Arizona State University
Ira A. Fulton Schools of Engineering
School of Electrical, Computer, and Energy Engineering
EEE 489: Senior Design Laboratory II

**To:**
Michael Kozicki (Professor)
Mehmet Balaban (Teaching Assistant)
Brian Skromme (Project Mentor) <skromme@asu.edu>

**From:**
Vanessa Martinez (Leader) <vimartin@asu.edu>
Paul Rayes (Software) <prayes@asu.edu>
Vu Truong (Software) <vu.truong@asu.edu>
Jossue Montes (Hardware) <montes.josh@gmail.com>

**Date:** 24 April 2015

**Project Title:** SLAM Air Quality Monitoring Robot: A Modern Implementation of Environmental Monitoring Systems

Enclosed is our final report for EEE 489 Senior Design Laboratory summarizing work done on a robot for monitoring cleanroom air particulate matter, using a modular add-on approach to potentially address multiple applications. This document describes the technical information, milestones, and budget. It also includes insights regarding the product and its role in a critical and fast-evolving industry.

# 1 TABLE OF CONTENTS

## 2 TABLE OF FIGURES

## 3 TABLE OF TABLES

## 4 EXECUTIVE SUMMARY

Cleanroom facilities must monitor the air quality in their clean rooms. However, research and interviews with clean room industry experts suggests that the methods of monitoring the quality of the air inside cleanrooms is not as effective as it could be. Hand-held sensors require a staff member to go throughout the cleanroom taking readings at various points, which removes a member of the production team from performing their normal duties and potentially disrupts cleanroom operations. Wall-mounted or otherwise stationary devices only collect air quality data within a limited area, necessitating the purchase of multiple units. This proposal is to design a small modular robot to solve these issues with an attractive selling price. The robot will navigate autonomously through a cleanroom, build a 2-dimensional map of the cleanroom environment, and present air quality measurements in a color-coded “heat map,” with additional details available if desired. This data is presented to the user on a web interface similar to many modern web applications. A modular approach means the robot could also have hardware add-on modules to potentially address other applications; this proposal focuses on air quality monitoring. A prototype has been built that is capable of monitoring air quality and displaying the heat map, but is not capable of autonomous navigation or operation in unknown environments. A scale model of the final product has been designed. The project is divided into four milestones: Milestone 1: Robot drives like an RC car (completed), Milestone 2: Display various sensor data on web interface (completed), Milestone 3: Hardware prototypes and test environment, and Milestone 4: Software. The cost of the robot has remained inexpensive at just under $700, not including the cost of the air quality sensing module.

## 5 INTRODUCTION

The existing air quality monitoring methods in clean rooms used by sensitive electronics manufacturers in semiconductor industry have areas where they can be improved. The two deficiencies this project will address are, air quality devices located at fixed locations and the hand-held requirement. The main problems with fixed air quality sensors are there has to be a large number of them in a clean room environment and a third party must be contracted to complete an air quality assessment. If not strategically placed, a large number of sensors can lead to ineffective readings of an environment. The sensors also have the potential to neglect areas at high risk for air contamination. The downside to hiring outside personnel to conduct air quality test is the cost. It is typically very expensive, which may lead to less frequent testing. This project proposes a more efficient and cost-effective way to address the surveillance inaccuracies and high costs semiconductor companies face in cleanroom air quality monitoring.

The industry-ready prototype of our product is affordable and fully automated. It will have a user-friendly setup procedure and will require little or no training to operate. It is a small modular robot capable of simultaneous localization and mapping (SLAM) of its environment. It will map out its environment onto a web interface, remain aware of its location, and monitor the air quality within the environment 2ft.-6ft above ground level. The platform will support various hardware add-ons. A robot charging dock or “home station” will allow for conducive cleanroom integration.

This prototype, however, will focus on air quality monitoring. The works-like prototype of the design will use an inexpensive sensor for proof of concept. The data obtained from the air quality sensor will be displayed on the web interface along with the map. This project proposes an autonomous air quality monitoring robot to address the costly air quality assessment challenges faced by the semiconductor industry.

This proposal discusses the following:

- Project Specifications
- Background research
- Technical details of the prototypes, air quality sensor, and software
- Analysis of ABET EC2000 criterion
- Project milestones and schedule
- Completed and planned tasks
- Available project resources
- Project budget

# 6 TECHNICAL NARRATIVE

This project is composed of two separate prototypes. The first "works-like prototype" does not look like the final product but is fully functional. The works-like prototype involves several distinct parts: The actual robot, the air quality sensor, the robot software, and the web interface. The robot contains the hardware, electronics, and sensors common to any application involving a SLAM robot. The air quality sensor measures the number of particles in the air and, for the works-like prototype, is an inexpensive commercially-available sensor. The robot software controls the various electronics and sensors. The web interface displays all relevant data to the user, and allows the user to give commands to the robot. The second "looks-like prototype" is 3D-printed scale model of what the final product could resemble. It is not functional and is purely for illustrative purposes.

## 6.1 Specifications

The current works-like prototype has the following features:

- Battery-operated
- Particle count sensor
- LCD
- 802.11n WiFi for wireless communication
- Web interface
- USB interface for setup and debugging

In addition, the current works-like prototype meets the specifications listed in Table 1.

The web interface has the following requirements for proper functionality:

- Windows, Linux, or Mac OS X
- Google Chrome 35+
- Node.js 0.12
- WiFi (802.11n with WPA2 security recommended)
- PlayStation 3 controller recommended for manual control
- 1280x720 or greater resolution

The USB debugging interface requires Ubuntu 14.04 or newer. It is not compatible with Mac OS X and has limited functionality in Windows.

Table 1: Works-like prototype specifications

| | **Minimum** | **Nominal** | **Maximum** |
|---|---|---|---|
| **Battery life (hours)** | 1.4 | 2.8 | 6.0 |
| **Detectable particle size[3] (um)** | 1.0 | | |
| **Detectable ISO classes[3]** | 3 | | 9 |

| | | | |
|---|---|---|---|
| **Detectable particle count[3] [4] (particles/m$^3$)** | 0 | | 98M |
| **Sensor measurement period [4] (seconds)** | | 30 | |
| **Particle count % noise[3]** | | 25 | |
| **Environment size[1]** | | | 120 x 120 cm |
| **Speed** | 0 | TBD | TBD |
| **Distance from two nearest landmarks[2] (cm)** | 15 | | 600 |
| **LIDAR range [5] (cm)** | 15 | | 600 |
| **Operating temperature[4] (ºC)** | 0 | | 45 |

Notes:
1. The environment size limitation was arbitrarily chosen to be the size of the current test environment, and is not due to any software or hardware limitations.
2. The localization algorithm requires that the robot is within sight of two landmarks. These limits are determined by the range of the LIDAR.
3. Particle count specifications are determined by the air quality sensor used, and is not a limitation of any other hardware or software. Determining the ISO class of a cleanroom is not possible with a single sensor; this specification is the range of classes that allow for particles of 1 um or larger [1] which is the range detectable by the current sensor [4]. Noise was determined experimentally.
4. The operating temperature is determined by the limitations of all internal components. The most restrictive of these is the air quality sensor [4].

See Appendix H for battery life calculations.

## 6.2 Air Quality

### *6.2.1 Market Research*

Organizations with a vested interest in airborne particle surveillance were contacted, as they are the foremost prospective clients if the robot ever enters a salable stage. Airborne particle surveillance represents an unavoidable operating expense for such organizations, since without bare-minimum contamination levels they must halt production, perhaps indefinitely.

Of the four organizations contacted, one is a research-driven facility, while the remaining three are business-driven. A research-oriented facility is usually part of an academic organization and used for educational purposes, while a commercial facility is typically a business that manufactures products for a profit. In either case, airborne particle surveillance takes place in a cleanroom environment, where sensitive electronics are manufactured by highly trained staff.

The key distinction is that an educational research facility allocates far less resources (such as capital) into its cleanroom and usually has a less-impressive cleanliness rating. The commercial facility invests far more into cleanroom maintenance and air testing equipment - the quality and reputation of its products (and indeed, of the entire business) is determined by an industry-standard air cleanliness rating. Since they operate for-profit, commercial facilities/organizations are the foremost client base for the robot, since they will depend far greater on reliable air testing and are capable of investing more capital.

The most recurring concerns or mentions and proposed solution are as follows:

a. *"The staff might trip over it due to its extremely small size."*
   This concern was brought up due to the size of the current looks-like prototype. The final product will have a chassis that is 2 feet long by 1.5 feet wide and stands 2.5 feet tall. It will be equipped with two flashing yellow alert LEDs that span the length of the left and right sides of the robot. Additionally, a (non-alarming) siren will sound periodically when the robot is nearby and in surveillance mode. Together, these precautions should give ample notice to any on-site staff that the robot is in the vicinity.
b. *"Airborne particle surveillance must not be restricted to just a few inches off the floor."*
   This concern was brought up due to the current placement of the air quality sensor on the works-like prototype and in earlier drafts of the looks-like prototype. The looks-like prototype has since been fitted with an extendable/retractable vertical arm that extends 4 feet upward from the body. Combined with the wheels, the chassis, the vertical arm, and the sensor itself, air testing can take place anywhere from 2’5” to 7’9” off the ground.
c. "*The wheels and motors might generate dust on their own, making true particle detection a lost cause."*
   Per this feedback, the motors in the final product will be sealed and located within the chassis. The LIDAR may be sealed in transparent polycarbonate.

### *6.2.2 Class Rating*

Consumer research, which involved scheduling field visits to several facilities, was conducted and analyzed extensively throughout the course of the project. All cleanroom facilities are issued an industry-standard Class rating that reflects the purity of their cleanroom environment(s). The Class rating is determined based on how many contaminant particles are detected in a set volume; for example, how many actual particulates per cubic inch. The usual culprits that contaminate a cleanroom environment are limited very small dirt or debris particles that measure on the order of tenths of micrometers in size.

Devices that test the air quality are priced according to their resolution, or the size of the smallest particle that can be recorded. Accordingly, devices with large particle resolutions cannot be used to issue high class ratings to a facility, since it is incapable of detecting very small or fine debris. The maximum environmental particle count requirement of a given Class rating is exacting[1]; however, the relationship between a desired Class rating and the required device resolution is relatively straightforward: i) Class 1000 ratings can only be issued using a device resolution of 0.5 microns or lower, ii) Class 100 ratings require a device resolution no larger than 0.3 micron, and iii) Class 1 ratings require 0.1 micron resolution device. The current SLAM robot will be equipped with a 1 micron resolution air sensor and all testing, simulation, and data-collecting will take place with this resolution in mind. Since the SLAM robot is intended to be modular by design, the end product can be custom tailored to the user's needs and can be outfitted with a sensors of any resolution.

