Trail Mix Dispenser – Design Document

Team 67
Mathew Jacob
Andrew Ma
Kanav Kariya
TA: Dongwei Shi

February 2018

Contents

1 Introduction

1.1 Purpose

In the modern world, an increasing number of people are living under high-stress conditions and their time is becoming more and more valuable. A common example of this is the college student who lives a high-stress lifestyle as a result of constant coursework, extracurricular activities, and personal endeavors. High-stress lifestyles leave little to no time for thinking about, preparing, and maintaining healthy dietary habits. For example, almost half of all college students in a study done in Gwangju, South Korea ate out at least once a day[6]. In addition, high levels of stress have been linked to increased eating of sweet fatty foods[5]. These two facts, in conjunction, would indicate that high-stress lifestyles lead to unhealthy dietary habits. Diet has a demonstrable impact on people's physical and cognitive function. In the afternoon, a calorie rich snack such as trail mix could provide a much needed boost to a student or young professional[3].

A common problem with home preparation is that it is difficult to accurately measure the amount of calories in a meal without a strong understanding of the ingredients and their masses. Many people that are invested in living healthy lifestyles own food scales now and it has been proven that measuring food before consumption has a positive impact on the outcome of diet plans[4]. Our position is that providing these people with low-effort, healthier, and customizable food options from their home would increase their quality of life. Our conclusion is that a well rounded trail mix with nuts[7] can be healthier and simpler to prepare than the majority of commercially available snacks.

Our goal is to create a trail mix dispenser that will allow the user to remotely create a quick and healthy snack to bring along wherever they go. Rather than taking time out of the user's day to think about and prepare a snack, they will be able to simply indicate the contents of their customizable trail mix from their phone. The trail mix dispenser, using Bluetooth, will receive this information and hastily and accurately dispense the desired contents, mix them together, and dispense them into the user's container ready for consumption.

1.2 Background

Our research found that there are a few options for automatic snack dispensing[1]. However, these dispensers do not measure the amount of food dispensed. There also exist many different forms of manual dry food dispensers, such as the cereal dispensers commonly seen in cafeterias, however none of these are automatically operated nor do they have an attached food scale. This means that health conscious people and want to count calories have to manually prepare and measure their food.

The importance of having healthy on-the-go options, as stated in the purpose, is growing from a convenience to a necessity in busy modern lives. The goal of our project is to automatically measure, prepare, and dispense the food for the user in their own home using their own ingredients.

This allows the user to regain control over their diet without losing more time.

1.3 High-level Requirements

- The dispenser must be able to dispense trail mix autonomously once request has been issued from the app
- The dispenser must dispense within ± 20g per ingredient
- The dispenser must be able to dispense within 5 minutes from start to finish

2 Design

2.1 Block Diagram

The entire design requires four functional units that are all interdependent for proper function: the power supply unit, the control unit unit, the actuation unit, and the wireless unit. The power supply unit's purpose is to take 120VAC 60Hz power from a standard wall outlet and convert it to manageable 12V DC, 5V DC and 3.3V DC to power the other modules. The wireless module will include an application that will communicate with the Bluetooth module which will in turn communicate with microcontroller using UART. The control unit module will control stepper motors and LEDs based on the current state, instructions from the Bluetooth module and signals from the weight sensor and limits switches.

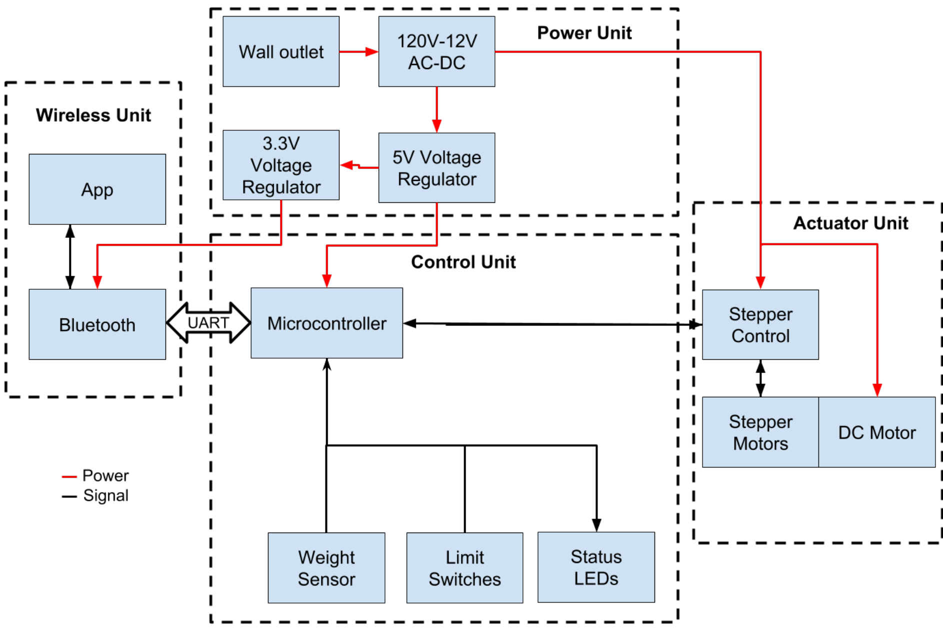

Figure 1: Block diagram of the Trail Mix Dispenser.

