Bluetooth Ouija Board

ECE 445 Design Document

Team 59: Luke Staunton, Nikhil Mathews, Oluwatobi Ijose
TA: Jacob Bryan

3/26/18

1. Introduction

1.1 Objective

We are designing and implementing a system that would allow us to successfully create the impression that there are spiritual forces communicating specific messages to users of a ouija board. By creating an iPhone controlled, Bluetooth ouija board, we will be able to perform this illusion by allowing the board to autonomously move the planchette towards the desired characters and make any participants feel the presence of the supernatural. A person with the designated app would be able to input a string of the user's choice, and the planchette on the board would maneuver to spell the word, guiding the users hand with the planchette which makes the illusion much more compelling, especially to non-believers. Since an electromagnet is being used, players will not be able to detect magnetic forces when the system is not in use.

Additionally, similar systems have been designed to automate chess boards, and hobbyists have used electromagnets. One good example we saw when researching was an automated chess board that moved its magnetic pieces with the use of a two axis robotic arm with a magnetic head, located beneath the box. That project also uses the ATMega250 microcontroller with Bluetooth LE to facilitate communication with an application. Despite this, our project will be the first ouija board that we were able to come across that is sophisticated enough to appear believable.

1.2 Background

There are many people who genuinely believe that there are actually spirits controlling what is output by a ouija board, and that there are many unseen supernatural elements to our world. Historically, ouija boards have been operated purely by the end user by phenomenon called "ideomotor effect"[1]. With the added benefit of our electromagnet, we can further influence the end users perception. We hope to play on the superstitions of these people, in order to solve the problem of a ouija board that is not actually moving on its own. We have seen a couple rudimentary versions of this project online[2], but they aren't very fluid and sometimes do have obvious signs such as clicking noises, or choppy delay between characters, which we plan to eliminate. While there is not an outright need for this solution, it does provide a fun trick that you can show your friends and can be much more convincing than the original ouija board.

1.3 High-level requirements

- A phone application that tells the system that the planchette is on the board, and receives an acknowledgement signal that the planchette is ready to be moved by the system.
- The user can input a string of characters on the phone application. These characters will be visited by the planchette in the order that they were entered.
- The system communicates back to the phone application the progress the planchette has made in the string, and communicates back to the user if the system temporarily or permanently loses control of the planchette.

2. Design

2.1 Block Diagram

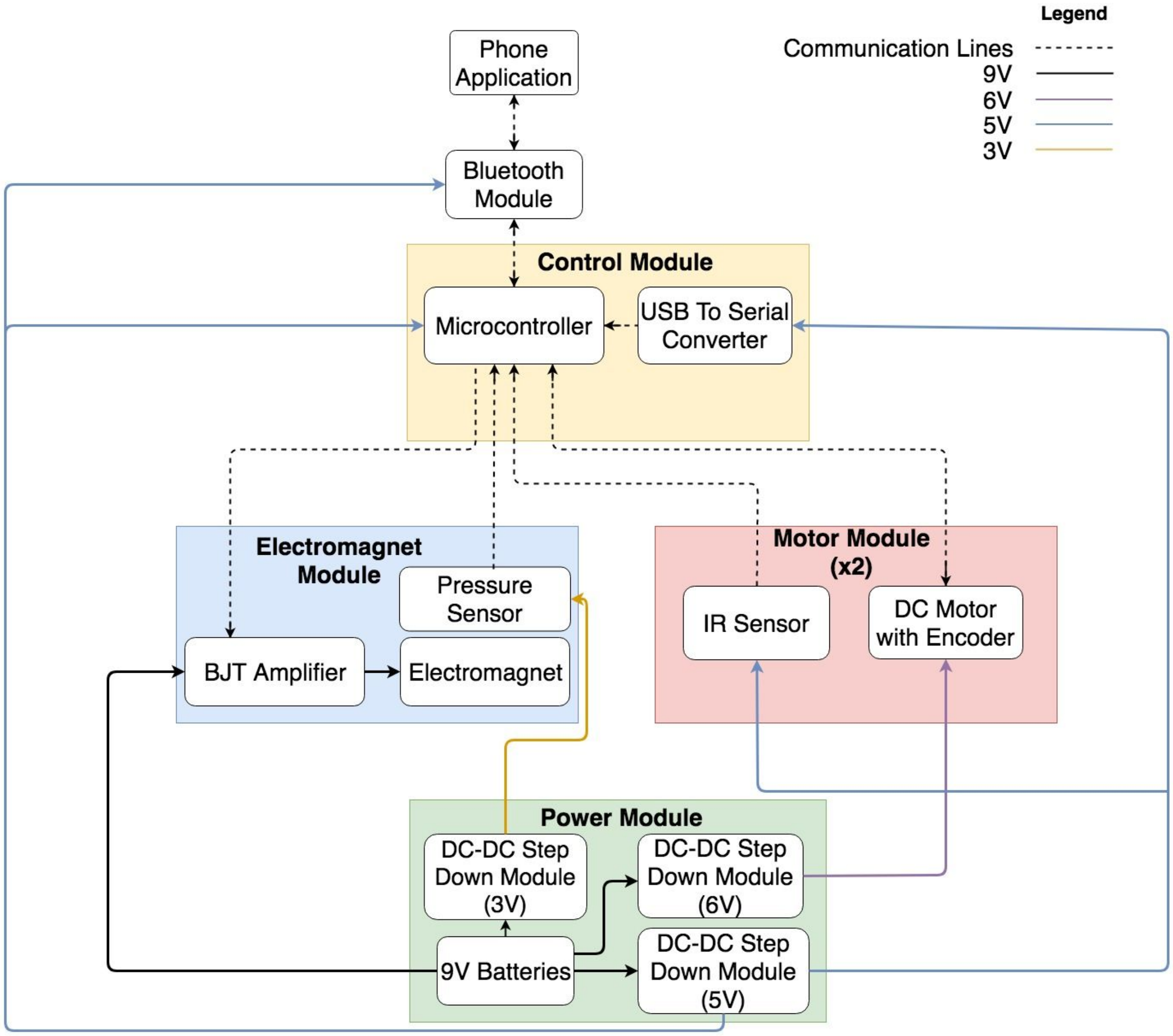


Figure 1 : High level block diagram of design

2.2 Physical Design

The physical design of our system, depicted in Figures 2-4, displays the critical mechanical components to our system, aside from the top of the ouija board, and the planchette that sits on the board, and the box that will house the rails and components.

