Molecular Communication Over the Air

ECE4012 Senior Design Project

Team Molecular
Project Advisor: Matthieu Bloch

Jacob Callahan, jcallahan8@gatech.edu
Nick Fahrenkrog, nick.fahrenkrog@gatech.edu
Constance Perkins, cperkins39@gatech.edu
Zhongyang Shi, zshi39@gatech.edu
Jun Xiang, jxiang30@gatech.edu
Siyan Yu, siyan_yu@gatech.edu

Submitted

December 14, 2017

Table of Contents

Executive Summary

Molecular communication is a form of communication that uses molecules instead of electromagnetic waves to transmit information. It has a variety of applications in biomedical, military and environmental areas. In situations like earthquakes where traditional EM waves are unable to propagate, molecules can travel past the environmental barriers and achieve successful communication. The most promising application is in the biomedical field, using molecular communications in drug delivery systems.

The project goal is to design and prototype a molecular communication system capable of sending text messages from one machine to another. The system consists of an electronic spray bottle that sprays alcohol to a sensor. The spray rate and concentration of molecules will be controlled by a microcontroller. The sensors that receive the transmitted alcohol measure the alcohol concentration and are connected to a different microcontroller which demodulates the signal. Further analysis of the gathered data will be done in MATLAB.

This prototype will be built at the macro-scale to allow focus on researching how to improve the speed and accuracy of this method of communication. The total cost to build a prototype of this device is $91.09, which covers two Arduino Uno microcontrollers, one spray bottle, two alcohol sensors (with different sensitivities and operating points), and the cost to manufacture PCB boards. Since the primary purpose of this project is research, the device will remain in a cycle of working product and implementing improvements. One prototype will be produced to show that text messages can be encoded into binary and transmitted by sending low or high concentration molecules. All improvements will focus on increasing the rate of communication.

Molecular Communication Over the Air

1. Introduction

Team Molecular will develop a wireless communications system that transmits data by adjusting the concentration of alcohol molecules in the air. The team requests $91.09 to fund the development of a prototype of the system.

1.1 **Objective**

The goal of this project is to develop a prototype communication system that uses a spray of alcohol to transmit data. The transmitter hardware consists of a bottle to spray the alcohol and a microcontroller that controls the spray bottle. The microcontroller encodes the inputted message as a binary sequence and then modulate the channel symbols into a carrier signal. The binary sequences manifest as pulses of alcohol being sprayed, the alcohol would then propagate through air and accelerated by a fan to the receiver. The receiver consists of sensors connected to a microcontroller that are capable of detecting the change in concentration of the sprayed alcohol. The microcontroller is programmed to be able to decode the sequence of pulses and restore the binary sequence from the inputted message and display it. The binary sequence encoding is such that a zero would represent a low concentration of alcohol at the receiver while a one would represent a high concentration of alcohol at the receiver. A block diagram of the envisioned prototype is featured in Figure 1.

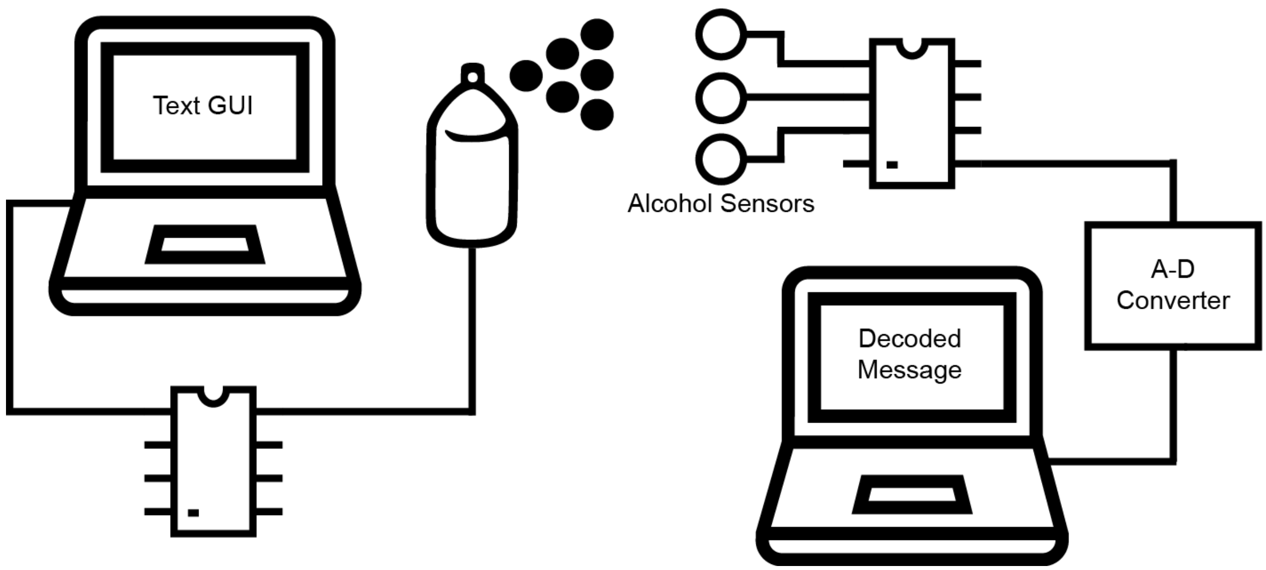


Figure 1. A diagram of the completed molecular communication prototype.

1.2 Motivation

The main motivation for this project is to develop a new way of wireless communication that can be implemented when traditional wireless methods cannot be used. Some areas where standard electromagnetic communication is difficult include inside tunnels and very small distances [1]. Tunnels are a problem because electromagnetic waves cannot bend around turns, but it is possible to make air flow around turns. Very small distances, on the order of nanometers, are a problem because the antenna size must be on the order of a wavelength of the transmitted signal. Therefore, the signal would have to be transmitted in the ultraviolet region to be detected by a nanometer-sized antenna, which is not feasible because these frequencies do not travel through most materials very well and are potentially hazardous to humans. However, it is feasible to make a device that can detect molecules at that scale.

1.3 Background

Currently, there is at least one research group that has built a working prototype. Our design draws some ideas from their work, including the general system architecture. This group consisting of Nariman Farsad, Weisi Guo and Andrew W. Eckford has demonstrated that it is possible to send text messages using this system, although it takes around three seconds to transmit a single bit. They used the ITA2 standard for their alphabet, which represents each character as a five-bit sequence. Therefore, it takes an average of fifteen seconds to transmit each character using their system. Part of the reason for this slow transmission rate are the resume and response times of the receiver. The response time is the amount of time the sensor takes to react to a change in concentration and the resume time is the amount of time the transmitter must wait before sending another bit. They also performed several tests which demonstrated that having a fan to blow the chemicals toward the sensor greatly improves accuracy of detecting the signal, even using a fan that generates turbulence. They also showed that their system had some nonlinearities, which caused some errors in reading the signal [1].

2. Project Description and Goals

The goal of this project is to create a functioning prototype of a wireless communication system that can transmit data by adjusting the concentration of particles in the air through a transmitter and a receiver. The transmitter consists of a GUI on a laptop to allow the user to input text messages and a microcontroller that controls a mechanized spray bottle. The receiver contains a sensor and microcontroller to record the change in concentration of the particles and convert the analog data to digital data and a laptop to decode and translate data into the text message. Project objectives include:

- Successfully transmitting a message from transmitter to receiver
- Building prototype easily and spending a smaller amount on parts

- Making the platform so the data collected is easily accessible for others to use for research

3. Technical Specifications

Major components of this system are the embedded transmitter and the embedded receiver. Table 1 displays the specifications of the microcontroller that will be implemented in both the transmitter and receiver, an Arduino Uno Rev3, Table 2 displays the specifications of the two alcohol sensors implemented on the receiver and Table 3 shows the specifications of the receiver PCB.

Table 1. Arduino Uno Rev3 Specifications [2]

Specification	Value
Microcontroller	ATmega328P
Operating voltage	5v
USB	14
Digital I/O	6
Analog Input Pins	6
Flash Memory	32 KB
SRAM	2 K
EEPROM	1 K
Clock Speed	16 MHz
Length	68.6 mm
Width	53.4 mm
Weight	25 g

Table 2. Alcohol sensor specifications

Specification	Value	
	MQ-3 [3]	MQ303A [4]
Diameter	16.8 mm	9.4 mm
Height (w/o pins)	9.3 mm	8 mm
Detection Range	25 - 500 ppm	20 - 1000 ppm
Heater Voltage	5V ± 0.1V (AC or DC)	0.9V ± 0.1V (AC or DC)
Circuit Voltage	5V ± 0.1V (DC)	3V ± 0.1V (DC)
Output Voltage	2.5V < Vo < 4.0V	-
Material	Plastic	Metal
Cost	$4.95	$5.50

Table 3. Printed circuit board specifications

Specification	Value
Length	70.79mm
Width	41.90mm
Height	26.50mm
Trace Width	0.3048mm
Number of Components	13 ● 2 of each 1k/5k/10kΩ Potentiometers ● 2 2N3904 NPN Transistors ● 1 TL071 Op-Amp ● 1 MQ-3 Alcohol Gas Sensor ● 1 MQ303A Alcohol Gas Sensor ● 2 Headers

Table 4. Paint Gun specifications [13]

Specification	Value
Input Voltage	120V A.C.
Operating Current	4.0A A.C.
Weight	1.45 kg
Length	?
Width	4.96 in.
Height	10.16 in.