2.2 Points Summary

Module Name	High Level Requirement	Points
Wireless Module	Dispenser must be able to receive the request accurately from the mobile application.	5
Control Module 1	The dispenser must dispense within 20g per ingredient.	15
Control Module 2	Dispensing time must be under 5 minutes from start to finish	10
Power Module	The voltages and currents are stepped down/up appropriately to power all other modules and facilitate intermodule communication	5
Actuator Module	All 5 steppers and the DC motor operate successfully in conjunction	15

Table 1: Points Summary

2.3 Physical Design

Our physical design consists of three separate processes: dispensing ingredients from the four containers, mixing them in a bowl, and serving them into a container of the users choice. All three of these processes will be carried out with DC and stepper motors.

Figure 2: Cereal Dispensers from Amazon

First, the dispensing will be performed by refitting the containers shown in figure 2, replacing the knobs with stepper motors. Once attached and coupled to the rotating mechanisms, these stepper motors will be able to steadily dispense the ingredients from the containers. The appropriate weight of each ingredient will be dispensed, one at a time, through a funnel into the weighing bowl, shown in figures 3 and 4.

Once all the ingredients have been dispensed into the bowl, a motor within the bowl will mix the ingredients. The bowl will then be tilted, pouring the trail mix into the user's container.

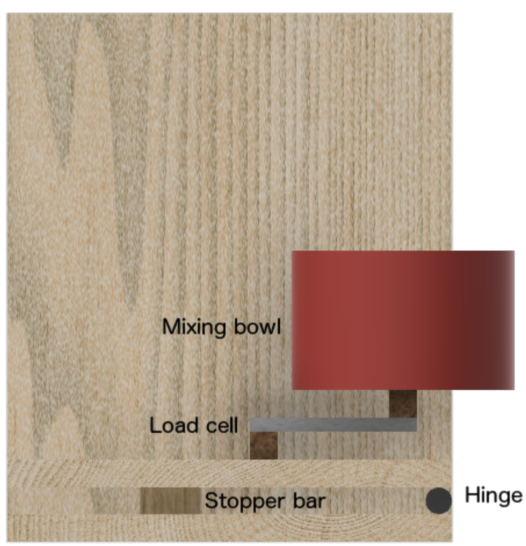

Figure 3: CAD Drawing of Mixing Platform – Front

Figure 4: CAD Drawing of Mixing Platform – Perspective

2.4 Block Design

2.4.1 Power Unit

2.4.1.1 120V-12V AC/DC Converter We use the 120V-12V AC/DC Converter to step the voltage down from the 120VAC power from the wall outlet to the 12V DC voltage that the stepper motors and the DC motor require. It is important that this component be able to handle a high enough current to drive all of the motors in our system. The converter takes the form of a barrel jack. We selected the 96PSA-A84W12V1, because it is a switched power supply which can give us up to 7A of current, efficiently giving us sufficient current to power the dispenser (see table 3). Overall, the circuit should consume at most about 56.4W of power (see equation 1).

$$
\begin{aligned}
P &= IV \\
P &= 12\text{V} \cdot 4.70\text{A} = 56.4\text{W}
\end{aligned}
\tag{1}
$$

Requirements	Verification
$V_{out} = 12\text{V} \pm 0.2\text{V}$	1. Plug the AC/DC Converter into the wall outlet. 2. Measure the voltage between V_{out} and GND and ensure it falls within 11.8V and 12.2V.

Table 2: Requirements and Verification for 120V-12V AC/DC Converter

Component	Maximum Current Draw	Quantity	Total Current Draw
Steppers	0.70	5	3.50
DC Motor	0.68	1	0.68
5V regulator	0.52	1	0.52
Total			**4.70**

Table 3: Summary of current drawn by devices operating at 12V.

2.4.1.2 5V Voltage Regulator The 5V voltage regulator is used to step the voltage down from 12VDC to 5VDC to power the microcontroller. The 12VDC will be sourced from the 120V-12V AC/DC Converter. We chose the TS30011-M050QFNR 5V 1A buck switching regulator because it can supply up to 1A of current which is much more than we require (see table 5). We chose a buck switching regulator since it would reduce the power wasted in converting from 12v to 5v. Note that for table 3, we treated the 5V regulator as a linear regulator to overestimate the current.

Requirements	Verification
V_{out} = 5V±0.2V	1. Power the 5VDC Voltage Regulator using 12VDC from the barrel jack on pin 3 in Figure 5. 2. Measure the voltage between V_{out} and GND and ensure it falls within 4.8V and 5.2V.

Table 4: Requirements and Verification for the 5V Voltage Regulator.

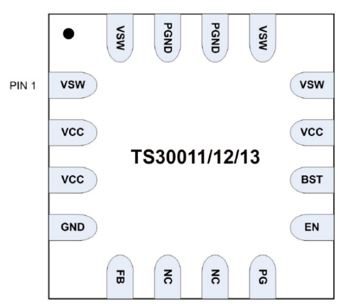

Figure 5: Pinout diagram of the 5V Voltage Regulator.

Component	Maximum Current Draw	Quantity	Total Current Draw
Atmega328P	0.20	1	0.20
Decoder	0.02	1	0.02
Bluetooth	0.20 (estimated)	1	0.20
3.3V Regulator	0.10	1	0.10
Total			**0.52**

Table 5: Summary of current drawn by devices operating at 5V.

2.4.1.3 3.3V Voltage Regulator The purpose of the 3.3V Voltage Regulator is to step the voltage down for use with the Bluetooth module. The input for the 3.3V Voltage Regulator will be 5VDC provided from the output of the 5V Voltage Regulator. We chose the PAM2305AAB330 3.3V 1A buck regulator because it can supply much more than the required current (see table 6). The buck converter although more efficient than a linear regulator happens to be less expensive so we opted to use it. Note that our calculations for table 5 overestimate the input current as if we had chosen a linear regulator.