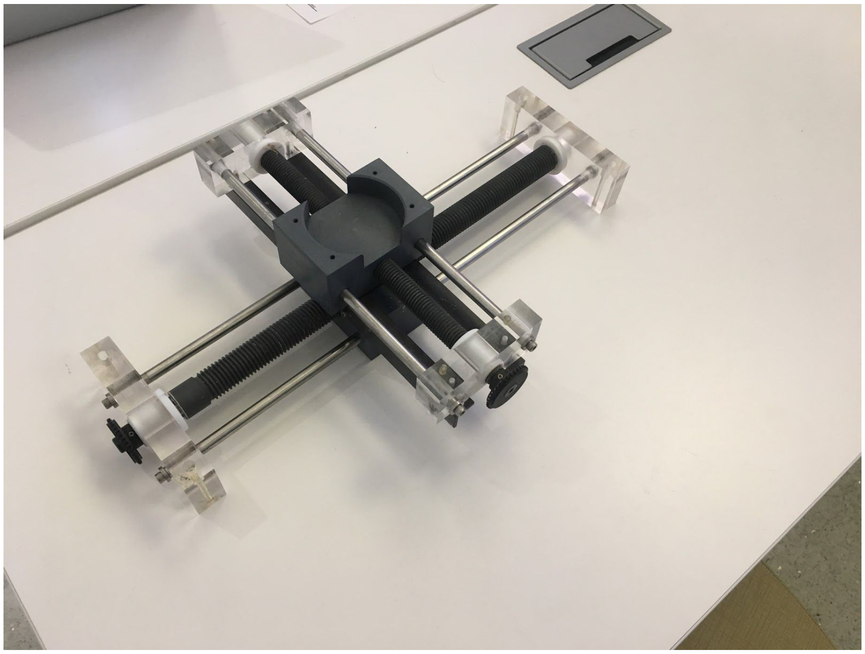


Figure 2: View of mechanical prototype

The internal dimensions of the box containing the rails will be about 20.25” x 13.25” x 4.5”. The two rails are able to move because of the external gears that drive the worm gears that push the module.

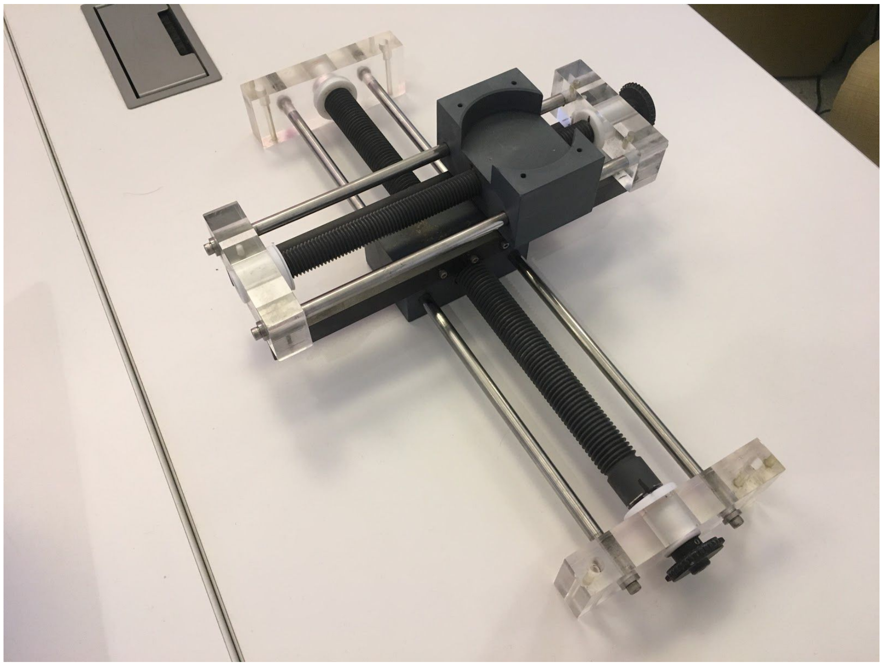


Figure 3: Alternative view of mechanical prototype

As one can see in figures 2 and 3, the longer rail moves the entire shorter rail in the x direction. The shorter rail then has the ability to move the electromagnet magnet module in the y direction. Together the two give us complete control of motion across the plane. The longer rail will have the servo motor attached to the top of the rail, and the shorter rail will have the motor attached below the rail.

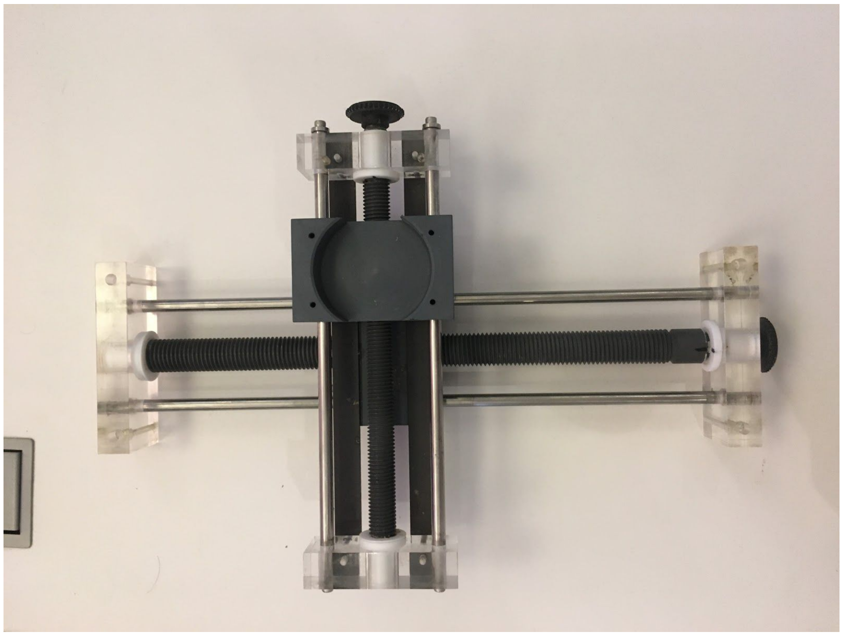


Figure 4: Top down view of mechanical prototype

2.3 Phone App

The phone app will be used to control the word that appears on the ouija board. Because this is a "magic" trick, the user will input a word, and the app will communicate with the bluetooth module on the board. The app will have a basic UI where the user can input an entire word the planchette follows where the user goes.

