Self Playing Harmonica

Group 66

Sean Jasin, David Zhang, Robert Zhu

ECE 445: Senior Design Laboratory

1. Introduction

1.1 Problem:

The harmonica is a versatile, simple, yet technically difficult instrument to play. There is a need for the background music of a live instrument, yet it is difficult to master the harmonica. Some lack the time to practice and learn the harmonica. For others, they may no longer be able to physically play the harmonica, or do not have access to training or musical education. Existing musical devices exist for keyboard and string instruments, but not for wind instruments. There is a need for a self-playing harmonica that can produce melodies without requiring manual lip or breath control.

1.2 Solution:

We propose a device that can play the harmonica. The self-playing harmonica consists of multiple subsystems. The power supply provides power at all required voltages for the MCU, air pumps, and electronic pneumatic valves. The harmonica-computer interface connects to both the harmonica and the MCU, and is responsible for controlling the airflow through the harmonica. It consists of pneumatic tubes, air pumps, and electronic valves. The MCU is responsible for controlling the pumps and valves in the harmonica-computer interface, as well as taking a MIDI file and converting it into a sequence of pump and valve motions. Lastly, songs are uploaded to the MCU through WiFi. We will create a website where the user can upload a MIDI file, and that file will then be available to play on the device.

1.3 Visual Aid:

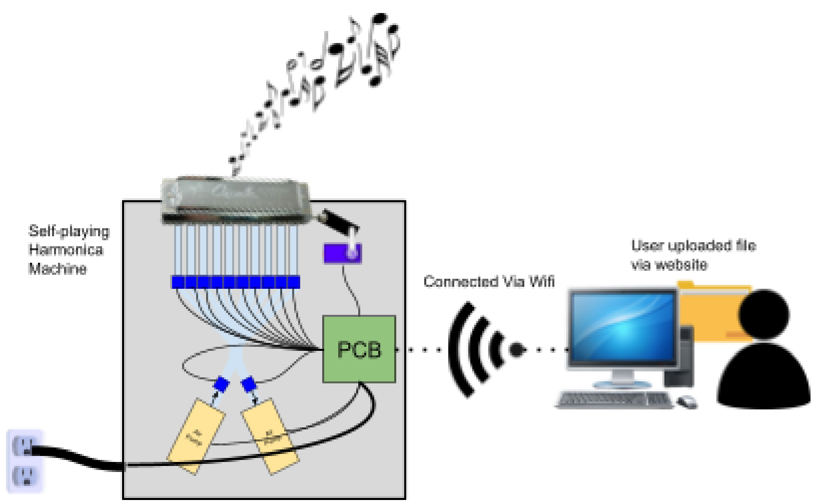

Figure 1. Visual aid for self-playing harmonica. Shows user input, harmonica control, and musical output.

1.4 High-Level requirements list:

1. Must be able to blow in and suck out air of all holes in a chromatic harmonica
2. Must be able to engage and disengage the slide of a chromatic harmonica.
3. Must be able to receive song data (MIDI) wirelessly