Table 5. Software Environment

Operational System	Windows
Transmitter Programming	Arduino IDE
Receiver Programming	Matlab 2017b, with Matlab Arduino Support Package

Table 4. Solid State Relay [14]

Specification	Value
Input Operating Voltage	3-32 V D.C.
Trigger Current	7.5mA @ 12V
Output Operating Voltage	24-380 V A.C.
Output Voltage Drop	1.6 V
Rated Output Current	40A
Isolation Strength	50MΩ/500V D.C.
Weight	105g

4. Design Approach and Details

4.1 <u>Design Approach</u>

4.1.1 Text GUI

The goal of this project is to transmit text messages over the air via chemical molecules. The text that the user wants to input into the system is collected by a GUI, Graphical User Interface, on a PC and then the GUI will send it via a serial port to the Arduino microcontroller. The text input GUI is programmed in C# and allows the user to input a message consisting of any characters in the English alphabet and allows the user to select the time bin, 10, 20, or 30 seconds, by selecting the radio buttons. The user will then send all the information chosen to the Arduino by pressing the "Send" button.

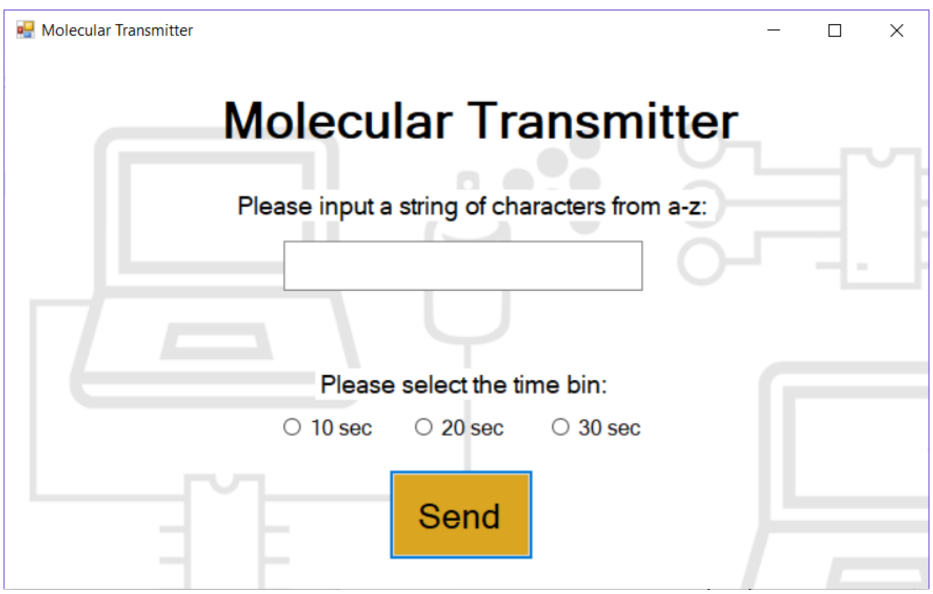


Figure 4.1. Appearance of the text input GUI

4.1.2 Microcontroller Programming in transmitter

The Arduino Uno is constantly check the USB serial port for any incoming text message from the GUI. Once it receives the message in the form of a character string, the Arduino will parse through the string. The first character of the strong is either a 1, 2, or 3, which the Arduino will use to set the time bin. First, a two-bit header, “10,” is transmitted to indicate that a transmission is about to start in the channel. Then the Arduino will parse through the rest of the string, associating each character with a five-bit binary sequence, Baudot-Murray code, according to the International Telegraph Alphabet No. 2 (ITA2) standard [6, 11]. This standard was chosen because of its relative short bit length and fixed length. In a loop, the Arduino will go through the binary code and control the motor of the spray gun depending on whether the bit is a 0 or a 1. A ‘0’ bit is modulated as no spray and a ‘1’ is modulated as spray. When the transmission is complete, the Arduino will once again transmit the two-bit header to indicate to the receiver that the transmission is over.

4.1.3 Solid state relay and spray bottle

The Hyper Tough paint gun was chosen as the sprayer to transmit the alcohol to the sensors [13]. It can store a variety of liquid chemicals. The paint gun requires a 120V A.C. input. To control this with an Arduino, we decided to use a solid state relay (SSR). [14] The SSR consists of a phototriac, a snubbing circuit and a zero-cross circuit. The snubbing circuit is there to absorb any voltage spikes from the induction motor. The zero-cross circuit is there to ensure the triac is not switched on until the input is very close to zero volts, which helps prevent large voltage spikes. The phototriac is controlled by two LEDs, which the Arduino can handle with ease.

4.1.4 Alcohol sensors and printed circuit board

The receiver includes two alcohol sensors connected to a microcontroller. We use the MQ-3B and the MQ303A for their availability and low cost. Each of the alcohol sensors acts as a variable resistor whose value is inversely proportional to the concentration of alcohol in the surrounding air. Adding a load resistor ($R_{L)}$ in series and measuring across it creates a voltage divider such that the output voltage V_{out} is related to the input voltage V_{in} as $V_{out} = V_{in} \times \frac{R_L}{R_L + R_s}$. In this configuration, it is clear that as the alcohol concentration increases, R_s decreases and V_{out} increases, approaching V_{in}. The reason for using more than one sensor is to compare readings between them to confirm accuracy. The microcontroller will be served to convert the analog voltage signal from the sensors to a digital signal which can be delivered to a computer via USB. The block diagram of the receiver is shown below in Figure 4.2.

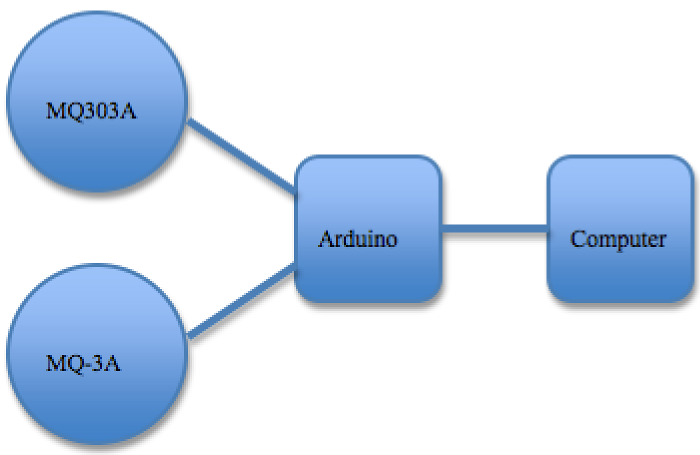


Figure 4.2. Graphic representation of the receiver

The MQ303A requires 1V across its heater. Unfortunately, 1V sources are difficult to find, so we decided to make one ourselves. The circuit, shown in Figure 4.3, consists of a voltage divider producing 1V to the positive terminal of an op-amp. The op-amp controls two transistors in parallel whose output is connected to the sensor heater and fed back to the negative terminal of the op-amp. The transistors are needed because the op-amp cannot provide enough current to the heater. The reason for having two transistors is to split the current load to prevent overheating.

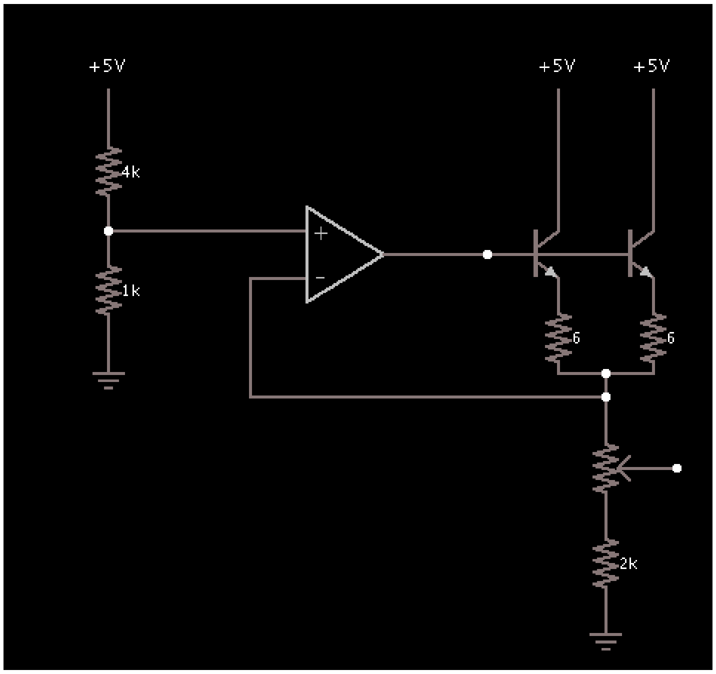


Figure 4.3. Schematic of the 1V source.

The printed circuit board was created to create a compact version of the receiver. A picture of the PCB and the wiring are shown in Figure 4.4. During testing, it was found that there was a 2V offset and the sensors were not very sensitive. Both problems can be attributed to the load resistances being too large, however we did not have time to test this.