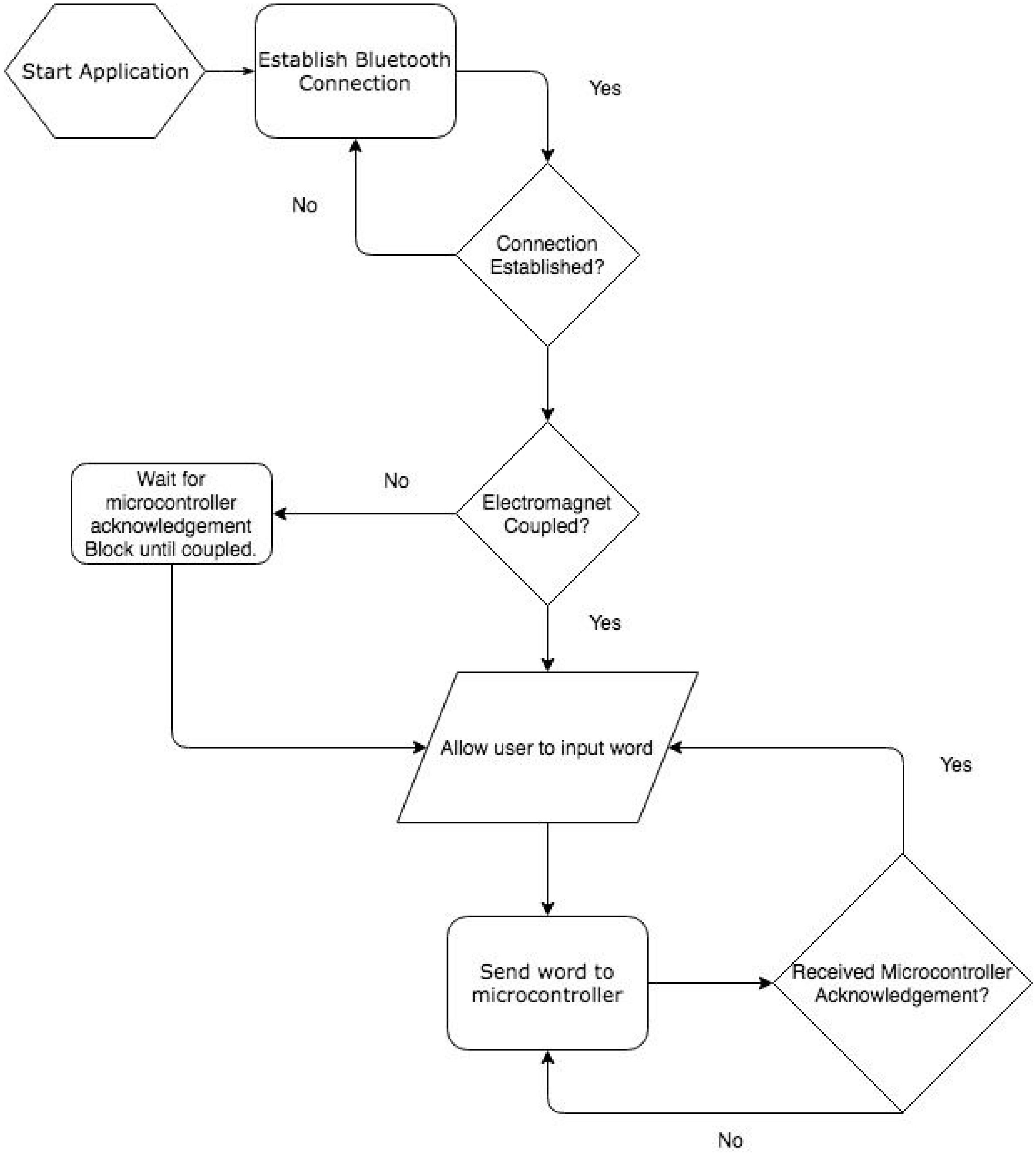

Start Application
Establish Bluetooth Connection
Yes
No
Connection Established?
Wait for microcontroller acknowledgement Block until coupled.
No
Electromagnet Coupled?
Yes
Allow user to input word
Yes
Send word to microcontroller
Received Microcontroller Acknowledgement?
No

Figure 5 : Phone application flow-chart

Requirement	Verification
- Phone application follows the state diagram in figure 5 with no error.	1. Try all state combinations and look at the phone screen to see that the correct next state behavior is achieved by application.
- Receive and display information about the state of the planchette (coupled or not) according to the microprocessor with no error, and a delay of less than five seconds.	1. Apply pressure to the pressure sensor and look at the phone screen to see that the that coupled acknowledgement is received from microcontroller. Visibly confirm coupling LED is on. 2. Remove pressure and start a stopwatch when LED shuts off. Verify visually on phone screen that decoupled acknowledgement is received before five second mark.
- Indicate to microcontroller that the planchette is on the board and it can begin to snake search to look for it.	1. Send "snake search" signal to microcontroller from phone app. 2. Observe visually that the snake search functionality begins on the electromagnet module.

2.4 Electromagnet Module

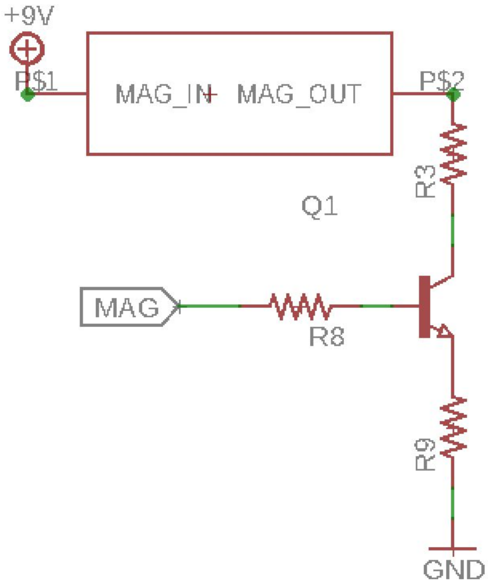


Figure 6: Schematic of electromagnet circuit

The electromagnet module encompasses the planchette, the underlying electromagnet used to move it and a pressure sensor to help prevent decoupling of the magnet. The planchette would contain hidden pieces of metal, so as to attract the electromagnet that is located under the board. We chose to use an electromagnet as opposed to a permanent magnet due to its controllability by electric current. We will be able to apply as much, or as little current as needed to successfully facilitate motion of the planchette. When a word is input in the phone app and communicated to the microcontroller via Bluetooth, the electromagnet would be "magically" repositioned by two moving rails. In order to prevent decoupling, a pressure sensor will be used to gauge the difference in pressure between when the planchette is directly overhead, or decoupled. Using some threshold value, we can determine if decoupling has occurred, and move the electromagnet accordingly. Also, we will need a transistor amplifier circuit to control the amount of current going through the electromagnet.

2.4.1 Pressure Sensor

The electromagnet module will incorporate a pressure sensor to effectively determine magnetic coupling. We will be using the NXP MPL3115A2 pressure sensor, which was chosen due to being affordable, and adhering to the I2C protocol for communication with the microcontroller. The sensor has a range of 10 kPa - 110 kPa, and takes a voltage input of 1.8-3.6V. The input to the sensor will come from the regulated power supply voltage, and it will output its pressure reading as a 24 bit digital output to the microcontroller. The calculations in equations [1-7] give the normal force applied on the pressure sensor. This should be equal to the force applied by the electromagnet when coupled, and will be used as a threshold to determine coupling. Using these equations, we determined that a pressure of 101.3+2.027x would be an adequate threshold value, where x would be the number of washers we use. Using 10 washers, we have a pressure lower bound of 121.57 kPa.

$$washer\ diameter\ =\ .027\ [m] \tag{1}$$

$$thickness\ of\ x\ washers,\ h\ =\ .00241x\ [m] \tag{2}$$

$$density\ of\ zinc\ =\ 7135\ [kg/m^3] \tag{3}$$

$$volume\ of\ x\ washers,\ v\ =\ \pi r^2 h\ =\pi(1.35)^2 * 2.41\ =\ .00138\ [m^3] \tag{4}$$

$$washer\ mass,\ m = \rho * v =\ 0.0985x\ [kg] \tag{5}$$

$$sensor\ area,\ A = 2(wl\ +\ hl + hw)\ = \tag{6}$$

$$2[(.005 * .003) + (.0011 * .003) + (.005 * .0011)]\ =\ .0000476\ [m^2]$$

$$pressure,\ P\ =\ F/A\ =\ mg/A\ =\ (9.8 * 0.0985)/.0000476\ =\ 20.27\ [kPa] \tag{7}$$

Using this as a lower bound, when the magnet is active, we know the pressure will be above this. thus, our threshold will be somewhere in the range of [pressure lower bound, lower bound+magnetic force]. The upper bound for our pressure range will be fully determined once the proper magnetic force is determined.

