

**Department of Robotics & Mechatronics Engineering**

**Preliminary Design Review**

Autonomous Wheelchair

Max Bronson, Denny Mannakulathil, Oscar Montealegre

Mechatronics System Design- Fall 2021

Mechatronics Engineering

Kennesaw State University

10/11/2021

# Executive Summary

This report is used to outline the development process of the Automated Wheelchair and its surrounding systems. The system implements Python, C++, and ROS to process data provided from both a Velodyne VLP-16 LiDAR Module and a Microsoft Kinect camera. Using this processed data, serial commands are sent to an Arduino 2560 which sends a variable voltage to the electric motors on the rear wheels of the wheelchair. The system is to be capable of point-to-point navigation without human intervention following a target being defined. The system will implement object avoidance to prevent collision with obstacles that may pass through the optimal path calculated by the wheelchair.

**Table of Contents**

[Executive Summary 2](#_Toc84890376)

[List of Tables 6](#_Toc84890377)

[1. System Overview 7](#_Toc84890378)

[1.1. Major Works 7](#_Toc84890379)

[1.2. System Block Diagram 8](#_Toc84890380)

[1.3. Major Parts 8](#_Toc84890381)

[2. Design Requirements & Specifications 9](#_Toc84890382)

[2.1. Dimensions 9](#_Toc84890383)

[2.2. Functions 9](#_Toc84890384)

[2.3. Capabilities 9](#_Toc84890385)

[3. Verification Approach/Plan 10](#_Toc84890386)

[3.1. Analysis 10](#_Toc84890387)

[3.1.1. Operation Time and Power Draw 10](#_Toc84890388)

[3.1.2. Testing Path-planning 10](#_Toc84890389)

[3.1.3. Sensor Data Accuracy Check 10](#_Toc84890390)

[3.2. Simulation 11](#_Toc84890391)

[3.2.1. 3D CAD Models of Bracket Parts 11](#_Toc84890392)

[3.2.2. Obstacle Avoidance in a Virtual Environment 11](#_Toc84890393)

[3.3. Test 11](#_Toc84890394)

[3.3.1. Sensors Testing for Calibration 11](#_Toc84890395)

[3.3.2. Motors Tests for Response to Commands 11](#_Toc84890396)

[3.3.3. Obstacle Avoidance Testing 11](#_Toc84890397)

[4. Basic Design & Trade Study Results 12](#_Toc84890398)

[4.1. Wiring Diagrams 12](#_Toc84890399)

[4.2. Analysis & Simulation Results 13](#_Toc84890400)

[4.2.1. Simulations Using ROS 13](#_Toc84890401)

[4.2.2. Kinect RGB and Infared Analysis 13](#_Toc84890402)

[4.3. Trade Study Items 15](#_Toc84890403)

[4.3.1. Microcontrollers 15](#_Toc84890404)

[4.3.2. Sensors 15](#_Toc84890405)

[4.3.3. Motors 16](#_Toc84890406)

[4.3.4. Battery 16](#_Toc84890407)

[4.3.5. Structure 17](#_Toc84890408)

[4.3. Bill of Materials 18](#_Toc84890409)

[4.4. Minimum Success Criteria 18](#_Toc84890410)

[4.5. Design Optimization/Improvement 18](#_Toc84890411)

[5. Other 19](#_Toc84890412)

[5.1. Available Resources 19](#_Toc84890413)

[5.2. Project Schedule & Budget 20](#_Toc84890414)

[5.3. Team Chart/Assignments 20](#_Toc84890415)

[References 21](#_Toc84890416)

[Appendices 22](#_Toc84890417)

[Appendix A: Velodyne VLP-16 22](#_Toc84890418)

[Appendix B: Microsoft Kinect V2 24](#_Toc84890419)

[Appendix C: Arduino 2560 25](#_Toc84890420)

[Appendix D: Bioenno Lithium-Ion Battery 26](#_Toc84890421)

**Table of Figures**

[**Figure 1:** Block Diagram 8](#_Toc84890422)

[**Figure 2:** Wheelchair Structure 9](#_Toc84890423)

[**Figure 3:** Wiring Diagram of Digital Signals 12](#_Toc84890424)

[**Figure 4:** Wiring Diagram of Power to New Sensors 13](#_Toc84890425)

[**Figure 5:** VLP-16 SLAM Early Implementation 13](#_Toc84890426)

[**Figure 6:** Kinect Rviz Capabilities Analysis 14](#_Toc84890427)

[**Figure 7:** Kinect Camera Capabilities Analysis 14](#_Toc84890428)

[**Figure 8:** Bill of Materials 18](#_Toc84890429)

[**Figure 9:** Project Budget 20](#_Toc84890430)

[**Figure 10:** Project Schedule 20](#_Toc84890431)

[**Figure 11:** VLP-16 Data Sheet 1 22](#_Toc84890432)

[**Figure 12:** VLP-16 Data Sheet 2 23](#_Toc84890433)

[**Figure 13:** Microsoft Kinect V2 Rendering 24](#_Toc84890434)

[**Figure 14:** Microsoft Kinect V2 Data Sheet 24](#_Toc84890435)

[**Figure 15:** Arduino 2560 Rev3 Wiring Diagram 1 25](#_Toc84890436)

[**Figure 16:** Arduino 2560 Rev3 Wiring Diagram 2 25](#_Toc84890437)

[**Figure 17:** Arduino 2560 Rev3 Technical Specifications 26](#_Toc84890438)

[**Figure 18:** Properties of Bioenno Battery 26](#_Toc84890439)

# List of Tables

|  |  |
| --- | --- |
| **Figure Name** | **Page #** |
| Table 1: Major Parts | 8 |
| Table 2: Verification Approach/Plan | 10 |
| Table 3: Microcontroller Trade Study | 15 |
| Table 4: LIDAR Trade Study | 16 |
| Table 5: Battery Trade Study | 17 |
| Table 6: Frame Trade Study | 17 |

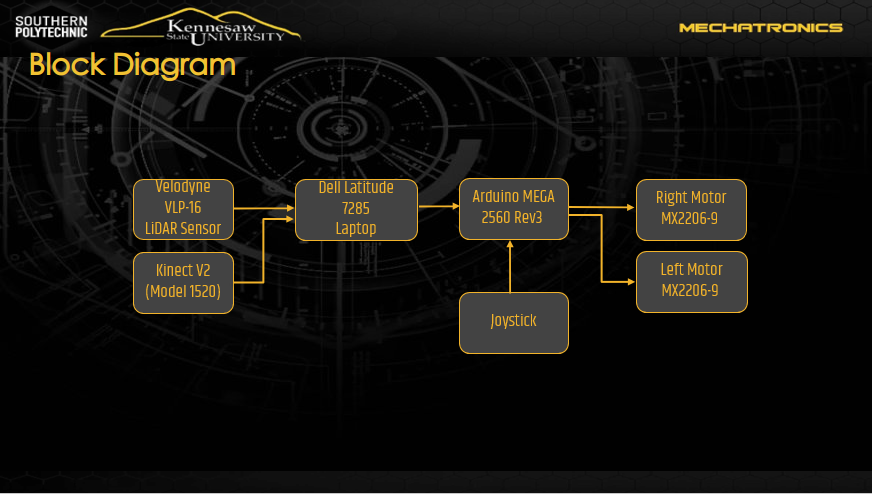
# System Overview

The automated wheelchair is a system that will implement Python, C++, and ROS programs to autonomously complete point to point navigation while dynamically avoiding both static and mobile obstacles. The system will implement a Velodyne VLP-16 LiDAR module and a Microsoft Kinect as the main sensors for collecting data about the world surrounding the wheelchair. Data processing will be completed using a Dell Latitude 7285 followed by an Arduino Mega 2560 sending instructions to both electric motors attached to the rear wheels of the wheelchair.