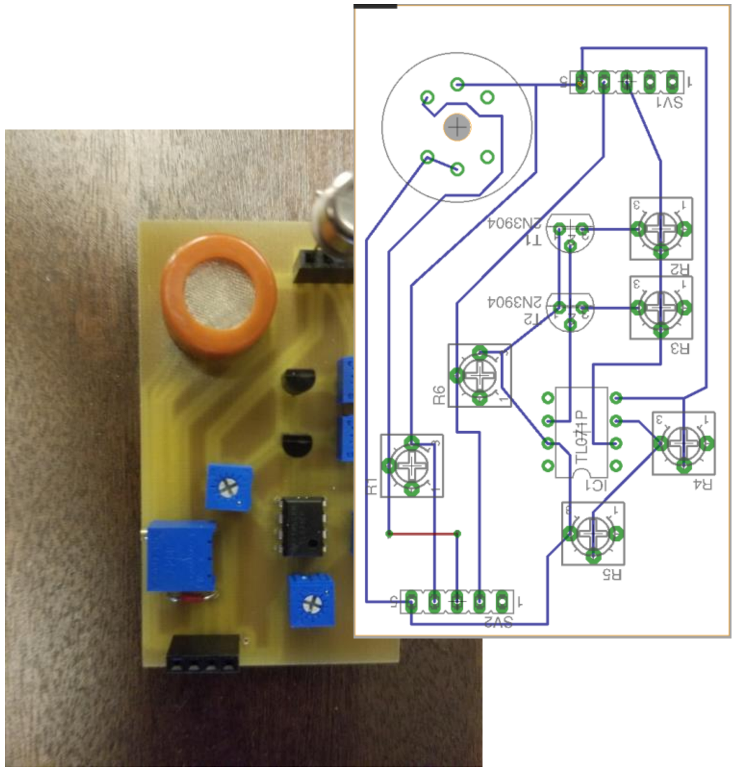


Figure 4.4. Picture of PCB and wiring.

4.1.5 Message decoding in receiver

With real-time concentration signal from receiver Arduino, Matlab program demodulates the signals back into binary symbols and decodes the symbols into characters. Then the computer GUI displays the characters. Here comes very important system information. Our Matlab program requires Matlab Arduino Support Package which allows Matlab to directly interact with Arduino. As an alternative, we could program with Arduino IDE but Matlab has much more built-in functions convenient for signal processing.

The demodulation replies on detecting concentration surge. Specifically, the program demodulates a surge in concentration as ‘1’. To detect a surge, the program finds peaks of the original signal. User needs to manually set up threshold on time and voltage to exclude close peaks. Here Matlab function, ‘findpeaks()’, is satisfactory. To demodulate ‘0’s, the program takes time difference of consecutive surges, and divided the time difference with user-determined time interval. The result should be rounded to integer and then minus one to compute the number of ‘0’s between two surges. An extra ‘10’ bit sequence is added to every starting and ending of new message for synchronizing the timing between the transmitter and the receiver.

The next step, decoding, is straightforward. Based on the ITA2 table, a binary-to-character conversion can be hard coded. It’s a necessary concern in the future that effectiveness of signal reception is highly related to time synchronization. When a peak is delayed in time due to disturbance, the actual time interval becomes shorter or longer than user-determined time interval We’ll discuss more in 4.2.

Codes and Standards

International Telegraph Alphabet No. 2 (ITA2) standard is one of the most significant standards for our design; it is a known encoding/decoding standard for digital communication. The encoder and decoder are programmed based on this standard.

4.2 Constraints, Alternatives, and Tradeoffs

4.2.1 Testing System Variables

The first system parameter is the time interval between each bit transmitted. Since we are using alcohol to transmit data, it takes time for the alcohol particles in the air to evaporate so that the channel is ready for the next transmission. As shown in Figure 4.5 and Figure 4.6, the 15-second time interval has a much cleaner alcohol reading than the 10-second time interval testing does. And the spikes are more distinguishable for later message decoding on the receiver side. During testing, the team achieved 10s interval time with fan and for a relatively short distance.

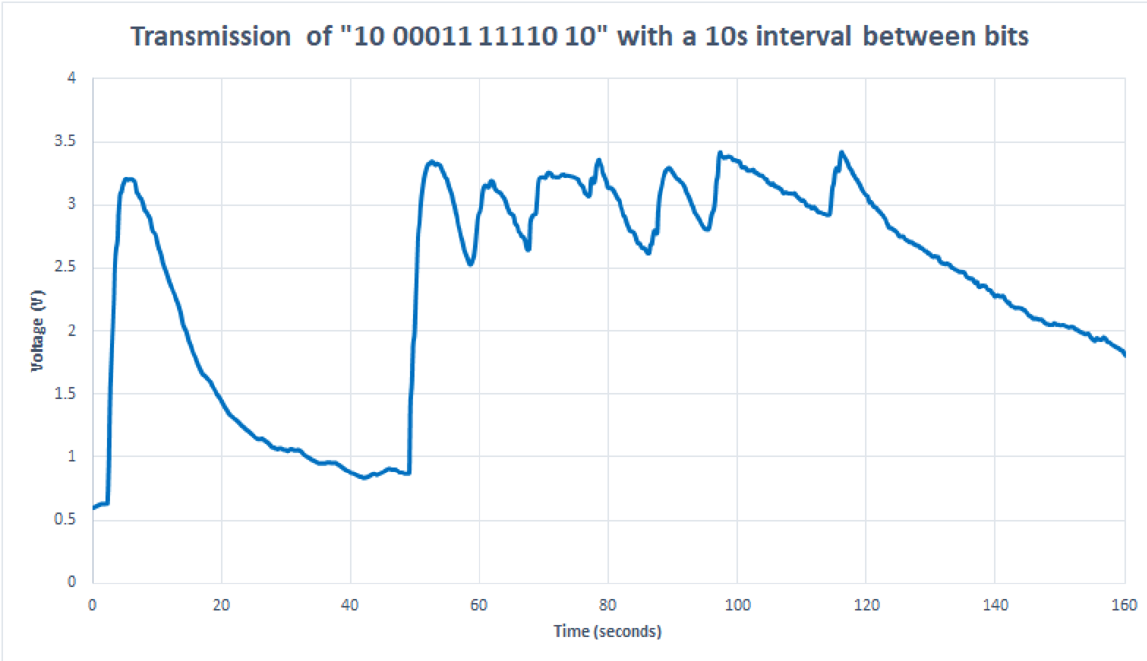


Figure 4.5. Results of transmitting a binary sequence pulled from the MQ-3B sensor with a 10 second interval between bits.

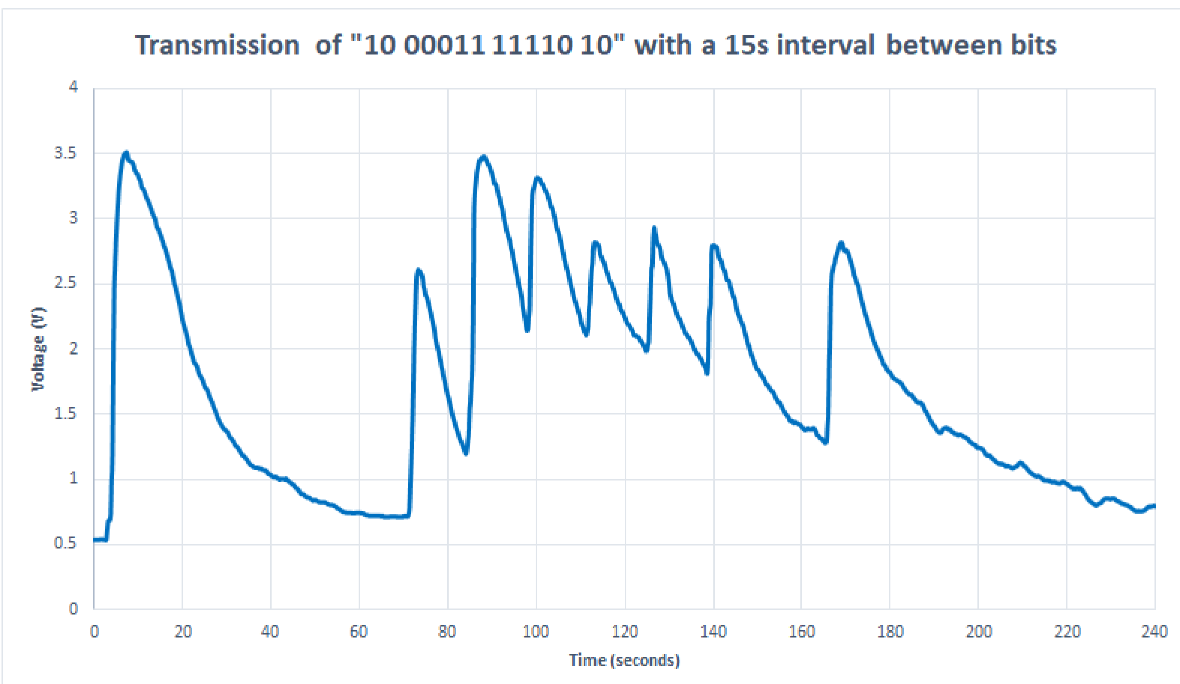


Figure 4.6. Results of transmitting a binary sequence pulled from the MQ-3B sensor with a 15 second interval between bits.

The second system parameter is the duty cycle of the spray gun motor. The team used pulse width modulation to obtain different duty cycles to control the power level of the motor which in turn, determines the amount of alcohol being sprayed by the spray gun. The team tested the voltage reading of different duty cycles at a transmission distance of 6 feet and collected 15 data points for each duty cycle. The probability density function plots in Figure 4.7 and 4.8 reflect the testing results and describe the relationship between the alcohol readings on the receiver side with their corresponding motor duty cycle. At this point of the project, the team only need to differentiate between 0 and 1, so any one of the duty cycles could be chosen to represent 1 depending on the channel condition. However, in our preliminary attempt towards utilizing multiple duty cycles, satisfactory separations have been shown. Therefore, potentially four-level transmission could be achieved, which will take advantage of four different duty cycles, including 0%, to have four distinguishable voltage levels on the receiver side so that the system can send two bits per pulse to achieve better transmission speed.