Requirement	Verification
● Microcontroller detects absolute pressure within the threshold of [121.57, 121.57+mag force] within a resolution of +/- 0.25%, as per sensor documentation. Initial calculation of magnetic force is determined in Section 2.4.2 below	1. Attach pressure sensor to microcontroller. 2. Apply various amounts of pressure and read output. 3. Program behavior for pressures above and below threshold. 4. Used for coupling detection.
● Maintain supply current at 30 $\mu A \pm 5\%$.	1. Ensure current to pressure sensor is within tolerance range with use of an ammeter.
● Maintain supply voltage range between 2.7 V - 3.3 V	1. Use oscilloscope to measure open circuit voltage of sensor and make sure it is within voltage range. 2. Using the same method as above, make new measurements once pressure sensor is pressurized and ensure voltage stays within the tolerance.

2.4.2 Electromagnet

The second part of this module is the physical electromagnet and housing. Our electromagnet will consist of 10 zinc washers, wrapped with a coil made out of copper wire. The electromagnet will be placed in a housing made of wood, and placed along the rails to facilitate movement. When the magnet is coupled, the pressure sensor will feel not only the force of the electromagnet, but the magnetic force due to the current running through it. As an initial reference, we will equate the magnetic force through the magnet to the normal force of the washers. Using results from Section 2.4.1 of this design, we see that we will need force equal to the result of equation 8, and that results in a pressure increase equal to the result of equation 10:

$$F = 9.8 * 0.0985\ [N] = .9653[N] \quad (8)$$

$$P = F/A = .9653 / .0000476 = 20.7\ [kPa] \quad (9)$$

$$total\ pressure = 121.57 + 20.27 = 141.84\ [kPa] \quad (10)$$

We will use a manipulation of the equation for the force in a solenoid wrapped piece of metal to determine the amount of current needed to attain that force. Our current design choices give N=40 turns of the wire around the piece of metal, and g = 10 mm of air gap between the electromagnet and the top of the board, the path of magnetic flux.

$$I = \sqrt{\frac{2g^2F}{N^2\mu A}} = \sqrt{\frac{2(10E-2)^2(.9653)}{40^2(4\pi E-7)(.0000476)}} = 44.9\ [mA] \tag{11}$$

Requirement	Verification
● Maintain a current of 44.9mA +/- 5% through electromagnet, as per the	1. Wrap coil around predetermined number of zinc washers. 2. Use oscilloscope to provide a current through wire. 3. Through intuition, determine needed current to successfully attract planchette through the board. 4. Update current requirement value based on experimentation.

2.5 Motor Module

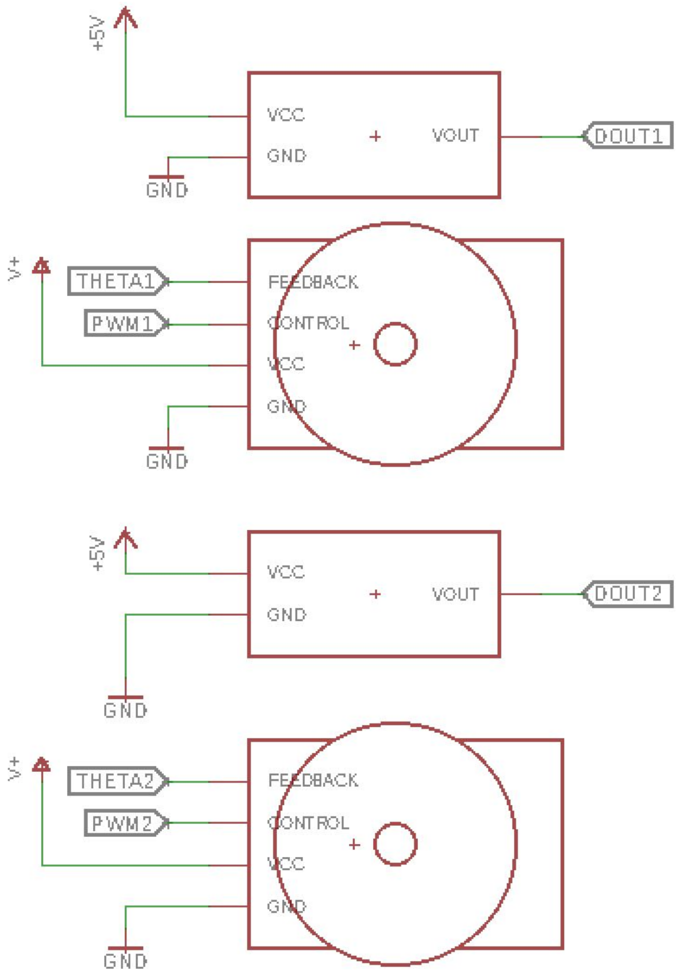


Figure 7 : Schematic of motor[3] module & distance sensor[4]circuit

Each motor module will be responsible for moving one of the positioning rails to the desired position, and will act as the interface between physical goals, and the DSP control algorithm loaded onto the microcontroller. This can be better understood in the microcontroller section. Positioning of the rail moves the electromagnet module and in turn, moves the planchette. The motor module will receive its power from the power module. The module will also take in a PWM signal from the microcontroller to actuate the servo motor and send the necessary data for the microcontroller to know both the angular position of the motor as well as the linear position of the rail.

The physical motor will be attached to each rail and have a gear that provides torque to a driving axel. That axel will be connected to two gears that will mesh with a rack gear inserted on the inside of the box containing the system.

2.5.1 DC Motor w/ Encoder

Requirement	Verification
● Rail stops within 1.25 cm radius of the desired location.	1. Set a desired location location from the microcontroller. 2. Measure distance of rail from desired physical location with a ruler along the major axis and the minor axis. Using pythagorean theorem calculate radius of error.
● Rail moves to desired position with no overshoot.	1. Remove top of ouija board. 2. Set a desired location from the microcontroller with x and y location greater than that of the initial positions of the rails. 3. Using a pencil record on the interior how far it traveled past the final x and y coordinates. If it moves past at all then it fails the test. 4. Repeat steps 2 and 3 with a desired location with a smaller x and y location than the current rail position.
● Relay physical position to microcontroller within a margin of 0.75 cm.	1. Attach oscilloscope lead at the positional data output. 2. Move the rail to an arbitrary position. 3. Read scope data and confirm measurement is accurate.

2.6 Bluetooth Module

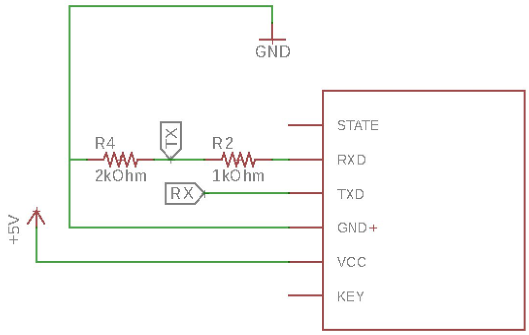


Figure 8: Schematic of Bluetooth[5] module circuit

This module will be purchased (HC-05 Arduino) and act as the means of Bluetooth communication to the phone application. It will use its TX and RX ports to communicate serially with the microcontroller. It will also be powered on the same power rail as the microcontroller. The HC-05 has a default data rate of 38,400 baud (bits per second), so communication rate won't be of much concern because the rate at which a user inputs a word is much less than that, and also the planchette will be moving very slowly by design. However, the main concern is validating that no words were dropped due to weak connection or some other problem. We may have to implement a feedback loop indicating to the phone application that the word was received, and if not, to attempt to transmit it again. Bluetooth LE is known to have some issues with dropping packets (which are just ignored), so we may have to implement a feedback loop to make sure we transmit the correct words and don't skip over any words.