## 1.1. Major Works

The Automated Wheelchair consists of three major categories of work: Mechanical, Electrical and Programming. The mechanical work in its current state is 35% complete. The wheelchair and the wire housing box are complete and in place. A mount for the Linux machine to sit on the back of the wheelchair has been designed and is currently in the process of being 3D printed. There are two additional sensor mounts currently in the design phase. The electrical work being conducted is 60% complete. All digital logic between the Arduino Mega and the wheelchair motors is complete. The power and digital signals for the addition of the VLP-16 and Kinect are still to be completed. Lastly, the programming is 30% complete. Packages and supporting information have been compiled. The utilization of the VLP-16 Module and Kinect Camera in Ubuntu has been established; the algorithm which will process the data produced by the sensors is currently being developed.

## 1.2. System Block Diagram

**Figure 1:** Block Diagram

## 1.3. Major Parts

#### Table 1: Major Parts

|  |  |
| --- | --- |
| **Part** | **Model** |
| Frame | Yurob Portable Electric Wheelchair |
| Microcontroller | Arduino Mega 2560 Rev3 |
| Sensors | Velodyne VLP-16, Microsoft Kinect V2 |
| Linux Machine | Dell Latitude 7285 |
| Battery | Bioenno Power 12V Lithium-Ion Battery |

# Design Requirements & Specifications

The following sections show the physical composition of the system, as well as the functions and capabilities of the wheelchair upon completion:

## 2.1. Dimensions

The figure below is a summary of the system in production. This summary outlines the physical dimensions expected upon the system’s completion. The height of the wheelchair will increase by ~6 inches due to the sensors being mounted on a stand above the head of the user.



**Figure 2:** Wheelchair Structure

## 2.2. Functions

The wheelchair is to be capable of maneuvering around both stationary and moving obstacles while navigating. Navigation from initial point to target point is to be accomplished autonomously and without any user intervention. The path-planning algorithm is to be capable of adjustment as new data, due to changes in environment, is collected by the sensors. The system will implement SLAM to both map its environment and localize its current location in a virtual representation of the world.

## 2.3. Capabilities

Based on the input of a desired destination, the wheelchair will be capable of autonomous navigation to points at least 50 feet from its current location, moving at a pace of 3 mph. The wheelchair will begin planning a path to the desired destination as it begins moving toward such desired destination; the wheelchair will be capable of maneuvering around corners, obstacles and through entryways on the planned path by modifying its originally planned path until reaching the desired destination.

# Verification Approach/Plan

To ensure the wheelchair system functions properly a series of tests and inspections must be conducted, both during development and before every operation of the system. Below is an outline of these tests and instructions on how to properly inspect the system to ensure successful operation.

#### Table 2: Verification Approach/Plan

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Systems** | **Analysis** | **Model** | **Simulation** | **Testing** |
| Mechanical | ✔ | ✔ | ✔ | ✔ |
| Motors |  |  |  | ✔ |
| Batteries | ✔ |  |  | ✔ |
| Microcontroller | ✔ |  |  | ✔ |
| Sensors | ✔ |  | ✔ | ✔ |

## 3.1. Analysis

### 3.1.1. Operation Time and Power Draw

The system will need to be tested for operation time while all sensors are running, and the wheelchair is moving. Upon assembly the wheelchair will run until the batteries are drained. Using the run time data, an estimated power draw will be calculated.

### 3.1.2. Testing Path-planning

Testing and analysis of the path-planning algorithms will be crucial in developing a successful system. This will be done through both physical tests and ROS simulation; the team will extensively navigate the wheelchair throughout hallways and simulate a testing bot in Gazebo. The testing & analysis will focus on ways to optimize the system’s path-planning capabilities, as well as confirmation of user comfort and safety.

### 3.1.3. Sensor Data Accuracy Check

Manual analysis of the data received from both the Kinect and the LiDAR sensor will be conducted to ensure there are no unexpected objects or gaps in the readings taken by the sensors. There will also be inspections of the data received as it is represented in rViz; this representation provides a straightforward way to ensure that the data is accurately representing the wheelchair’s environment.

## 3.2. Simulation

### 3.2.1. 3D CAD Models of Bracket Parts

Each mount that will be 3D printed is to be 3D modeled in Solidworks, to ensure it is correctly dimensioned. This will allow for simulations of how the parts will fit together when printed and assembled.

### 3.2.2. Obstacle Avoidance in a Virtual Environment

Using iz & Gazebo, the system’s algorithm can be tested under a far more dynamic set of circumstances. The prebuilt worlds provided by Gazebo will allow for testing the developed program in situations not easily recreated in the areas around KSU. Using this simulation tools, we can more accurately predict how the wheelchair will function in unpredictable and complex conditions.

## 3.3. Test

### 3.3.1. Sensors Testing for Calibration

Testing of the calibration before operation will be crucial in creating a consistent and reliable system. Both the LiDAR sensor and Kinect can be recalibrated, should the data received be inaccurate or unpredictable. This testing will be done by placing the sensors in a consistent start position and confirming accurate data. Should the data not be as expected, recalibration will be conducted, and the test will be restarted.

### 3.3.2. Motors Tests for Response to Commands

Manual commands will be sent from the joystick on the wheelchair to ensure the motors respond correctly when told to maneuver forward, back, and turn. Forced commands from the Arduino will confirm proper communication between the microcontroller and the motors.