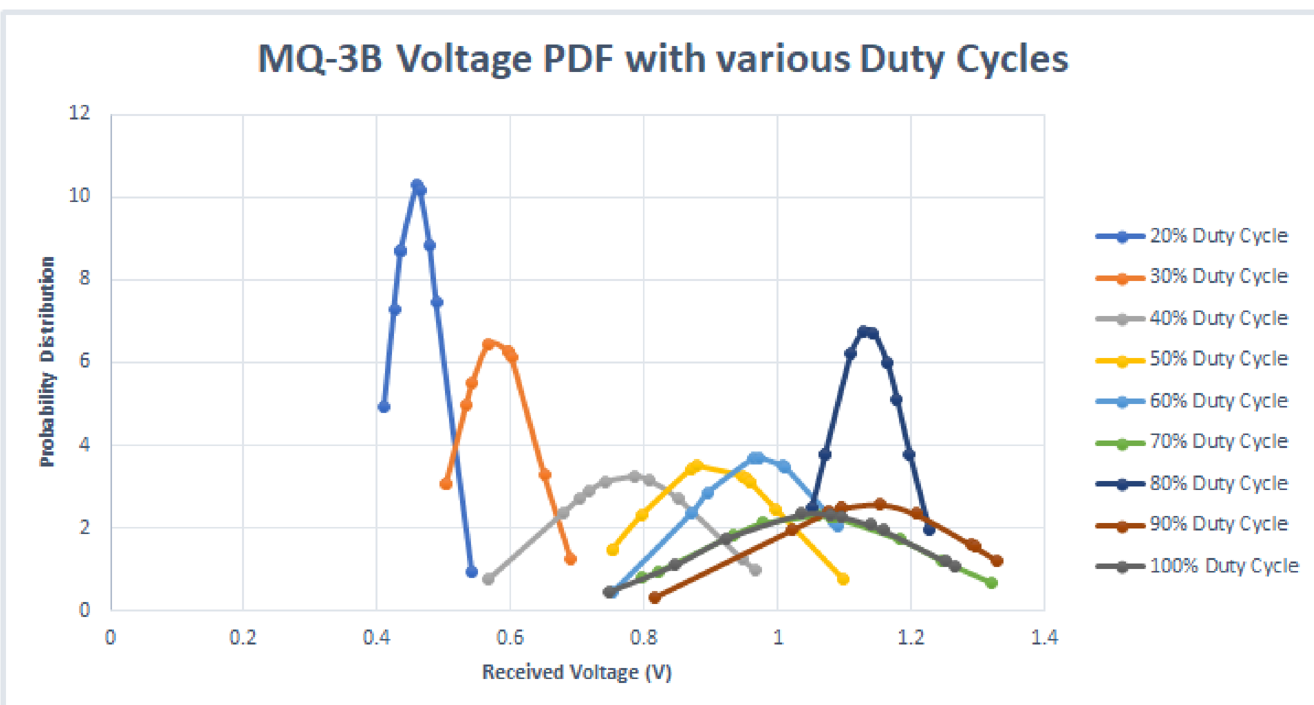


Figure 4.7. Normalized probability distribution for the MQ-3B sensor for different duty cycles with limited data samples.

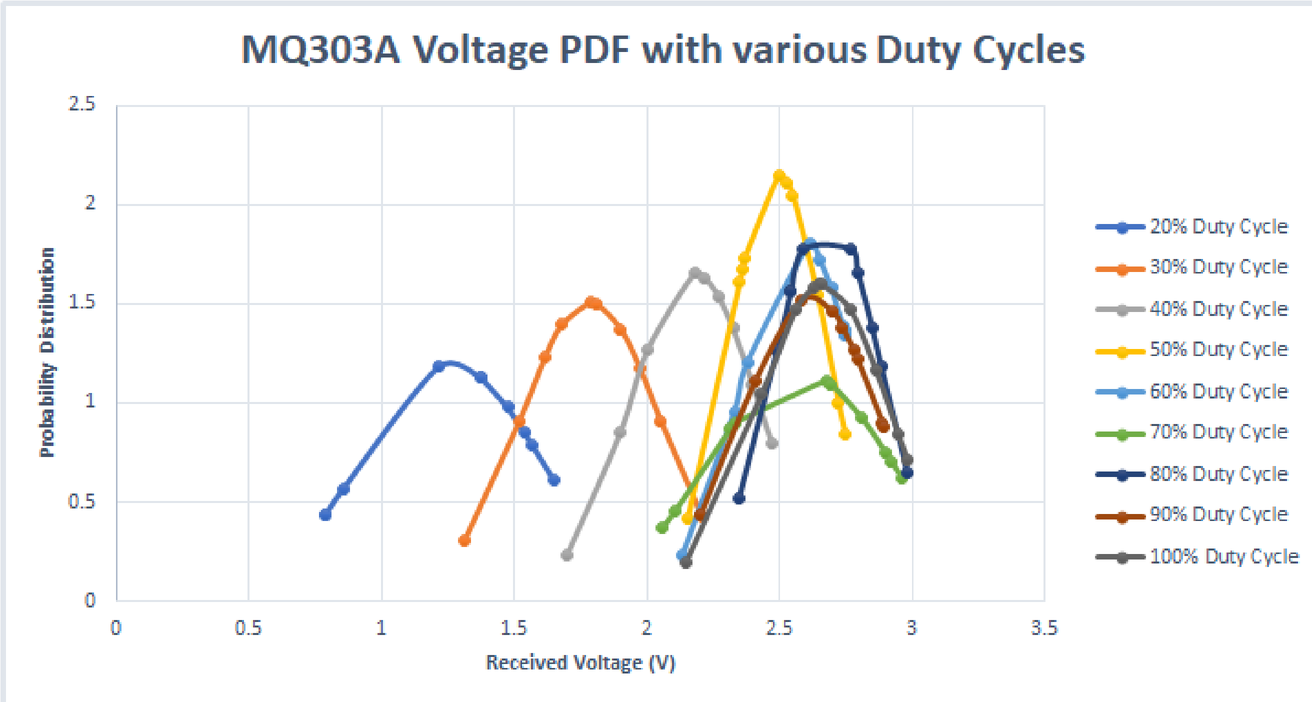


Figure 4.8. Normalized probability distribution for the MQ303A sensor for different duty cycles with limited data samples.

The last parameter controlled by the transmitter side is the alcohol concentration. In Figure 4.9 with 3.5% alcohol concentration and Figure 4.10 with 6.5% alcohol concentration, the team did experiments with different duty cycles ranging from 20% to 100%. The alcohol reading values on the receiver side are reflected on the graph. It is obvious that 6.5% alcohol concentration gives a more consistent voltage reading on the receiver side since a higher concentration is less vulnerable to constantly changing channel conditions. Therefore, the team decided to adopt 6.5% concentration for further testing.

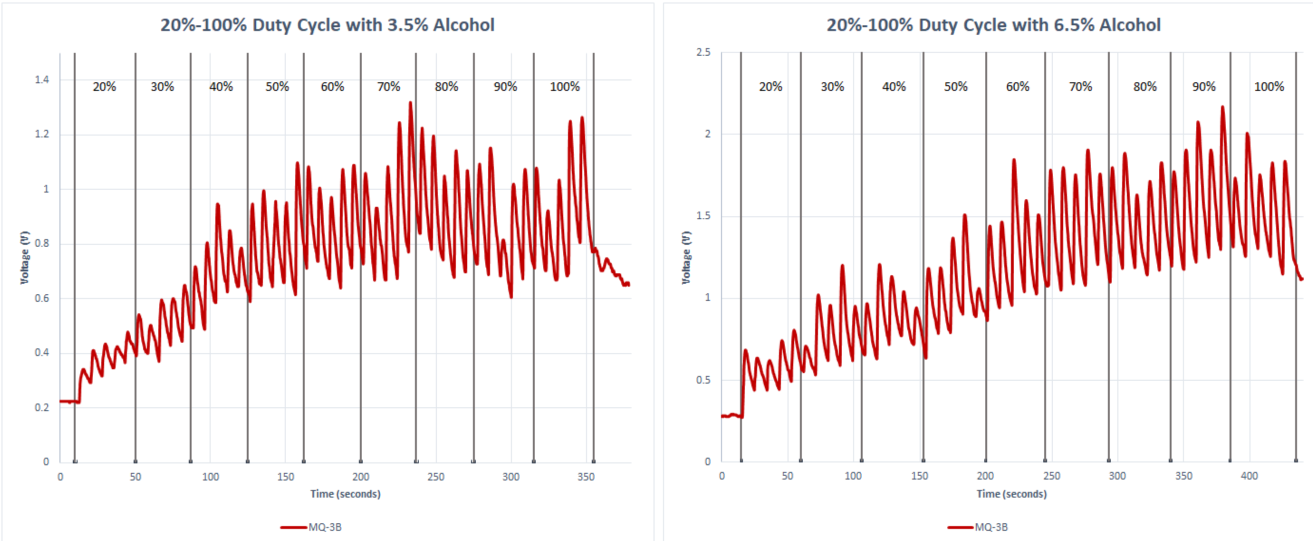


Figure 4.9. Results of increasing the power of the motor and transmitting five 1's at that power while using a 3.5% isopropyl alcohol solution and a 6.5% isopropyl alcohol solution respectively.

Load Resistance

The MQ-3B gas sensor offered the ability to tune its sensitivity by varying the resistance of a load resistor. In the technical documentation, Sparkfun recommended a 200 KΩ load resistance. Through experimentation, five resistances between 2 KΩ and 200 KΩ were used to determine which load resistance offered the best sensitivity for the system. From this testing, the 2 KΩ resistor was selected.