## 6.3 Robot Operation

In the ideal product the robot will operate in four modes: Explore, Map (Autonomous), Map (Manual), Monitor, and Home. The current works-like prototype supports only the Map (Manual) mode.

### *6.3.1 Explore*

In the Explore mode the robot will explore a new and unknown environment. It will leave its charging dock to begin to map and record a 2-dimensional view of obstacles (walls, machinery, cabinets, etc) in the entirety of a cleanroom. As it roams the environment it will identify walls and corners and save their locations, to be used as landmarks for localization. It will also perform localization as it explores to keep track of where it is and where identified landmarks are. A human operator will be required in this mode to ensure the map is being built correctly, and to help the robot identify locations it has already visited. The map of the environment will be displayed on the web interface as it is being generated. After the entire cleanroom is mapped the operator can specify which locations are more or less important for air testing, which locations it should not enter, and in which locations WiFi should be disabled. It will then calculate the best path to scan the entire facility in the least amount of time and with minimal disturbance to cleanroom personnel.

### *6.3.2 Map (Autonomous)*

In this mode, the robot will use the map created in Explore mode to conduct air quality monitoring of the cleanroom. It will traverse the calculated best path, monitoring air particulate count using the air quality sensing module. The data collected from the module will be wirelessly transmitted to a web interface and displayed onto the 2-D map of the environment in the associated locations. It will also intelligently determine when an area should be re-tested, such as if the measured particle count is over a certain threshold defined by the user. If the contamination levels are indeed unacceptably high, the robot can send an alert to on-hand staff to investigate.

#### *6.3.3 Map (Manual)*

In this mode, the robot is directly controlled by a human operator, and performs air quality monitoring as in the Map (Autonomous) mode. Air data may be gathered in real time or collected at the end of a run.

#### *6.3.4 Monitor*

In this mode the robot remains stationary and plots air quality in a single location as it changes over time. This is useful if the particle count in a location measured in Map mode is over a certain threshold and additional monitoring is desired to rule out false positives or normal variations in air quality levels.

#### *6.3.5 Home*

In the Home mode the robot will return to its charging dock by the shortest path that will minimally disturb cleanroom personnel. While in Map (Autonomous) or Explore mode it will enter this mode automatically once the battery charge is low. Once it is fully charged it will automatically resume its previous mode. An operator may also command it to enter this mode, in which case it will not automatically resume its previous mode.

### 6.4 Robot Hardware

All hardware on the robot connects to the BeagleBone Black through a custom circuit board. A block diagram of this board is shown in Figure 1**.** The schematic and layout are in Appendix A. In this diagram the red lines indicate power flow and black lines indicate data flow. Dotted lines indicate a component is not in use in the current works-like prototype. The pins used to connect each component to the BeagleBone Black are shown in Appendix B. The first version of this board was on a prototyping board, which contains all hardware shown in the diagram with the exception of the switching regulator.

A BeagleBone Black (BBB) running Ubuntu is the main computer, and connects directly to all the hardware. All the electronics are powered by a 2-cell 7.4V Lithium Ion battery, either directly, through a 5V regulator, or through a 3.3V regulator internal to the BBB. There are two batteries for increased battery life. The current prototype uses a linear regulator; a switching regulator is also present but is not fully operational.  A power LED indicates whether the robot is powered on. The connection protocol used for each device is also shown. The connection to the user's computer is over standard WiFi, and thus requires a WiFi network to be set up. This will be the case in many, but not all, environments. The WiFi adapter is connected to the BeagleBone Black through a USB hub to maximize the available power to the adapter. The motor driver controls the speed and direction of the motors. The LCD displays basic status information such as the IP address assigned to the robot. All hardware except the switching regulator is fully assembled.

There are two optical encoders on each wheel for a total of four sensors, which measure the speed and direction of rotation of the wheels. The third wheel adds stability. The inertial measurement unit (IMU) measures acceleration, angular velocity, and magnetic fields, which are used to determine the robot's relative motion. These two components are functional but are not used in the current prototype. A webcam was present in the initial prototype but was removed as few industry representatives found appeal in a livestream of the robot's path.

The LIDAR includes both a laser rangefinder and motor, and measures the distance to obstacles 360° around the robot approximately twice times per second [5]. This is used to detect walls and corners in the environment for localization. This laser is rated at Class 1, indicating it is safe for use near humans [5].

The air quality sensor is the Shinyei PPD42NS, a simple, commercially available sensor that measures airborne particle count and reports a value every 30 seconds [4]. It lacks both the high resolution and sophisticated data collection/delivery desired by industry experts consulted, but it is functional and integrates well with the works-like prototype.

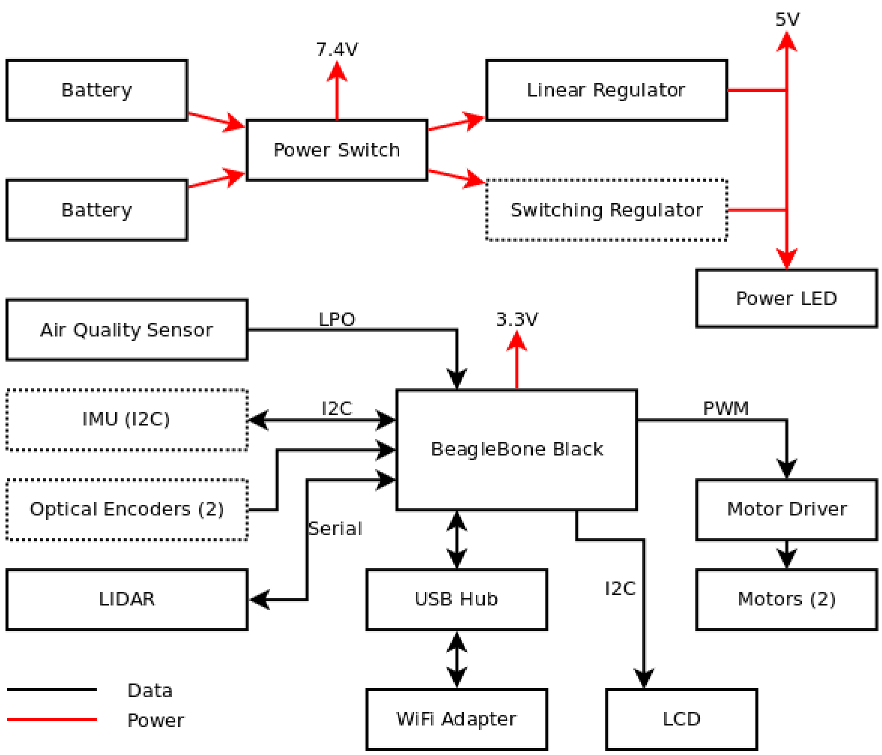



Figure 1: System block diagram of robot circuit board

## 6.5 Software

The software is divided into two pieces: The robot software, and the web interface. The robot software performs the majority of the tasks, including hardware communication and localization. The web interface displays the robot's status, selected mode of operation, the map of the environment, sensor data, and allows the user to control the robot.

The robot software is written in Javascript, which was chosen because it is the same language the web interface is required to use, it has a large number of existing modules, and its asynchronous event-based design is ideal for robotics applications. It is composed of several modules, which are shown in Figure 2 along with with how they work together. The web interface software is organized in a similar fashion. These modules communicate with each other using one-way events, similar to hardware interrupts. The current software contains all modules shown in the figure with the exception of the Pose module, which is still being built. Modules labeled "Hardware" communicate directly with external hardware but perform no calculations on the data received; modules labeled "Fixtures" contain pre-known data; and modules labeled "Stores" compute and store data from other modules.

A final product would have add-on software modules corresponding with add-on hardware modules, such as the air quality sensor, and would have built-in modules for mapping and decision-making.

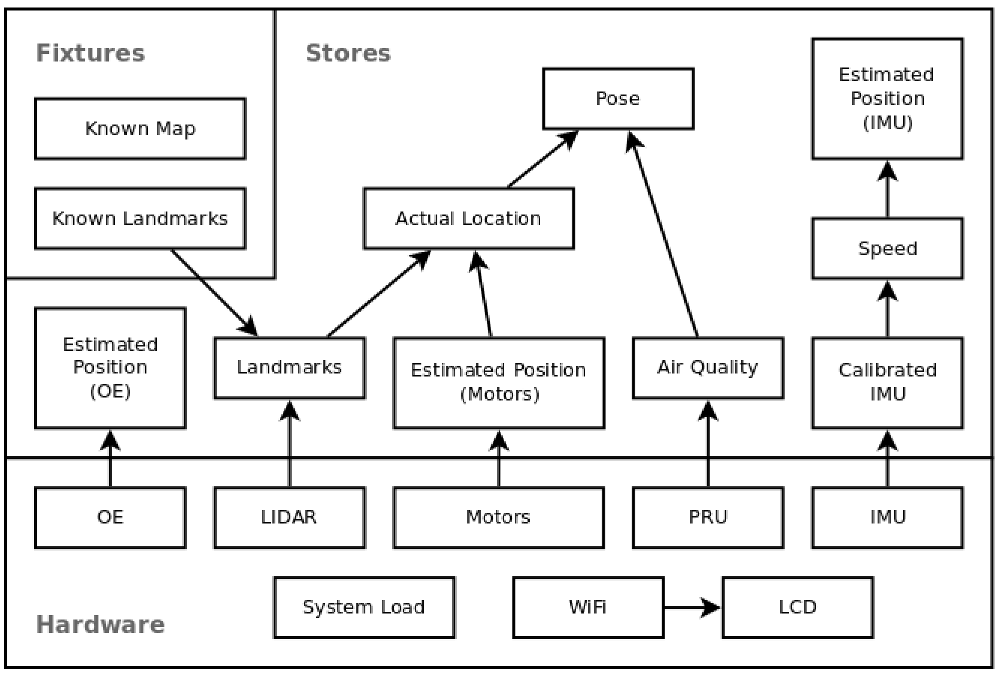


Figure 2: System block diagram of software modules

## 6.6 Looks-Like Prototype

The looks-like prototype was designed in Autodesk Inventor 2015 and is shown in Figure 3. After thoroughly reviewing the concerns and recommendations of cleanroom experts, several concept designs were drafted and fabricated into a 3D model. Most of the underlying circuitry and electrical components will be sealed away inside the robot's frame. The parts seen on the exterior are the LIDAR, the air quality sensor itself, and a pair of yellow LEDs to alert nearby staff. The LEDs were decided on after some industry representatives raised concerns over staff being inadequately warned of the robot being nearby. The sizes of many parts were arbitrarily chosen but satisfy the concerns raised by cleanroom representatives. Most importantly, the 3D looks-like prototype also takes into account the concern of whether particle surveillance will be limited to a single height - the extendable arm can raise the air sensor over twice the height of the robot to monitor air from 2’5” to 7’9”. All the dimensions (lengths, widths, heights) assigned to the 3D model adhere to the dimensions of the ideal final product. All dimensions of the looks-like prototype were scaled down to 1/10 their original value to reduce printing time and costs.

