Automated Boba Station

Team 49 - Hunter Huynh, Timothy Ko, Jordan Wu
ECE 445 Project Proposal - Spring 2019
TA: John Kan

Contents

1 Introduction

1.1 Objective

Boba, a popular drink among millenials, has prices that are still largely dictated by the manual labor involved in making it, so shops still require many employees. Unlike coffee, making boba tea requires handling both solids (boba, etc), and liquids (tea, syrup, milk) [1]. With the large variety of recipes, human workers are prone to make mistakes. Finally, taste consistency is hard to achieve without an automated solution, leaving drinks sometimes oversweet.

Our goal is to develop an automated boba station would have multiple dispensers connected to a U-shaped gutter that would all drip into a cup. We would have multiple dispensers for cold liquids and solids for toppings, simplifying the entire process of creating a boba drink, which will improve the efficiency of boba stores. In addition, because boba is currently made manually, you most likely either make too much or make too little for the cup size. This product will dispense predetermined amounts for each liquid/solid, customizable by a web user interface, reducing food waste and money.

1.2 Background

The creation of a boba drink comes in a multitude of configurations. However, it generally consists of just solids and liquids. In general, the process of creating a boba drink consists of putting the toppings (solids) specified by the order in the cup, pouring the tea and milk (milk tea may already be mixed together with a specific concentration of milk and tea), then adding the user specified amount of sugar and ice (such as 0%, 25%, 50%, 75%, or 100% of the normal amount) [1] [5]. For this project, we will just focus on the basic drink, milk tea with boba, which consists of cold brewed tea (black tea or green tea), milk (almond milk, regular milk, etc), sugar syrup, and tapioca pearls (boba).

Because the creation of boba drinks is currently all manual and based off of percentages, there is no unified concentration of each component of the drink. For example, 25% sugar level varies employee to employee and drink to drink because it is done manually. The concentration of milk and tea varies as well as the amount of toppings is in a drink. If done automatically, the concentrations will be unified per configuration. Eventually, we’d like this solution to allow for more permutations of drinks by adding in more dispensers with different ingredients and connecting it to this system.

1.3 High-level Requirements

- Have at least two dispensers, one for tapioca pearls and one for the milk tea.
- The station must be able to dispense a user-specified amount of tea, sugar syrup, milk, and boba with no more than ±10% error in mass.
- Have a web interface to control the amount of liquids/solids dispensed in each dispenser.

2 Design

This automated boba station will have two dispensers: one for tapioca pearls and one for milk tea. We will be using a screw conveyor to dispense the tapioca pearls and a solenoid valve to dispense the milk tea. An employee can manually refill the containers holding the milk tea and tapioca pearls. There will then be a sloped gutter in which the dispensers will dispense to which direct the ingredients into the cup, which is placed at the bottom end of the platform on top of another raised platform with a pressure sensor underneath. Each dispensers will be powered by 12 V and controlled by a power controller. A Raspberry Pi will be then controlling the power controllers for each dispenser as well as connect to Wi-Fi and open up a web server to serve the web UI to control the amount of liquids/solids dispensed in each.