Distance

The distance between the transmitter and receiver is rather important in that it can determine the quality of the signal the alcohol sensors were able to pick up. A good quality signal in our testings would be one that has a larger range of increase and decrease, because the more 'clear' the increase or decrease there is, the better the decoding would be in terms of differentiating between a '1' and a '0'. A quick comparison between Figure 4.11 and Figure 4.12 offers the conclusion that when we increase the distance from 6 ft to 8 ft, the average sensor voltage is significantly lower. Moreover, the transition between '1's and '0's are not as clear as we would like for signal processing purposes.

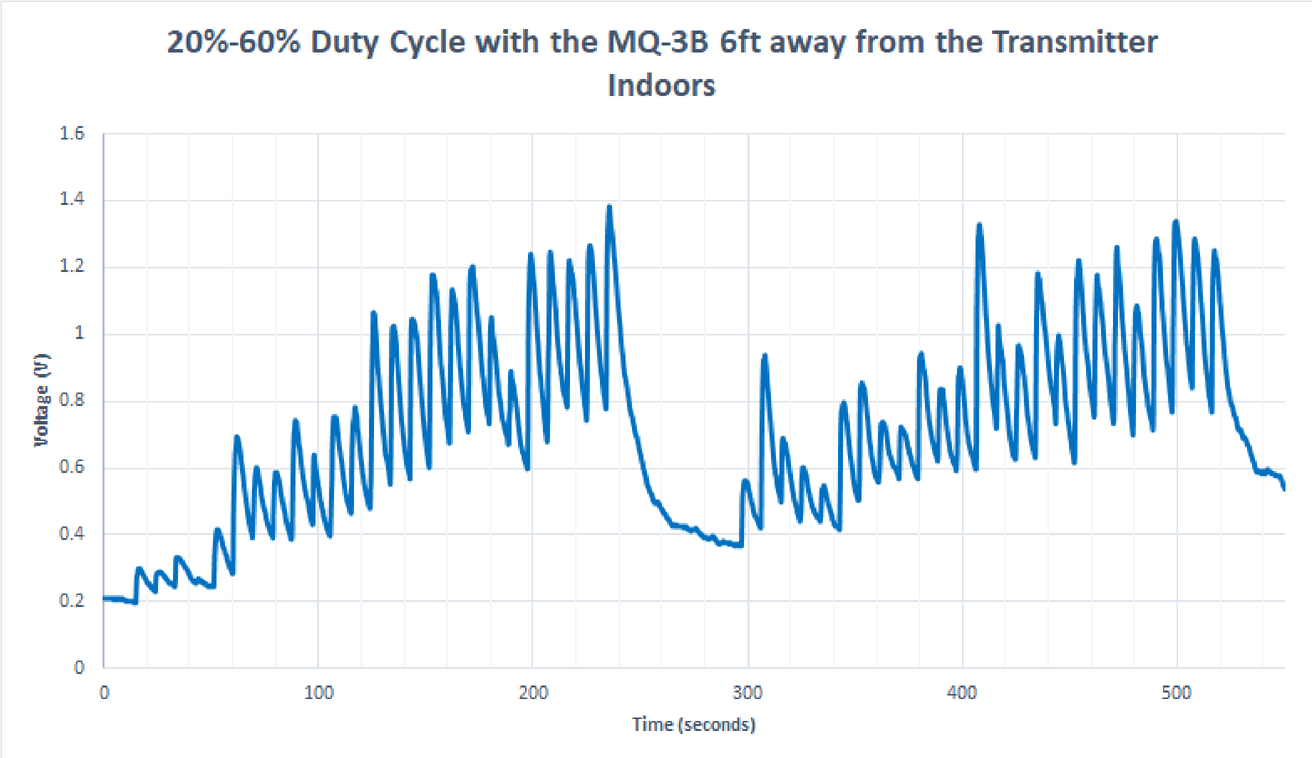


Figure 4.11. Results of increasing the power of the motor in 10% intervals from 20%-60% two times with a 60 second wait time between the two trials. The receiver was 6 ft away from the transmitter indoors.

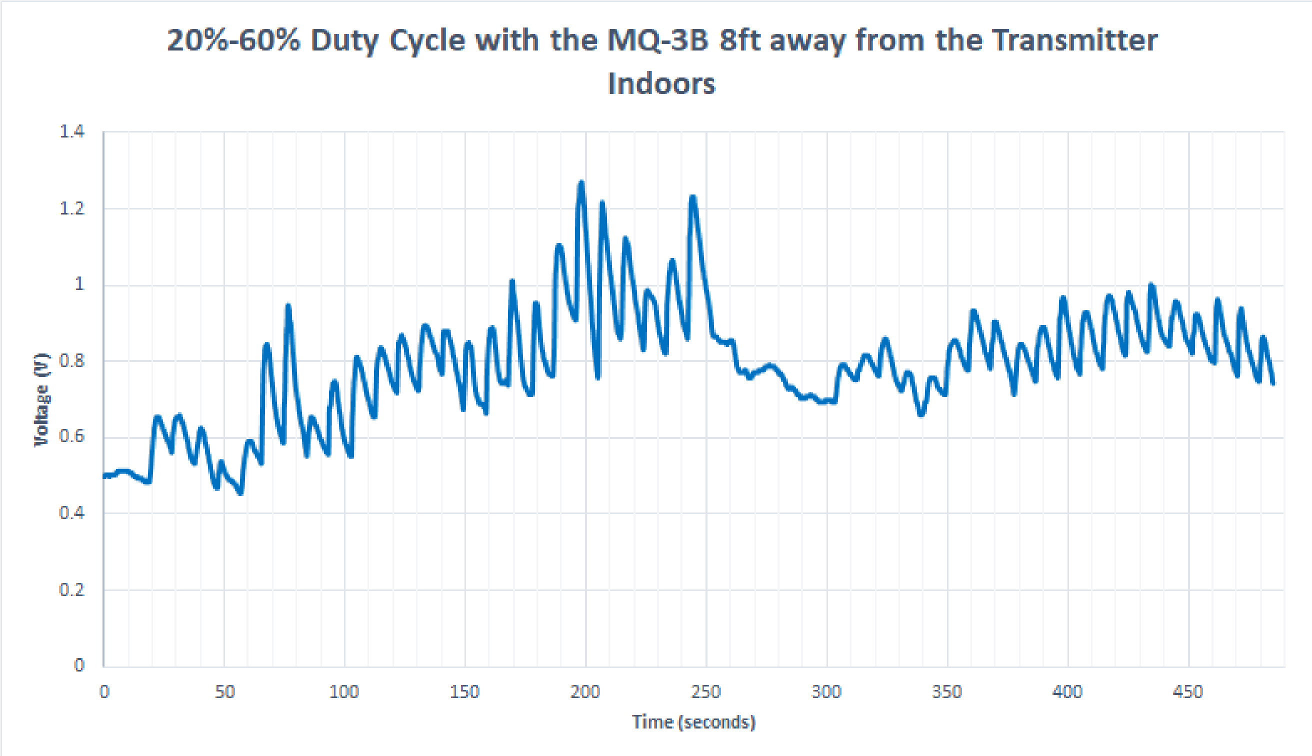


Figure 4.12. Results of increasing the power of the motor in 10% intervals from 20%-60% two times with a 60 second wait time between the two trials. The receiver was 8 ft away from the transmitter indoors.

Environment

Experiments done in Figure 4.11 and 4.12 are both conducted indoors, where there is minimal turbulence in channel propagation. While it can already be established that increasing the distance between the transmitter and receiver would lead to lower average sensor voltage, another reason why we did not opt to conduct any experiments at a distance more than 8 feet was the physical limitations of our indoor testing ground. To go beyond 8 feet would mean for us to move the entire prototype outdoors where we might encounter noise such as cross breeze or even wind in the opposite direction of the propagation. Comparing Figure 4.11 and Figure 4.13 we can observe that there are significant turbulence to the transmission process once the prototype is operating outdoors. The signal isn't uniform and hence decoding it would become tricky.