Figure 3: 3D render of looks-like prototype

## 6.7 Comparison of Initial Prototype, Current Prototype, and Ideal Product

The prototype initially proposed during the Fall of 2014 would be capable of localization and mapping in unknown environments and would display air quality as a color-coded heat map superimposed over the environment map. The current prototype only supports operation in pre-known environments. This change was made due to limited time, and because it was deemed less important for the prototype, as it is not unique to this project. An earlier version of the prototype also had a functioning webcam, which was removed due to a lack of USB bandwidth to support it and limited interest from cleanroom experts. The ideal final product would be cleanroom compatible, support add-on modules, and support multiple modes of operation. See Appendix C for a table detailing these differences.

## 6.8 ABET EC2000 Criterion

Per ABET EC2000[2], the project was designed with a sense of ethics and responsibility in mind. To this end, the following considerations were taken into account throughout the project.

### *6.8.1 Economic*

Depending on the specific needs of the consumer, air testing devices can cost from $1500 (on the low-end, educational facility side) to as far as $20,000+ (large-scale commercial). The consumer allocates further resources towards maintenance/repair costs, coupled with staff training in the case of sophisticated equipment. Evidently, reliable, accurate air testing is responsible for a significant chunk of the side costs to the consumer. The SLAM robot is designed with the end user's expenses in mind. Its creators believe that its unique assembly and novel method of operation can significantly reduce side costs for the consumer, allowing the consumer to allocate a larger portion of resources towards main production.

#### *6.8.2 Sustainability*

A small, efficient, economical 7.4V lithium ion battery powers the robot when it is not plugged in via USB to a computer. Plugging in enables battery charging, and the battery can of course be replaced at the end of its lifespan. So long as the robot is properly cared for, the lifetime of an individual unit is comparable to that of most consumer electronics.

#### *6.8.3 Manufacturability*

The SLAM robot's workhorse is the small BeagleBone Black (BBB) single-board computer that serves as the main controller. It, along with the ensemble of electronic components mentioned earlier are assembled together through basic electrical soldering. The coding and software that drives the robot is completely open source (non-proprietary) and the electronic components are extremely cheap to buy in bulk; unfortunately, the BBB is not available for purchase in large quantities. This is a non-issue, however, since other single-board computers can be used – the BBB simply presented convenience. As a final note, all the components (including the BBB) can be recycled, provided that a non-lead based electrical solder is used – the current version uses lead.

#### *6.8.4 Health and Safety*

The LIDAR component of the robot uses a laser to map the environment around it. There is potential for damage to human retinas if the user looks directly into this component. Since the robot is intended for no human intervention, it is unlikely that this is a threat. Also, the laser is emitted in brief, spaced-apart pulses so it isn't firing continuously, making even intentional contact with retinas exceedingly difficult. Additionally, there is a common concern from industry experts that the robot may accidentally bump into researchers/workers during its normal mode of operation. With sufficient coding this should not be a concern, as the autonomous model should not have difficulty recognizing objects in its path and altering its trajectory. Even at top speed, an impact with a human body could not possibly result in any serious injury. An impact at top speed might have just enough momentum to knock over a full glass of water, for example.

# 7 MANAGEMENT PLAN

The design and construction of the two prototypes was divided into four milestones. Three of the four milestones have been met; the fourth requires completing the software for the final demonstration. The primary resources used in the development of these prototypes are TechShop Chandler, time spent by the team members, and expertise of industry experts. The cost of the parts in the works-like prototype is slightly under $700; approximately $900 was spent on parts due to replacements and spare parts. The cost of the looks-like prototype is approximately $5 for material and seven hours to 3D print.

## 7.1 Tasks and Milestones

The milestones with their tasks are listed below, with the percent completed, start date, and finish date, sorted by finish date. If the percent completed is 100% then the finish date is the actual date, otherwise it is the scheduled date. See the Gantt chart in Appendix I for all scheduled tasks.

- Ongoing tasks: Research, planning, and reporting
  - Research applications (100% Sept 5 - Sept 30)
  - Plan tasks, hardware, software, and goals (100% Sept 5 - Dec 18)
  - Research air quality requirements (81% Sept 5 - Feb 24)
    - Remaining: additional patent research
- Milestone 1: Robot drives like an RC car
  - Basic robot hardware and overhead (100% Summer - Sept 12)
    - Test hardware connection to BBB. Develop code for web interface to display desired and actual speed and direction. Develop Arduino code for motor drivers.
  - Robot drives like an RC car (100% Sept 12 - Nov. 20)
    - Hardware configuration of motors to BBB. Set motor speeds. Use gamepad to drive robot. Display feedback on web interface.
- Milestone 2: Sensors
  - Display data from optical encoders, IMU (100% Sept 30 - Dec. 16)
    - Optical encoders: Read the speed of the wheel from optical encoder. Convert data to RPMs. Use an interrupt instead of timer.
    - IMU: Obtain orientation and acceleration. Display orientation and acceleration.
  - Display LIDAR data (100% Oct 2 - Apr 3)
    - Purchase and test LIDAR
    - Obtain and display environment data in a 2-D plane
  - Display air quality data (100% Nov 12 - Apr 10)
    - Purchase and test air quality sensor
    - Obtain and display air quality data
- Milestone 3: Hardware prototypes and test environment
  - Looks-like prototype (93% Feb 17 - Apr 27)
    - Determine requirements, design, and 3D print
    - Remaining: Assembly 3D printed parts
  - Custom PCB (100% Jan 26 - Apr 13)
    - Research, design, fabricate, assemble, and test circuit board.
  - Test Environment (100% Dec 19 - Feb 3)

        - Decide on materials, and design and build environment.
- Milestone 4: Software
    - Calculate & display robot speed and orientation (100% Nov 26 - Feb 25)
        - Develop code to convert RPMs from optical encoders to speed.
        - Develop code to use x, y, and z acceleration data from IMU to calculate speed.
        - Calculate orientation from IMU
    - Calculate & display robot position (95% Feb 18 - Apr 27)
        - Calculate estimated position from robot speed
        - Calculate position from estimate, LIDAR data, and existing knowledge of environment
        - Remaining: Complete testing and troubleshooting of code
    - Display map with robot, environment, and A.Q. data (100% Feb 3 - April 27)
        - Display map on web interface
        - Display robot location
        - Display air quality heat map
- Optional/low-priority tasks
    - Display debugging data and robot status on LCD (100% Sept 12 - Apr 7)
    - Display streaming video on web interface (0%) - removed

### 7.2 Justification of task delays from original proposal

The timeline has changed significantly from the original proposal. The primary changes and their justifications are listed below.

Milestone dates: The original proposal had significant time spans between each milestone completion date. This was in anticipation of working on each milestone individually. However, due to team member availability and purchasing times of some hardware, milestones 2, 3, and 4 were worked on simultaneously, resulting in delayed completion times of milestones 2 and 3.

LIDAR: Purchasing the LIDAR took significantly longer than expected. This delay was significant and impacted many other tasks, however was not controllable.

AQ Sensor: This task was temporarily put on hold to focus on development of the web interface, as that was determined to be a more difficult and time-consuming task.

Looks-Like Prototype: This task was postponed to complete additional research. No other tasks depend on this and thus this delay had no impact.

Custom PCB: This task could not be completed until testing of all hardware, including the LIDAR, had completed. No other tasks depend on this as a prototype board was used instead and thus this delay had no impact.

Robot position: This task was delayed due to the LIDAR, and the algorithm being more complex than anticipated.

Map display: This task was started early to compensate for delays in other tasks.

## 7.3 Resources

### *7.3.1 Facilities*

Facilities available for the project are TechShop Chandler; ASU libraries, study rooms, and conference rooms; as well as team members' personal homes.

### *7.3.2 Capabilities*

Josh has extremely relevant lab experience in a cleanroom facility at ASU's CSSER cleanroom. He was responsible for conducting market research through interviews (some on-site) and collecting product feedback from industry. He also helps handle many of the hardware aspects of the project, including the wiring and electrical soldering of the SLAMbot's underlying circuitry, designing the physical occupation grid, and designing and 3D-printing the looks-like prototype.

Paul is primarily in charge of the software design of the robot and web interface. Paul worked on an autonomous robot in the past, and thus has experience with the types of problems involved in robotics. He has significant software experience in the languages used in the project, including web interface design. He also has some knowledge working with CAD tools to design both circuit boards and mechanical parts.

Vu is primarily focusing on the software designs of the robot, his role includes: writing code for the robot to perform various functions, researching, and debugging code errors. His knowledge in programming will be used to configure and port components to the BBB. He will manipulate data from both the IMU and LIDAR to assist in the code which in turn will be used in determining the location of the robot. Whereas, his researching and debugging skills are used for troubleshooting obstacles that show up upon coding.

Vanessa is the team lead of the SLAM Air Quality Monitoring Robot team. Her primary responsibility is in scheduling, organizing, and leading team and team-advisor meetings. She develops the content for the written and presentation aspects of the project. She handles team conflicts and concerns. She focuses on hardware testing and verification of electrical connections of components. She also supports the team in research.

### *7.3.3 Expertise*

- Industry cleanrooms: Freescale Semiconductor, Particle Measuring Systems
- Research cleanrooms: ASU CSSER and ASU MTW cleanrooms
- Individuals: Michael Kozicki, Arthur Handugan and Stefan Myhajlenko (ASU CSSER), Tim Poet & Steven Salatino (Freescale), Mark Strnad (MTW), Rick Sparber (South Mountain Community College)

## 7.4 Budget

### *7.4.1 Equipment*

No equipment purchases are necessary for this project, as all necessary equipment is available for use at TechShop Chandler. Some equipment used is a soldering iron, 3D printer, multimeter, oscilloscope, logic analyzer, laser cutter, and other miscellaneous tools (screwdrivers, wire cutters, etc).