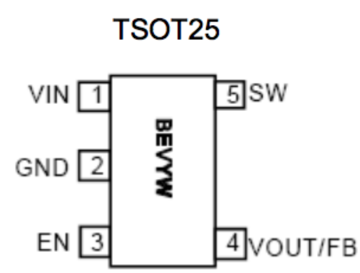

Figure 6: Pinout diagram of the 3.3V Voltage Regulator.

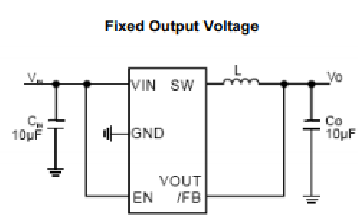

Figure 7: Fixed output voltage configuration for the 3.3V Voltage Regulator.

Component	Maximum Current Draw	Quantity	Total Current Draw
Level Shifter	0.10	1	0.10
Total			**0.10**

Table 6: Summary of current drawn by devices operating at 3.3V.

Requirements	Verification
V_{out} = 3.3V±0.2V	1. Power the 3.3VDC Voltage Regulator using 5VDC from the lab kit and complete the circuit as shown in Figure 7 2. Measure the voltage between V_{out} and GND and ensure it falls within 3.1V and 3.5V.

Table 7: Requirements and Verification for the 3.3V Voltage Regulator.

2.4.2 Control Unit

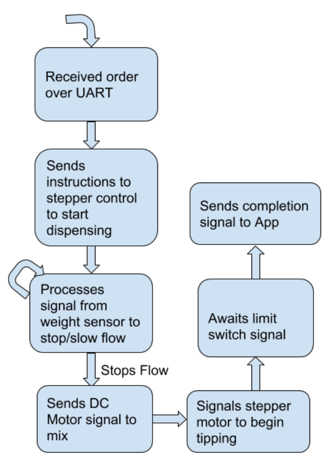

Figure 8: Control Unit flowchart.

2.4.2.1 Microcontroller The microcontroller functions as the central hub for the processing hub of the operation. It is powered by 5VDC provided by the 5V Voltage Regulator. It receives data using UART from the Bluetooth module to begin the process of dispensing the trail mix. It then sends signals to the stepper control based on the information provided from the Bluetooth module and the weight sensor and limit switches, operating in a closed feedback loop. We selected the Atmega328P chip because it has 23 GPIO inputs and also has 10-bit resolution analog to digital conversion. This is sufficient to interact with all of our circuits and, in addition, we can use Arduino bootloader and libraries, making our design easier to implement.

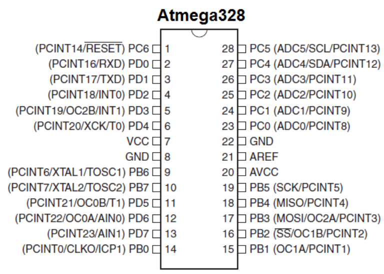

Figure 9: Pinout diagram of the microcontroller.

Requirements	**Verification**
$V_{in} \leq 1$V must be read as a digital 0.	1. Power microcontroller with 5VDC from the power supply in the lab kit. 2. Attach all digital input pins to 1VDC by creating a simple voltage divider circuit from the 5VDC. 3. Upload code setting all digital pins to INPUT and print the values via Serial. 4. Ensure all values are 0.
$V_{in} \geq 3.5$V must be read as a digital 1.	1. Power microcontroller with 5VDC from the power supply in the lab kit. 2. Attach all digital input pins to 3.5VDC by creating a simple voltage divider circuit from the 5VDC. 3. Upload code setting all digital pins to INPUT and print the values via Serial. 4. Ensure all values are 1.
Digital output 0 must correspond to $V_{out} \leq 1$V	1. Power microcontroller with 5VDC from the power supply in the lab kit. 2. Upload code setting all digital pins to OUTPUT LOW. 3. Probe each pin using a multimeter ensuring $V_{out} \leq 1$V for all pins.

Table 8: Requirements and verification for the microcontroller module.

Requirements	Verification
Digital output 1 must correspond to $V_{out} \geq$ 3.5V	1. Power microcontroller with 5VDC from the power supply in the lab kit. 2. Upload code setting all digital pins to OUTPUT HIGH. 3. Probe each pin using a multimeter ensuring $V_{out} \geq$ 3.5V for all pins.
ADC in analog in must have at least 8-bit resolution at 5V.	1. Power microcontroller using 5VDC from the lab kit. 2. Connect the analog input pin to a DC power supply and turn it on. 3. Increment the voltage from the DC power supply by 0.02V. 4. Upload code to print the read value from the analog input pin via Serial. 5. Ensure that the read value incremented by 1. 6. Repeat steps 3-5 until the voltage is at 5V.

Table 9: Requirements and verification for the microcontroller module continued.

