cargo weights
February	March	Determine current energy costs	Mapped route of streets and elevations for the test trip
February	March	Determine price cut goal	Log of the demand along the hilly test trip, capable of being adjusted for various rider & cargo weights
February	March	Determine where sesors will be placed	List of cargo weights to be considered in the tests
February	March	Determine control system	List of rider weights to be considered in the tests
March	March	Determine Light Sensor Requirements	List of Wheel Rotations Sensor Requirements, Decision Matrix created for evaluating COTS Digital Resistorss (empty but evaluation criteria is established)
2-Apr	2-Apr	Find light Sensor(s) that meets/exceeds sensor requirements	Completed Decision matrix for the Digital Resistorss
March	April	Determine Digital Interface Requirements	List of requirements for each Digital Resistors to be connected to a microcontroller or similar input device
March	April	Determine tests for Light Sensors	List of tests and expected outcomes for the Digital Resistors (probably a short list)
March	April	Determine tests for micro proccessors	List of tests and expected outcomes for the microcontroller with regards to at least the

AMEER M. BAKER

Ameer.Baker@gmail.com

Current Address
Permanent Address
1919 3rd St NW 13910
Gilbert St.
Washington, DC 20001 Upper
Marlboro, MD
(240) 305 - 9770 (240)
305 - 9770

Education

HOWARD UNIVERSITY, WASHINGTON D.C.

College of Engineering, Architecture and Computer Science
Candidate for B.S. in Electrical Engineering June 2012
Major GPA: 2.91

Work Experience

June 2011 – July 2011 **Student Academic Enrichment Program**- Washington, DC

SAEP Peer Counselor

- Worked with a team of 40 / Responsible for 18 students each
- Attended classes with students and provided support to instructors
- Involved and empowered students so they get the most out of their experience.
- Facilitated daily recreation and study time, as well as impromptu activities to build a positive community

December 2010 – Present **Loose Screws Technologies**- Washington, DC

Co Chief Executive Officer

- Perform day-to-day administrative and technical operations.
- Fundraising through different outlets to support initial cost of operations.
- Handle day-to-day HR operations and consult with customers about company performance.
- Helpdesk and technical support alongside web development for select clientele.

June 2009 – July 2011 **Howard University Service Center**- Washington, D.C.

IT Practitioner

- Managed multiple technical/ mission critical calls daily and consistently met high service standards
- Provided networking/ desktop support
- Responsible for updating and maintaining 200+ workstations
- Trouble shooting, administration and support of Windows XP, Windows 7

Further Skills

- Knowledgeable in Multi-project coordination, Product Improvement, IT Strategy Development, Client relationship management, Cisco Packet Tracer, and troubleshooting & Issue resolution.
- Strong knowledge in Microsoft, digital logic design input/output and structured program development, K-Maps, Pspice, Binary and Decimal logic.
- Collaborated with project teams to deliver software and hardware projects efficiently and effectively.

Chidi Michael Ekeocha

2251 Sherman Avenue, Apt136E, Washington D.C 20001 | (202)560-9790| michael.ekeocha@gmail.com

OBJECTIVE

To acquire the valuable skills and knowledge presented to me by Howard University, to make a considerable difference to Electrical Engineering, the Engineering Community, and the World at large.

EDUCATION

Howard University, Washington D.C, USA Expected Date of Graduation: May 2012 Classification: Senior
Degree: B.S in Electrical Engineering. GPA: 3.58

WORK EXPERIENCE

Systems Administrator, Computer Learning and Design Center September 2008 to present

- Manage a team of operators that run checks on the network status in the labs.
- Created a Dynamic Network Map, to allow remote interaction with workstations on the network.
- Setup machines to PXE boot; making use of a Linux based sever, and thin clients.
- Wrote a Chat application in C# to allow for internal communication between users.
- Wrote and Active Directory tool in C#, to automate the account creation process for Operators.
- Designed Administrator application that allows for the remote execution of scripts.

Software Developer in Test, SQL Server: Microsoft® Internship program May 2011 to August 2011

- Designed and tested a repository to house DML and DDL objects that mimic the various combinations of a T-SQL statement's syntax.
- Extended a local tool that provided Programmatic Interfaces for Modeling Objects and Data (PIMOD), upon which the repository was built.
- Wrote Tests for SQL Server 2011 improvements, using the repository to demonstrate its functionality, as well as provide a larger coverage area with which to test this improvement.
- Designed a coverage measurement tool to measure the degree of coverage that a test has against the T-SQL surface area.

Software Developer, SQL Server: Explore Microsoft® Internship program May 2010 to August 2010

- Worked in a team of three, to develop a replacement to an obsolete Scripting tool called DSCRIPT.
- This tool, KIND. , allowed for scripting of WINDBG processes, which were slow and repetitive.
  It provided compile time execution of scripts, thus yielding a speed boost that DSCRIPT didn't have.
  In addition scripts which were previously in clear text were now replaced with encrypted Dll. Files.
- Wrote a tool to parse PDB dumps and return generated code that comprises the internal data structures of the dump.

Calculus Tutor, Capstone Institute (Cp^2) September 2009 to June 2010

- Teach/Tutor a Class of about 40+ Students.
- Provide supplementary teaching/Tutoring services via Connect Yard.
- Work with a team that develops new and effective ways of tutoring students.
- Learned group management and interaction processes, through Capstone seminars.

Supervisor, Candace Water, Nigeria August 2007 to July 2008

- Was responsible for the maintenance of the machines and generators
- Managed sales and recorded the operations of the company.

SKILLS

- **Software/Languages**: MS Office, Win Server 2003/2008, Xilinx, Modelsim, Pspice, AD bulk users/ Export, Acronis, Rocks, HTML, C++ ,Jscript, C# , T-SQL, JAVA, VHDL