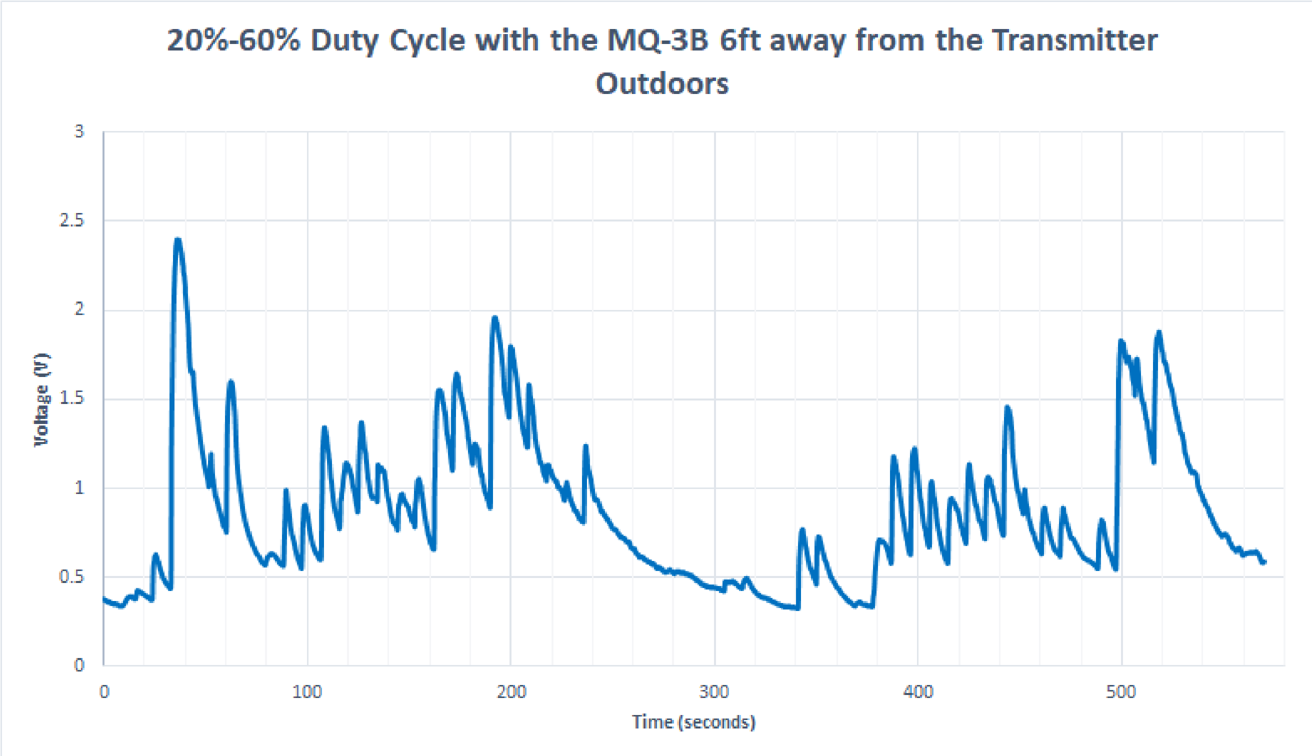


Figure 4.13. Results of increasing the power of the motor in 10% intervals from 20%-60% two times with a 60 second wait time between the two trials. The receiver was 6 ft away from the transmitter outdoors.

4.2.2 Design tradeoffs

One significant project decision is made on the transmitter side of the prototype. Instead of using a commercial electronic liquid spray bottle, the team considered building an automated spray bottle from a mechanical spray bottle with a handle, a TowerPro MG995 servo and a servo base [8]. The first advantage of building an automated spray bottle from scratch is the cost; the components needed in the design are easily accessible mechanical parts and electronic parts and are more economical than the commercial electronic spray bottle. The second advantage is the built automated spray bottle will be easier to maintain and troubleshoot since it involves only primitive hardware and software units. However, building a transmitter can cause unexpected delays and create unnecessary technical challenges for the project, thus diverting focus away from the main goal of the design project: molecular communication. Given that the commercial electronic spray can immediately transmit liquid once connected to a microcontroller makes it a better candidate for the project prototype.

In the proposal, the receiver was originally designed to have three gas sensors, each with different characteristics, to increase the likelihood that each bit would be read correctly. However, over the course of the semester, this was reduced from three sensors to two sensors during the design phase, and then from two sensors to one sensor during the testing phase. During testing, it was found that the signal processing algorithm worked best only using input from the MQ-3b gas sensor because its sensitivity levels worked best within the conditions of the system. Furthermore, only using one gas sensor will reduce the cost of the system by $10 and make it smaller. However, since there is only one gas sensor on the board, it is more prone to fail if that gas sensor gets damaged, and it no longer benefits from the diversity in sensor characteristics.

4.2.3 Constraints and drawbacks of the current system

The current system is not waterproof, so we have to put cover on the receiver circuits. Our prototype doesn’t have an isolation system that isolate the channel from wind perturbation. User should find external controlled environment to perform controlled tests. The receiver software is not designed to counteract significant error. For example, if a concentration surge is manually introduced at time 10s while time interval is 30s, it will be captured even at an inappropriate timing.

Our demodulation method relies on time-synchronous protocol. Air turbulence causes errors on the time difference between two peaks, thus the system failed to find correct number of ‘0’s. In such synchronous failure case, the system fails to demodulate correct number of ‘0’s, thus gets longer or shorter length of bit sequence. The difficulty comes with the fact that we currently can’t determine number and location of missing bits in our one-way communication system.

5. Schedule, Tasks, and Milestones

At the beginning of the semester, the tasks were divided up as followed: Zhongyang and Nick would be responsible for the receiver, Constance and Siyan would be responsible for the transmitter, Jun would be responsible for processing the voltage signals into messages, and Jacob would be responsible for printing the PCBs and any miscellaneous needed hardware help. All testing would be done at a system level and include everyone from the team.

Within the receiver side, Nick and Zhongyang both worked to design and build a prototype that connected that gas sensors to an Arduino, and Nick was also responsible for the Matlab code to read voltages and for creating a GUI. Designing the circuit, writing the Matlab code, and creating the GUI was all straightforward with little difficulty. However, there were some challenges when building the prototype that lead to lengthy debugging. These challenges included that the voltage divider originally created to power one of the gas sensors with 1V heater voltage did not function properly, dead sensors were hard to quickly identify, and tuning the circuit parameters to increase sensitivity.

Jacob built a 1V voltage regulator and printed a PCB of the receiver circuit. The design of the voltage regulator was quite simple since it was a standard circuit, however one challenge when assembling the circuit was determining that two transistors should be included to prevent overheating. Designing and printing the PCB was more challenging since Jacob had not used Eagle CAD before. Once the PCB was milled, not all of the circuit elements had been connected properly so he had to manually solder. The final product included a 2V offset in one of the gas sensors which after extensive debugging, was never diagnosed, and the sensitivity of the other gas sensor was not as high as on the breadboard.

Within the transmitter side, Constance and Siyan built and redesigned a GUI in C# to communicate with the Arduino and assigned the time interval, wrote code to control the motor from the Arduino, designed the communication protocol, and wrote code for testing scenarios. All of these tasks they did together and found getting the GUI to communicate with the Arduino for the first time as the most challenging task. To complete the GUI, they had to build a debugging LED so that it could be tested separately from the sprayer. The next most challenging task was controlling the motor from the Arduino for various duty cycles. For this part they used pulse width modulation to obtain different power level for the motor. It took several hours of revisions and debugging before this was functional. The other tasks they worked on still presented challenges, but were less difficult.

Lastly, Jun wrote the Matlab code to take in real-time sensor voltages and output characters and generated the probability distributions for sensor voltages at various duty cycles. This was the most challenging part of the senior design project and took several iterations to get something working. At first the code was only able to process the message after the entire message had been transmitted. Eventually the code was reworked to be able to be processed in real-time, however a new problem was introduced: variations in the time bins. The algorithm worked by making assumptions about the relative time between a pulse, so when fluctuations between ±3s and ±5s occurred, extra bits would be included in the demodulation. This was eventually resolved through a more tolerant relative time binning algorithm. Generating the probability distributions for sensor voltages at various duty cycles was much less difficult.

Figures 5.1, 5.2, and 5.3 below, show graphically the timeline that tasks were actually completed and milestones.

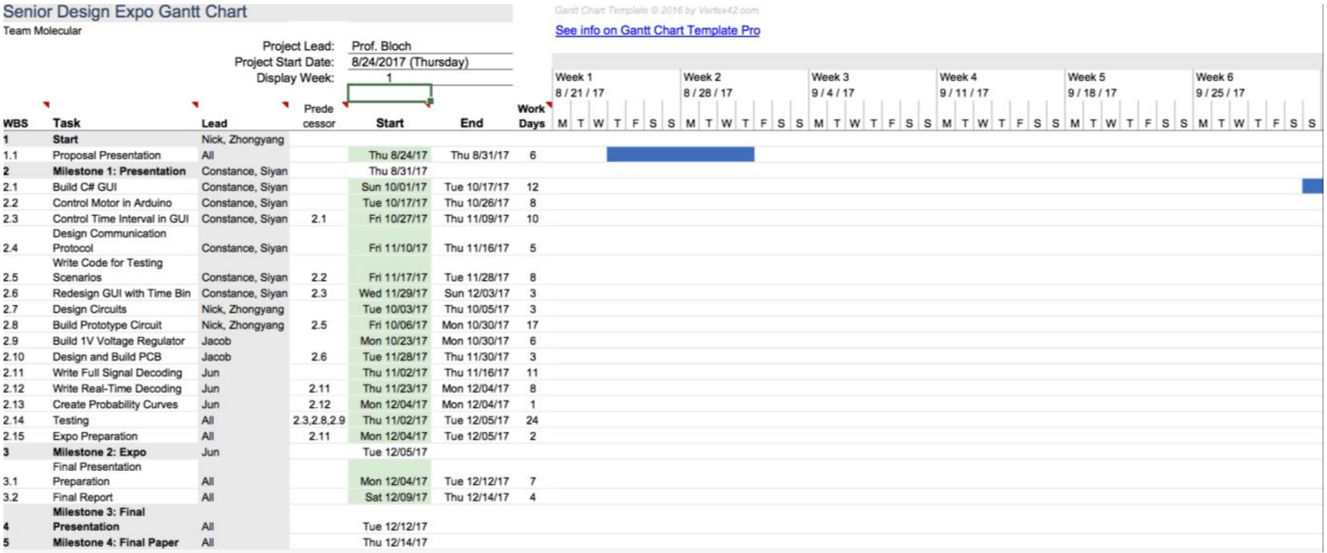


Figure 5.1 Gantt chart showing activity between weeks one and six. The one month gap was the duration of waiting for parts to arrive.