### 3.3.3. Obstacle Avoidance Testing

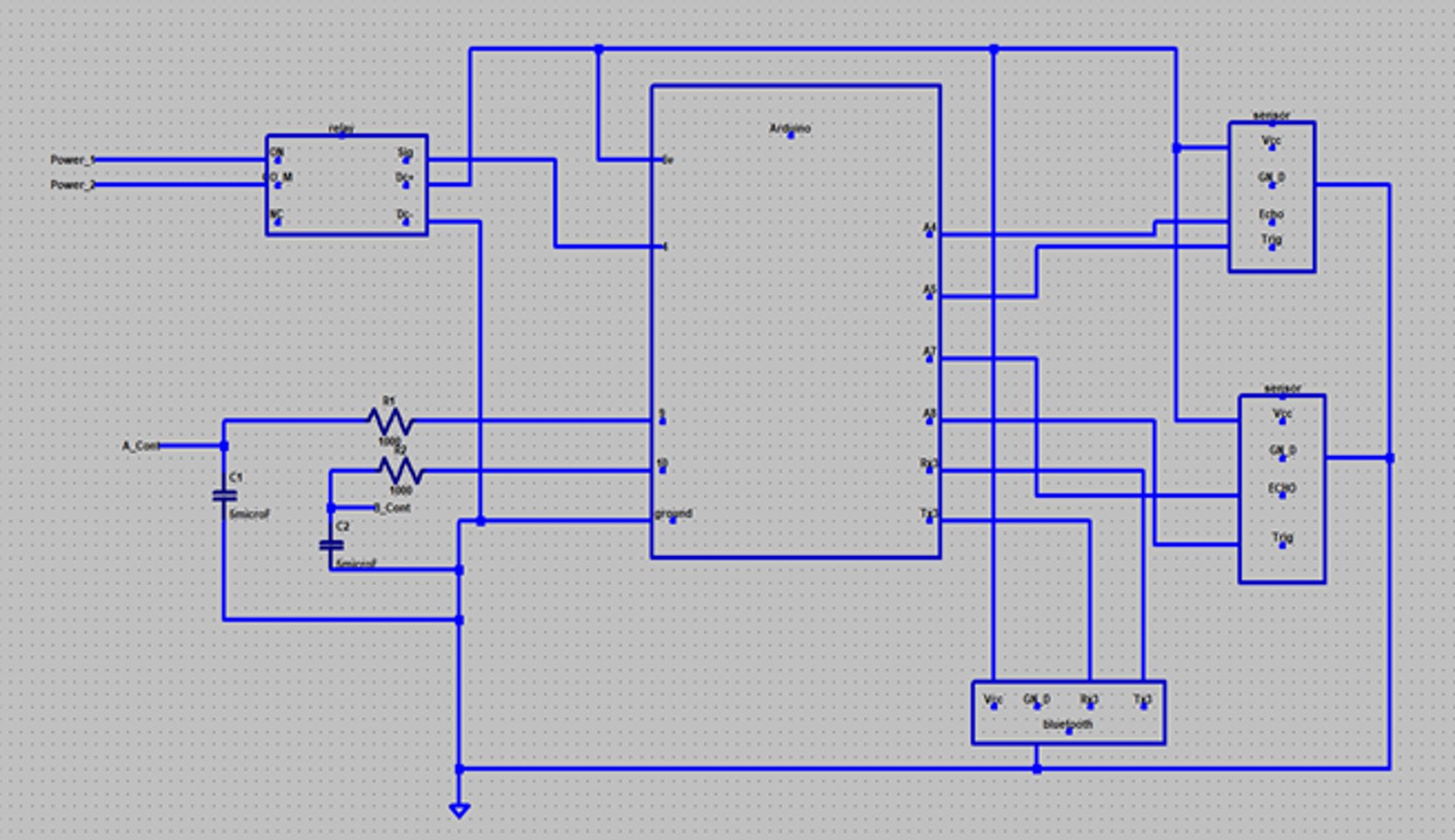
Before official presentations of the wheelchair, a forced test of the obstacle avoidance algorithm will be conducted. Ensuring that the wheelchair does not collide with obstacles on the wheelchair’s path will be crucial in preventing injury or damage to the user, the obstacle and the wheelchair.

# Basic Design & Trade Study Results

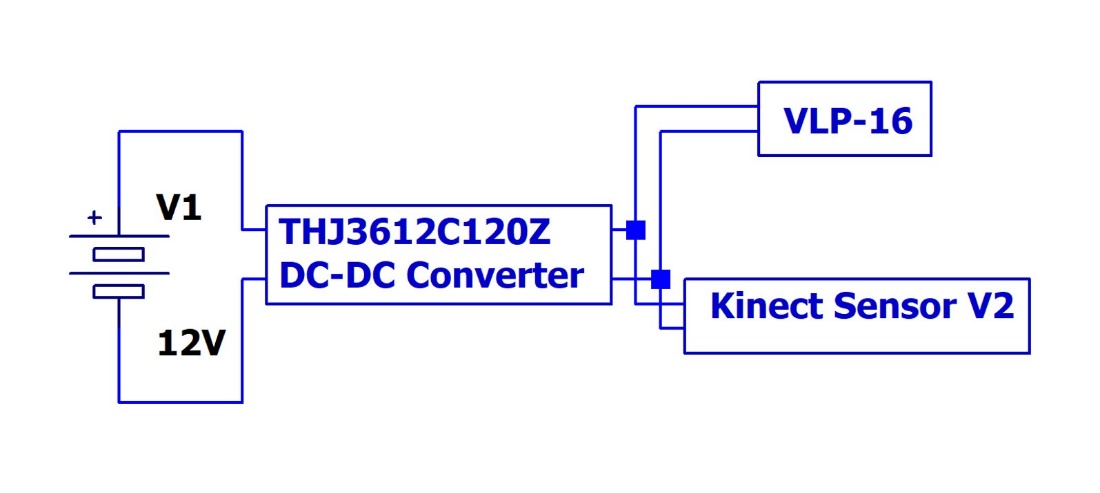
The beginning of this project has been centered around the collection of information, in order to choose desired features, as well as the software & hardware that will allow the group to accomplish such desired features. In the following section there is a collection of both design specifications of the wheelchair and its surrounding systems, as well as a series of trade studies that will show how the components being used were selected.

## 4.1. Wiring Diagrams

The wiring of the wheelchair system has been split into two drawings. The first drawing shows the digital signals from the Arduino to the wheelchair itself. This system will be connected to the laptop using the Arduino’s integrated USB connection. The second drawing is the power diagram of the sensors and battery being added to the current system.