Requirement	Verification
● Can properly communicate a single word (10 bytes/second) with the phone application from within 5 meters.	1. Send signal from the phone app and utilize a measuring tape to check the maximum distance that the BT module can receive signals from.
● Can receive entire words in the correct order from application without dropping packets over 80% of uses (BLE has no mechanism to recover lost packets).	1. Send multiple words via phone app. 2. Verify visually on phone screen that bluetooth module sends back acknowledgement signal for each word. 3. Ensure they're received in the correct order on the microcontroller queue.

2.7 Microcontroller Module

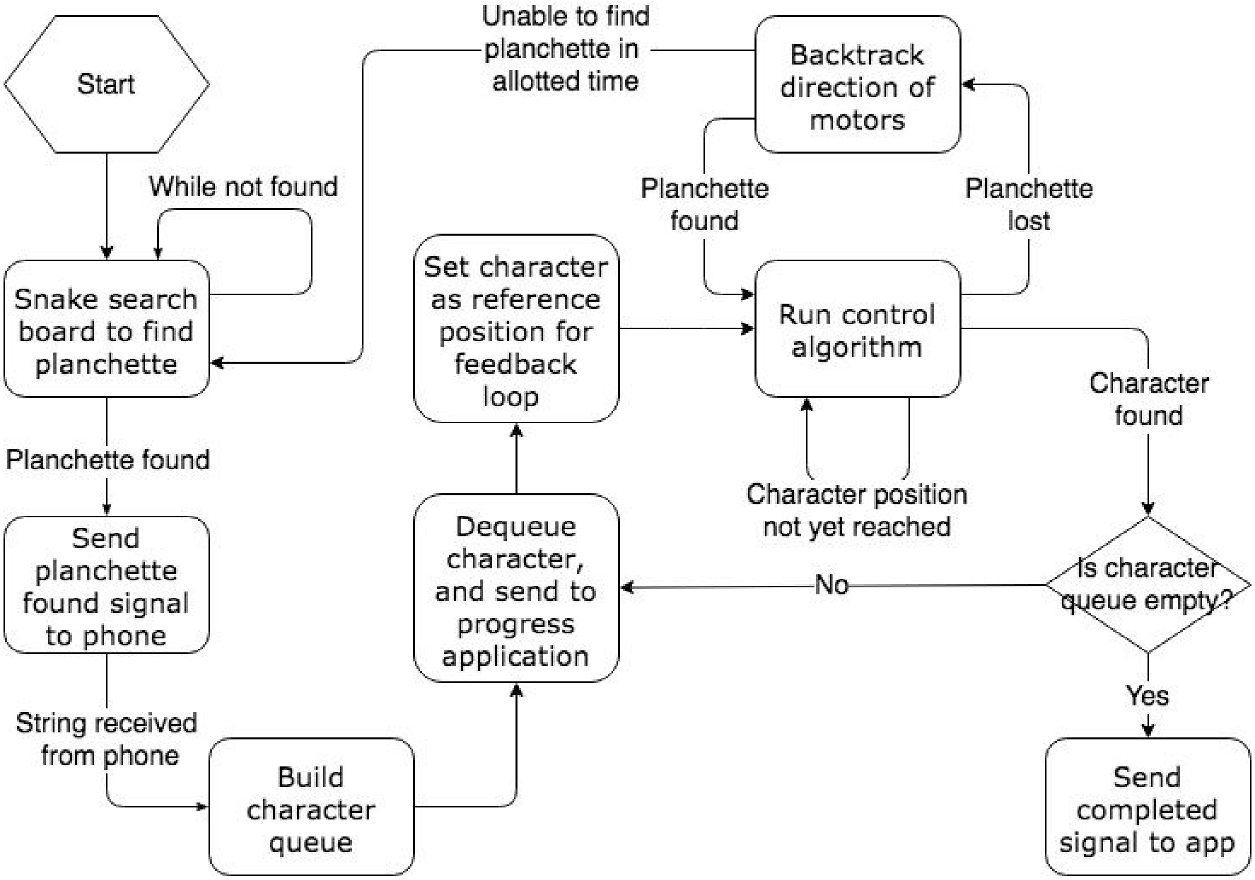

Start
While not found
Snake search board to find planchette
Unable to find planchette in allotted time
Backtrack direction of motors
Planchette found
Planchette lost
Set character as reference position for feedback loop
Run control algorithm
Character found
Planchette found
Send planchette found signal to phone
Dequeue character, and send to progress application
Character position not yet reached
No
Is character queue empty?
Yes
String received from phone
Build character queue
Send completed signal to app

Figure 9: Flowchart for microcontroller behavior

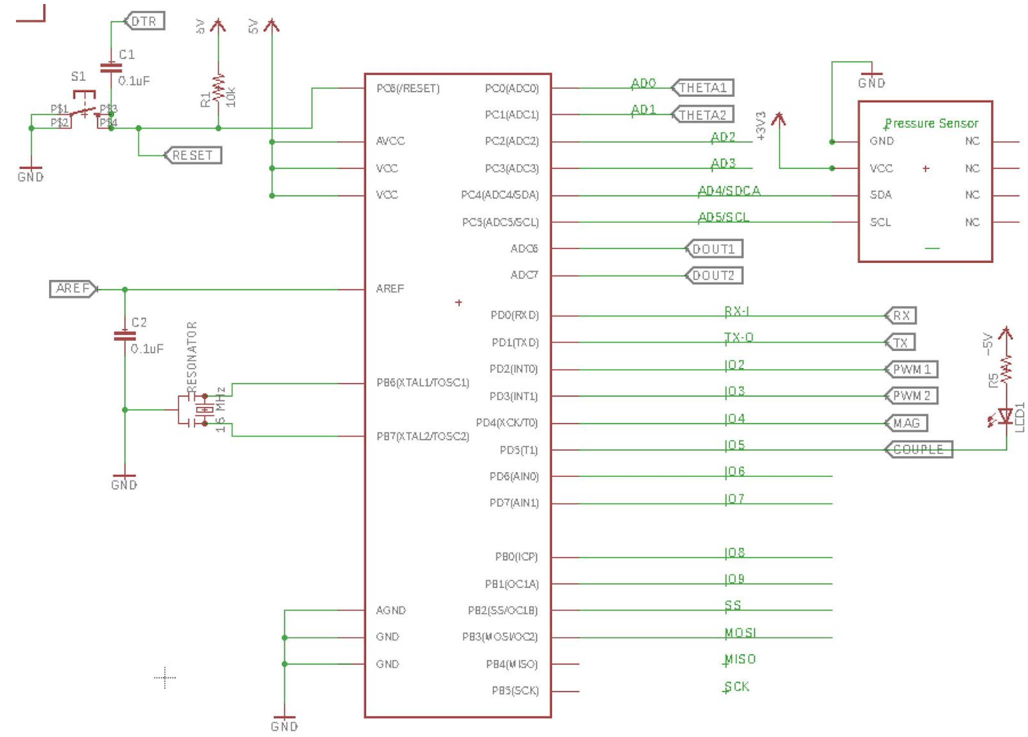


Figure 10: Schematic for microcontroller[6], pressure sensor[7] & related routes

The role of the microcontroller module is to process information received by the various modules and use that to control them. Notably, receive and process the needed string from the Bluetooth module, activate the electromagnet and determine if it is coupled, and use the prior two sets of information to set the proper feedback loop parameters.