2.1 Block Diagram

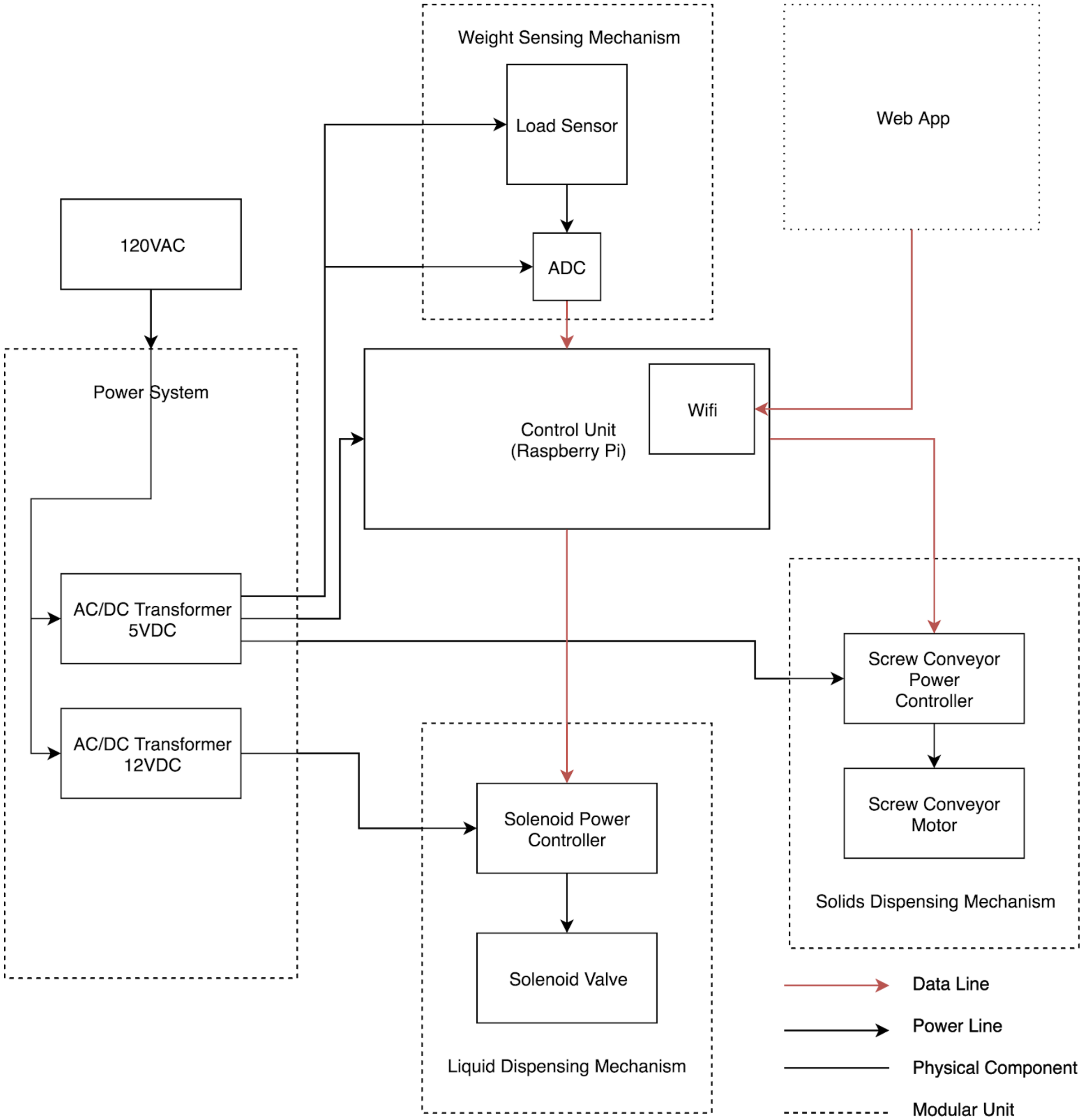

Figure 1. Basic Block Diagram

2.2 Physical Design

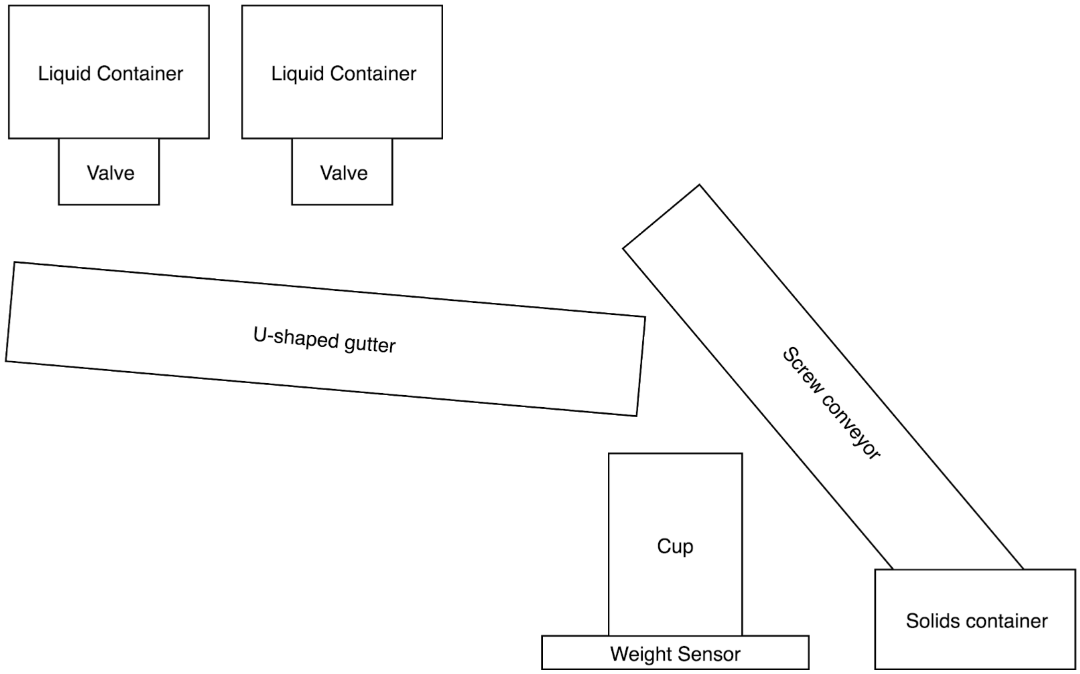

Figure 2. Physical Design Sketch

2.3 Power Supply

A power supply will be used to supply power to the controllers, Raspberry Pi, load sensor, and dispensers. It must be able to convert the wall outlet voltage of 120 V AC to +12 V DC for the solenoid and to +5 V DC for the Raspberry Pi and motor.

2.3.1 AD/DC Transformers

Two AD/DC transformers will be used to transform the wall outlet voltage to lower voltages so that the controllers, Raspberry Pi, load sensor, and dispensers can be powered.

Requirement 1: One transformer must be able to provide 5 V +/- 5% from a 120 V AC source.
Requirement 2: Another transformer must be able to provide 12 V +/- 5% from a 120 V AC source.

2.4 Liquid Dispensing Mechanism

2.4.1 Solenoid Valve

This will be used to control whether or not liquid is being dispensed. It should be normally closed when no voltage is applied, and open when sufficient voltage is applied across its terminal. The solenoid valves we will be using only opens when 12 V DC is applied. Therefore, the GPIO (General-purpose input/output) signals are unable to directly open the valves.

Requirement 1: Valve must open when 12 V +/- 5% is applied.
Requirement 2: Valve must be gravity fed (allow flow of liquid by gravity alone when open), as no other liquid pressurizing mechanism is in place.

2.4.2 Solenoid Control

The solenoid control will amplify the GPIO signal from the Raspberry Pi to 12V via a transistor so that the solenoid valve can be opened and closed.