**Figure 3:** Wiring Diagram of Digital Signals

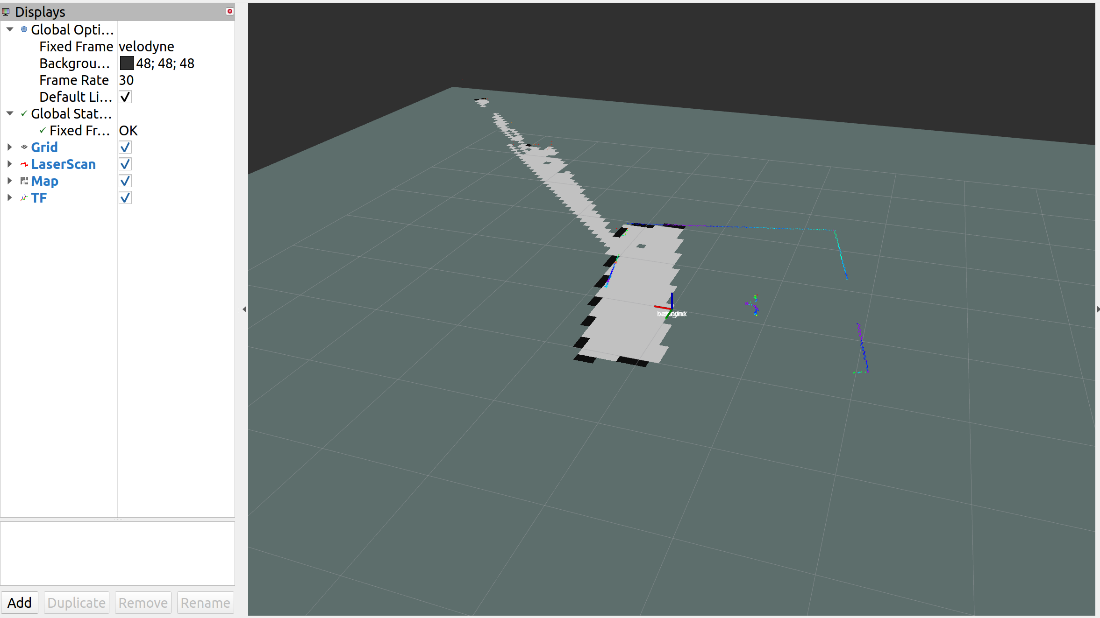


**Figure 4:** Wiring Diagram of Power to New Sensors

## 4.2. Analysis & Simulation Results

### 4.2.1. Simulations Using ROS

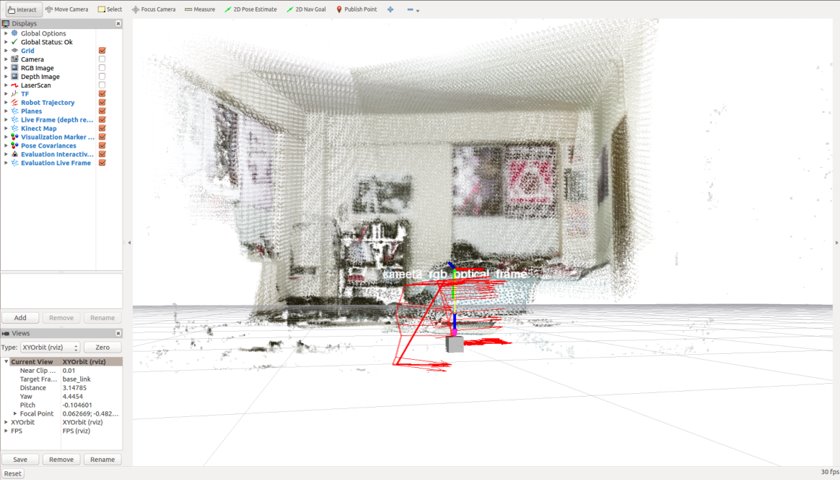
The LiDAR sensor in its current state requires some editing in order to function properly when used with ROS SLAM and gmapping. The sensor’s output does not contain any odometry data making it difficult to calculate changes in both location and orientation. The sensor is currently capable of producing LaserScan data that is used to create a 2D digital representation of the data collected by the LiDAR.



**Figure 5:** VLP-16 SLAM Early Implementation

### 4.2.2. Kinect RGB and Infared Analysis

As demonstrated in the images below, the Kinect sensor is capable of both RGB and Infrared data collection. Figure 6 shows (through the use of rViz) how the Kinect’s infrared visualization can be blended with its RGB visualization and used to 3D map its surroundings while retaining the original color of the objects scanned. Figure 7 demonstrates the with which different filters can be applied to what is visualized by the Kinect’s sensors in order to further improve any image processing algorithms, such as the object detection & avoidance algorithm the group is incorporating to this project.



**Figure 6:** Kinect Rviz Capabilities Analysis



**Figure 7:** Kinect Camera Capabilities Analysis

## 4.3. Trade Study Items

### 4.3.1. Microcontrollers

The team had to select a microcontroller to orchestrate the operation of the two electric motors, based on the computation of the sensors’ data by the Linux workstation. The team agreed to retain the Arduino Mega 2560. This is due to its 54 digital I/O pins (of which 15 provide PWM output) giving us the freedom to implement as many features as desired. Another advantage the Arduino Mega provides is its clock speed of 16 MHz. For the computation of the sensors’ inputs and the control of the Arduino Mega, the team vetted the set up implemented by the ‘Intelligent Wheelchair’ team (the team who worked on the wheelchair project last semester); such set up was composed of a Raspberry Pi 3.0 B & a 7” touchscreen attached to the wheelchair by a mount. The team then performed a trade study to compare the capabilities & features provided by the forementioned setup to the capabilities & features provided by a 2-in-1 laptop, the criteria used for such examination was processor speed, usability, memory, and form factor:

#### Table 3: Linux Machine Trade Study

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Criteria** | **Weight** | **Dell Latitude 7285** | | **Raspberry Pi 3.0** | |
| **Rating** | **Weighted** | **Rating** | **Weighted** |
| **Processor Speed** | 0.5 | 4 | 2 | 3 | 1.5 |
| **Memory** | 0.2 | 4 | 0.8 | 2 | 0.4 |
| **Usability** | 0.1 | 5 | 0.5 | 3 | 0.3 |
| **Form Factor** | 0.1 | 2 | 0.2 | 5 | 0.5 |
| **Total** | 1 |  | 3.5 |  | 2.7 |

We prioritized processor speed & RAM because the wheelchair’s user might often find themselves in a situation that requires intensive data analysis and near immediate reaction by the computer, such as when performing path planning & needing to avoid obstacles above & below the floor. We also preferred the laptop’s usability & form factor, while any 2-in-1 laptop remains folded as a tablet, it allows the touchscreen of the laptop to be used as a user interface; whenever necessary, the laptop can be...

### 4.3.2. Sensors

The team performed a trade study to compare various LiDAR sensors offered by our mentor Chris. The criteria used for such examination were range, sample frequency, samples per period, size, weight, and cost:

#### Table 4: LIDAR Trade Study

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Criteria** | **Weight** | **Velodyne VLP-16** | | **RPLiDAR A2** | |
| **Rating** | **Weighted** | **Rating** | **Weighted** |
| Range | 0.3 | 5 | 1.5 | 1 | 0.3 |
| Samples / period | 0.3 | 4 | 1.2 | 2 | 0.6 |
| Size & Weight | 0.25 | 3 | 0.75 | 4 | 1 |
| Cost | 0.15 | 1 | 0.15 | 5 | 0.75 |
| Total | 1 |  | 3.6 |  | 2.65 |

The team chose the Velodyne Puck (VLP-16) due to its greater range of visualization, its greater number of samples per period of rotation, and despite its slightly greater size, the Velodyne Puck is still very versatile for our intents and purposes. The only criteria in which the RPLiDAR A2 greatly outperforms the Velodyne Puck is in cost, with an average price comparison of 1:10 when purchased new.

### 4.3.3. Motors

Since the essential structure of our project is an electric chair, we chose to retain the electric motors originally assembled with the wheelchair. This, due to the fact that we confirmed that such motors have the power to carry the wheelchair’s user around, as well as the additional weight of the components we are attaching to the wheelchair for our project.

### 4.3.4. Battery

In order to power all the additional components being implemented throughout our project, the team realized the need to implement an additional battery. The team performed a trade study to compare various batteries offered by our mentor Chris. The criteria used for such examination is wight, charge capacity (Amp-hrs), physical size and cost:

#### Table 5: Battery Trade Study

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Criteria** | **Weight** | **Bioenno 12V** | | **Eco-Worthy 12V** | |
| **Rating** | **Weighted** | **Rating** | **Weighted** |
| **Weight** | 0.2 | 3 | 0.6 | 4 | 0.8 |
| **Amp-hrs** | 0.1 | 3 | 0.3 | 3 | 0.3 |
| **Form Factor** | 0.2 | 4 | 0.8 | 2 | 0.4 |
| **Cost** | 0.5 | 5 | 2.5 | 2 | 1 |
| **Total** | 1 |  | 4.2 |  | 2.5 |

The team preferred the Bioenno 12V battery because although it is a bit heavier, it provides a better size, shape and connector facilitating the attachment of the battery to the wheelchair structure and the implementation of the battery into the electrical system, while still providing as much charge capacity.

### 4.3.5. Structure

The team performed a trade study to compare the current prototype’s structure with the structure of other electric wheelchairs available in the market. The criteria used for such examination is weight, footprint, motor interface and cost:

#### Table 6: Frame Trade Study

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Criteria** | **Weight** | **Yurob Protable** | | **Electra 7** | | **Falcon Portable** | |
| **Rating** | **Weighted** | **Rating** | **Weighted** | **Rating** | **Weighted** |
| **Weight** | 0.2 | 4 | 0.8 | 2 | 0.4 | 2 | 0.4 |
| **Foot Print** | 0.2 | 3 | 0.6 | 2 | 0.4 | 2 | 0.4 |
| **Motor Interface** | 0.3 | 4 | 1.2 | 3 | 0.9 | 3 | 0.9 |
| **Cost** | 0.3 | 5 | 1.5 | 2 | 0.6 | 3 | 0.9 |
| **Total** | 1 |  | 4.1 |  | 2.3 |  | 2.6 |

The team preferred to retain the current structure due to its aluminum alloy composition, which makes it very lightweight while still being very rigid. Retaining the current structure also saves the team what would likely be over $1000. The only disadvantage the team noticed is the lack of comfort of the textile used for the user to sit on, we therefore added some cushioning to alleviate the situation.

