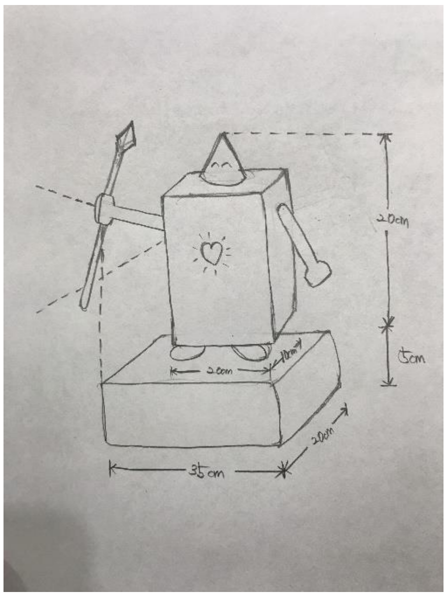
Design Document

Group 7

- Trevon
- Cong
- Will
- Jeon

Project Description

We intend to design a robot thespian from the classic film *Alice in Wonderland*. The character we want to design is a card, one of the queen's guards in the underworld. The card will be able to move its spear in a vertical motion, light up its card suit, and playback pre-recorded messages.

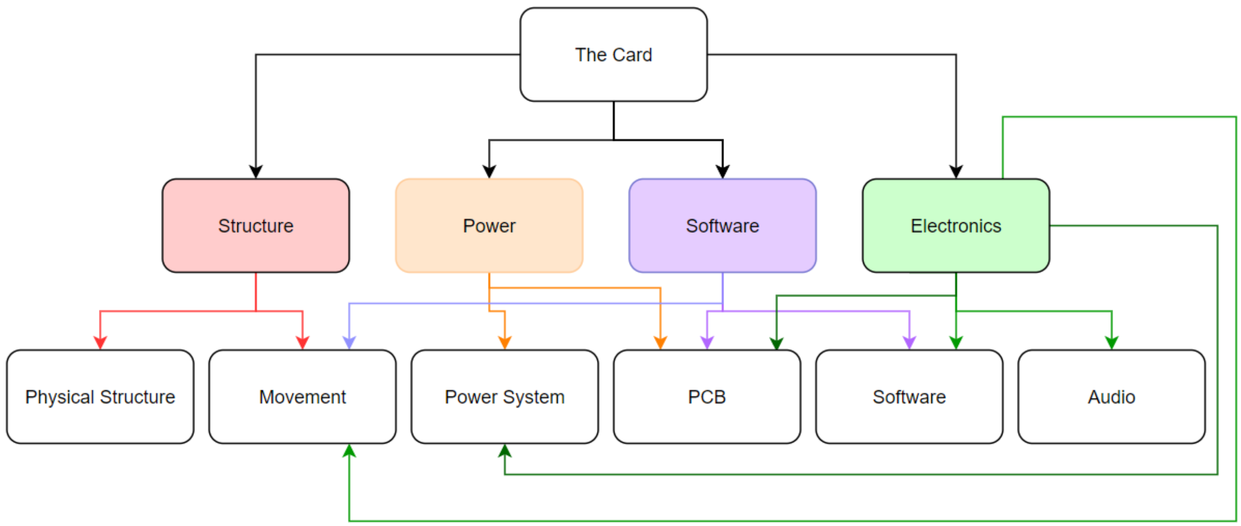


System Design

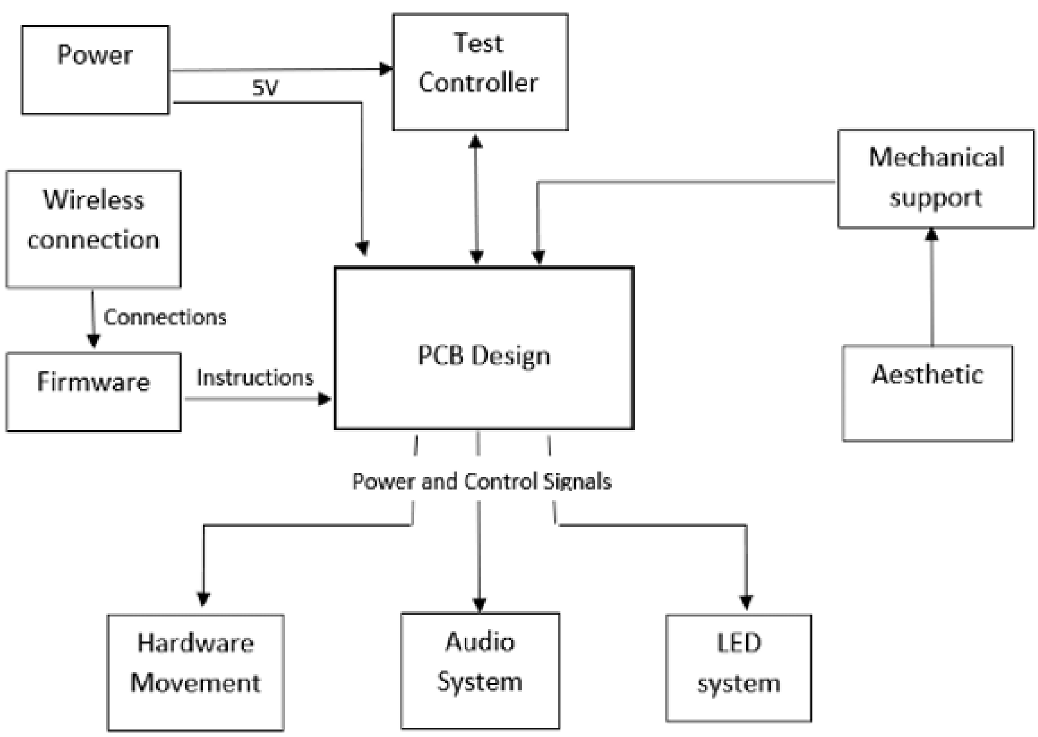
The goal is to create a robot thespian with a moveable body, a head, two feet, arms, and a spear. Since the character itself is 2-D, we want to mimic that visual from the film.

The inputs of the system are: 120 volts AC input power, instructions from the director, and test mode button. The outputs are movement from the thespian, and a speaker. We currently have 6 subsystems: power, movement, PCB Design, structural, hardware, and software.