2.4.2.2 Weight Sensor This module consists of two components, a load sensor and an amplifier. The load sensor is a 4-wire sensing device that has a voltage output that varies linearly with the mass placed on the sensor. This device will be powered with 12VDC from the 12V voltage regulator and the measurement leads will be connected to the amplifier. An example of 4-wire sensing with an amplifier can be seen in Figure 11. The amplifier will amplify the voltage across the sensor ($V_{\text{in}}^{+} - V_{\text{in}}^{-}$) based on the gain given by equation 2. We selected the 114990093 load cell from Seeed Studio because it allows us to measure masses between 0 and 3kg within ± 0.9g. We selected the INA125 instrumentation amplifier since it will allow us amplify the signal from our load cell precisely, reducing noise.

$$G = 4 + \frac{60\text{k}\Omega}{\text{R}_\text{G}} \tag{2}$$

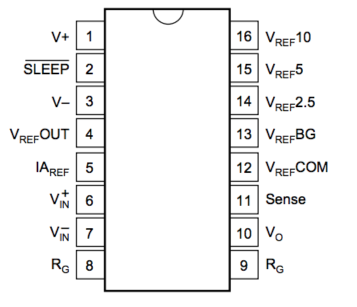

Figure 10: Pinout for the weight sensor amplifier

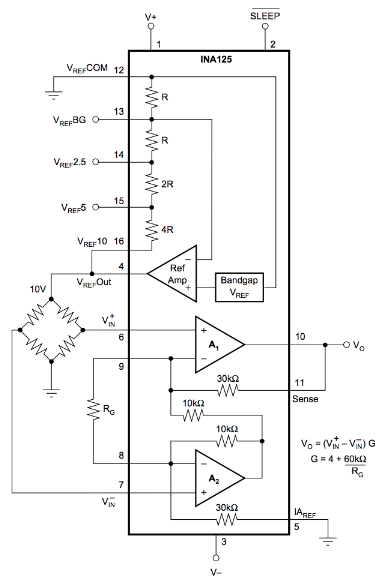

Figure 11: Schematic for 4-terminal sensing with the INA125 amplifier.

Requirements	Verification
Must have less than ±5g combined error	1. Power the load bar using the 12VDC from the lab kit. 2. Attach the measurement leads to an oscilloscope. 3. Mount the load bar such that weights can be placed on the non-mounting end. 4. Place a 250g mass on the load bar. 5. Measure and record the voltage on the oscilloscope. 6. Replace the 250g mass with a 500g mass. 7. Measure and record the voltage on the oscilloscope. 8. Replace the 500g mass with a 750g mass. 9. Measure and record the voltage on the oscilloscope. 10. Ensure that the average of the voltage read for the 750g and 250g masses is within ±1% of the 500g masses read voltage.
Must be able to support a maximum load of 1kg	1. Power the load bar using the 12VDC from the lab kit. 2. Attach the measurement leads to an oscilloscope. 3. Mount the load bar such that weights can be placed on the non-mounting end. 4. Measure and record the voltage on the oscilloscope. 4. Place a 500g mass on the load bar. 5. Measure and record the voltage on the oscilloscope. 6. Place another 500g mass on the load bar. 7. Measure and ensure that the read voltage is twice the difference in voltage of the 500g mass and the original taring reading.

Table 10: Requirements and verification for the load sensor.

2.4.2.3 Limit Switches The limit switch is primarily used for the stepper motor that tips the bowl. Once it reached the desired angle, the switch is hit and this information is passed on to the microcontroller to stop the bowl from tipping further. The switches will have to be debounced to prevent multiple fluctuations when they are triggered.

Requirement	**Verification**
Limit switch is properly debounced	1. Connect 5V to the first terminal and GND to the third terminal 2. Move the switch so that the common terminal is connected to GND 3. Probe the common terminal and GND using an oscilloscope with single shot capture 4. Set a trigger for 2.5V 5. Close the switch 6. Ensure that the transition was properly debounced (i.e. only one transition from low to high)

Table 11: Requirements and verification for limit switches

2.4.2.4 Status LEDs There will be two status LEDs will indicate whether the design is powered on and whether it is processing, respectively. The power status LED will be attached to the 5V voltage regulator. The processing status LED will be powered and controlled by the microcontroller.

Requirement	**Verification**
LED illuminates at 15mA	1. Connect an LED, a 330 Ω resistor and a 5VDC power supply from the lab kit in series. 2. Turn on the power supply. 3. Verify that the LED illuminates.

Table 12: Requirements and verification for status LEDs.

2.4.3 Actuator Unit

2.4.3.1 Stepper Control The stepper control is designed by us and contains five driver modules, one for each stepper motor. This module is important because it allows us to separately control all five stepper motors using fewer GPIO pins on our microcontroller. The driver modules are based on the SparkFun Easy Driver and can be seen in Figure 12. The driver that we created takes 5VDC as a reference level for the digital logic signals STEP and DIR. A rising edge on STEP will create output signals from OUT1A, OUT1B, OUT2A, and OUT2B that correspond to a single full step. DIR determines the direction of the step. The STEP signal is sent to the proper motor driver using a 3:8 decoder, meaning that it only takes five GPIO pins (STEP, DIR, SELECT0, SELECT1, SELECT2) to control five motors. The overall schematic with connections to the microcontroller can be seen in Figure 13. We chose to use 5 of the A3967SLBTR stepper driver because it is the IC used in the SparkFun EasyDriver and is relatively easy to use. It only requires STEP and DIR inputs from the MCU to control motor movement.