#### *7.4.2 Robot Works-Like Prototype Supplies*

The main hardware and electronics required for the works-like prototype are listed in Table 2. Some additional parts were also ordered for spares, replacements, testing before deciding on which part would work best, or were required for a previous version of the prototype. All parts purchased for the works-like prototype are listed in Appendix D.

The LIDAR costs over $100, however for this project it is the best way to accomplish environmental mapping. Infrared sensors do not provide enough accuracy for the project, and ultrasonic sensors cost nearly as much as a LIDAR while still not providing as much accuracy and requiring additional code to use. The LIDAR is both highly accurate and easier to use. Most competing LIDARs with 360° sensing cost significantly more, making this one a good price.

The air quality sensor in the final product will need to be more accurate than this one. For a class 100 cleanroom, that will be between $300-500 for a 0.3um sensor; existing solutions for cleanrooms cost over $1k. For a class 1 cleanroom, that will be between $5k-10k for a 0.1um sensor; existing solutions for cleanrooms cost approximately $20k. For the initial works-like prototype the $15 sensor with 1um resolution will suffice, and is accurate enough for a class 1000 cleanroom[1].

Table 2: Main works-like prototype parts

| **Item** | **Vendor** | **Price/each** | **Description** |
|---|---|---:|---|
| Chassis and Motors | DFRobot | 34.35 | Forms the mechanical base of the robot with two motors and a caster |
| BeagleBone Black | Amazon | 55 | Central single-board computer |
| Proto Cape | SparkFun | 9.95 | Prototype for custom circuit board |
| LIDAR | RobotShop | 398.99 | Environmental mapping |
| Crius CO-16 OLED LCD | eBay | 7.18 | Displays debug data to user |
| Dust Particle Sensor | Seeed Studio | 15.90 | Detects airborne particles |
| Motor Driver | SparkFun | 8.95 | Powers the two motors |
| Battery | BatterySpace | 14.95 | x2 |
| Battery Charger | BatterySpace | 19.95 | |

| | | |
|---|---|---|
| Other parts listed in Appendix D | 244.71 | |
| Miscellaneous additional supplies and small electronics already on hand | | |
| **Total** | **824.88** | |

*7.4.3 Robot Looks-Like Prototype Supplies*

The looks-like prototype was manufactured using a 3D printer at TechShop Chandler. The cost of material was under $7. A class fee of $65.00 was required to use the 3D printer; however, this fee is waived for students. Details of all parts can be seen in Appendix G.

*7.4.4 Test Environment*

Material for the physical 2D environment, which is used for testing and demonstrations, needed to be purchased. It is built from a pegboard base (4 feet by 4 feet) and wood walls to construct a crude maze replica of the environment constructed in the software. The total cost was $81.23; see Appendix F for a listing of all parts used.

*7.4.5 Custom Circuit Board*

Manufacturing of the custom circuit board was done by hand at South Mountain Community College with the assistance of Rick Sparber, a professor in their department of Engineering. The cost was thus only the board itself and the electronics on it, or $23.35. All list of all parts on the board is in Appendix E.

### 7.5 Justification of budget differences from original proposal

The original proposal specified a total cost of under $700. The current works-like prototype costs under $700, which meets that budget. The additional costs of approximately $200 are replacement parts, the test environment, and parts that are no longer used on the prototype.

## 8 Conclusion

The semiconductor industry powers the field of new and improved electronics, which enjoy a permanent, lasting place in our lives. One of the biggest operating costs facing the semiconductor industry arises from the need for high cleanliness and purity of their manufacturing facilities. Facilities allocate a significant amount of resources annually to meet the standards of cleanliness. They may sacrifice utility and accuracy for lower costs when it comes to monitoring the sterility of their environments. Devices that are manufactured in unclean environments will underperform or fall short of consumer expectations.

Existing solutions primarily suffer from one of two disadvantages. Many are fixed to one location, thus several are needed to give a broad picture of air quality in an environment. Hand-held models exist, but these require a person to carry them around and can be prohibitively expensive. The features that differentiate of our product from existing solutions are that it is less expensive and can record air quality data throughout the environment. The robot is fully automated and made for next to zero user intervention. Also, its hands-off mode of operation keeps facilities running smoothly without causing any kind of production delay.

Although semiconductor manufacturing facilities constitute the largest portion of the clientbase, the uses for the SLAM robot can extend far beyond environmental monitoring. The initial prototype will not be autonomous, however this will be an even more compelling solution as it could be set up and then simply left alone. It will not require a human to carry it, nor will it require any specialized training, and it can be used instantly with little or no setup. The most appealing aspect of this is the lower cost, which will allow industries to pull resources back towards main production. Given the existing solutions, we believe this robot will revolutionize the way air quality is surveyed in environments, provide the best method of verifying clean room air particulate levels, and create more sustainable and hygienic working spaces.

# 9 REFERENCES


1. Airborne Particulate Cleanliness Classes in Cleanrooms and Clean Zones, FED-STD-209E, 1992.
2. Criteria For Accrediting Engineering Programs, ABET, Baltimore, MD, 2012, pp. 3.
3. D. Molloy. *Beaglebone: GPIO Programming on ARM Embedded Linux*. [Online] Retrieved 2014, December 4. Available: http://derekmolloy.ie/beaglebone/beaglebone-gpio-programming-on-arm-embedded-linux/
4. Shinyei PPD42NS datasheet, n.d. Available: http://www.seeedstudio.com/wiki/images/4/4c/Grove_-_Dust_sensor.pdf
5. Robopeak, “Low Cost 360 degree 2D Laser Scanner (LIDAR) System: Introduction and Datasheet,” RPLIDAR datasheet, May 2014, Revision 7

# 10 Appendix A: Printed Circuit Board Schematic and Layout

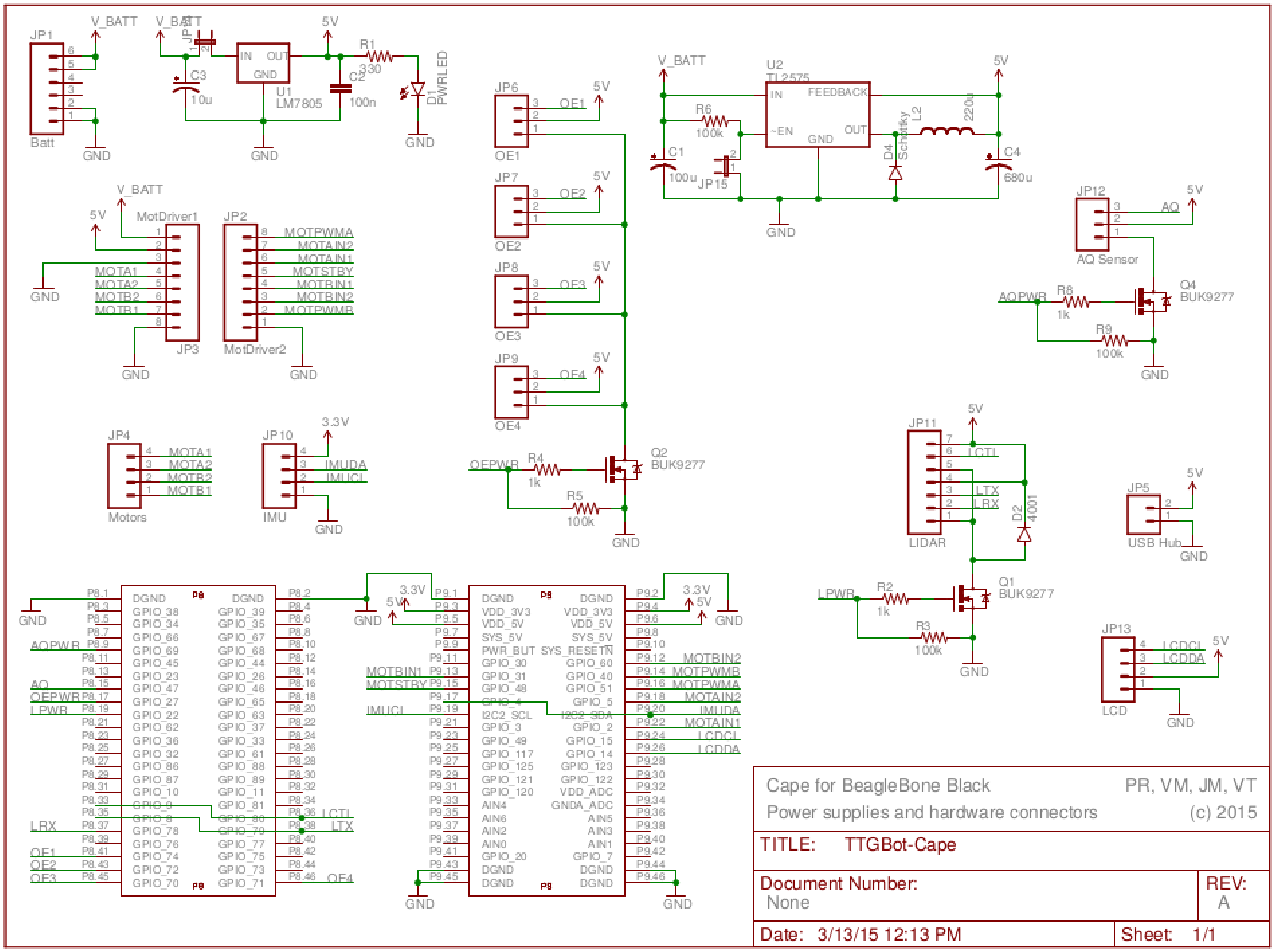


Figure 4: Schematic of robot PCB

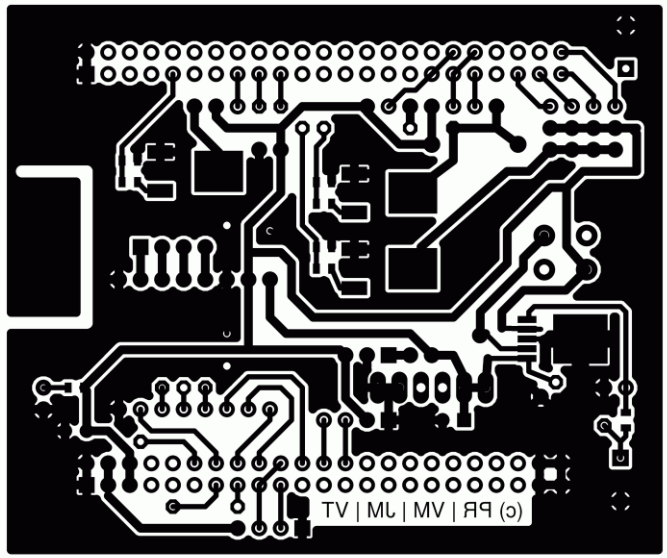


Figure 5: Layout of robot PCB

## 11 Appendix B: BeagleBone Black pin allocation

The default pin allocation for the BBB allocates pins to an HDMI interface, MMC flash storage, and user LEDs[3]. The HDMI interface has been disabled, thus those pins are available for use; the MMC interface and user LEDs are still enabled, thus those pins may not be used. The pins used to interface with the robot's hardware are listed in Table 3. Device pins that are not listed are not electrically connected to the BBB.