Hierarchical Design



Block Diagram

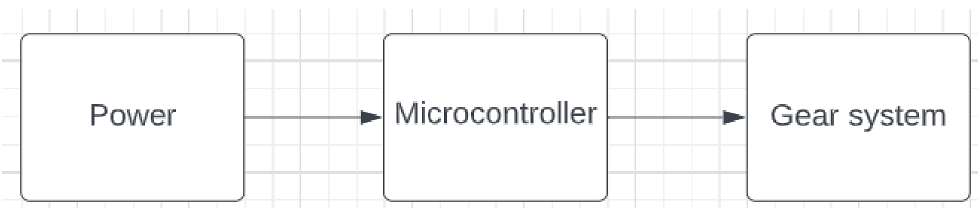


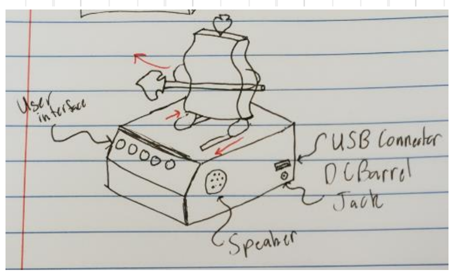
Sub-system Designs

The power sub-system will use 120 volts which will be converted to 5 volts DC. This will be converted through a transformer before being sent to the microcontroller.

The movement sub-system encompasses all the mechanical movements we want our thespian to have. The two current primary points of movement are the arms, which will both be attached to the spear for the guard, and the body. The plan is to laser cut both and glue them to the spear. The arms will move in a forward motion to simulate how an actual spear would be used in combat. Additionally, this helps satisfy the requirement to have the thespian grow and shrink in size.

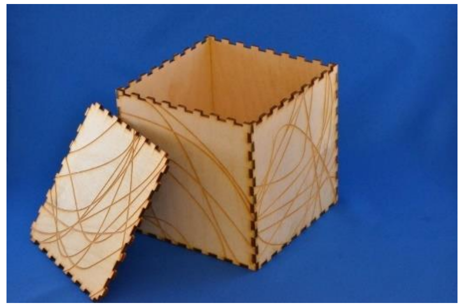
On another note, the body will be able to twist back and forth. The plan is to have a moveable track under the feet. We will implement a gear system to control the forward and backward movement of the thespian. The figure below visualizes this effect.



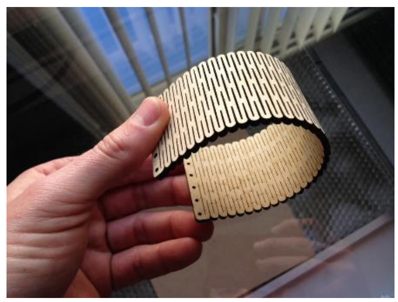


PCB Design is another sub-system. We have to design and head to the Hive to create it. The PCB can route…

For the structural sub-system, we will go through design, fabrication, and testing. We will create the base first and use a puzzle-piece pattern to interlock the base together, like the example below.



Also, for the body of the card, it will be created with wood. We are considering a design that is flexible. The pattern is repeated throughout the structure. Certain designs favor durability, while others favor flexibility.



For the hardware design, we wanted to supply our microcontroller with 5 V converted from 120 V to be supplied to an Arduino microcontroller.

For the software design, we wanted to be able to connect to the director seamlessly. We have implemented an Arduino uno and it is able to light LEDs, play audio, and utilize pushbuttons.

Constraints and Alternatives

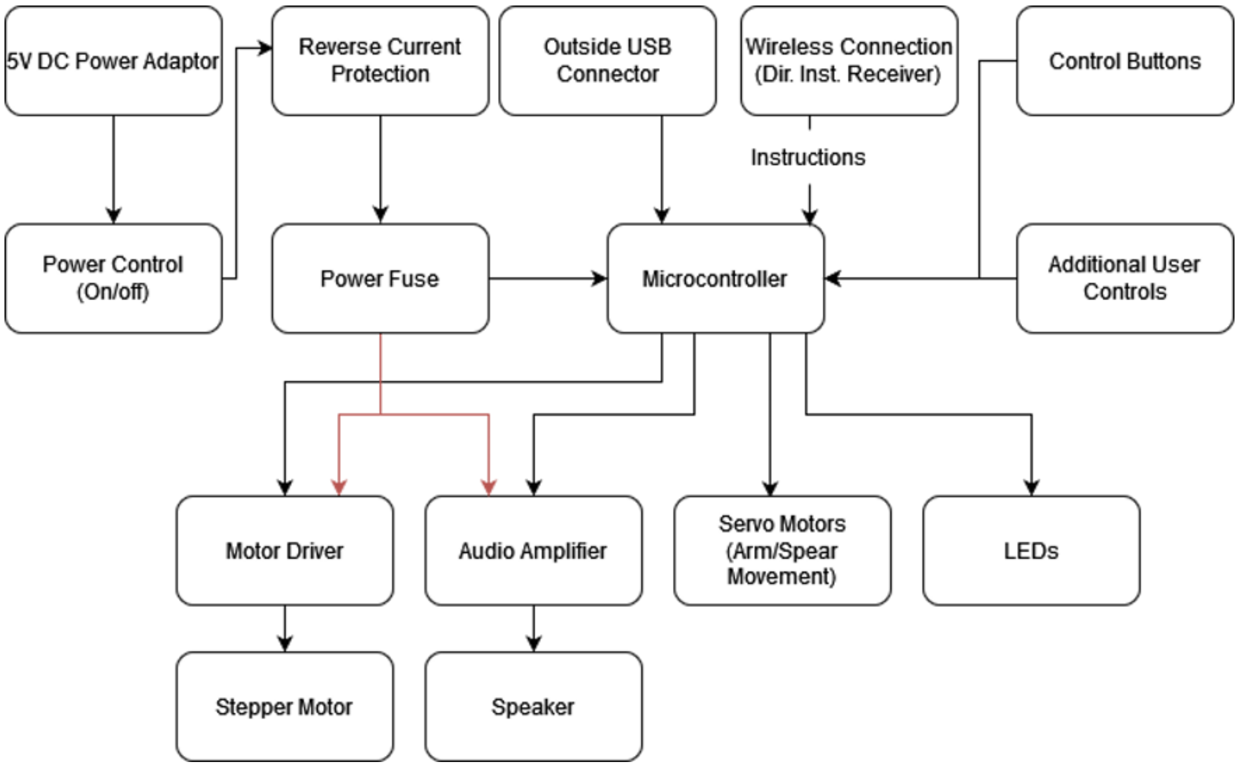
The biggest threat to our project is maintaining our project schedule. Keeping on track with design choices and the process is important to finishing the project on time.