The microcontroller has a special “snake search” function that can be seen as an element of the block diagram in figure 9. This function starts the electromagnet module off in one corner of the board and then moves it all the way to the other side of the major axis very slowly. When it reaches the other side it moves slightly lower on the minor axis, then pans back on the major axis. It does this until it has recognized that the plachette has been coupled.

The feedback loop, and its parameters, can be better understood by reading the section titled Control loop calculations. Although we don’t yet know all of the coefficients, we will be able to calculate them based on physical measurements in our manufacturing process.

The microcontroller also turns on and off a “couple” LED that is lit when the planchette is registered as coupled and is turned off when it is registered as decoupled.

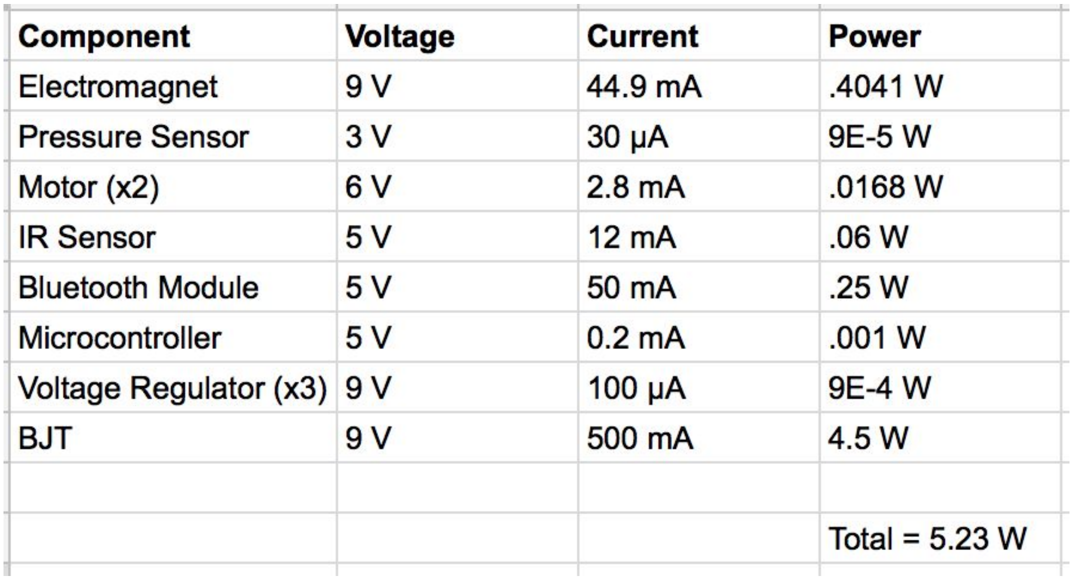
Requirement	Verification
● Can build a queue of ten characters that need to be visited by the planchette based on input from the phone application.	1. Send a string of ten characters to the microcontroller. 2. Confirm that the rails move to the positions corresponding to the characters by measuring the locations they stop at. Also confirm that the order these positions are reached is consistent with the input.
● Output to motor changes within 160 ms after a change in the distance signal.	1. Attach an oscilloscope lead to the output to the motor, and a second oscilloscope lead to the input from the distance sensor. Set the oscilloscope to roll mode. 2. Place an obstruction in from of the distance sensor to spoof the appearance of the position being off. 3. Confirm that the duty cycle being sent to the motor changes within 160 ms from the change in the distance sensor signal.
● Performs snake search function upon request of the phone application.	1. Have the phone application send the signal to perform the sweep function while pressurizing pressure sensor. 2. Confirm that the electromagnet module is search across the x and y directions. 3. Remove pressure from the pressure sensor and confirm that this stops the sweeping.

2.8 Power Supply Module

2.8.1 9V Batteries

We will be using two 9V batteries to power our ouija board. Initially, we thought of using 120V ac wall power for the board, but we then realized that the primary goal of this illusion is to be convincing, and with a visible power cord, we would be missing that goal. The batteries will feed into the voltage regulator so that all components and peripherals adhere to their respective voltage inputs. In order to meet current requirements, we will be placing 5 Duracell MN1604 9V batteries in parallel. Each battery has a current capacity of 580 mAh, and can output current of up to 25mA. Thus placing them in parallel would ensure a total current capacity of 2.9

amp-hours, and a total current of 1.25A enough to power all our components, according to parallel addition of current.



Component	Voltage	Current	Power
Electromagnet	9 V	44.9 mA	.4041 W
Pressure Sensor	3 V	30 µA	9E-5 W
Motor (x2)	6 V	2.8 mA	.0168 W
IR Sensor	5 V	12 mA	.06 W
Bluetooth Module	5 V	50 mA	.25 W
Microcontroller	5 V	0.2 mA	.001 W
Voltage Regulator (x3)	9 V	100 µA	9E-4 W
BJT	9 V	500 mA	4.5 W
			Total = 5.23 W

Assuming same power draw each hour, this means we consume 5.23 Watti-Hours. Using equations 1 and 2 below, we see that we should have our system powered for up to 5 hours. This is a max calculation, assuming 100% runtime for each component of the project.

$$Power\ Rating\ =\ 580\ mAh * 9V\ * 5\ batteries\ =\ 26.1\ WH \qquad (1)$$

$$Maximum\ runtime =\ 26.1\ WH\ /\ 5.23\ WH\ \approx 5\ hours \qquad (2)$$

Requirement	Verification
● Provide up to 5 hours of run time	1. Connect batteries to system 2. Continually all parts of system 3. Ensure the batteries are functional for up to hours.