Table 3: BeagleBone Black pin allocations

| Device | Device pin name | Device pin number | BBB pin name | BBB pin mode | BBB pin number | Description |
|---|---|---|---|---|---|---|
| IMU | SDA | | I2C2_SDA | 3 - I2C2_SDA | P9_20 | Serial data (I2C) |
| | SCL | | I2C2_SCL | 3 - I2C2_SCL | P9_19 | Serial clock (I2C) |
| TB6612FNG Motor Driver | STBY | | UART4_TXD | 7 - gpio0[31] | P9_13 | Standby (disable outputs) |
| | PWMA | | EHRPWM1A | 6 - ehrpwm1A_mux1 | P9_14 | Speed of motor A (PWM) |
| | PWMB | | EHRPWM1B | 6 - ehrpwm1B_mux1 | P9_16 | Speed of motor B (PWM) |
| | AIN1 | | GPIO1_28 | 7 - gpio1[28] | P9_12 | Direction of motor A |
| | AIN2 | | UART4_RXD | 7 - gpio0[30] | P9_11 | Direction of motor A |
| | BIN1 | | GPIO1_16 | 7 - gpio1[16] | P9_15 | Direction of motor B |
| | BIN2 | | GPIO1_17 | 7 - gpio1[17] | P9_23 | Direction of motor B |
| Left Optical Encoder | LOE1 | 1 | GPIO2_12 | 7 - gpio2[12] * | P8_39 | O.E. signal 1 for left wheel |
| | LOE2 | 2 | GPIO2_13 | 7 - gpio2[13] * | P8_40 | O.E. signal 2 for left wheel |
| Right Optical Encoder | ROE1 | 1 | GPIO2_10 | 7 - gpio2[10] * | P8_41 | O.E. signal 1 for right wheel |
| | ROE2 | 2 | GPIO2_11 | 7 - gpio2[11] * | P8_42 | O.E. signal 2 for right wheel |
| HD44780 LCD | DB4 | 11 | TBD | | | Data bus bit 4 |
| | DB5 | 12 | TBD | | | Data bus bit 5 |

| | | | | | | |
|---|---|---|---|---|---|---|
| | DB6 | 13 | TBD | | | Data bus bit 6 |
| | DB7 | 14 | TBD | | | Data bus bit 7 |
| | RS | 4 | TBD | | | Register select |
| | E | 6 | TBD | | | Enable |
| | V0 | 3 | TBD | | | Contrast voltage |
| | R/W | 5 | TBD | | | Read/Write (tied to GND for write) |
| RPLIDAR | RX | | UART5_TXD | 4 - uart5_txd | P8_37 | Receive |
| | TX | | UART5_RXD | 4 - uart5_rxd | P8_38 | Transmit |
| | MOTOCTL | | UART3_CTSN | 2 - ehrpwm1A | P8_36 | Motor speed control (PWM) |
| SM-PWM-01A Dust Particle Sensor | P1 | 4 | GPIO1_13 | 7 - gpio1[13] * | P8_11 | 1-2um particle density |
| | P2 | 2 | GPIO1_12 | 7 - gpio1[12] * | P8_12 | 3-10um particle density |

* The optical encoders and dust sensor are connected to PRU-capable pins, as they involve real-time processing of data. Thus, we could use the PRU for either of these devices if necessary by changing the pin mode and using a separate assembly program. These pins also work as regular GPIOs.

## 12 APPENDIX C: COMPARISON OF PROPOSED PROTOTYPE, CURRENT PROTOTYPE, AND IDEAL PRODUCT

Table 4: Comparison of proposed prototype, current prototype, and ideal product

| Features | Proposed Prototype | Existing Prototype | Ideal Product |
|---|---|---|---|
| Cleanroom friendly and does not produce contaminants | No | No | Yes |
| Looks-like prototype | Yes | Yes | No |
| Add-on modules | No | No | Yes |
| Explore mode | Yes | No. This is not unique to this project and was removed due to time constraints to give additional time to other areas. | Yes |
| Map (Autonomous) mode | No | No | Yes |
| Map (Manual) mode | Yes | Yes | Yes |
| Monitor mode | No | No | Yes |
| Home mode | No | No | Yes |
| Webcam | Yes | No. Removed due to limited USB bandwidth and as cleanroom experts showed no interest in a live video stream | No |

## 13 APPENDIX D: EXPENDITURES FOR WORKS-LIKE PROTOTYPE

Table 5: Parts purchased for works-like prototype

| Item | Vendor | Part No. | Price/each | Qty. | Description |
|---|---|---|---|---|---|
| Chassis and Motors | DFRobot | ROB0049 | 34.35 | 1 | Forms the mechanical base of the robot with two motors and a caster |
| Chassis upper deck | DFRobot | DFR0310 | 3.50 | 1 | Forms the mechanical top of the robot |
| Optical Encoders | DFRobot | SEN0116 | 2.90 | 2 | Tells how fast the wheels are spinning. |
| Arduino Uno | TechShop Chandler | | 30.00 | 1 | Used for testing hardware as most example code is written for the Arduino; not used in prototype |
| BeagleBone Black 1 | SparkFun | DEV-12076 | 44.95 | 1 | Central single-board computer; died for unknown reasons |
| BeagleBone Black 2 | Adafruit | 1876 | 55.00 | 1 | Replacement for first BBB, died due to incorrect wiring in prototype |
| BeagleBone Black 3 | Amazon | | 55.00 | 1 | Replacement for second BBB |
| Proto Cape | SparkFun | DEV-12774 | 9.95 | 1 | Prototype for custom circuit board |
| IMU | Adafruit | 1714 | 19.95 | 1 | Provides info about the orientation and acceleration of the robot |
| LIDAR | RobotShop | RB-Rpk-01 | 398.99 | 1 | Environmental mapping |
| LCD | Sparkfun | LCD-09568 | 29.95 | 1 | Provide debugging information, not used as it was too big |
| Crius CO-16 OLED LCD | eBay | | 7.18 | 1 | Smaller, replaces above LCD |

| | | | | | |
|---|---|---|---|---|---|
| IR sensor | SparkFun | BOB-10901 | 9.95 | 1 | Not accurate, using LIDAR instead |
| Webcam | Amazon | B000Q3VECE | 13.97 | 1 | Streaming video, not used |
| Micro Servo | DFRobot | SER0006 | 3.46 | 1 | Rotate webcam for better view |
| Dust Particle Sensor | Seeed Studio | SEN12291P | 15.90 | 1 | Detects airborne particles |
| Motor Driver | SparkFun | ROB-09457 | 8.95 | 1 | Powers the two motors |
| USB Hub | Amazon | B002FFT8Z6 | 5.34 | 1 | Gives more USB slots and provides more power to WiFi adapter than BBB can |
| Battery | BatterySpace | CU-LC-14430S2 | 14.95 | 2 | |
| Battery Charger | BatterySpace | CH-L7405 | 19.95 | 1 | |
| Power switch | Circuit Specialists | 8014 | 1.19 | 1 | |
| M3x10 standoffs | DFRobot | FIT0066 | 1.50 | 1 | Mechanical connections of various parts |
| M3x20 standoffs | DFRobot | FIT0182 | 1.90 | 1 | Mechanical connections of various parts |
| M3x6 standoffs | DFRobot | FIT0065 | 1.40 | 1 | Mechanical connections of various parts |
| Logic Level Converters | SparkFun | BOB-12009 | 2.95 | 2 | Translate logic levels to different voltages, not used in current prototype. Previous |
| 6” jumper wires | SparkFun | PRT-12796 | 1.95 | 1 | Connects various electronics to custom PCB |
| Female headers | SparkFun | PRT-00115 | 1.50 | 3 | Connector on custom PCB for various electronics |
| Male headers | SparkFun | PRT-00116 | 1.50 | 3 | Connector on custom PCB for various electronics |
| Miscellaneous additional supplies and small electronics already on hand | | | | | |

| | | |
|---|---|---|
| Total, not including custom PCB or air quality add-on module. Parts for final product will be cheaper due to bulk discounts. | 824.88 | |

## 14 APPENDIX E: EXPENDITURES FOR CIRCUIT BOARD

Table 6: Parts purchased for circuit board

| Item | Vendor | Part No. | Price/each | Qty. | Comments |
|---|---|---|---|---|---|
| 3x5” Presensitized Copper Clad Board | Circuit Specialists | 603 | 7.60 | 1 | The PCB itself |
| 5V Linear Regulator | Digi-Key | LM7805 | | 1 | Already had on hand |
| 10uF capacitor | Digi-Key | | | 1 | For linear regulator, already had on hand |
| 100nF capacitor | Digi-Key | | | 1 | For linear regulator, already had on hand |
| Male/Female Headers | | | | | Already purchased for prototype PCB |
| 2x23 stacking headers | Ada-Fruit | 706 | 4.95 | 1 | Set of two, used to connect PCB to BBB |
| Switching buck regulator | Digi-Key | 296-21824-1-ND | 2.39 | 2 | |
| 220uH Inductor | Dig-Key | 811-2101-ND | 0.85 | 2 | For switching regulator |
| 100uF Capacitor | Digi-Key | 495-6004-ND | 0.73 | 2 | For switching regulator |

| | | | | | |
|---|---|---|---|---|---|
| 470uF Capacitor | Digi-Key | 495-6026-ND | 0.77 | 2 | For switching regulator |
| Schottky Diode | Digi-Key | RA 13V1CT-ND | 0.66 | 2 | For switching regulator |
| 330 Ohm SMT resistor | Digi-Key | | 0 | 1 | Already had on hand |
| LED | Digi-Key | | 0 | 1 | Already had on hand |
| 1kOhm SMT resistor | Digi-Key | | 0 | 3 | Already had on hand |
| 100kOhm SMT resistor | Digi-Key | | 0 | 4 | Already had on hand |
| SMT Power NMOSFET | Digi-Key | BUK9277 | 0 | 3 | Already had on hand |
| Diode | | 4001 | 0 | 1 | Already had on hand |
| Total, not including tax or shipping | | | 23.35 | | |