## 4.3. Bill of Materials

Below is a detailed Bill of Materials estimating the cost required to build the Automated Wheelchair.



**Figure 8:** Bill of Materials

## 4.4. Minimum Success Criteria

For the system to be considered a success a series of requirements must be achieved. The wheelchair must navigate to a target point with an error of less than or equal to 4 feet. This means that the wheelchair must be less than 4 feet from the target when navigation is complete. Secondly, the wheelchair must avoid walls and stationary objects by not coming closer than 2 feet to the obstacle. The wheelchair must be capable of avoiding a moving object that passes in front of it as it moves down the planned path. Finally, the wheelchair is to be able to navigate to a point at least 50 feet away at an average rate greater than 2 mph.z

## 4.5. Design Optimization/Improvement

Upon completion of the project there will still be several opportunities for both expansion and improvement. The first of these expansions in capability would be to implement voice commands that would allow more impaired users to easily utilize the system. Using the public Amazon Alexa voice recognition algorithm and the microphone array on the Microsoft Kinect, voice commands could be an added functionality.

The manual joystick currently on the wheelchair is restricted to x and y axis navigation. This joystick could be replaced with a superior one to provide more precise movement commands as well as the ability to rotate about the z axis if desired.

While this project implements D\*Lite as the path planning algorithm set, there are many alternatives that have different advantages. A GUI that allows the user to select from a variety of path planning options based on the current need could be implemented to improve the dynamic ability of the wheelchair to meet the users' needs.

The current wiring of the digital signals is not user friendly. It is in a place that interferes with the comfort of the user when they sit while also being at risk of dislodging connections if bumped. A reworking of this circuit would create a more comfortable user experience while also improving the reliability of the system.

Even with functional obstacle avoidance there is always room for improvement. Expanding the types of obstacles that the wheelchair recognizes and avoids would create a safer system and prevent any unexpected collisions due to a failure to notice an obstacle on the path. Specifically, the system could be improved in a way that allows it to recognize sudden changes in elevation that might result in either tipping or uncontrollable acceleration of the wheelchair.

# Other

This section covers available resources, project schedule, and team responsibilities.

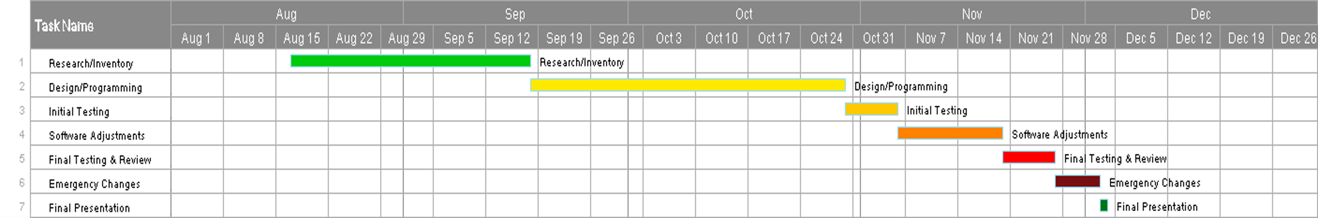
## 5.1. Available Resources

|  |  |
| --- | --- |
| Mentors:   * Chris Voicu | Facilities:   * Engineering Lab Q-118 * Senior Design Lab |
| Hardware   * LIDAR Sensor * Ultrasonic Sensors * Arduino MEGA * Wheelchair Frame * Kinect Camera * Dell Latitude 7285 | Software   * Solidworks * AutoCAD Electrical * Ubuntu 20.04 * C++/Python/ROS |

## 5.2. Project Schedule & Budget



**Figure 9:** Project Budget



**Figure 10:** Project Schedule

## 5.3. Team Chart/Assignments

Max Bronson is the team leader as well as one of the programming leads, in charge of the LiDAR interfacing, ROS, and Arduino Programming. Oscar Montealegre is the other programming lead, focusing on Kinect Interfacing, Object Identification, and ROS programming. Denny Mannakulathil is in charge of 3D design, which makes him responsible for the modeling, drawing, and 3D printing of all parts needed.

# References

[LIDAR-based autonomous wheelchair](https://ieeexplore.ieee.org/document/7894082)

[Autonomous wheelchair design reference](https://journals.sagepub.com/doi/10.5772/55477)

[Python reference #1](https://www.youtube.com/watch?v=XCKWZAtKTnU&list=PLGs0VKk2DiYzguDvh5xk2XoX9V1VKP5Hv)

[3D design reference](https://www.tinkercad.com/things/eF5YDip8WDO-lidar-scanse)

[SolidWorks reference](https://help.solidworks.com/2018/english/SolidWorks/sldworks/HIDD_OPTIONS_IMPORT_VRML_2.htm)

[D\* lite reference](https://github.com/Sollimann/Dstar-lite-pathplanner)

<https://ieeexplore.ieee.org/document/7894082>

[ROS tutorial for beginners](https://youtube.com/playlist?list=PLk51HrKSBQ8-jTgD0qgRp1vmQeVSJ5SQC)

[OpenCV with use of python reference #1](https://www.youtube.com/playlist?list=PLS1QulWo1RIa7D1O6skqDQ-JZ1GGHKK-K)

[OpenCV with use of python reference #2](https://www.youtube.com/watch?v=oXlwWbU8l2o)

[ROS reference #1](https://www.youtube.com/playlist?list=PLk51HrKSBQ8-jTgD0qgRp1vmQeVSJ5SQC)

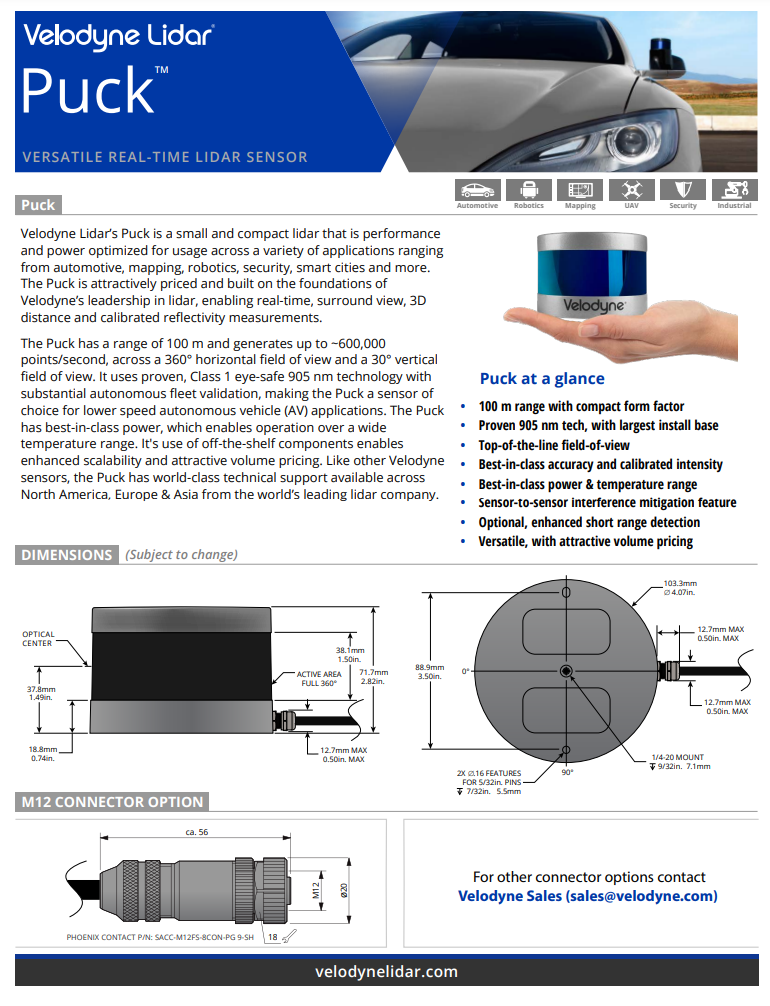
[ROS reference #2](https://www.youtube.com/playlist?list=PL8dDSKArO2-m7hAjOgqL5uV75aZW6cqE5)