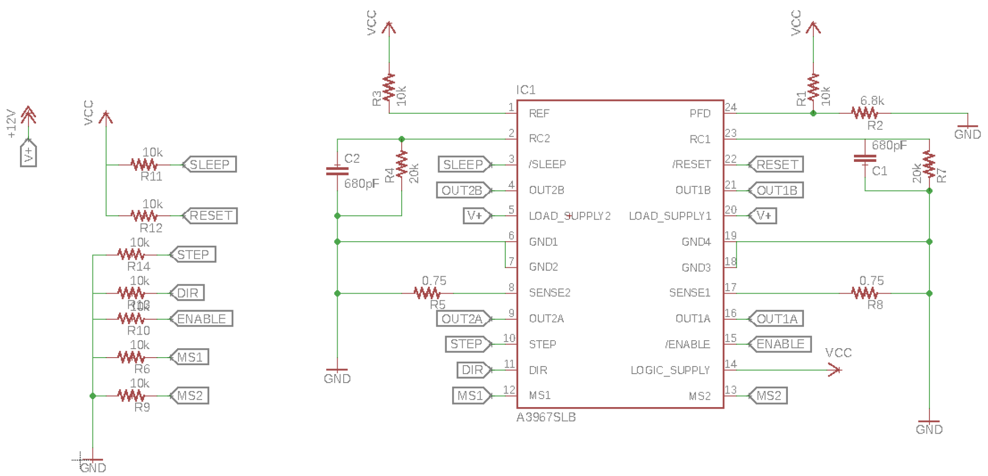

Figure 12: Schematic for our stepper motor driver based on the SparkFun EasyDriver.

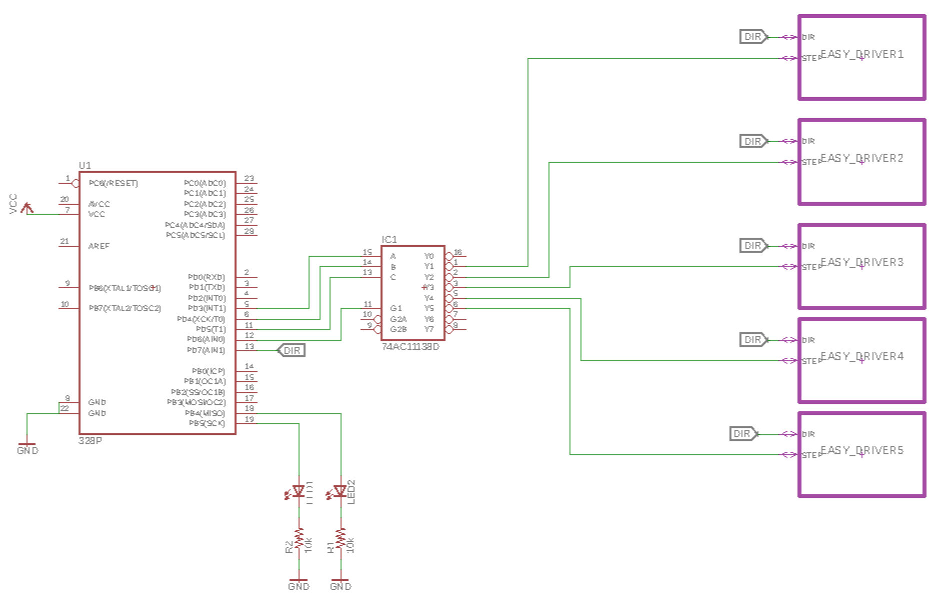

Figure 13: Schematic for our stepper control connected to the MCU.

Requirements	Verification
Must be able to separately control 5 stepper motors	1. Power the stepper motor control with 12VDC and 5VDC from the lab kit and connect the output leads to the motors. 2. Provide the select bits for one of the motors with 5VDC and GND. 3. Give the equivalent of a single rotation in the form of a 5V square wave with a frequency of 100Hz to the STEP input. 4. Ensure the motor completed a single rotation. 5. Repeat steps 2-4 for the remaining motors.
Must be able to step motors at a rate equivalent to $\geq$10rpm for 60 seconds	1. Power the stepper motor control with 12VDC and 5VDC from the lab kit and connect the output leads to the motors. 2. Mark a point on each of the motor shafts. 3. Record the motors. 4. Provide the select bits for one of the motors with 5VDC and GND. 5. Give the control a 5V square wave with a frequency equivalent to 10rpm calculated from Equation 3 for 60 seconds. 6. Repeat steps 4-5 for the remaining motors. 7. Watch the recorded camera footage and ensure all motors completed 10 rotations.

Table 13: Requirements and Verification for the Stepper Control.

2.4.3.2 Stepper Motors In our design there will be five stepper motors in total. Four of these motors will be used to dispense the ingredients from the containers and one will be used to control a pulley that will tip the mixing bowl over into the final container. The stepper motors in the containers will have to produce a minimum torque at a certain speed in order to dispense the ingredients. The stepper motor used to tip the bowl will have to produce a minimum torque as well as rotate in both directions to tilt the bowl up and back down. We will be using Adafruit's model 324 12V, 200 step bipolar stepper motors since they will allow us to have fine control over movement in both directions with sufficient torque at a maximum current of 700mA.