# 15 APPENDIX F: EXPENDITURES FOR TEST ENVIRONMENT

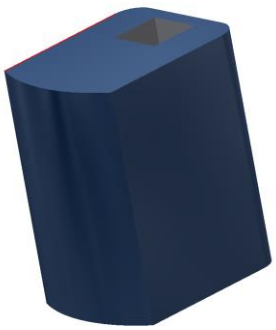
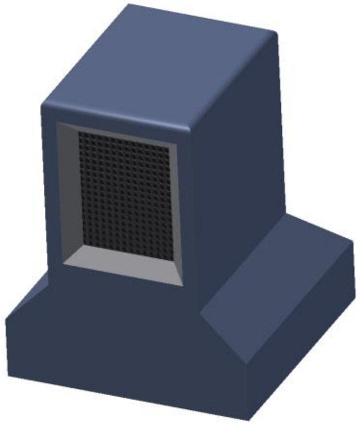
Table 7: Parts purchased for test environment

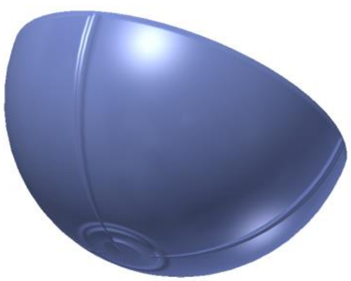
| Item | Vendor | Dimension | Price/each | Qty. | Description |
|---|---|---|---|---|---|
| PegBoardBase | Home Depot | 3/16"x4'x8' | 17.98 | 1 | Brown pegboard base; Cut it in half |
| Wooden Walls | Home Depot | 1"x8"x8' | 11.46 | 4 | Pine; Forms the outside walls of environment |
| Wooden Supports | Home Depot | 1"x2"x8' | 5.58 | 2 | Pine; Used to elevate pegboard base structure |
| Wood glue | Home Depot | 8 fl. oz. | 3.95 | 1 | Central single-board computers |
| Unbranded Brad Nails #17 | Home Depot | 1" | 2.30 | 1 | Small diameter, Zinc-Plated Steel |
| Total, not including hand tools utilized to cut and assemble the parts (most were borrowed). | | | 81.23 (w/o tax) | | |

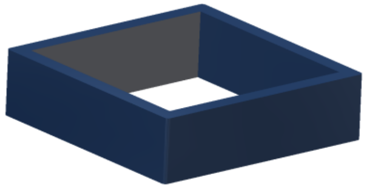
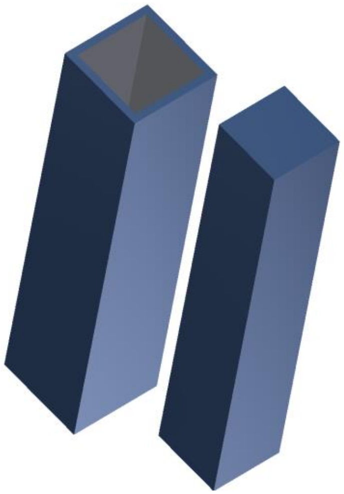
## 16 Appendix G: Details of Looks-Like Prototype

All parts of the looks-like prototype were printed with PLA plastic. The cost is calculated at $0.15/gram by TechShop Chandler.

Table 8: Parts for looks-like prototype.

| Part | 3D Software Model | Theoretical Dimensions (Intended) | Printed Dimensions (Actual) | Qty. | Print Cost |
|---|---|---|---|---|---|
| Body |  | 24 x 18 x 30” | 2.4 x 1.8 x 3.0” | 1 | $3.00 |
| AQ sensor |  | 4 x 4 x 5” | 0.4 x 0.4 x 0.5” | 1 | < $0.50 |

| | | | | | |
|---|---|---|---|---|---|
| Lidar |  | 4 x 3 x 3" (approx) | 0.4 x 0.3 x 0.3" | 1 | < $0.50 |
| Wheels (front) |  | Diameter: 6.0"<br>Thickness: 2.0" | Diameter: 0.6"<br>Thickness: 0.2" | 2 | < $0.50 |
| Wheel (rear) |  | Diameter: 6.0" | Diameter: 0.6" | 1 | < $0.50 |

| | | | | | |
|---|---|---|---|---|---|
| Arm base |  | 7.25 x 7.25 x 7.25"<br>Hole: 6.25 x 6.25" | 0.725 x 0.725 x 0.725"<br>Hole: 0.625 x 0.625 | 1 | < $0.50 |
| Extendible arms |  | Outer: 6.0 x 6.0 x 28"<br>Hole: 5.0 x 5.0"<br><br>Inner: 4.75 x 4.75 x 28" | Outer: 0.6 x 0.6 x 2.8"<br>Hole: 0.5 x 0.5"<br><br>Inner: 0.475 x 0.475 x 2.8" | 1 | $1.00 |
| Alert lights |  | 8.0 x 1.0" | 0.8 x 0.1" | 2 | < $0.50 |
| Total cost of printing | | | | | < $7.00 |

## 17 Appendix H: Power Consumption

Current draw of all parts was measured with a digital multimeter in various modes of operation and is listed in Table 9. All components (with the exception of the motors) were powered through a 5V linear regulator. The multimeter was placed in series between the regulator and multimeter to include power loss within the regulator. As most components cannot operate normally without being connected to the BBB, it was included in the measurements, and later subtracted.

The efficiency of the linear regulator is determined by the following equation. Output voltage is fixed at 5V, input voltage is the battery operating voltage, which varies and was chosen as 8 V for the worst case.
Efficiency = Output Voltage / Input Voltage = 5 / 8 = 63 %

We also calculated what the power consumption of each component would be with a switching regulator with efficiency of 77%, which is a common efficiency among several commercially available parts researched.

Based on these measurements we calculated the total current draw and expected battery life with a 650mAh battery with both a linear and switching regulator, shown in Table 10. We also calculated the total current draw and expected battery life with MOSFET switches added on the power supply to the optical encoders, air quality sensor, and LIDAR.

The robot has two 650mAh batteries and a linear regulator, thus expected battery life in the worst case (all components busy) is 1.4 hours, and expected battery life in the best case (idling with hardware switches) is 6.0 hours.

Table 9: Current draw of all components

| Device | Current (mA), measured as drawn from the battery | Current if regulator efficiency was 77% instead of 63% | Current goes through regulator? |
|---|---|---|---|
| BBB (idling) | 180 | 147 | Y |
| *BBB (busy)* | *320* | *262* | *Y* |
| WiFi (disabled) | 40 | 33 | Y |
| *WiFi (connected)* | *100* | *82* | *Y* |
| Optical Encoders | 100 | 82 | Y |

| | | | |
|---|---|---|---|
| AQ sensor | 80 | 65 | Y |
| LIDAR (not spinning) | 50 | 41 | Y |
| *LIDAR (spinning)* | *160* | *131* | *Y* |
| Servo | 10 | 8 | Y |
| Webcam | | | Y |
| LCD | 0 | 0 | Y |
| IMU | 0 | 0 | Y |
| Motor Driver (on standby) | 0 | 0 | Y |
| *Motors* | *100* | *100* | *N* |
| Note: Italicized rows indicate a busy state for that component. Non-italicized rows indicate an idle state, or that the component was measured in only one state. | | | |

Table 10: Total current draw of all components

| Scenario | Current (mA) | Battery Life (hrs) |
|---|---|---|
| Idling with no hardware switches (Linear) | 460 | 1.4 |
| Idling with no hardware switches (Switching) | 376 | 1.7 |
| Idling with hardware switches (Linear) | 220 | 3.0 |
| Idling with hardware switches (Switching) | 180 | 3.6 |
| All components busy (Linear) | 870 | 0.7 |
| All components busy (Switching) | 730 | 0.9 |

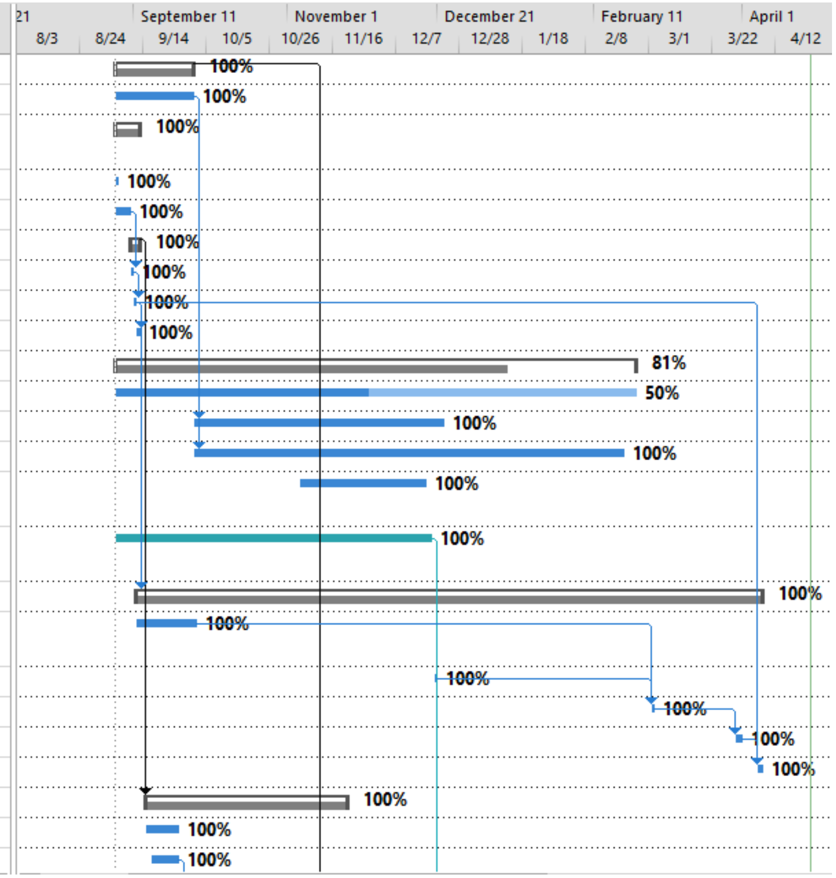
## 18 Appendix H: Code

All code is located in a private Git repository. Contact any of the team members with your BitBucket username for access to the repository.