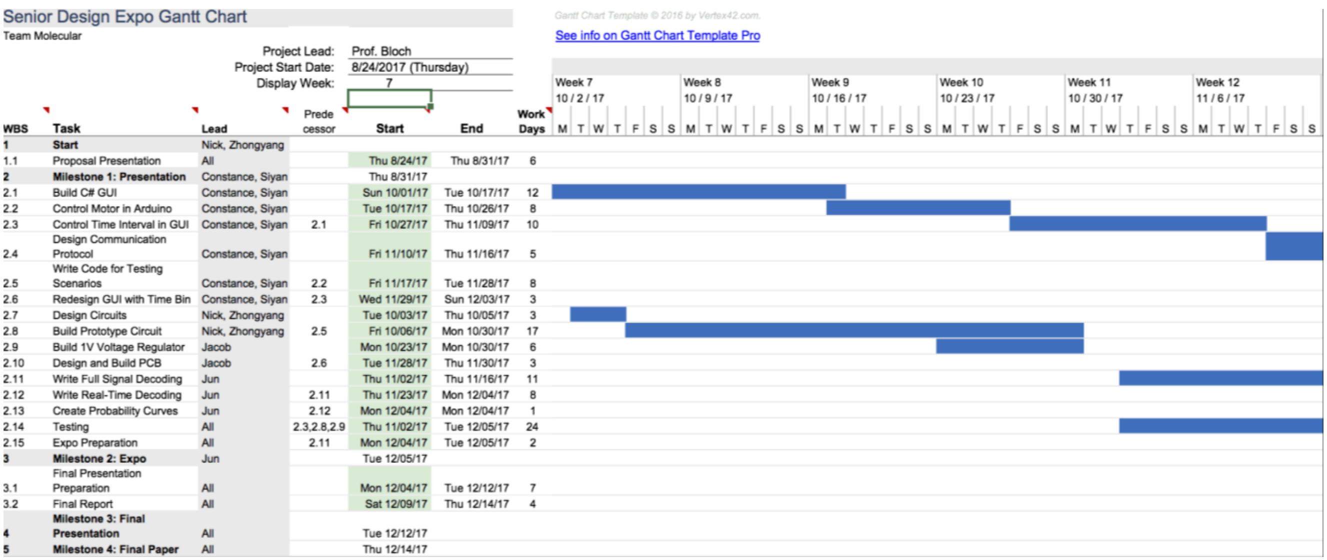
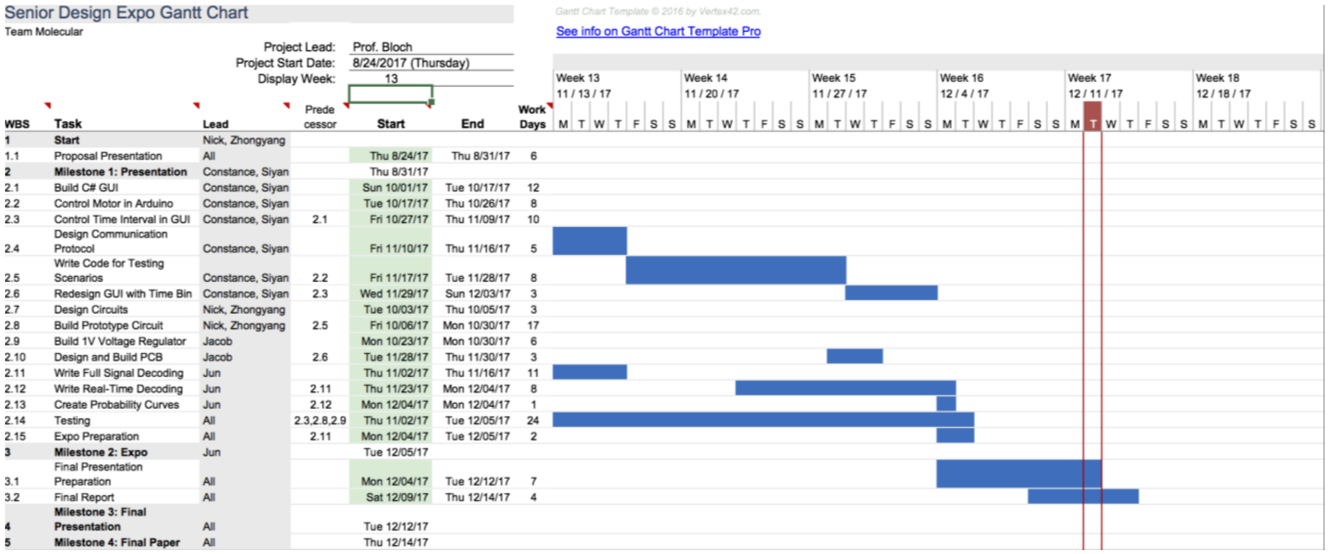


Figure 5.2 Gantt chart showing activity between weeks seven and twelve.



WBS	Task	Lead	Predecessor	Start	End	Work Days
1	**Start**	Nick, Zhongyang				
1.1	Proposal Presentation	All		Thu 8/24/17	Thu 8/31/17	6
2	**Milestone 1: Presentation**	Constance, Siyan		Thu 8/31/17		
2.1	Build C# GUI	Constance, Siyan		Sun 10/01/17	Tue 10/17/17	12
2.2	Control Motor in Arduino	Constance, Siyan		Tue 10/17/17	Thu 10/26/17	8
2.3	Control Time Interval in GUI	Constance, Siyan	2.1	Fri 10/27/17	Thu 11/09/17	10
2.4	Design Communication Protocol	Constance, Siyan		Fri 11/10/17	Thu 11/16/17	5
2.5	Write Code for Testing Scenarios	Constance, Siyan	2.2	Fri 11/17/17	Tue 11/28/17	8
2.6	Redesign GUI with Time Bin	Constance, Siyan	2.3	Wed 11/29/17	Sun 12/03/17	3
2.7	Design Circuits	Nick, Zhongyang		Tue 10/03/17	Thu 10/05/17	3
2.8	Build Prototype Circuit	Nick, Zhongyang	2.5	Fri 10/06/17	Mon 10/30/17	17
2.9	Build 1V Voltage Regulator	Jacob		Mon 10/23/17	Mon 10/30/17	6
2.10	Design and Build PCB	Jacob	2.6	Tue 11/28/17	Thu 11/30/17	3
2.11	Write Full Signal Decoding	Jun		Thu 11/02/17	Thu 11/16/17	11
2.12	Write Real-Time Decoding	Jun	2.11	Thu 11/23/17	Mon 12/04/17	8
2.13	Create Probability Curves	Jun	2.12	Mon 12/04/17	Mon 12/04/17	1
2.14	Testing	All	2.3,2.8,2.9	Thu 11/02/17	Tue 12/05/17	24
2.15	Expo Preparation	All	2.11	Mon 12/04/17	Tue 12/05/17	2
3	**Milestone 2: Expo**	Jun		Tue 12/05/17		
3.1	Final Presentation Preparation	All		Mon 12/04/17	Tue 12/12/17	7
3.2	Final Report	All		Sat 12/09/17	Thu 12/14/17	4
4	**Milestone 3: Final Presentation**	All		Tue 12/12/17		
5	**Milestone 4: Final Paper**	All		Thu 12/14/17		

Figure 5.3 Gantt chart showing activity between weeks seven and eighteen. The red line indicates the current day at the time of finishing this report.

6. Final Project Demonstration

During the EXPO, both the transmitter and receiver were placed in a horizontal fashion on the table, approximately five feet apart. Demonstration of a successful transmission required one user to operate both laptops that are a part of the system.

1. On the laptop connected to the receiver, the user must run the MATLAB code that collects the data from the sensor and decodes the bits sent over the channel.
2. Then, on the laptop connect to the transmitter, the user must input a message consisting of a-z characters in the text box on the text input GUI.
3. The user can also select the time bin using the radio buttons on the text input GUI. The options for the time bin are 10, 20, or 30 seconds, with the default time bin being 20 seconds. The user then clicks the gold "Send" button.
4. The transmitter Arduino Uno will then parse through the string of characters sent from the text input GUI, using the first character, a 1, 2, or 3, to determine the time bin and assigning the rest of the characters to their 5-bit ITA2 standard code.

5. The Arduino Uno will first transmit a “1” bit by turning on the motor at 20% PWM and then a “0” bit by not turning on the motor at all for the given time bin.
6. As the Arduino Uno parses through the 5-bit code per character, it will turn on the motor on and off depending on the bit. A troubleshooting LED it will flash green for “1” and red for “0.” A small, USB powered fan is used to help propagate the alcohol solution over to the receiver.
7. The receiver will detect the influx of isopropyl alcohol using two alcohol sensors. These sensors are connected to another Arduino Uno which will then relay the voltage data from the sensors to the receiver laptop.
8. As the message is transmitted, the receiver laptop will update real time; a line graph GUI shows when a “1” bit is decoded by the receiver and another GUI will display the characters that are decoded.

With the project being more research-oriented, there would be minimal adjustments to the prototype in terms of its hardware and external design. Instead, more time would be dedicated to the correct implementation of microcontroller programs and efficiency of the overall process.

7. Marketing and Cost Analysis

7.1 <u>Marketing Analysis</u>

This system would be useful anywhere standard methods of electromagnetic communication are not feasible. One example would be in micro- and nano-scale technology where the antenna would need to be larger than the actual device. It is, however, possible to make a chemical sensor at this scale, making this technology useful for communication between nanoscale devices.

Currently, only prototypes of this system exist, and it will probably not be marketable for several years as it is still in the very early stages of development. It is still large-scale, so not useful for nanoscale technology and it is still very slow.

7.2 Cost Analysis

The total cost for building the prototype was $91.09. The breakdown of the main material costs is shown in Table 7.1.