[ROS reference #3](https://www.youtube.com/playlist?list=PLJNGprAk4DF5PY0kB866fEZfz6zMLJTF8)

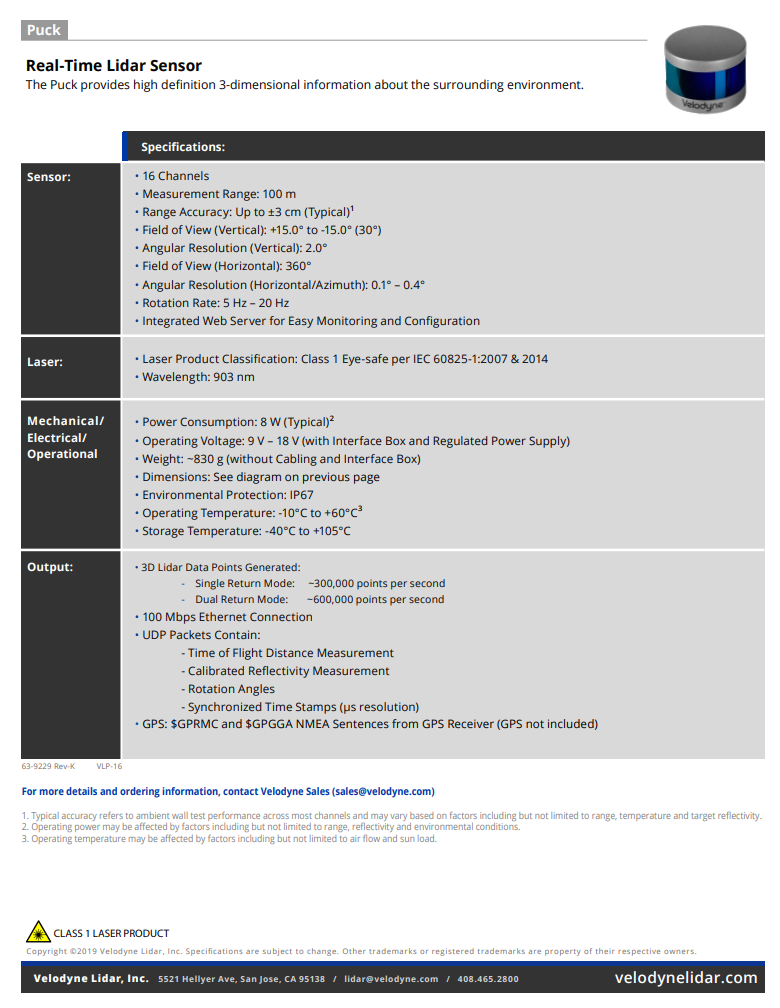
[Arduino reference](https://www.youtube.com/playlist?list=PLGs0VKk2DiYw-L-RibttcvK-WBZm8WLEP)