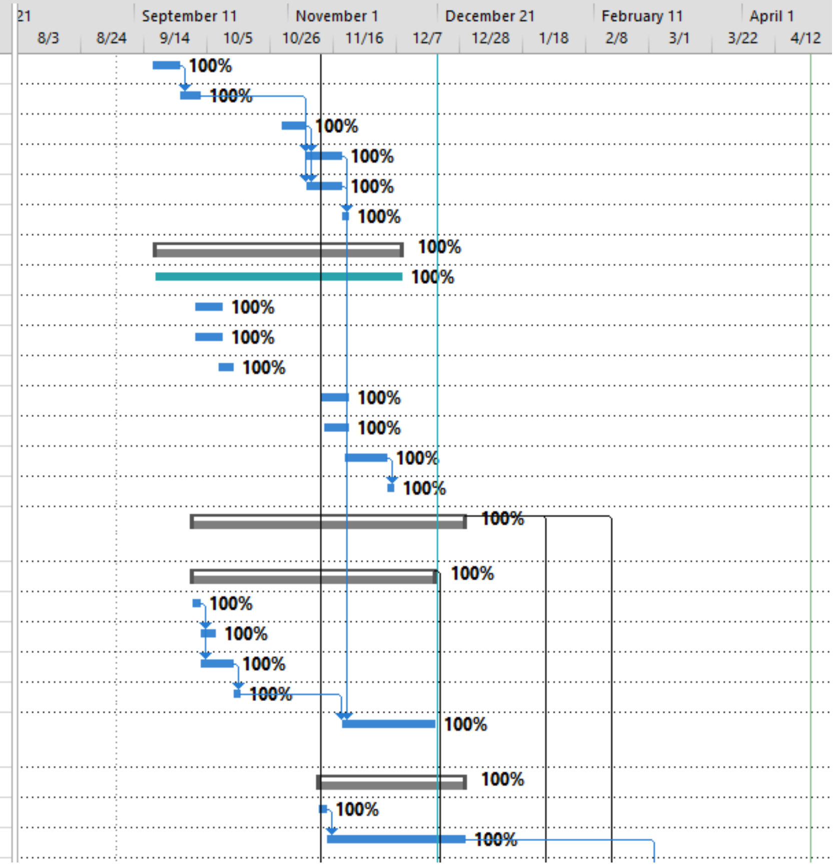
Repository URL: https://bitbucket.org/the7thGhost/ttgbot

# 19 APPENDIX I: GANTT CHART

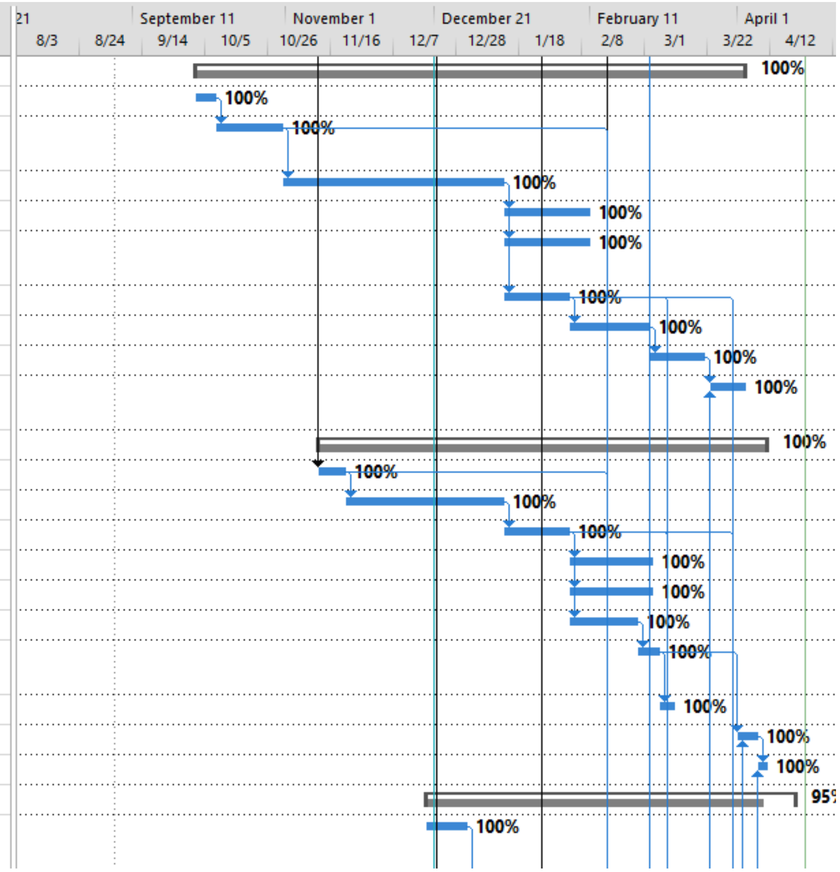
| | Task Name | Durati | Start | Finish | Predecesso | Resource Names |
|---|---|---|---|---|---|---|
| 21 | ◢ Research applications | 18 days | Fri 9/5/14 | Tue 9/30/14 | | |
| 22 | Decide on application | 18 days | Fri 9/5/14 | Tue 9/30/14 | | |
| 23 | ◢ Basic robot hardware and overhead | 6 days | Fri 9/5/14 | Fri 9/12/14 | | |
| 24 | Summer tasks | 1 day | Fri 9/5/14 | Fri 9/5/14 | | |
| 25 | Install Ubuntu | 3 days | Fri 9/5/14 | Tue 9/9/14 | | Paul |
| 26 | ◢ Get WiFi @ ASU | 3 days | Wed 9/10/14 | Fri 9/12/14 | | Paul |
| 27 | Get WiFi working | 1 day | Wed 9/10/14 | Wed 9/10/14 | 25 | |
| 28 | Connect to ASU's WiFi | 1 day | Thu 9/11/14 | Thu 9/11/14 | 27 | |
| 29 | Write WiFi instructions | 1 day | Fri 9/12/14 | Fri 9/12/14 | 28 | |
| 90 | ◢ Research AQ Requirements | 120 days | Fri 9/5/14 | Tue 2/24/15 | | |
| 93 | Research AQ patents | 120 days | Fri 9/5/14 | Tue 2/24/15 | | Vanessa |
| 91 | Air quality competition | 60 days | Wed 10/1/14 | Mon 12/22/14 | 22 | Josh |
| 92 | Talk with A.Q. experts | 100 days | Wed 10/1/14 | Fri 2/20/15 | 22 | Josh |
| 94 | Research clean room standards | 30 days | Wed 11/5/14 | Tue 12/16/14 | | Josh |
| 112 | Plan tasks, goals, Gantt chart, schedule | 75 days | Fri 9/5/14 | Thu 12/18/14 | | |
| 38 | ◢ LCD-low priority | 145 days | Fri 9/12/14 | Tue 4/7/15 | 28 | |
| 39 | Buy wires & connectors | 14 days | Fri 9/12/14 | Wed 10/1/14 | | Paul |
| 40 | Buy different LCD | 1 day | Sat 12/20/14 | Sat 12/20/14 | | |
| 41 | Wire LCD | 1 day | Mon 3/2/15 | Mon 3/2/15 | 39,40 | Paul |
| 42 | Code for LCD | 2 days | Mon 3/30/15 | Tue 3/31/15 | 41 | Vu |
| 43 | Show info on LCD | 2 days | Mon 4/6/15 | Tue 4/7/15 | 42,28 | Paul |
| 30 | ◢ Robot drives like an RC car | 49 days | Mon 9/15/14 | Thu 11/20/14 | 26 | |
| 37 | PS3 Ctrlr | 9 days | Mon 9/15/14 | Thu 9/25/14 | | Vu |
| 31 | Buy connectors for motors | 7 days | Wed 9/17/14 | Thu 9/25/14 | | Paul |



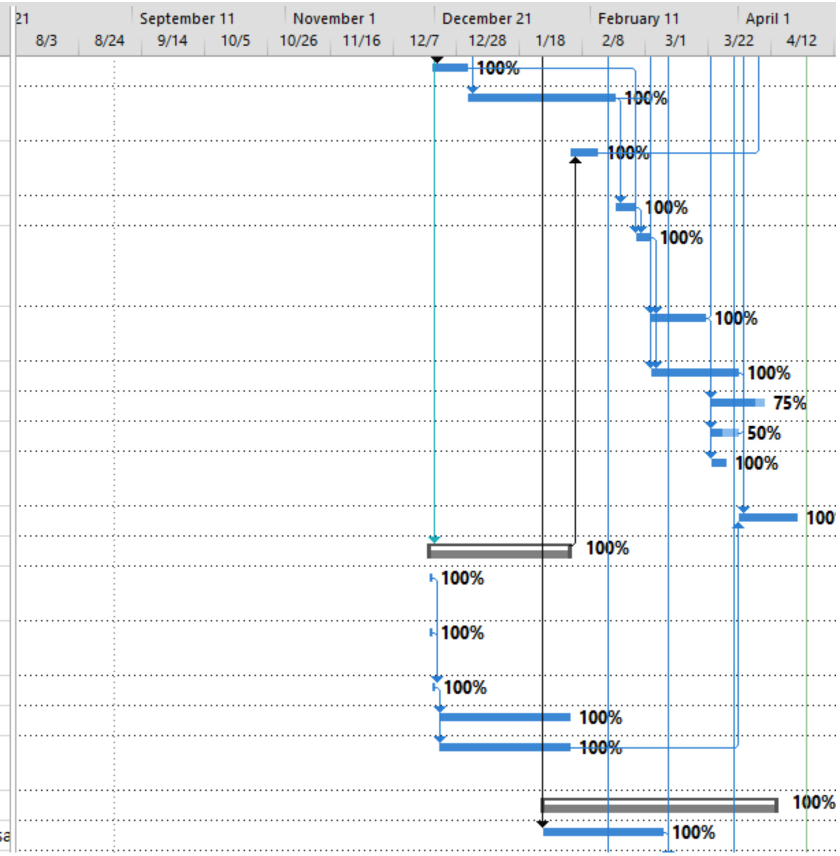
| | Task Name | Duration | Start | Finish | Predecessors | Resource Names |
|---|---|---|---|---|---|---|
| 31 | Buy connectors for motors | 7 days | Wed 9/17/14 | Thu 9/25/14 | | Paul |
| 32 | Wiring | 5 days | Fri 9/26/14 | Thu 10/2/14 | 31 | Josh |
| 33 | Disable HDMI Framer | 6 days | Thu 10/30/14 | Thu 11/6/14 | | Paul |
| 34 | Port Arduino code | 8 days | Fri 11/7/14 | Tue 11/18/14 | 33,32 | Vanessa |
| 35 | Get device tree working | 8 days | Fri 11/7/14 | Tue 11/18/14 | 32,33 | Paul |
| 36 | Set Motor Speeds | 2 days | Wed 11/19/14 | Thu 11/20/14 | 34,35 | Vu |
| 12 | Fall Progress reports | 58 days | Thu 9/18/14 | Mon 12/8/14 | | |
| 20 | Videos 1-10 | 58 days | Thu 9/18/14 | Mon 12/8/14 | | |
| 13 | Presentation 1 | 7 days | Wed 10/1/14 | Thu 10/9/14 | | |
| 14 | Quad Chart 1 | 7 days | Wed 10/1/14 | Thu 10/9/14 | | |
| 15 | Memo 1 | 3 days | Thu 10/9/14 | Mon 10/13/14 | | |
| 17 | Memo 2 | 7 days | Wed 11/12/14 | Thu 11/20/14 | | |
| 16 | Presentation 2 | 6 days | Thu 11/13/14 | Thu 11/20/14 | | |
| 18 | Final report draft 1 | 10 days | Thu 11/20/14 | Wed 12/3/14 | | |
| 19 | Final report | 2 days | Thu 12/4/14 | Fri 12/5/14 | 18 | |
| 54 | Display data from optical encoders, IMU | 62 days | Tue 9/30/14 | Mon 12/29/14 | | |
| 55 | Optical encoders | 59 days | Tue 9/30/14 | Fri 12/19/14 | | |
| 56 | Test optical encoders | 3 days | Tue 9/30/14 | Thu 10/2/14 | | Vu |
| 57 | Wire Optical Encoders | 3 days | Fri 10/3/14 | Tue 10/7/14 | 56 | Josh |
| 58 | Get data from O.E. | 7 days | Fri 10/3/14 | Mon 10/13/14 | 56 | Paul,Vu |
| 59 | Convert data to RPMs | 2 days | Tue 10/14/14 | Wed 10/15/14 | 58 | Paul,Vu |
| 60 | Use interrupt instead of timer | 23 days | Wed 11/19/14 | Fri 12/19/14 | 59,35 | Vu |
| 61 | IMU | 32 days | Tue 11/11/14 | Mon 12/29/14 | | |
| 62 | Wire IMU | 3 days | Tue 11/11/14 | Thu 11/13/14 | | Josh |
| 63 | Read IMU data | 29 days | Fri 11/14/14 | Mon 12/29/14 | 62 | |