2.8.2 Voltage Regulator

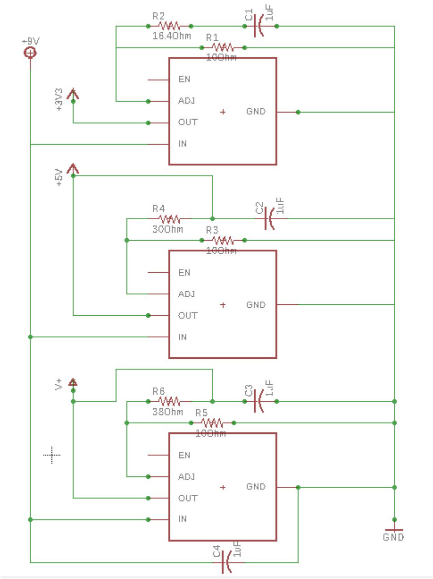


Figure 11: Schematic for Voltage Regulator[8] and related routes

Our battery will be able to deliver 9V, however this will be too much for many of our components. Thus we will need three voltage regulators to step down that voltage. We will need one to go from 9V to 5V for our IR sensors, Bluetooth module, and microcontroller. The second will go from 9V to 3.3V for the pressure sensor, and the final regulator will step down to 6V for the motors. The datasheet for the regulator necessitates that each capacitor be at least 1μF. In order to obtain the necessary “bucked” voltages, the equations below were used to obtain resistances necessary. We chose a small 10 Ω resistor for R2, and V_{ADJ} is to be set to 1.25V based on the data sheet.

$$V_{out} = V_{ADJ}(1 + R_1/R_2) \quad (3)$$

$$3.3 = 1.25(1 + R_1/10) \rightarrow R_1 = 16.4\Omega \quad (4)$$

$$5 = 1.25(1 + R_1/10) \rightarrow R_1 = 30\Omega \quad (5)$$

$$6 = 1.25(1 + R_1/10) \rightarrow R_1 = 38\Omega \quad (6)$$

Requirement	Verification
- Supply 5V +/- 5% to bluetooth module, ATmega328P, and IR sensors.	1. Perform LTSpice simulation on regulator schematics with different parameters to ensure that voltage stays within range.
- Supply 6V +/- 5% to each DC motor.	1. Complete same procedure as above.
- Supply 3.3V +/- 5% to pressure sensor.	1. Complete same procedure as above.
- Output of 0V +/- .05V when enable is low.	1. When we have the actual chip, use regulator. 2. Use voltmeter to measure output.

Control Loop Calculations:

$$\theta(t) \equiv motor\ angle, \quad \tau(t) \equiv motor\ torque, \quad F(t) \equiv force\ on\ EM\ module,$$
$$a(t) \equiv acceleration\ of\ module, \quad \alpha(t) \equiv angular\ acceleration\ of\ module,$$
$$v(t) = velocity\ of\ module, \quad \omega(t) \equiv angular\ velocity\ of\ motor,$$
$$u(t) \equiv input\ reference\ signal, \quad x(t) position\ of\ module$$

For the following equations, all proportionality constants will be left in terms of subscripts of k, these constants will be tuned based upon physical measurements in the assembly process. Assuming the DC motor to be a single pole system gives us the following equation to start our model, which can be manipulated into a transfer function:

$$(1):\ \tau(t) = k_0 * \alpha(t) = k_1 * \left(u(t) - \omega(t)\right)$$

$$(2):\ \ddot{\theta}(t) = k_1/k_0(u(t) - \dot{\theta}(t))$$

Using Laplace transforms:

$$(3):\ \mathrm{s}^2 * \Theta(s) = \frac{k_1}{k_0} * \left(U(s) - s * \Theta(s)\right)\ and\ (4){:}\ \mathrm{T}(s) = k_0 * s^2 * \Theta(s)$$

$$(5){:}\ \left(s^2 - \frac{k_1}{k_0} * s\right)\Theta(s) = \frac{k_1}{k_0} * U(s)$$

$$(6){:}\ \frac{\Theta(s)}{U(s)} = \frac{\frac{k_1}{k_0}}{s^2 - \frac{k_1}{k_0} * s}\ and\ (7){:}\ \frac{\mathrm{T}(s)}{U(s)} = k_1 * \frac{s^2}{s^2 - \frac{k_1}{k_0} * s}$$

The torque being put on the is related to the force the worm gear puts on the EM module by a ratio R. If the motion of the electromagnet module is not only governed by the force provided by the worm gear but also some friction gives us the following model:

$$(8): F(t) = a(t) * k_2 = k_3 * v(t) + k_4 * R * \tau(t)$$

$$(9): \ddot{x}(t) * k_2 = k_3 * \dot{x}(t) + k_4 * R * \tau(t)$$

Using Laplace transform:

$$(10): s^2 * k_2 * X(s) = \ k_3 * s * X(s) + k_4 * R * \mathrm{T}(s)$$

$$(11): \frac{X(s)}{\mathrm{T}(s)} = k_4 * \frac{R}{s^2 * k_2 - k_3 * s}$$

Substituting in equation (7) into (11):

$$(12): P(s) = \frac{X(s)}{U(s)} = \frac{k_4 * R * s^2}{(s^2 - \frac{k_1}{k_0} * s) * (s^2 * k_2 - k_3 * s)}$$

Although this is the most accurate model, based on experiments run on our current worm gear and module configuration, it is excessive. We expect that we will more realistically use equation (6) multiplied by an additional constant. This is because in our experimentation the linear position of the module is a direct scalar multiple of the angular position of the gear. This equation can be seen below in equation (13):

$$(13): H(s) = \frac{X(s)}{U(s)} = k_5 * \frac{\frac{k_1}{k_0}}{s^2 - \frac{k_1}{k_0} s} = \frac{k_6}{s^2 + k_7 s}$$

By using the reduced model for an analog controller with two poles defined as G(s) and using a negative feedback loop like that in figure 12, to get overall system transfer function of equation (15).

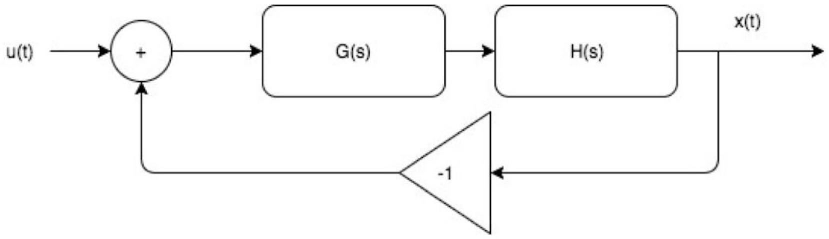


Figure 12: Control loop

$$(14): G(s) = \frac{sb_0}{s^2 + \ sa_1 + a_0}$$

$$(15): \frac{H(s)G(s)}{1 + H(s)G(s)} = \frac{k_6 * b_0}{(s^3 + (k_7 + a_1)s^2 + (a_0 + k_7 * a_1)s + a_0 * k_7)}$$

By using the Routh-Hurwitz stability criterion we can select values for a_0, a_1 and b_0 to stabilize the system. We can then use the bilinear transform on equation 14 to get difference equation that will be executed on our microcontroller. The value of the period T in the bilinear transform will be no greater than 160 milliseconds.

3. Tolerance Analysis

The most important tolerance we must maintain is that the planchette stay coupled to the magnet. It is important that our phone app, electromagnet module, and motor module work in unison to maintain this extremely important property of our project. The first aspect that needs to be accounted for is that the current that goes through the magnet. To determine this initially, we worked backwards from the normal force of the washers on the pressure sensor, to determine the current through the magnet. Using Newton's Second Law, as detailed in the above sections, the electromagnet (10 washers with a coil of 40 turns wrapped around it) applies 0.9563 N of force on the pressure sensor. Using the equations for mmf in a magnetic circuit, we found that if there is a current of 44.9 mA going through the magnet, we would be able to have enough force to attract the planchette.