An alternative to using flexible wood would be just using non-flexible wood. It would make the design much simpler and easier to fabricate.

In the post CDR (above picture), we decided to change the bendable wood design to a box format. We had concluded that the bendable wood would be too difficult to test and implement given the time constraints of this semester. Simplifying our design allowed us to finish the project in a timely manner. Additionally, it made it more structurally sound and reliable to work on.

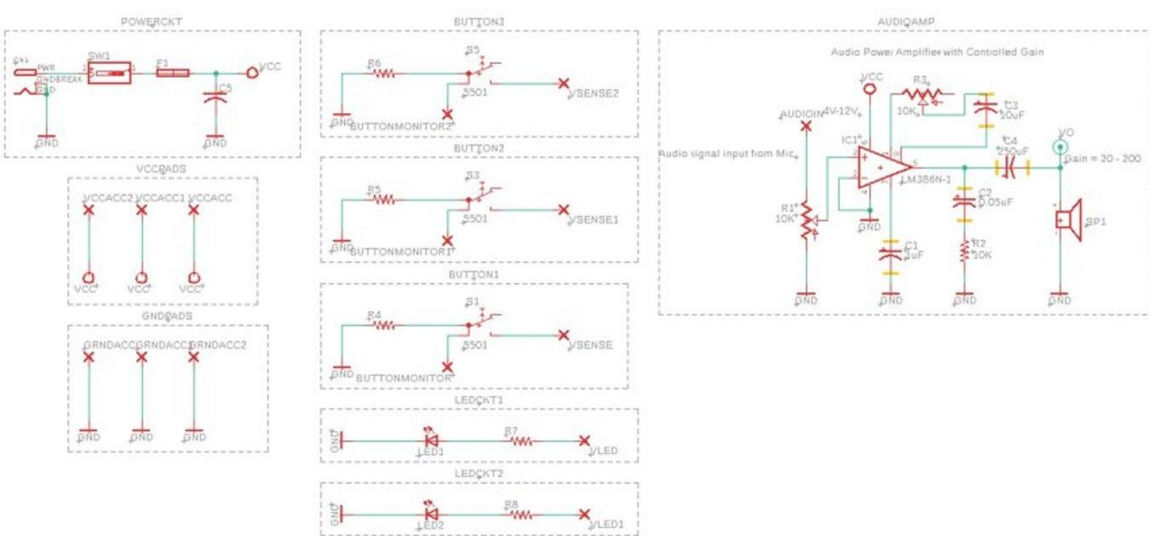
Electronic Design (Will)

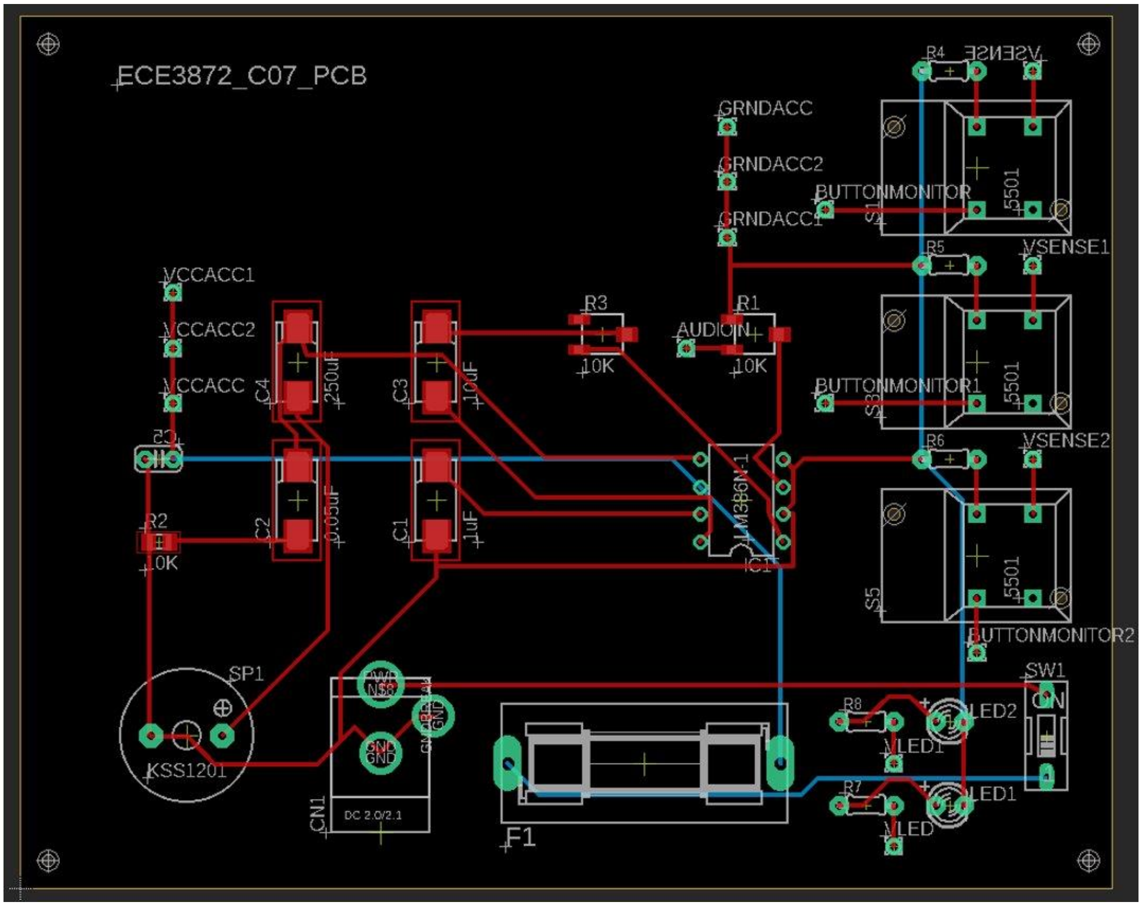
We will be employing an Arduino Uno as our microcontroller working in tandem with a few separate circuits and devices. Below, a high-level block diagram is provided to give an idea of the structure of the design. This structure has stayed the same through new simulations and tests, but a flexible mindsight will still be employed moving forward.