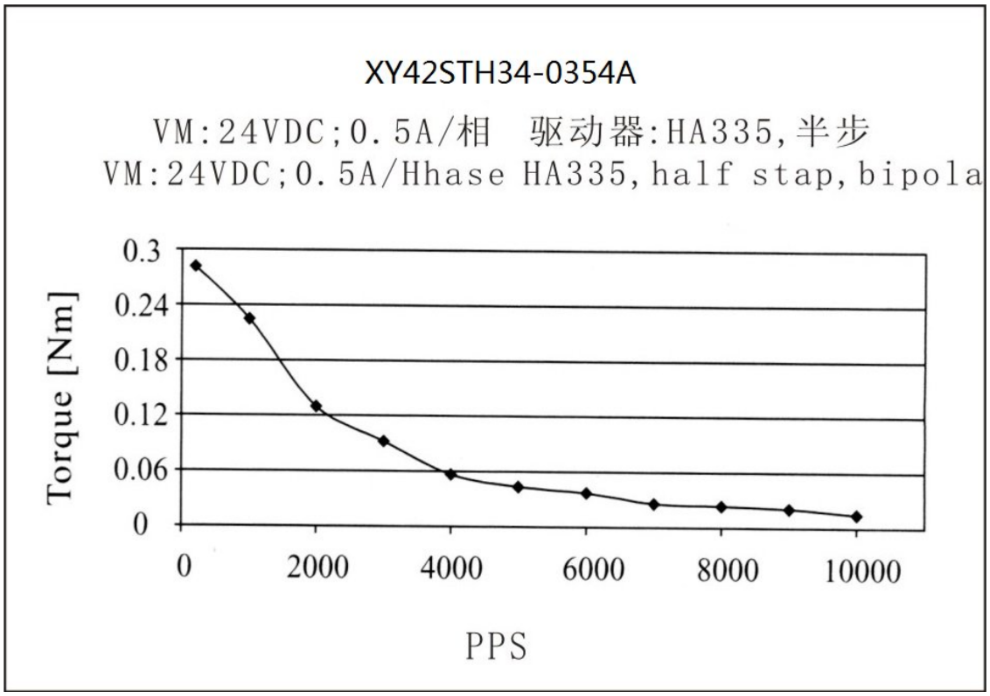

Figure 14: Torque vs PPS (pulses per second) for the Stepper Motors

$$PPS = N\frac{\text{Rotations}}{\text{Minute}} * 200\frac{\text{Steps}}{\text{Rotation}} * \frac{1}{60}\frac{\text{Minute}}{\text{Second}} \tag{3}$$

Requirements	**Verification**
Draws $\leq$ 1A when motor static.	1. Power the stepper motor control with 12VDC and 5VDC from the lab kit and connect the output leads to the motors. 2. Using a multimeter, measure the current drawn to power the motor and ensure it is $\leq$ 1A.
Must have a torque greater than 16oz-in at an angular speed of 10rpm.	1. Calculate the number of pulses per second (PPS) required to achieve 10rpm using Equation 3. 2. Compare with the provided Torque vs PPS chart on the datasheet or in Figure 14 and confirm that the torque is greater than 16oz-in.

Table 14: Requirements and Verification for Stepper Motors

2.4.3.3 DC Motor The DC motor is used to mix the dispensed ingredients into a more homogeneous mix such that it more closely resembles commercial trail mix. It will be powered with 12VDC from the 12V voltage regulator and controlled by the microcontroller. Our DC motor will be controlled using a transistor and pulse width modulation. An example provided by Adafruit can be seen in Figure 15. We will be using the FIT0492-A since it will provide us with 124 oz-in of torque with a maximum current of 0.68A.

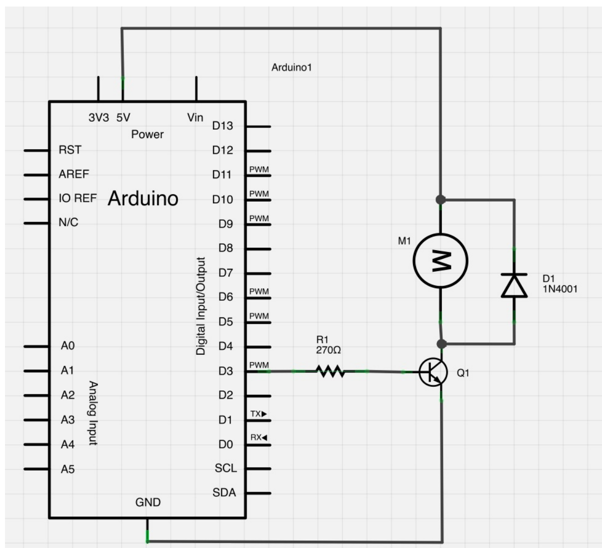

Figure 15: Example control circuit for DC motor by Adafruit.

Requirements	Verification
Must be able to spin with 1kg of opposing force.	1. Couple the DC motor to the fins that are shown in Figure **??**. 2. Connect the DC motor as shown in Figure 15 replacing the Arduino with a function generator. 3. Place the 1kg mass such that it is opposing the rotation of the fin. 4. Set the function generator to a 5V 400Hz square wave with a 50% duty cycle. 5. Ensure the mass moves. If it does not, slowly increase the duty cycle until it does.

Table 15: Requirements and verification for the DC motor.

2.4.4 Wireless Unit

2.4.4.1 Bluetooth The Bluetooth module relays Bluetooth information over UART to the microcontroller. It receives instructions from the Android application. We chose the HC-05 Bluetooth module since it will allow us to communicate with it via UART and already has readily available libraries. The controller takes a higher input voltage (i.e. 5V) and regulates it down to 3.3V and thus operates at 3.3V for logic, requiring level shifters.