# Appendices

## Appendix A: Velodyne VLP-16

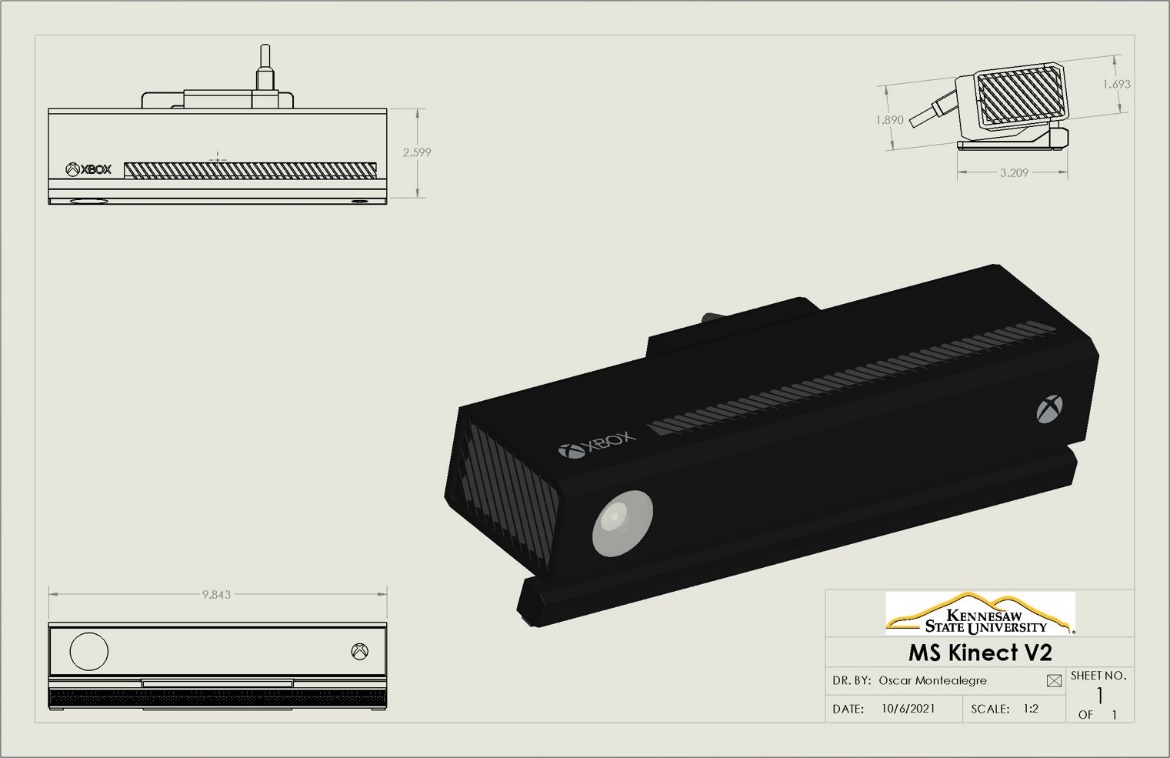


**Figure 11:** VLP-16 Data Sheet 1

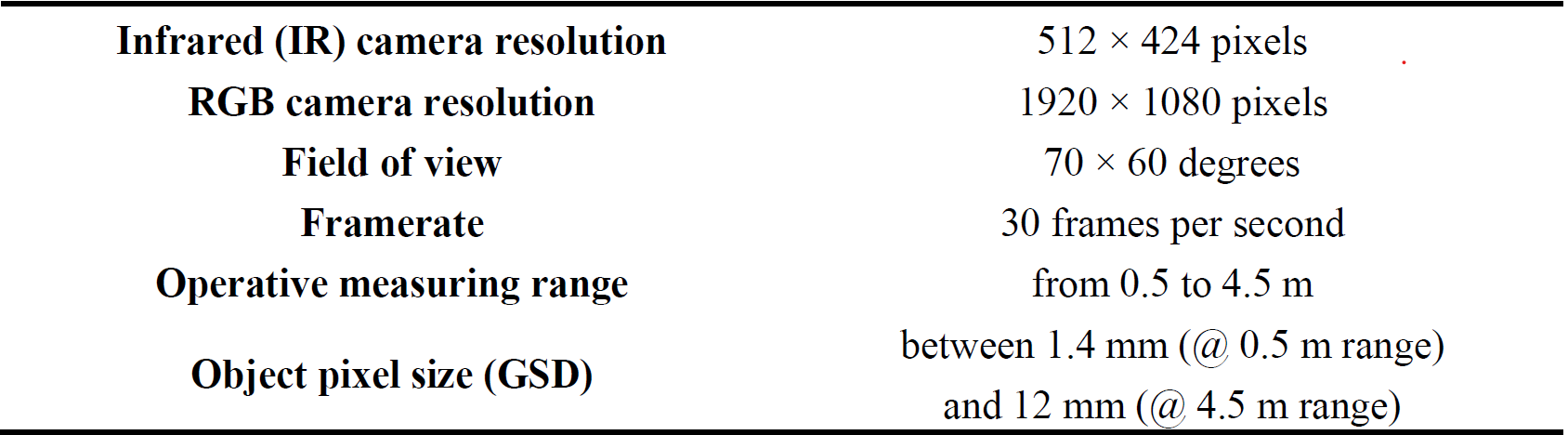


**Figure 12:** VLP-16 Data Sheet 2

## Appendix B: Microsoft Kinect V2

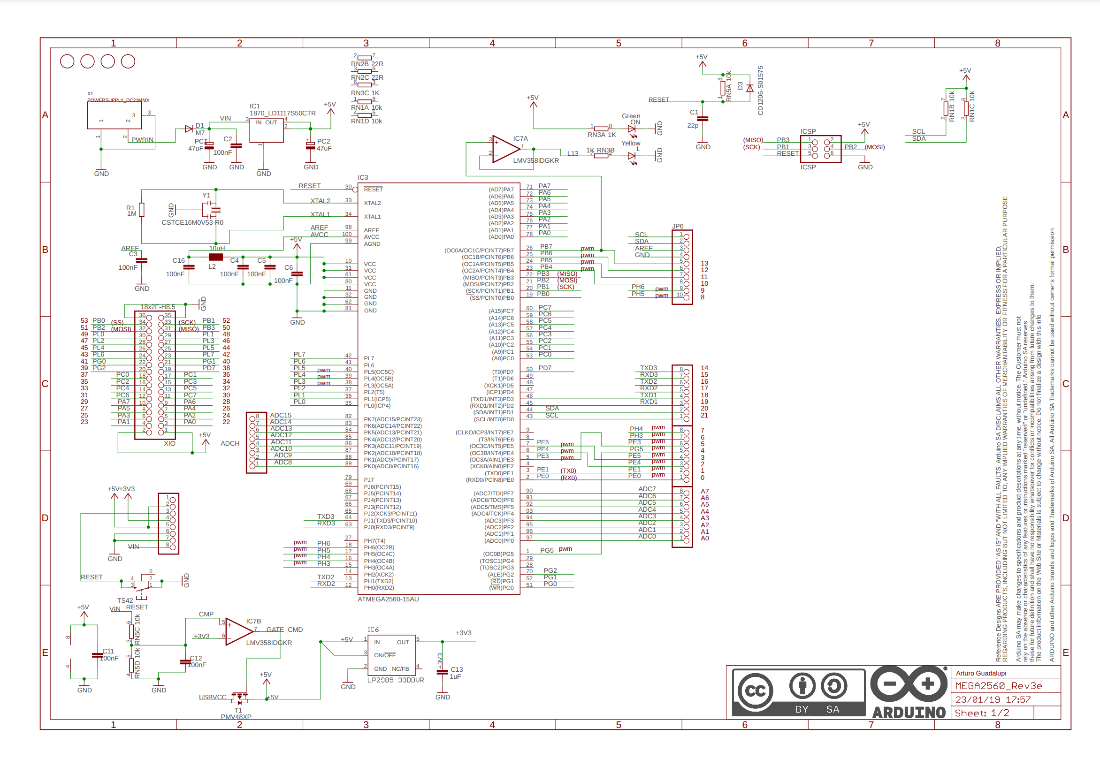


**Figure 13:** Microsoft Kinect V2 Rendering

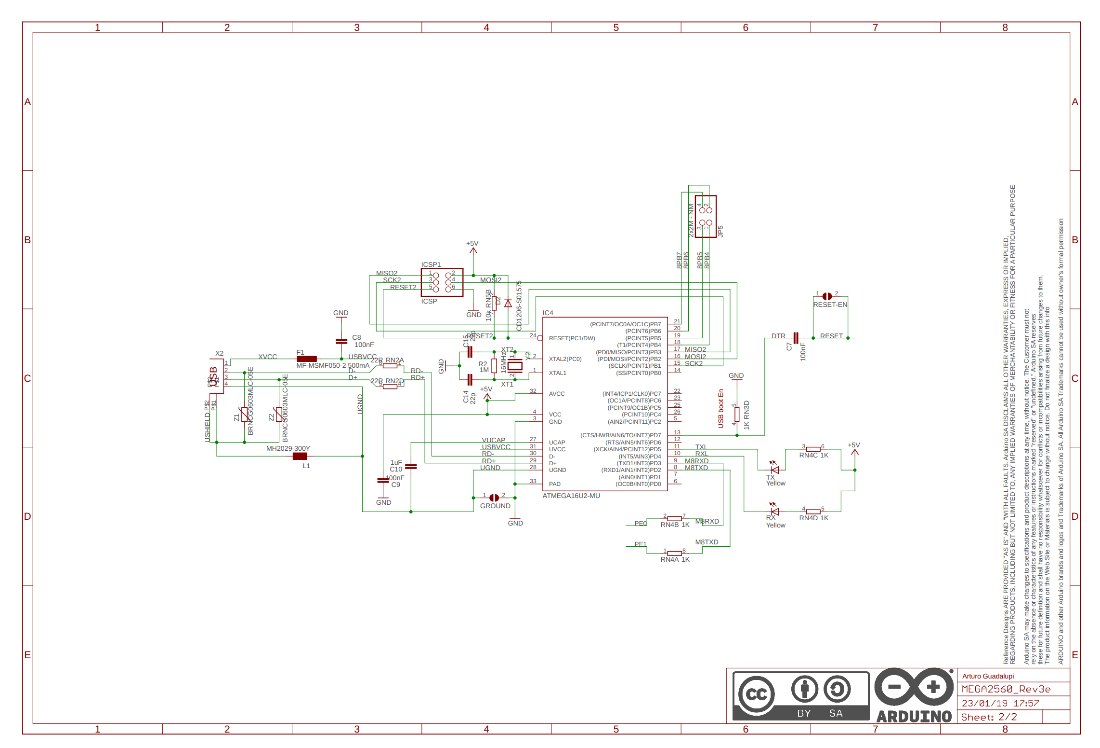