Lines with arrows are used to denote input/output relationships in the diagram, with outputs being grouped at the bottom and inputs grouped in the top two lines.

We have designed the following PCB (schematic and board pictured below) to act as a landing pad and routing place for all electrical components.





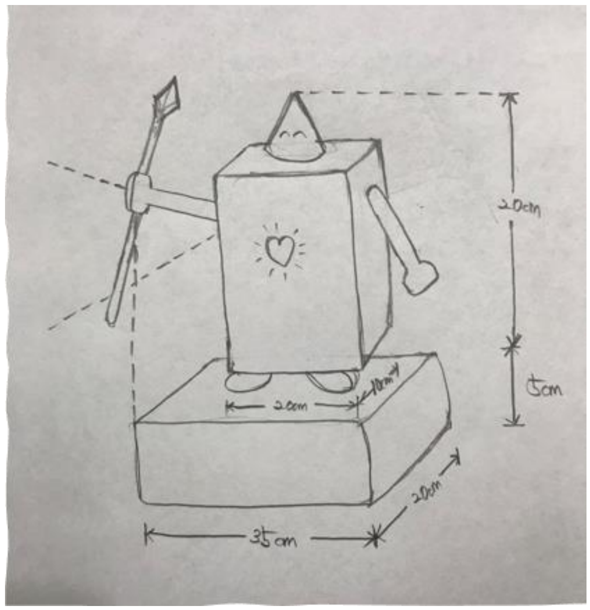
Currently, the board has integration for pushbuttons, LEDs, audio system, servo motors, and necessary power circuitry. We included voltage supply and ground pads on the board to allow for future components to be included if need be.

Our next step will be to finish breadboarding the stepper motor circuit and the wifi module integration. After that, the final steps will be to assemble and solder the PCB and necessary components in the final container for the project.

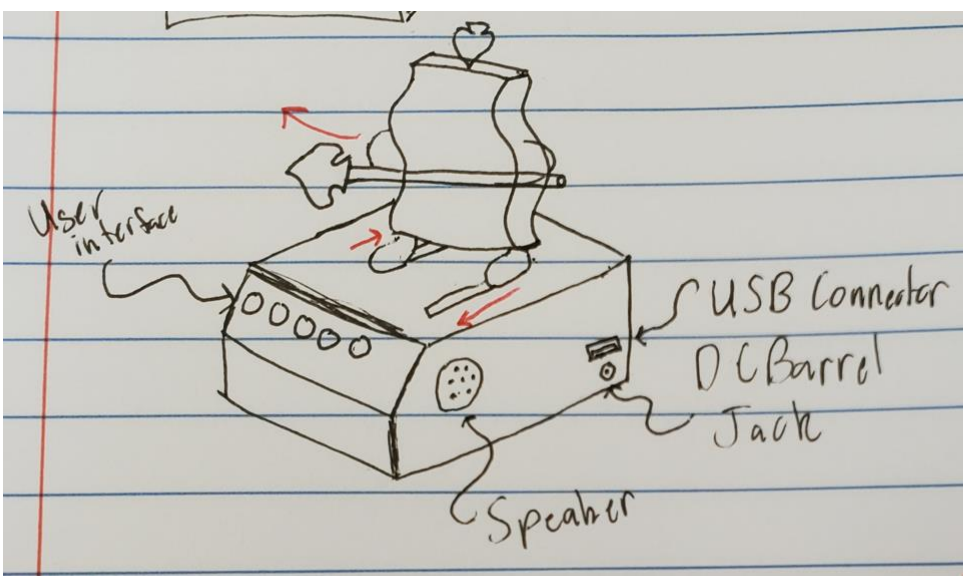
Mechanical Design (Will)

Our mechanical design utilizes servo and stepper motors and a gear system that will allow the card soldier to “march” forward and thrust his spear out and up. Seeing as our application will not require any unique movement or conversion of movement, we will be sourcing our gears from the Senior design lab or credited vendor, such as McMaster-Carr or Mouser. The servo motor(s) will be used to rotate gears on the left and right side of the card soldier at his shoulder joints. We will use a combination of gears and rods to translate this movement into his thrust. The stepper motor will be used to drive the tracks that move the feet in opposite directions. This will be accomplished by threading gears perpendicular to and on opposite sides of the upward facing stepper gear. This will convert the plane of movement as well as produce synchronized, opposite-direction translational movement for the feet of the soldier. The figures below provide context to the current design.

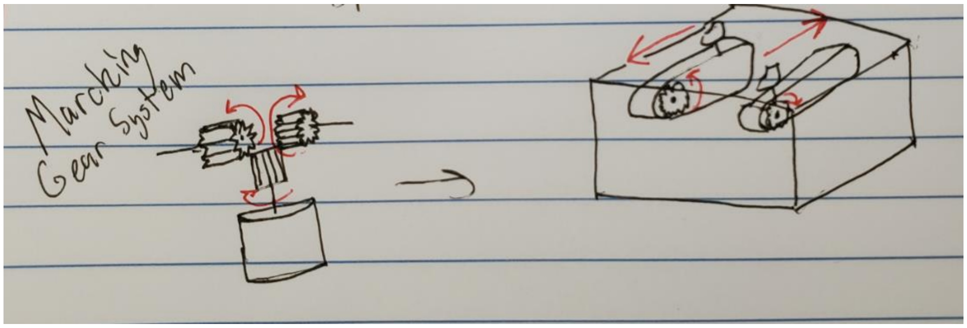
This was our earliest sketch of what the card soldier might look like, giving us a place to build upon.



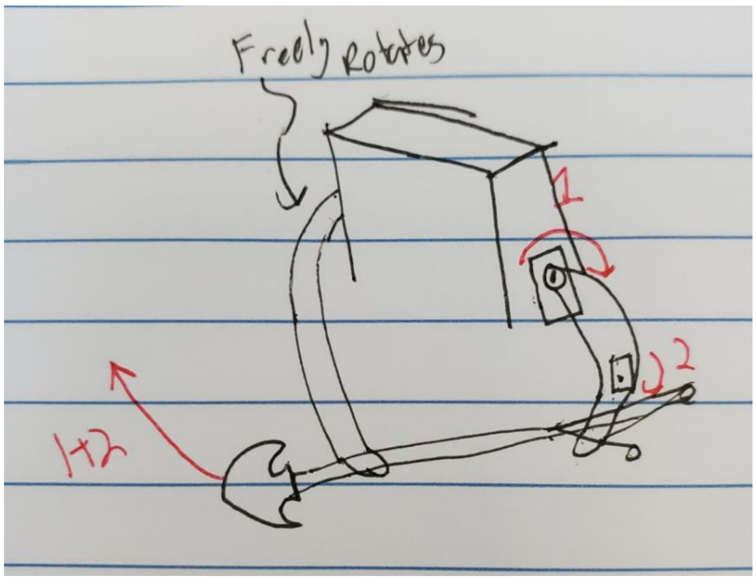
A more recent sketch begins to indicate movements and features on our design, such as the marching motion and the spear thrusting motion.