Requirement: Must be able to output 12 V +/- 5% to the solenoid valve.

2.5 Solid Dispensing Mechanism

The solid dispensing mechanism is used to dispense the topping, tapioca pearls. Because tapioca pearls must be held in a liquid solution (water or sugar water), as it would clump up together if not, we require a way to dispense just the pearls out without the liquid. Our primary solution right now is using a screw conveyor to scoop the pearls out from its container and move it up into the gutter, while leaving the liquid in the container.

2.5.1 Screw Conveyor Motor

The screw conveyor will be driven by a 5 V DC stepper motor to allow for precise control of the screw conveyor.

Requirement: Motor must have enough torque to drive the conveyor.

2.5.2 Screw Conveyor Control

The GPIO signal from the control unit will need to be converted to drive the stepper motor.

Requirement: Must be able to output 5 V +/- 5%.

2.6 Weight Sensing Mechanism

The weight sensing mechanism is used to measure the weight of the cup and liquids/solids dispensed in it. This data will be used in coordination with the dispensing mechanism to control how much of that certain ingredient we have already dispensed and need to dispense based on the configuration.

2.6.1 Load Sensor

The load sensor will be mounted under the platform the cup sits on and outputs an analog signal. We can use the TAL221 load sensor for this purpose, which outputs voltages 0.7 V +/- 0.15 V [4].

Requirement: Sensor must be accurate to < 5g to reliably measure liquids. Sensor must support up to 500g, the typical weight for a drink.