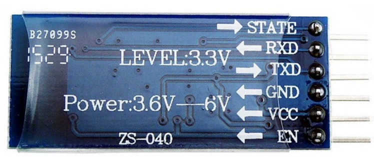

Figure 16: Pinout of Bluetooth Module

Requirement	Verification
Emulates a UART connection	1. Connect BT module TX/RX pins to level shifters 1A pins 2. Set DIR = 0V on the TX shifter and DIR=3.3V on the RX 3. Connect level shifters 1B pins to Arduino GPIO pins 4. Connect power pins on the BT module to 5V and GND 5. Power on Arduino and program with a sketch that relays input from the HC-05 over USB serial 6. Connect to HC-05 module from a unix computer over BT 7. Open a serial session on the computer and type "Hello World!" 8. Verify that the message was received by the Arduino
Can be used from up to 3 meters away	1. Connect BT module TX/RX pins to level shifters 1A pins 2. Set DIR = 0V on the TX shifter and DIR=3.3V on the RX 3. Connect level shifters 1B pins to Arduino GPIO pins 4. Connect power pins on the BT module to 5V and GND 5. Power on Arduino and program with a sketch that relays input from the HC-05 over USB serial 6. Move computer 3m away from BT module 7. Connect to HC-05 module from a unix computer over BT 8. Open a serial session on the computer and type "Hello World!" 9. Verify that the message was received by the Arduino

Table 16: Requirements and Verification for Bluetooth

2.4.4.2 App The Android application will serve as the users control unit. The user will be able to pick from some popular pre-loaded recipes and design their own custom trail mix. The app can eventually be extended to provide the user with usage and nutrition statistics as well as become a platform for sharing recipes.

Requirements	Verification
Communicates accurately with bluetooth module at 3m distance	1. Use a measuring tape to measure a 3m distance 2. Establish a bluetooth connection between your mobile phone and the module 3. Open the app and transmit orders 4. Ensure that the right orders are received
Supports trail mix recipes with custom weights	1. Open the app and use the new recipe option 2. Input custom weights and click enter 3. Make sure the right recipe is transmitted to the module

Table 17: Requirements and Verification for App

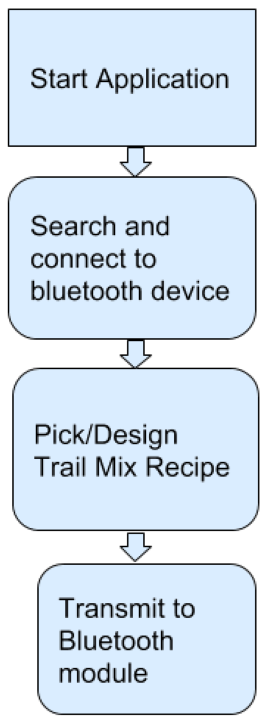

Figure 17: Flowchart for the Android application.

2.4.4.3 Level Shifter This module will allow the 3.3V Bluetooth module to communicate with the 5V powered microcontroller by shifting the voltage. This is essential to successful communication using UART between the two devices. It will require power from both 5VDC and 3.3VDC, which will be provided by the appropriate regulators. We will be using two of these, because despite being bidirectional they can only perform a single direction of communication at a time. We chose to use two 74LVCH2T45DC modules since they are compatible with 5v and 3.3v and will allow us to facilitate simultaneous bidirectional communication.

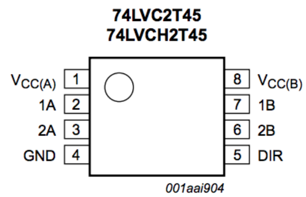

Figure 18: Pinout of the 5V to 3.3V Level Shifter.

Requirements	Verification
Shifts digital 1/0 reference voltage from 5V±0.2V to 3.3V±0.2V	1. Supply 5VDC to VCCA from lab kit power supply. 2. Supply 3.3V to VCCB from DC power supply. 3. Connect 5VDC to DIR 4. Set 1A = 0 by connecting to GND. 5. Set 2A = 1 by connecting to 5VDC. 6. Verify that 1B = 0 using DMM with V=0V±0.2V. 7. Verify that 2B = 1 using DMM with V=3.3V±0.2V.
Shifts digital 1/0 reference voltage from 3.3V±0.2V to 5V±0.2V	1. Supply 5VDC to VCCA from lab kit power supply. 2. Supply 3.3V to VCCB from DC power supply. 3. Connect GND to DIR 4. Set 1A = 0 by connecting to GND. 5. Set 2A = 1 by connecting to 3.3VDC. 6. Verify that 1B = 0 using DMM with V=0V±0.2V. 7. Verify that 2B = 1 using DMM with V=5V±0.2V.

Table 18: Requirements and verification for level shifter.

2.5 Tolerance Analysis

A large part of our project is ensuring we have an accurate measurement of the mass of trail mix already dispensed. In our high-level requirements we listed a tolerance range of 20g for each individual ingredient. This tolerance was determined based on the precision of the stepper motors, the precision of the weight sensor, and the dimensions of the fins in our dispenser. In this tolerance analysis we will focus on the precision of our weight sensor. The accuracy of the load cell within our weight sensor is given as a percentage of the maximum rated output. This was given as 0.03% with a maximum rated output 2.0mV/V. We will be using an excitation voltage of 12VDC for our load cell using a 12V voltage regulator. Therefore the maximum rated output is

$$2.0\text{mV/V} * 12\text{V} = 24\text{mV}$$

We will be using an amplifier to set the maximum voltage to 5V, approximately 208.3. The load cell has a maximum load of 3kg and the rated output scales linearly, with 0kg corresponding to an output of 0V and 3kg corresponding to 5V. Our microprocessor has a 8-bit ADC with a maximum voltage of 5V. Using this we can find calculate the bit precision of the load cell without error as

$$\frac{3\text{kg}}{2^8\text{bits}} = 11.72\text{g/bit}$$

From this point on in the analysis we are focused on the combined error of our load cell. The combined error is interpretted as percentage deviation of the measured voltage for a given mass. This means that the maximum error after amplification is

$$24.0\text{mV} * 208.3 * \frac{0.03}{100} = 1.50\text{mV}$$

which in terms of bit-error is

$$1.50\text{mV} * \frac{2^8\text{bits}}{5\text{V}} = 0.077\text{bits}$$

Because this value is less than a bit, we can conclude that the error will not appear in after the ADC. However, the resolution of our device falls at 11.72g/bit. In terms of our project, this means a potential error of 11.72 grams per ingredient.