The following sketch is the mock-up of how the marching motion will be driven by one motor on tracks.



Finally, the following sketch illustrates how the spear thrust might work.



Our next step is to start implementing physical models/prototypes of these designs to work out any problems that might arise while still achieving our initial goal of marching and thrusting the spear.

In the post CDR iteration of the Card, we updated it to have one moving arm instead of attaching both arms together. Originally, it was supposed to have two arms where one has a servo attached and the other arm is attached to the opposite arm. In conclusion, we decided to have one moving arm and one stationary arm. It made it significantly easier to work with and didn't draw away from the visual effect.

Software Design (Cong)

In software design, there are 4 different states including: OFF, READY, TESTMODE, and OPERATION shown in Figure 3.1. In which, Operation mode had 2 sub-states of Mechanical Moving and Visual Effects. Layer Software Architecture is developed to add more details on how the system works during Ready state in Figure 3.2. To connect main and sub-states, we come up with state diagram, which is preliminary ideas to design software architecture in Proteus in Figure 3.3. In our design, a schematic is used to allocate features and components such as 2 motors, speaker, 3 pushbuttons, battery, and PCB. The board Arduino Uno is used to develop our software with functions of playSpeaker(), fadLed(), and playMotor(). The program architecture design provides more details how codes can be applied in Figure 3.4.

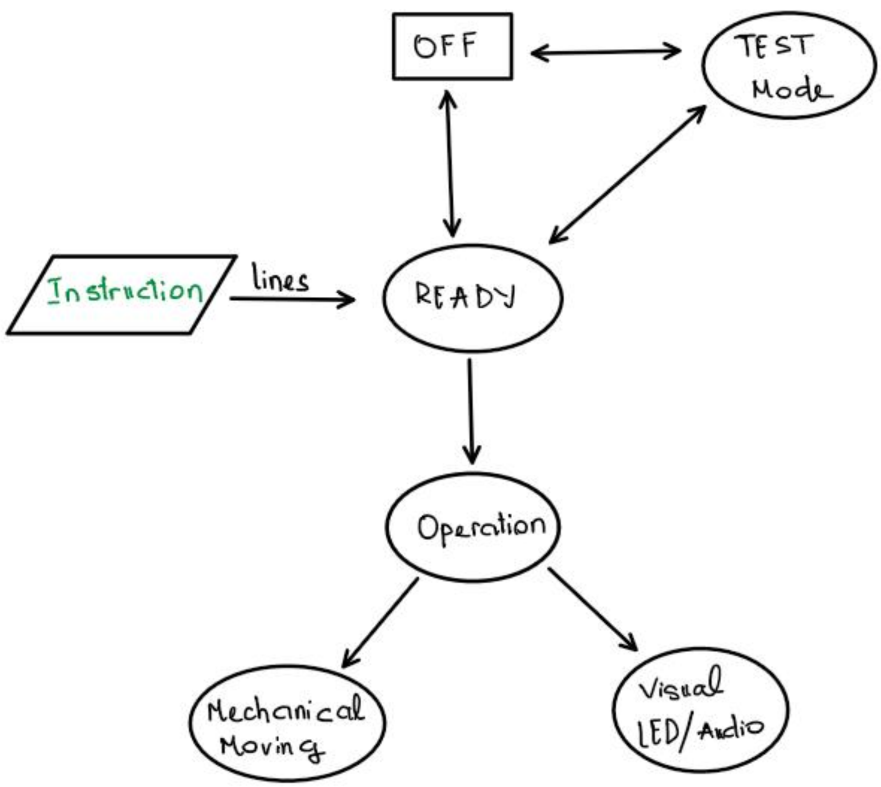


Figure 3.1: Software state machine with states OFF, READY, TESTMODE, and OPERATION

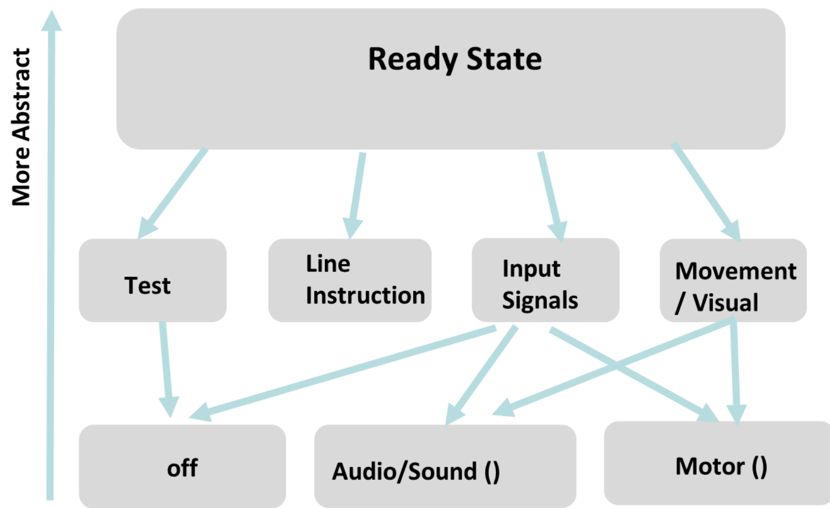


Figure 3.2: Layer software architecture with states OFF, READY, TESTMODE, and OPERATION