Table 7.1. Cost to build prototype

Product Description	**Quantity**	**Unit Price ($)**	**Total Price ($)**
Arduino Uno Rev3	2	$24.95	$49.90
MQ-3B	1	$4.95	$4.95
MQ303A	1	$5.50	$5.50
Electric Spray Gun	1	$19.98	$19.98
Solid State Relay	1	$8.81	$8.81
1V Voltage Regulator	1	$1.95	$1.95

The total number of labor hours were calculated assuming an 8-hour work day for the worst-case-scenario time allotted in the schedule. Total labor costs were calculated by taking the number of work hours, the hourly salary assuming each engineer earns $47.41/hour [10], and the number of engineers working on that section. The total development costs were calculated assuming the fringe benefit to be 30% and the overhead to be 120%. Developing and producing the circuits board will have the highest number of labor hours due to prototyping a unique board for the transmitter and the receiver. The overall cost of producing this prototype will be $405,825.83 as shown in Table 7.3.

Table 7.2. Development costs

Project Component	Labor Hour	Labor Cost	Part Cost	Total Component Cost

Transmitter	240	$22756.80	$24.95	$22,781.75
Receiver	240	$22756.80	$35.40	$22,792.20
Circuit Board	424	$20101.84	0	$20,101.84
PC Host Environment	168	$7964.88	0	$7,964.88
Data analysis	240	$68270.40	0	$68270.40
Total	**1312**	**$141,850.70**	**$60.35**	**$141,911.07**

Table 7.3. Total Development Costs

Parts	$60.35
Labor	$141,850.72
Fringe Benefits, % of Labor	$42,555.22
Subtotal	$184,466.29
Overhead, % of Matl, Labor & Fringe	$221,359.54
Total	**$405,825.83**

Over a five-year period, it is assumed 50 units will be sold. Table 7.4 shows the calculation of the expected profit and selling price assuming this system goes into production. Fringe benefits were assumed to be 30%, overhead was assumed to be 120% and sales expense was assumed to be 2%.

Table 7.4. Selling Price and Profit per Unit

Parts Cost	$150
Assembly Labor [12]	$20
Testing Labor	$35
Total Labor	$55
Fringe Benefits, % of Labor	$16.50
Subtotal	$221.50

Overhead, % of Matl, Labor & Fringe	$265.80
Subtotal, Input Costs	$487.30
Sales Expense	$10
Amortized Development Costs	$25
Subtotal, All Costs	$522.05
Profit	$400
Selling Price	***$922***

8. Conclusion

By December 5th, the team had successfully built the hardware and software components of the transmitter and receiver prototypes. The prototype, though has a rather stringent selection of environmental criteria, is fully functional in terms of message exchange using alcohol as its medium. The prototype is inexpensive and replicable, and serves as a proof that molecular communication at macro scales is feasible. The transmission speed exceeds the expectation by 10 times and the transmission distance exceeds the expectation by 2.5 times, as shown in Table 8.1.

Table 8.1. Project status compared with proposal specifications

Item	Proposed Spec	Achieved Spec
Minimum Transmission Speed	0.01 bits/second	0.1 bits/second
Maximum Transmission Distance	100cm	250cm

The team also defined and tested various system parameters quantitatively as described earlier in the paper.

In terms of future work, using different power levels to transmit 2 bits per pulse to increase transmission speed and reduce bit error rate seems very promising with the current limited testing samples. More testing needs to be conducted to better explore the communication channel and it would be ideal to increase the transmission distance between the receiver and transmitter. With regards to real world applications, the team would also like to research ways to work around the environmental factors that might affect an outdoor transmission, in order to actually use the prototype with minimal considerations of environmental limitations. Lastly the system should be independent of computers and should only utilize microcontrollers in the next generation prototype.

9. Leadership Roles

The leadership roles that each member carried out for both 4011 and 4012 semesters are summarized in Table 9.1.

Table 9.1. Leadership roles for each team member

Team member	**Leadership role**
Jacob Callahan	Lead hardware engineer
Nick Fahrenkrog	Expo coordinator
Constance Perkins	Lead firmware engineer, Webmaster
Zhongyang Shi	Documentation coordinator
Jun Xiang	Lead data analyst
Siyan Yu	Lead embedded system engineer

Technical roles are summarized in Figure 9.1. Team members' names are placed next to the component of the prototype that they mainly contributed through, programming or building wise.

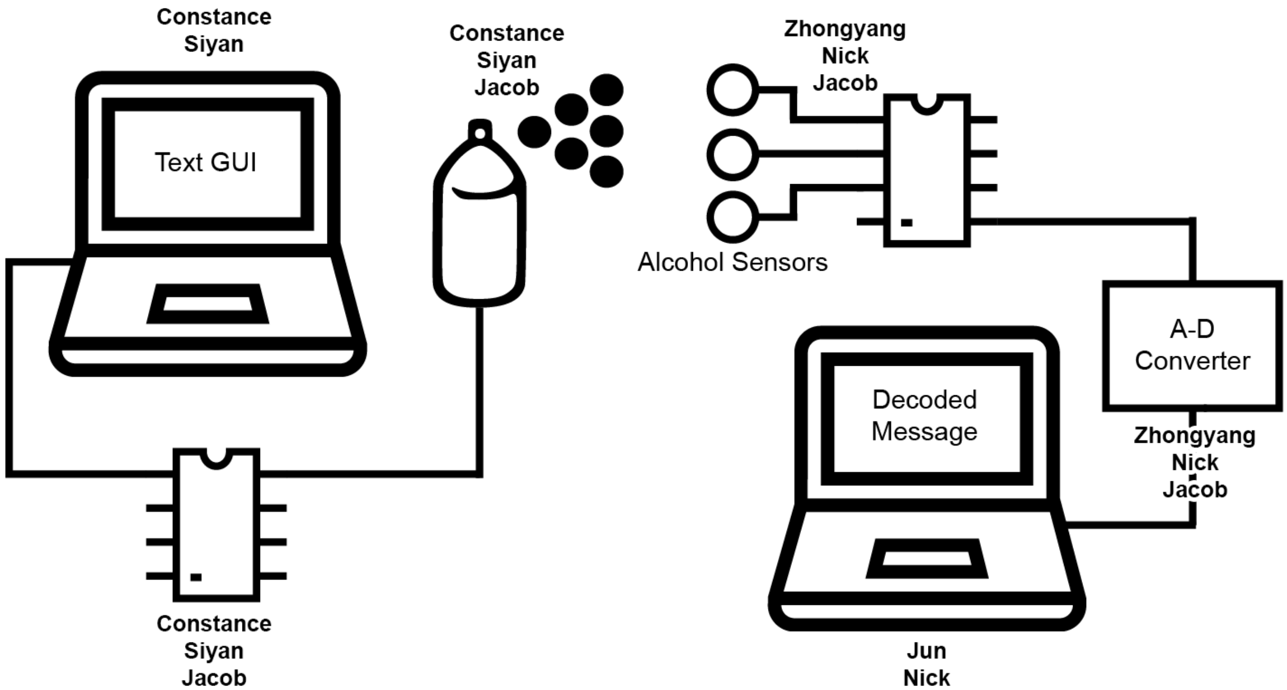


Figure 9.1. Block diagram labeled with team members responsible for component

10. References

[1] Farsad, Nariman, Weisi Guo, and Andrew W. Eckford. "Tabletop Molecular Communication: Text Messages through Chemical Signals." *PLOS ONE*. Public Library of Science, 18 Dec. 2013. Web. 11 Apr. 2017.

[2] "Arduino - ArduinoBoardUno", Arduino.cc, 2017. [Online]. Available: https://www.arduino.cc/en/Main/ArduinoBoardUno. Web. 11 Apr. 2017.

[3] Seeed Studio, "Grove - Gas Sensor (MQ3)," MQ-3 datasheet, Sept. 2015

[4] Seeed Studio, "MQ303A Alcohol Sensor," MQ303A datasheet, Jan. 2009

[5] Zhengzhou Winsen Electronics Technology Co., Ltd, "Hot-wire Type Gas Sensor (Model：MR513)," MR513 datasheet, June 2015. [Revised Aug. 2016].

[6] The International Telegraph and Telephone Consultative Committee (CCITT) (1988) International Telegraph Alphabet No. 2. Blue Book, Fascicle 7–1.

[7] Frbiz, "Duroblast electric Sprayer images, view Duroblast electric Sprayer photos of item 42681500," 2011. [Online]. Available: http://amazingnokout.en.frbiz.com/images/p42681500 duroblast_electric_sprayer.html

[8] TowerPro, "High Speed Metal Gear Dual Ball Bearing Servo" MG995 datasheet

[9] Scharf, L. L. (2002). Statistical signal processing: detection, estimation, and time series analysis. Reading, Mass.: Addison-Wesley.

[10] "Occupational Employment and Wages, May 2016," U.S. Bureau of Labor Statistics, 03-Mar-2017. [Online]. Available: https://www.bls.gov/oes/current/oes172071.htm. Web. 11 Apr. 2017.

[11] "The "Baudot" Code". dataIP Limited. Archived from the original on 26 August 2010. Retrieved 9

October 2010

[12] "Electronic/Soldering Technician Job - Latitude Inc. - Marietta, GA." *Indeed.com.* Indeed, 13 Apr. 2017. Web. 14 Apr. 2017.

[13] “Hyper Tough 4.0Amp Electric Spray Gun.” [Online]. Available: https://www.walmart.com/ip/Hyper-Tough-4-0Amp-Electric-Spray-Gun/671795748#read-more. [Accessed: 12-12-2017].

[14] “Solid State Relay - 40A (3-32V DC Input.” [Online]. Available: https://www.sparkfun.com/products/13015. [Accessed: 12-12-2017].