**Figure 14:** Microsoft Kinect V2 Data Sheet

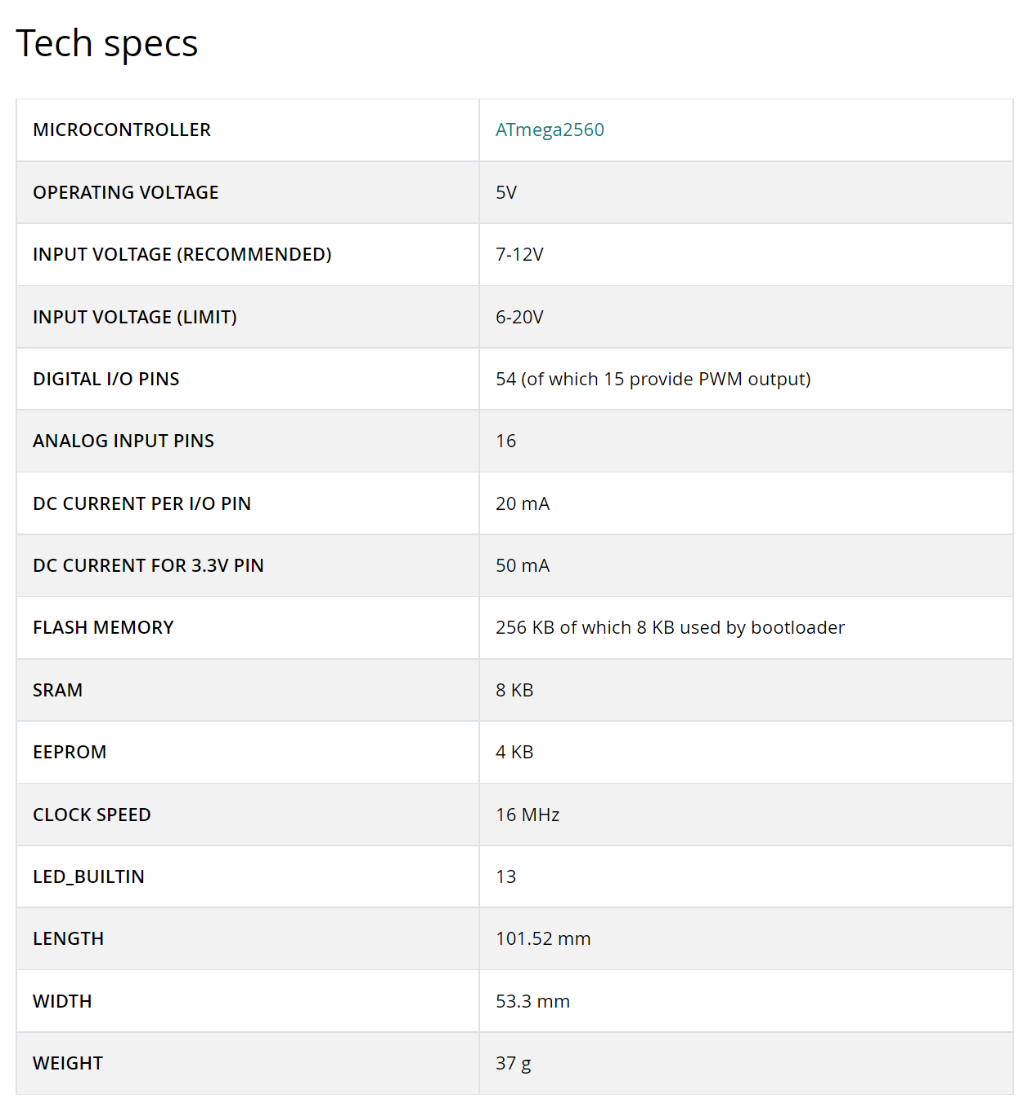
## Appendix C: Arduino 2560



**Figure 15:** Arduino 2560 Rev3 Wiring Diagram 1

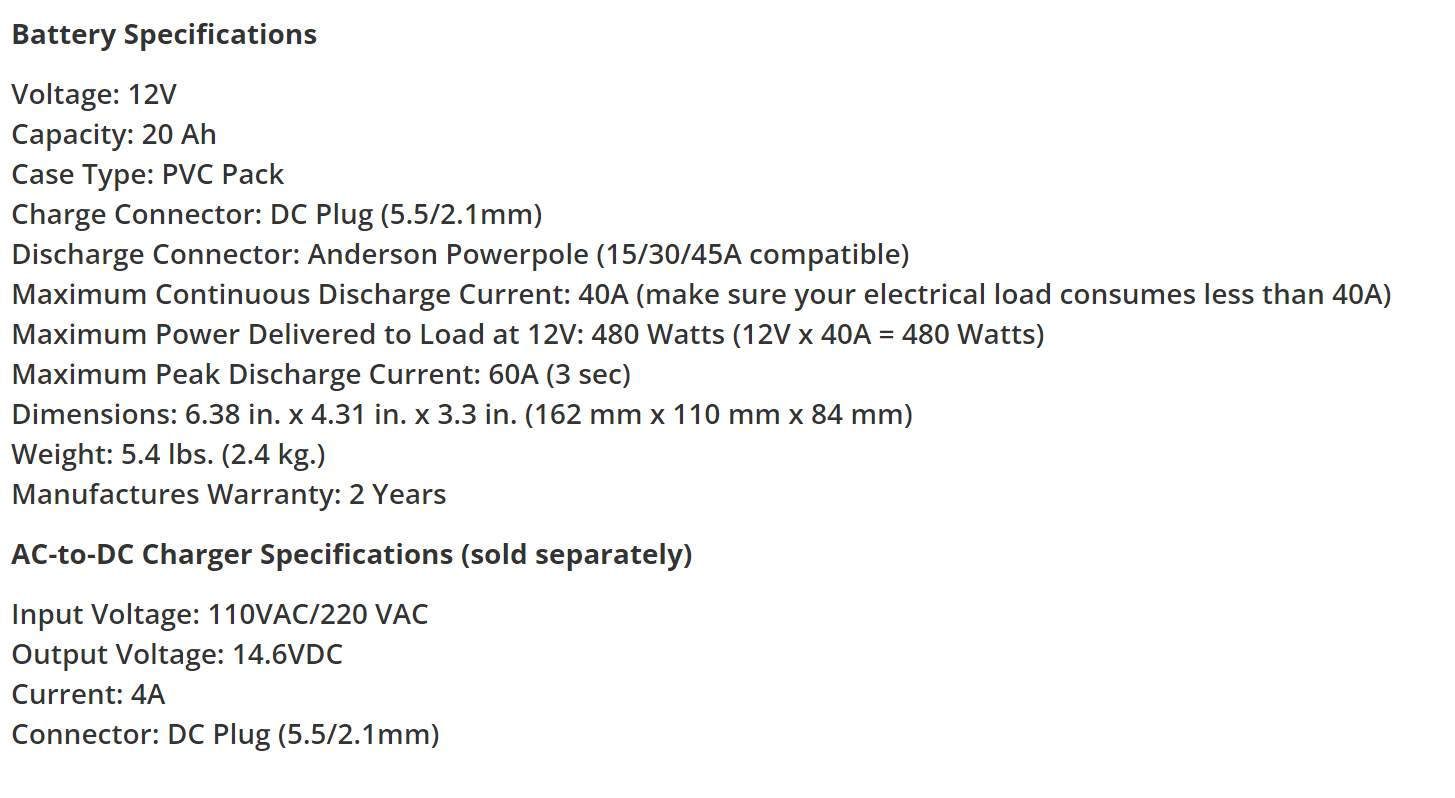


**Figure 16:** Arduino 2560 Rev3 Wiring Diagram 2



**Figure 17:** Arduino 2560 Rev3 Technical Specifications

## Appendix D: Bioenno Lithium-Ion Battery



**Figure 18:** Properties of Bioenno Battery