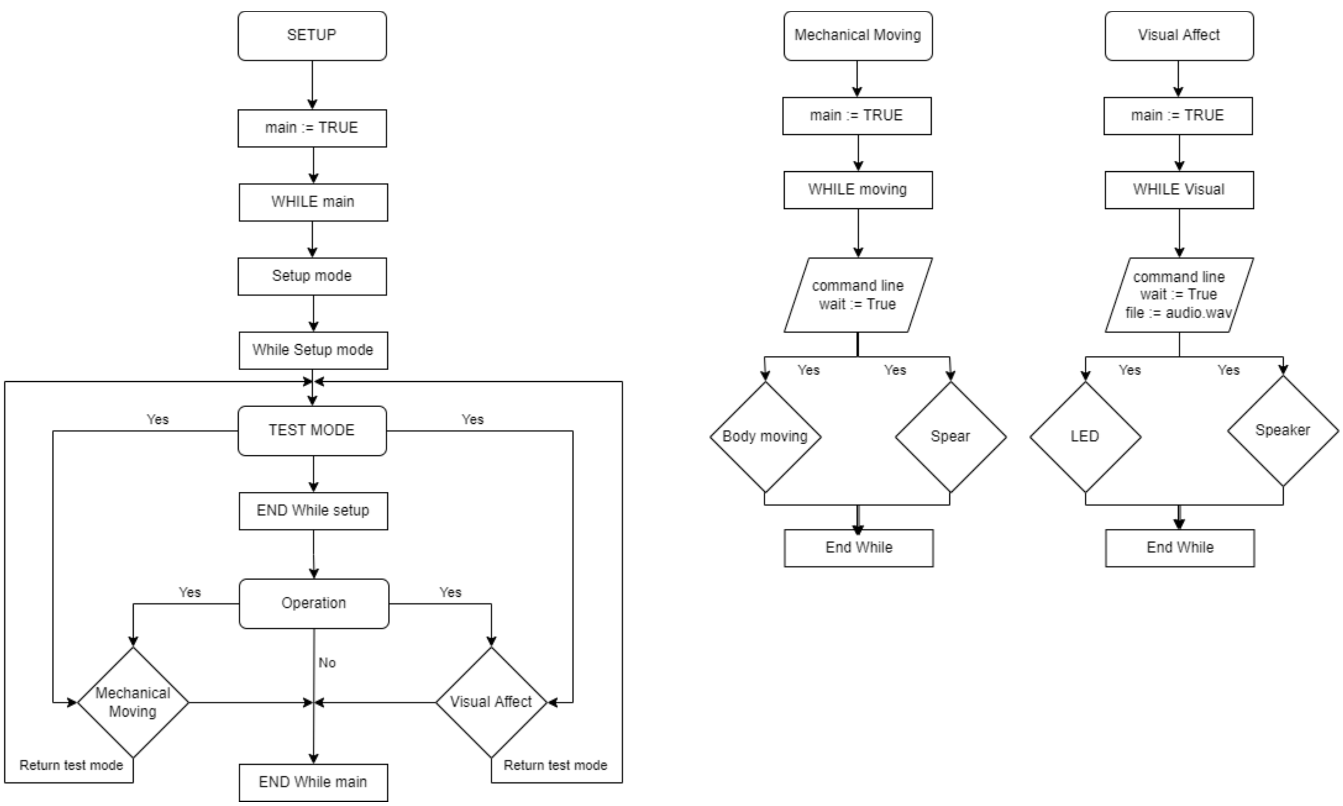
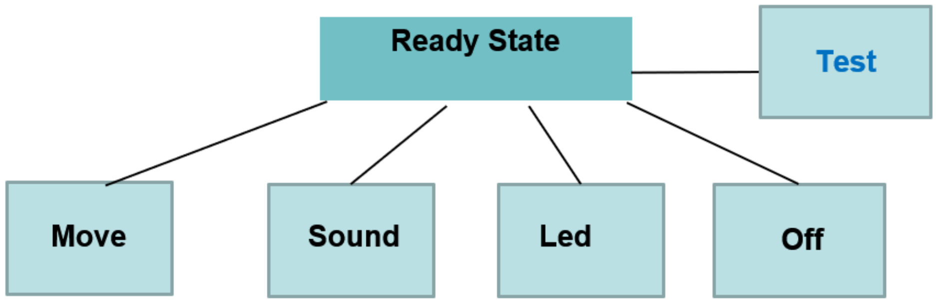


Figure 3.3: Software architecture includes READY, TESTMODE, and OPERATE



State Machine Manager Pseudocode

```
main(){
Initialize
If notready, test(); break;
while (1){
switch(u){
        case 1: u = move();break:
        case 2: u = sound();break;
        case 3: u = led(); break;
        default: u = off(); break; }}
```

Figure 3.4: Software architecture design includes OFF, READY, TESTMODE, and OPERATION

In the final iteration of the Card's code (post-CDR), we wanted to give the robot more flair and personality. We decided to use our 3 buttons for 3 different actions that go along with the story.

- The buttons signify 3 different actions the Card can do: Introduce, Marching, and Alert.
  - Introduce: This is to introduce the Queen of Hearts. The card moves its arm to hit its spear on the checkerboard floor and starts playing a fanfare.
    - Code Explanation: The song and the command to move the spear are in a for loop and it is set to play 16 notes before banging the spear. It loops 3 times.
  - Marching: The Card will play a short tune to signal the beginning of the march and then move its legs a few times.
    - Code Explanation: The Card plays a short, predetermined song melody and then calls the command for moving it's legs at a slow speed.
  - Alert: There is an emergency in the Queen's palace and the Card will signal the alarm by playing the same note four times. It then moves its legs quickly to get to the Queen and thrusts its spear to ward off any enemies.

- Code Explanation: The alert melody is called. It has 4 of the same notes in an array. The Arduino calls the music function and plays the song and then the robot moves its legs. Lastly. The card moves its spear up and down 3 times.

Software Simulation

In software, we use Arduino Uno via Arduino IDE to develop a program that can control motors, sounds, and 2 LEDs. Our codes include parts of define constants to support initialization, initialized parts with information about pins (speaker, 2 LEDs, motors, buttons), define notes to play melody on speaker, loop of program, and 3 functions of playSpeaker(), playMotor(), and fadLed(). We use one fading Led at heart of Card, and another one at head of Card for decorating such as eye. Speaker plays melody while motors move the card's spear (up down) and head (rotate). During the test, 3 buttons are used to check functions of Led, Speaker, and Motor. First, button_1 turns on speaker. A playSpeaker() function is called to play a short melody. Next, we use button_2 to trigger both Led and play a long melody. The melody is 1-2 min is long enough to express the card in motion with designed movements such as spear and head move, LED fade at heart. The button_3 is used to test motor. It can control a DC motor which is used to rotate the head of card. The results are come as expected. Also, Test is successful by DC power vs. batteries including 3.3 V and 9 V.