Increasing the bits of our ADC will increase the precision, however it will also increase the influence of the combined error. Redoing all the calculations above using a 10-bit ADC, which is available in the microcontroller we chose, we can find that the precision is narrowed to 2.93 g/bit and the error is 0.308 bits, which means that it would still not appear after our ADC.

We found this analysis very insightful as we know now that we can safely increase the bits used in our ADC from 8 to 10 without introducing the error from the load cell and amplifier.

3 Cost and Schedule

3.1 Cost Analysis

3.1.1 Labor

Average hourly rate: \$40
Number of hours to complete: 60
Cost per partner = \$6000
Total labor cost = \$18000

3.1.2 Parts

Name	\$/unit	Quantity	Total cost
Load Cell	\$10.15	1	\$10.15
Load Cell Amplifer	\$6.89	1	\$6.89
Stepper	\$14.00	5	\$70.00
DC Motor	\$11.90	1	\$11.90
Bipolar Stepper Driver	\$3.50	5	\$17.50
Level shifter	\$0.50	2	\$1.00
3x8 Decoder	\$0.41	1	\$0.41
5V regulator	\$1.08	1	\$1.08
12V 7A Power supply	\$30.88	1	\$30.88
Atmega328	\$2.01	1	\$2.01
Bluetooth Module	\$7.69	1	\$7.69
3.3V regulator	\$0.58	1	\$0.58
3-pack cereal dispensers	\$27.99	1	\$27.99
1-pack cereal dispenser	\$18.64	1	\$18.64
Total			\$206.72

Table 19: Parts Cost

3.1.3 Grand Total

Item	Cost (\$)
Labor	18000
Parts	201.74
Grand Total	18201.74

Table 20: Total Cost

3.2 Schedule

Week	Mathew	Andrew	Kanav
2/26/18	Order parts Talk to Machine Shop		
3/5/18	Implement changes from Design Review		
3/12/18	Implement power circuit	Implement power circuit	Test power circuit
3/19/18	SPRING BREAK		
3/26/18	Test MCU	Test weight sensor Test LEDs + Limit switches	Start App Development
4/2/18	Implement closed loop feedback	Test motors	Implement closed loop feedback
4/9/18	Implement prototype	Implement prototype	Test wireless unit Implement prototype
4/16/18	Implement feedback from mock demo		
4/23/18	Prepare presentation Help with paper	Prepare final paper	Prepare presentation Help with paper
4/30/18	Present!		

Table 21: Weekly Work Schedule

4 Discussion of Ethics and Safety

One of the most important safety concerns in our project is that we will be working with 120VAC wall outlet power and stepping it down to a lower DC voltage that we will then regulate. This 120VAC line will be very dangerous when live and as such we will abide by the lab safety standards. This means that first and foremost we will routinely check for both damaged or frayed equipment. We will also be sure not to bring food or drink into a lab setting to avoid damaging any equipment. Also, we will be sure to practice caution around the system when it is live, such as practicing the one-hand method. Also, given that none of us have significant experience with working with high voltage converters and regulators we will follow Section 6 of the IEEE Code of Ethics[2] and use off-the-shelf components.

As a result of our project being used for measured food preparation, it is important to disclose any inaccuracies in our design. In accordance to Section 1 and Section 3 of the IEEE Code of Ethics we will inform the user of any errors in the measurement of food. We will not mislead or misinform the user as to the accuracy of our design so that they can make informed dietary decisions.

In conclusion, we are responsible for all the design decisions, which means it is also our responsibility to disclose any issues that may endanger the user as stated in Section 1 of the IEEE Code of Ethics [2]. We believe that if designed properly we will be able to minimize and most likely completely mitigate these issues in our design.

References

[1] In: 22 (Feb. 2018), p. 2018. URL: `https://smile.amazon.com/SnackMan-Motion-Activated-Treat-Candy-Dispenser/dp/B008Z8VXPK?sa-no-redirect=1`.

[2] "Code of Ethics". In: *IEEE* (Feb. 2018). URL: `https://www.ieee.org/about/corporate/governance/p7-8.html`.

[3] R. Kanarek and D. Swinney. "Effects of food snacks on cognitive performance in male college students. Appetite". In: 14.1 (1990), pp. 15–27.

[4] L. Lipsky et al. "Power of Food Scale in association with weight outcomes and dieting in a nationally representative cohort of U.S. young adults." In: *Appetite* 105 (2016), pp. 385–391.

[5] G. Oliver, J. Wardle, and E. Gibson. "Stress and Food Choice: A Laboratory Study." In: *Psychosomatic Medicine* 62.6 (2000), pp. 853–865.

[6] H. Ro. "Factors in Food Selection and Eating Out Behavior of College Students." In: *Korean Society of Food Culture* 22 (Feb. 2018), pp. 241–249. URL: `http://www.koreascience.or.kr/article/ArticleFullRecord.jsp?cn=SSMHB4_1999_v14n3_241`.

[7] E. Ros. "Health Benefits of Nut Consumption. Nutrients". In: 2.7 (2010), pp. 652–682.