2.6.2 Load Sensor ADC

Since the load cell outputs an analog signal, it needs to be amplified and converted to a digital signal via an ADC to be read by the Raspberry Pi.

Requirement: Must be able to read voltages 0.7 V +/- 0.15 V and convert it to a digital value.

2.7 Control Unit

The control unit controls the different dispensing mechanisms, when to open and close as well as takes input from the weight sensing mechanism to make decisions. It will store the configurations set by the user through web UI it serves and manage the system accordingly.

Requirement: Control unit must connect to the wifi in order to change configurations.

2.7.1 Raspberry Pi

We've chosen the Raspberry Pi as the control unit for this entire system. It has a wifi chip built in along with a Linux OS so we could open up a port and run a web server that serves our web UI in controlling the amount of liquids/solids dispensed by each dispenser through its GPIO pins. It may have enough processing power and memory in order to host a Web UI for the user to make changes to recipes. If not, we can host a separate server on a cloud platform like AWS for the web UI, which would then make requests to the Raspberry Pi with just the recipes. This will be powered by 5VDC through a transformer from 120VAC wall outlet.

Requirement: Must have GPIO pins to control the other mechanisms. Must be connected via Wifi or other networking apparatus to allow user to access Web UI hosted on the Pi.

2.8 Risk Analysis

The solids dispensing mechanism is the component that poses the most risk to the completion of this project. The ability to dispense solid boba is fundamentally important to this project (to meet the 2nd high level requirement). Further, boba tea without boba is just milk tea. However, boba can be messy to work with. It is small pieces of solid material that must be stored in water to prevent it from drying [1]. The mechanism must be able to dispense solid boba while also dispensing as little of the water that it is stored in as possible. The mechanism must also be able to store a large amount of boba. The mechanism must not leak any water to any of the electrical components. Further, the mechanism must be able to dispense the right amount of boba as instructed by the Control Unit. Our current solution using the screw conveyor is not finalized. Other solutions are also being considered, but should also require some sort of motor similar to the one we have described.

3 Safety and Ethics

3.1 Ethics

While working on this project, we will abide by the IEEE Code of Ethics and the ACM Code of Ethics in their entirety. For this project, it is important to commit ourselves to #1 of the IEEE Code of Ethics: "to hold paramount the safety, health, and welfare of the public, to strive to comply with ethical design and sustainable development practices, and to disclose promptly factors that might endanger the public or the environment" [2]. We will do so by following safety precautions described in the next section. For instance, we will use certified transformers to convert 120 V AC to 5 V DC and 12 V DC. Further, we will also be sure to commit ourselves to #7 of the IEEE Code of Ethics: “to seek, accept, and offer honest criticism of technical work, to acknowledge and correct errors, and to credit properly the contributions of others” [2].

3.2 Safety

The main safety hazard is the use of a 120 V AC power from a wall outlet. To mitigate this hazard, we will use certified transformers (such as the Schumacher PC-6 DC Power Converter) to immediately transform this to 5 V DC and 12 V DC and will never use 120 V directly in any way. We will also abide by United States regulations regarding electrical devices such as those described by NISTIR 8118r1 [3]. From the end user’s perspective, there are few safety concerns since the user just has to interact with the web interface and then grab a cup of boba.

4 References

[1] “Everything You Need to Know About Bubble Tea”, 2016. [Online]. Available: https://www.souschef.co.uk/blogs/articles/everything-you-need-to-know-about-bubble-tea [Accessed February 6, 2019]

[2] “7.8 IEEE Code of Ethics”. [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8.html [Accessed February 6, 2019]

[3] “A Guide to United States Electrical and Electronic Equipment Compliance Requirements”, 2017. [Online]. Available: https://nvlpubs.nist.gov/nistpubs/ir/2017/NIST.IR.8118r1.pdf [Accessed February 6, 2019]

[4] “TAL221: MINIATURE LOAD CELL” [Online]. Available: https://cdn.sparkfun.com/assets/9/9/a/f/3/TAL221.pdf [Accessed February 7, 2019]

[5] “Milk Tea & Iced Tea” [Online]. Available: http://www.sunnysboba.com/milk-tea--iced-tea.html [Accessed February 7, 2019]