Link: https://youtu.be/vs_DEEJ0wTE

Interface Design (CONG)

The interfaces between sub-systems include inputs and outputs. Among those output will be inputs of another sub-system. Power system is a basic system to take current power input from 5v DC adapter or 9V Batteries to generate output of 3.3V V-out and 9v V-in. The output voltages are the input for Hardware systems. Namely, power source drives PCB with all components such as LEDs, speaker, and motors, as well as Arduino Uno and all electric modules. Software system has input of line instructions to give outputs of signals to control electrics including motors, servo, LEDs, and speaker. Therefore, electronic components need two input at the same time: power input from Power subsystem and signal input from Software Subsystem to make movements. Outputs of electronic operation is the input of movement subsystem as well as visual/audio subsystem. The output of movement subsystem is to change physical structural and make the spear moved and the head of card rotated. Structural subsystem with gears and racks converts simple moves of motors to actions of the card. Similarly, the outputs of audio subsystem are fading LED at the card's heart, bright LED at the card's eye, and melodies playing during moving actions. Buttons or switches are necessary to make an input to enter test state with manually testing functions of electric components. The whole system begins with inputs from power subsystem as well as instruction line to generate outputs that are inputs in sequence of software subsystem and hardware subsystem so that eventually the final outputs are card's movements with sound and LEDs at heart and head.

System Requirement Number	System Requirement	Sub-System Name	Sub-System Requirement Number	Sub-System Requirement
SYS_1	System will have a wriggle body allow walking forward	Material System	Sub 1.1	Body will be the material that have flexibility for wriggling also have endurance for constant movement (bend back and forth)
			Sub 1.2	Body should meet the requirements for the size
			Sub 1.3	Body should be tested for 5 minutes about mechanical simulation
SYS_2	System will have a walking ability and turns head left and right.	Movement System	Sub 2.1	Motors will be placed in main box for making walking motion (principle like conveyor belt)
			Sub 2.2	Motor will be placed in head for turning left and right
SYS_3	System will have a LED lighting system.	LED system	Sub 3.1	Lighting system at spear so LED will be placed at the edge of the spear
SYS_4	System will have head, body, two arms, two legs, and a spear. Main box to support whole body.	Structural System	Sub 4.1	Appropriate for each parts also meet the requirements (turn off)
			Sub 4.2	Appropriate for each parts also meet the requirements (turn on)
SYS_5	System's PCB include resistor, connector, LED that has wire to accept power source, and motor/servo	Hardware System	Sub 5.1	The components and wires are placed properly (not exceed temperature)
			Sub 5.2	The PCB is preperly fit with the requirement size
			Sub 5.3	The system have buttons can support each movements and sounds
SYS_6	System has a software make movement according to the several buttons	Software System	Sub 6.1	The software should support every movements properly
			Sub 6.2	Prepare the proper device for supporting software
SYS_7	System has a speaker to play different sounds corresponding head moving, opear moving, or repeat a welcome saying	Sound System	Sub 7.1	The system have a speaker

Integration

We need to assemble sub-systems so that it can work together correctly and smoothly. At first, we need to test each sub-system separately to confirm every system is good. This idea is so important because it help troubleshoot more easily in terms of each part, which is helpful to solve major issues of the whole systems.

At first, we can test power, buttons, and other mechanical mechanism so that we can implement motors, speaker, LEDs lights, PCB via breadboard. Then, we can attach parts together and connect with power so that it can run in test mode. Finally, we can integrate software in design to control the card.

Software High-Risk Parts: control the Arduino Uno or microcontroller is the most difficult one to drive the whole systems. After pass test mode, sub-systems exchange information and wait lines from servers. It means network connection management is required to be solid so that to make the system work efficiently.

Hardware High-Risk Parts: PCB is main risk of the hardware design because its quality decides how smooth the whole system works. Also, motors are necessary to be noticed since they are moving and may connect not well with gear. We may reduce the risks by regularly test and visually check connections.

Mechanical High-Risk Parts: our design has two step motors that go in vertical and horizon directions for spear. Also, we try to make the body move in wave shape. Those moves require strong base together with complex ramps. In limited rooms, it may affect designation and cause accidents. To deal with those risks, we need to manage good connections and make sure it is clear in moving directions.

Repository Management: We have discus and exchange information only via Discord and GitHub that includes our electronic system files (schematic, PCB, and Electric design). The GitHub also includes our software code. Our Team shares files in our central Microsoft Teams SharePoint folder

Configuration Controls: Our UI connect series of pushbuttons and power switch so that it can supply power and signals to integrated parts via PCB. Microprocessor is the most main part to drive our design PCB and other attached parts.

Test Plan (John)

Specify the requirements and Verification Groupings

1. Test 1

This will be an external inspection of the machine to verify the system meets requirements.

Procedure	System Requirement	Pass/Fail Criteria
Use measuring tape to find length, width, and height of machine with power turned off	Sys 4.1	Must be less than 35cm x 35cm x 35cm when off
Use measuring tape to find length, width, and height of machine when power is turned on at full extension.	Sys 4.2	1. Must be less than 75cm x 75cm x 75cm when on and fully extended 2. Ensure the range of movement of the spear vertically
Inspect the material that has flexibility for wriggling also have endurance for constant movement (bend back and forth)	Sys 1.1	Must have flexibility and endurance
Use measuring tape to find length, width, and height of material	Sys 1.2	Must be less than 20cm x 10cm x 1cm
Inspect the material for more than 3 minutes in simulation mechanical motion	Sys 1.3	No damage or any visible transformation in appearance

Table 1. Description and procedure of sub-system verification provided by Test 1

2. Test 2

This will be an internal inspection of the machine to verify system requirements.

Procedure	System Requirement	Pass/Fail Criteria
Inspect that there are no exposed conductors inside the machine and the connections are made using proper connectors	Sys 5.1	
Use measuring tape to find length, width, and height of PCB	Sys 5.2	Fits PCB assignment setting
Inspect the components also setting of PCB	Sys 5.3	4-40 machine screws, at least 0.125 holes in the PCB

Table2. Description and procedure of sub-system verification provided by Test 2.

3. Test 3

This will demonstrate the proper operation of the system.