| | Task Name | Duration | Start | Finish | Predecessors | Resource Names |
|---|---|---|---|---|---|---|
| 44 | ◢ Display LIDAR data | 129 days | Thu 10/2/14 | Fri 4/3/15 | | |
| 45 | Test short-range IR sensor | 5 days | Thu 10/2/14 | Wed 10/8/14 | | Vanessa |
| 46 | Decide on Sensors-want to use LIDAR | 16 days | Thu 10/9/14 | Thu 10/30/14 | 45 | |
| 47 | Buy LIDAR | 49 days | Fri 10/31/14 | Mon 1/12/15 | 46 | Vanessa |
| 48 | Wire LIDAR | 21 days | Tue 1/13/15 | Tue 2/10/15 | 47 | Paul,Vu |
| 49 | Physically attach LIDAR to robot | 21 days | Tue 1/13/15 | Tue 2/10/15 | 47 | Paul |
| 50 | Test LIDAR | 16 days | Tue 1/13/15 | Tue 2/3/15 | 47 | Vu |
| 51 | Read LIDAR output | 19 days | Wed 2/4/15 | Mon 3/2/15 | 50 | Vu |
| 52 | Rewrite LIDAR code | 14 days | Tue 3/3/15 | Fri 3/20/15 | 51 | Pul |
| 53 | Display object positions on web interface | 10 days | Mon 3/23/15 | Fri 4/3/15 | 70,52 | Paul |
| 95 | ◢ Display air quality data | 105 days | Wed 11/12/14 | Fri 4/10/15 | | |
| 96 | Determine necessary data | 7 days | Wed 11/12/14 | Thu 11/20/14 | 21 | Josh |
| 97 | Research and buy sensors | 34 days | Fri 11/21/14 | Mon 1/12/15 | 96 | Josh,Paul |
| 98 | Test sensor | 16 days | Tue 1/13/15 | Tue 2/3/15 | 97 | Vu |
| 99 | Wire sensors | 20 days | Wed 2/4/15 | Tue 3/3/15 | 98 | Paul,Vu |
| 103 | Physically attach to robot | 20 days | Wed 2/4/15 | Tue 3/3/15 | 98 | Paul |
| 104 | Get PRU working | 17 days | Wed 2/4/15 | Thu 2/26/15 | 98 | Paul |
| 105 | Get data from sensors using PRU | 5 days | Fri 2/27/15 | Thu 3/5/15 | 104 | Paul |
| 100 | Display AQ on web interface | 3 days | Fri 3/6/15 | Tue 3/10/15 | 105 | Paul |
| 101 | Save data with poses | 5 days | Wed 4/1/15 | Tue 4/7/15 | 105,68 | |
| 102 | Display AQ on map | 3 days | Wed 4/8/15 | Fri 4/10/15 | 76,101 | |
| 64 | ◢ Localization | 85 days | Thu 12/18/14 | Mon 4/20/15 | | |
| 72 | Research how to get position from IMU data | 7 days | Thu 12/18/14 | Wed 12/31/14 | | Paul |



| | Task Name | Durati | Start | Finish | Predecesso | Resource Names |
|---|---|---|---|---|---|---|
| 75 | Convert RPMs to speed | 5 days | Sat 12/20/14 | Wed 12/31/14 | 55 | Vu |
| 69 | Calculate orientation from IMU data | 35 days | Thu 1/1/15 | Wed 2/18/15 | 72 | Paul |
| 76 | Display accurate map on web interface | 7 days | Wed 2/4/15 | Thu 2/12/15 | 106 | Paul |
| 73 | Get speed from IMU data | 5 days | Thu 2/19/15 | Wed 2/25/15 | 69 | Paul |
| 66 | See which speed is more accurate and decide which to use-OE is | 3 days | Thu 2/26/15 | Mon 3/2/15 | 73,75 | |
| 70 | Get position estimate from OE and motor control speed | 14 days | Tue 3/3/15 | Fri 3/20/15 | 66,69,63 | Vu |
| 74 | Calculate Initial Position | 21 days | Tue 3/3/15 | Tue 3/31/15 | 66,69,63 | Vu |
| 65 | Display robot poses on map | 14 days | Mon 3/23/15 | Thu 4/9/15 | 70 | |
| 68 | Keep track of all our poses | 7 days | Mon 3/23/15 | Tue 3/31/15 | 70 | |
| 71 | Display robot's driven path on web interface | 5 days | Mon 3/23/15 | Fri 3/27/15 | 70 | Paul |
| 67 | Calculate actual position | 14 days | Wed 4/1/15 | Mon 4/20/15 | 74,110 | Paul |
| 106 | Build environment | 30 days | Fri 12/19/14 | Tue 2/3/15 | 112 | |
| 107 | Decide on environment materials | 1 day | Fri 12/19/14 | Fri 12/19/14 | | |
| 111 | Plan environment size and shape | 1 day | Fri 12/19/14 | Fri 12/19/14 | | |
| 108 | Design environment | 1 day | Sat 12/20/14 | Sat 12/20/14 | 107,111 | |
| 109 | Build environment | 28 days | Mon 12/22/14 | Tue 2/3/15 | 108 | |
| 110 | Map of environment in software | 28 days | Mon 12/22/14 | Tue 2/3/15 | 108 | Vanessa |
| 77 | Custom PCB | 56 days | Mon 1/26/15 | Mon 4/13/15 | | |
| 78 | Design PCB Schematic | 30 days | Mon 1/26/15 | Fri 3/6/15 | 54 | Paul,Vanessa |



| | Task Name | Duration | Start | Finish | Predecessors | Resource Names |
|---|---|---|---|---|---|---|
| 79 | Design PCB Layout | 4 days | Mon 3/9/15 | Thu 3/12/15 | 78,98,50 | Paul |
| 80 | Fabricate PCB | 1 day | Fri 3/13/15 | Fri 3/13/15 | 79,85 | Paul,Vanessa |
| 81 | Drill PCB | 11 days | Mon 3/16/15 | Mon 3/30/15 | 80 | Vanessa |
| 82 | Populate PCB | 6 days | Tue 3/31/15 | Tue 4/7/15 | 81 | Vanessa |
| 83 | Test PCB | 3 days | Thu 4/9/15 | Mon 4/13/15 | 82 | Paul,Vu |
| 1 | Spring progress reports | 70 days | Fri 1/30/15 | Thu 5/7/15 | | |
| 11 | Video 1 | 1 day | Fri 1/30/15 | Fri 1/30/15 | | |
| 2 | Video 2 | 1 day | Tue 3/3/15 | Tue 3/3/15 | | |
| 9 | Report 3 | 6 days | Fri 3/13/15 | Fri 3/20/15 | | |
| 10 | Presentation 3 | 6 days | Thu 3/19/15 | Thu 3/26/15 | | |
| 5 | Video 3 | 1 day | Thu 3/26/15 | Thu 3/26/15 | | |
| 3 | Video 4 | 1 day | Tue 4/7/15 | Tue 4/7/15 | | |
| 6 | Documentation | 19 days | Mon 4/13/15 | Thu 5/7/15 | | |
| 7 | Final report | 7 days | Fri 4/17/15 | Mon 4/27/15 | | |
| 4 | Video 5 | 1 day | Fri 4/24/15 | Fri 4/24/15 | | |
| 8 | Final Demo | 6 days | Fri 4/24/15 | Fri 5/1/15 | | |
| 84 | Looks-Like Prototype | 44 days | Tue 2/17/15 | Fri 4/17/15 | | |
| 85 | Design chassis | 30 days | Tue 2/17/15 | Mon 3/30/15 | 54,46,96 | Josh |
| 88 | Design A.Q. module | 30 days | Tue 2/17/15 | Mon 3/30/15 | | Josh |
| 86 | Fabricate chassis | 11 days | Tue 3/31/15 | Tue 4/14/15 | 85,79,98,50 | Josh,Paul |
| 89 | Fabricate A.Q. module | 11 days | Tue 3/31/15 | Tue 4/14/15 | 88 | Josh,Paul |
| 87 | Assemble chassis | 3 days | Wed 4/15/15 | Fri 4/17/15 | 86 | Josh |