However, upon going to the lab and physically hooking up the magnet we found that we would need far more turns of the coil to effectively attract the planchette, and we found the necessary current to be about 3.5 A, using a ¼ in. diameter, 3 in. long steel bolt with an estimated 400 turns of wire around it. We will be further experimenting with a different number of turns of the coil, different shaped metal objects to wrap the coil around and different air gap ratios between the board and electromagnet to determine exactly what works as a threshold for magnetic coupling.

To further investigate the event of magnetic decoupling, we will need to examine how far away the planchette can be from the magnet before this event. We have developed the following theory for the necessary magnet strength to keep the planchette coupled. Using the maximum translational speed of the electromagnet module and the sampling rate of the pressure sensor, we can calculate a minimum distance that the magnet needs to stay coupled within in order to guarantee that our system can recognize decoupling. Both worm gears will be driven at a maximum speed of 160 rpm. The worm gear on the major axis translates at .636 cm/revolution and the minor axis translates at .508 cm/revolution. This caps the speed of the electromagnet module at 2.17 cm/s. Since we will be sampling the pressure sensor at least every 200 ms, this means the most that the module can move between any two samples is .435 cm. In order to fully develop the relationship between motor speed and magnetic strength, more testing will need to be done with the fully developed motor-rail system.

4. Cost

Part	Unit Cost	Units Needed	Sub Total Cost
Atmel ATMega328P	$2.50	1	$2.50
HC-05 Arduino Wireless Bluetooth Receiver	$7.69	1	$7.69
Sharp GP2Y0A41SK0F Analog Distance Sensor	$9.55	2	$19.10
Honeywell SSCMNNN030PA2A3 Pressure Sensor	$31.88	1	$31.88
Parallax 900-00360 DC Servo Motors	$24.99	2	$49.98
Texas Instruments LP38692MP Voltage Regulator	$1.58	3	$4.74
FLORA SEWABLE 3-PIN JST WIRING A (wiring for distance sensors)	$1.95	4	$7.80
LED DURIS S 5 GREEN 540NM 2SMD	$0.73	2	$1.46
IC USB SERIAL FULL UART 20SSOP (serial processor)	$2.12	2	$4.24
Resistors of values 2k, 1k, 10k, 10, 39, 30, 16.2, 27 ohms	$0.04	80	$3.20

Capacitors of value .1uF, 47 pF, 1uF, 10uF	$0.20	15	$3.00
SWITCH PUSHBUTTON SPST 1A 30V (Push button)	$0.74	3	$2.20
OSC XO 16.000MHZ HCMOS TTL SMD (16MHz Oscillator)	$1.31	3	$3.93
MOSFET N-CH 60V 0.31A SOT323 (Mosfet)	$1.31	3	$3.93
Our Labor	$50.00/hr	3 x 11 weeks x 12 hours	$19,800
Total Cost			$19,944.76

5. Schedule

Week	Luke	Nikhil	Tobi
2/26	Order components parts and refine physical design. Work on PCB layout. Complete slide deck and presentation.	Initiate programming phone app backend code. Make list of high level software requirements.	Use wire for extensive testing of needed current and force.
3/5	Attach sensors and motors to mechanical prototype. Take physical measurements to calculate parameters of controller.	Continue programming phone backend; begin programming phone app for receiving Bluetooth signals. Research microcontroller Bluetooth protocol.	Write drivers for I2C communication protocol between the microcontroller and the pressure/IR sensor
3/12	Write control loop subroutine. Iteratively test with original parameters to refine the model.	Complete Bluetooth feedback on microcontroller. Verify that phone app and microcontroller can communicate.	Use physical measurements to determine clearance between electromagnet housing and components
3/19	**Spring Break**	**Spring Break**	**Spring Break**
3/26	Finish PCB layout. Test motors on physical rails to confirm they reach the correct point. Have functioning control loop. Integrate pressure sensor subroutines developed by Tobi into control loop.	Continue working on integrating Bluetooth signals and phone backend. Develop beginnings of user interface.	Test batteries in conjunction with voltage regulators to ensure proper functionality

4/2	Integrate Bluetooth subroutines to get characters actually communicated from phone.	Test receiving and sending appropriate signals to and from microcontroller	Begin process of soldering PCB. Continue battery safety and power testing.
4/9	Debug integration of various subroutines. Ensure adherence of code to flowchart.	Test full functionality of phone app with Bluetooth and microcontroller.	Aid in other aspects of the project, namely connectivity between sensors, microcontroller, and application
4/16	Debug control loop integration with sensors and micro.	Debug phone application and bluetooth signals.	Prepare appropriate method of presentation for mock demo.
4/23	Prepare mock presentation.	Divide up speaking points for mock presentation and prepare all documents/work needed for mock presentation.	Continue refining the product demonstration.
4/30	Work on final report sections relating to motor modules, microcontroller, and PCB layout	Work on final presentation.	Divide up sections for report. Aid in both final report and presentation.

6. Ethics and Safety

The core safety concern for our project is power consumption. We will be using 9V batteries, potentially with multiple in series. This allows us to achieve greater than 9 volts, but also introduces the potential of supplying too much voltage to our components. Cautious use when the system is plugged and diligent awareness of which traces are carrying high voltage currents prevents crises (or unsafe conditions). The most care should be taken with the motors, as they are the component that would require the greatest amount of power. Other components such as sensors or the microcontroller are fairly low powered and low strength, and do not pose much risk.

The primary ethical concern associated with our project is the potential for it to ruin one’s reputation. Our project can be described most simply as an illusion that a supernatural being is attempting to communicate with the user. Although it may sound laughable to some, others are easily convinced that such spirits may be trying to tell them things. This proposes the risk that the project we design could be used to maliciously deceive people in such a way that harms their reputation. We’ve reviewed the IEEE guidelines for ethics[9] and although we cannot fully prevent someone from misusing our project to deceive or hurt someone’s reputation, we have considered this in our design and thus plan to make it apparent that this product is purely for entertainment purposes.

References

[1] A. Romano, "How Ouija boards work. (Hint: It's not ghosts.)", *Vox*, Sept. 29 2017. [Online]. Available: https://www.vox.com/2016/10/29/13301590/how-ouija-boards-work-debunked-ideomotor-effect. [Accessed: 19- Feb- 2018]

[2] “Animated ‘Haunted’ Ouija Board,” *instructables.com*, Nov. 7 2007. [Online]. Available: http://www.instructables.com/id/Animated-Haunted-Ouija-Board. [Accessed: 21- March- 2018]

[3] Parallax, “Parallax Feedback 360° High-Speed Servo”, 900-00360 datasheet, Sept. 2017.

[4] Sharp, “Distance Measuring Sensor Unit”, GP2Y0A41SK0F datasheet.

[5] GM Electronic, “HC-05 Bluetooth Module User’s Manual”, HC-05 datasheet.

[6] Atmel, “Atmel-42735B-ATmega328/P_Datasheet_Complete”, ATmega328/P datasheet, Nov. 2016.

[7] Honeywell, “TruStability Board Mount Pressure Sensors”, SSCMNNN030PA2A3 datasheet.

[8] Texas Instruments, “LP3869x-ADJ 1-A Low Dropout CMOS Linear Regulator With Adjustable Output”, LP3869 datasheet, Dec. 2004 [Revised February 2016].

[9] "IEEE IEEE Code of Ethics", *Ieee.org*, 2018. [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8.html. [Accessed: 19- Feb- 2018]