Procedure	System Requirement	Pass/Fail Criteria
Use power switch to turn system on and machine functions.	Sys 3.1 / Sys 7.1	The LED will be activated with sound
Press “move” switch	Sys 2.1	Motor will be activated so that the robot starts to move walking
Press “turn” switch	Sys 2.2	The head will be turned left and right

Table 3. Description and procedure of sub-system verification provided by Test 3.

4. Test 4

This will be an inspection of the software processing to meet system requirements.

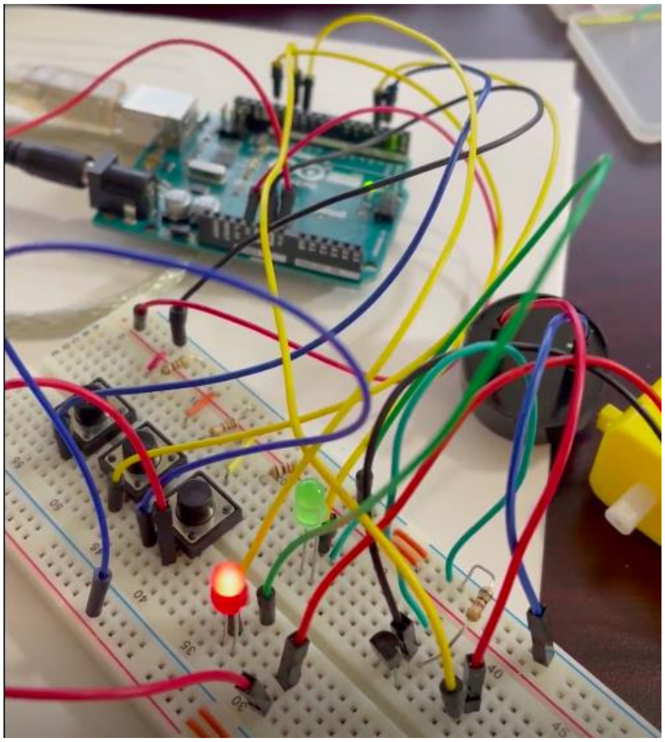


Figure 1. Test Setup

Procedure	System Requirement	Pass/Fail Criteria
Use power switch to turn system on and machine functions.	Sys 3.1 / Sys 7.1	The LED will be activated with sound
Press “move” switch	Sys 2.1	Motor will be activated so that the robot starts to move walking
Press “turn” switch	Sys 2.2	The head will be turned left and right

The software should support every movement properly	Sys 6.1	Each movement comes out following unput of each button
Prepare the proper device for supporting software	Sys 6.2	Use appropriate device

Table 4. Description and procedure of sub-system verification provided by Test 4.

5. Test 5

Test that all electrical connections defined are safely and properly connected with correct voltages and do not overheat.

Procedure	System Requirement	Pass/Fail Criteria
Use power switch to turn system on and machine functions.	Sys 3.1 / Sys 7.1	The LED will be activated with sound
Press “move” switch	Sys 2.1	Motor will be activated so that the robot starts to move walking
Press “turn” switch	Sys 2.2	The head will be turned left and right
Let the machine run for more than 60 seconds.	Sys 5.1 / Sys 5.3	Voltage readings are within .2V of expected value.
Use thermal imager to take temperature of components.	Sys 5.1	No component exceeds 80 degrees Celsius.

Table 5. Description and procedure of sub-system verification provided by Test 5.

Verification Type

	Requirement Number													
Test Number	Sys 1.1	Sys 1.2	Sys 1.3	Sys 2.1	Sys 2.2	Sys 3.1	Sys 4.1	Sys 4.2	Sys 5.1	Sys 5.2	Sys 5.3	Sys 6.1	Sys 6.2	Sys 7.1
1														
2				A										
3		I	I		I		I	I	I	I	I			
4						D						D	D	D
5	T													
6														

Verification Type	
Analysis	A
Inspection	I
Demonstration	D
Test	T

Integrated Master Schedule (Post CDR)

CDR													
Test													
Physical Structure													
Test structural integrity									Jeon				
Power System													
Test voltages of components									Will				
Test Arduino Power										Will			
Test PCB Power											Will		
Test Audio Power												Will	
Software													
Test Director Commands									Cong				
Prove commands are being Parsed										Cong			
Test movements											Cong		
Test Audio												Cong	
Test Lights												Cong	
PCB													
Test Audio on PCB									Jeon				
Test power										Jeon			
Test buttons and switch											Trevon		
Movement													
Test spear motors									Trevon	Trevon			
Test Card motors										Trevon			
Audio													
Test audio files											Jeon		
System Integration											Trevon	Will	
System Test											Cong	Jeon	
Final Inspection and Demonstration													

Bill of Materials

No	Name	Description	Manufatu	Part Number	Quantity	Price	Link	Order Status	Total Cost	Cost to Produce
1	5v DC Power Adapter	AC/DC WALL MOUNT AD	WSU050-4000-13		1	$15.71	WSU050-4	Special Order	$45.11	$124.42
2	Plastic Tracks	TRACK SET	2234-TRAKX40		2	$19.90	TRAKX40 C	Special Order		
3	Motor Driver	Motor Driver (Solder on)	ROB-14451		1	$9.50	SparkFun M	Special Order		
4	Arduino Uno	Sparkfun Arduino SMD R3	DEV-11021		1	*$26.97*	Amazon.co	ECE Parts Shop		
5	Wifi module	Sparkfun Wifi module	WRL-17146		1	*$5.50*	WiFi Modu	ECE Parts Shop		
6	Servo Motor	SERVOMOTOR RC 4.8-	SER0002		bulk pkg	*$9.95*	Servo - Ge	ECE Parts Shop		
7	Stepper Motor	Bipolar Stepper Motor Pe	108990003		2	*$17.50*	Stepper M	ECE Parts Shop		
8	LED Lights		COM-11448		10	*$5.50*	LED - 3mm	ECE Parts Shop		
9	Speaker	Through-Hole Speaker	PRT-20660		1	*$2.95*	Through-H	ECE Parts Shop		
10	Gears	PLASTIC PINION GEAR SE	LS-00021		58	*$6.29*	ECE Parts S	ECE Parts Shop		
11	USB Female Connector				1	*$2.28*	ECE Parts S	ECE Parts Shop		
12	USB Male Connector				1	*$2.37*	ECE Parts S	ECE Parts Shop		