2. Design

2.1 Block Diagram:

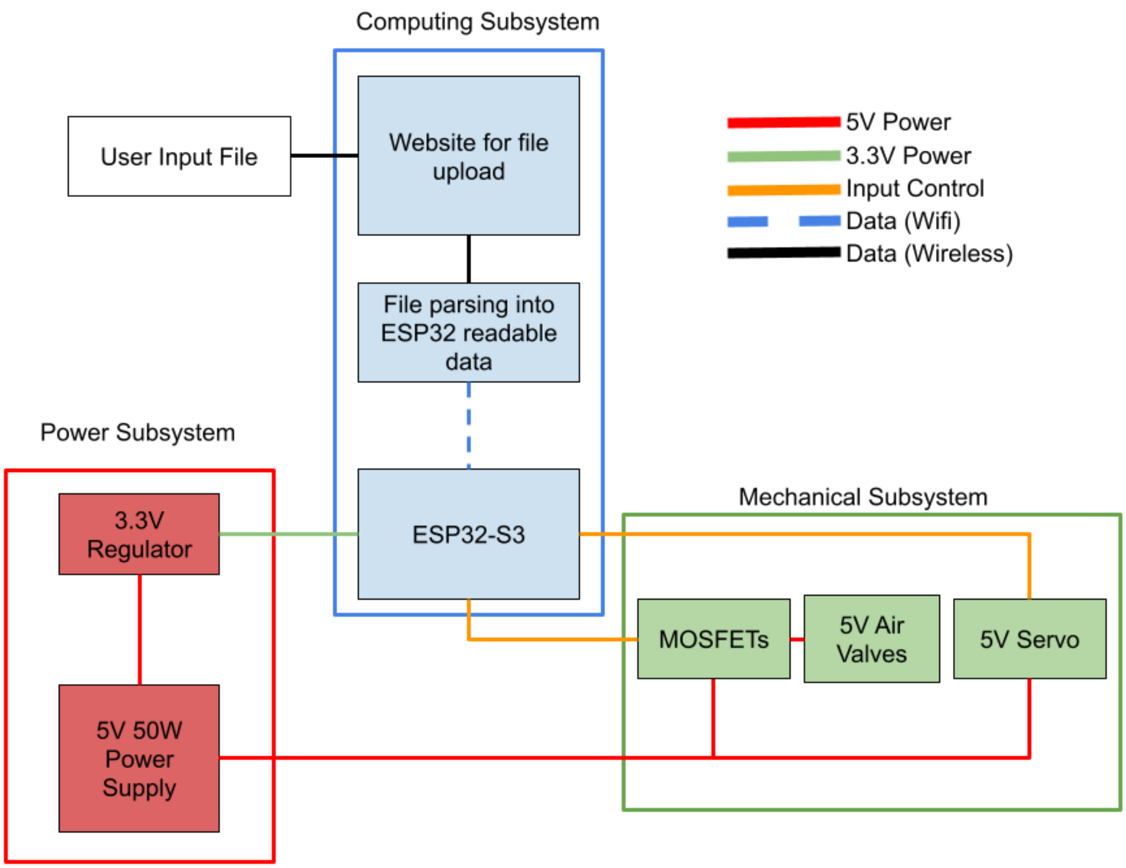

Figure 2. Block diagram of the device, showing all signal paths with color-coded signal type.

2.2 Subsystem Overview:

2.2.1 Power Subsystem

The power subsystem must be capable of supplying 3.3V and 5V power to the device. The 3.3V power supply is for the MCU, and the 5V power supply is used by the pneumatic valves and the servo. We will utilize a 5V 50W power supply that plugs into a wall outlet. We will then convert the 5V power supply into the signal voltage, 3.3V, using a voltage regulator on the PCB.

2.2.2 Computing Subsystem

The computing subsystem will use an ESP32 for its WiFi capabilities. The MCU has 2 functions: mechanical control and MIDI upload. The mechanical control will take MIDI inputs and will output the signal to control the motors (solenoids and servos) on the mechanical subsystem.

2.2.3 Mechanical Subsystem

The mechanical subsystem consists of all mechanical components of the device. This includes air pumps, solenoids, and each of their respective drivers. The drivers will allow the MCU to control the motors that dictate the airflow and the note being played.

2.3 Subsystem requirements:

2.3.1 Power Subsystem

The power subsystem must be able to provide 5 +/- 0.5V at a continuous current of 3A. The power system must also be able to provide 3.3V +/- 0.5V at a continuous current of up to 0.5A. The power system must be capable of handling current spikes in order not to overload the components of the PCB. Therefore, we will utilize a robust power supply and voltage regulator that is capable of handling at least double the required current.

2.3.2 Computing Subsystem

The computing subsystem will consist of a computer, website, the PCB, and wifi capabilities connecting the computer to the PCB.

The website will consist of a file upload from a website on the PC to the ESP32 using the wifi.h library and a wifi connection.

The PCB will have to get its power from the power subsystem, and it will provide all outputs via GPIO pins to the mechanical subsystem. The PCB will have an antenna for wireless communication, LEDs to display the state of the chip, and buttons for user control. Once the PCB receives the file from the computer through wifi, the ESP32-S3 will process the file and turn it into physical GPIO outputs.

The ESP32 will control the servo via a PWM signal through a standard GPIO output. To operate the 6V air valves, the ESP32 will send a 3.3V signal to MOSFets, connecting the air valves to a 6V power rail. Finally, the ESP32 will control the two air pumps using PWMs, also using MOSFets to control the on/off of the air pump.

2.3.3 Mechanical Subsystem

The mechanical subsystem will consist of the three main mechanical components - air valves, servo, and pump.

The air pumps we will use take PWM signals as inputs, so the drivers must be able to output PWM signals with varying duty cycles. We will utilize MOSFETsas motor drivers to convert the 3.3V PWM output from the ESP32-S3 into a 5V PWM signal for the pump.

There will be two DC 6V pumps. One of the pumps will be pumping air towards the harmonica, creating positive pressure. The other pump will be sucking air away from the harmonica, creating negative pressure. These pumps will be controlled by the ESP32. Both of these pumps will draw around 500mA. These pumps will provide airflow through ¼" rubber tubing. A 10-port air manifold will be required for the pumped air to be distributed to all 10 holes of the harmonica to be of relatively equal strength.

There will be eleven 2-position 3-way air valves. These air valves have two positions and three ports - one input and two outputs. Depending on the electrical stimulus, the air valves will be in one position or another. Ten of these air valves will be connected to tubes connected to the harmonica. These air valves will connect airflow from the pumps to the holes of the harmonicas or to the open air. The other air valve will control which pump is connected to the harmonica holes, determining if the air will be negative or positive pressure.

The servo will actuate the lever on the side of the 10-hole chromatic harmonica. What this does is switch the harmonica from whole notes to accidentals (ex. G to G#). This servo must have enough torque to actuate the lever, which will be enough, as the servo we are using has a stall torque specification of 2 kg*cm, which should be more than enough to actuate the lever. This servo is controlled by a PWM signal from the ESP32-S3.

2.4 Tolerance Analysis:

2.4.1 Component failure

2.4.1.1 ESP32

Our system makes use of 2 voltage lines, one at 5V and another at 3.3V. The logic found on the ESP32 is 3.3V, and the power input V_{cc} is only 3.3V-tolerant due to the FLASH chip it utilizes. When considering the fabrication of the device, the voltage input of the ESP32 must not coincide with the 5V lines through any means to avoid damaging the ESP32.

2.4.1.2 MOSFET Back EMF (Pump speed control)

Another concern of our system comes from how we will control airflow. More specifically, we will PWM 1 (or 2) pumps to control the amount of airflow through the pneumatic system. The problem arises when the motor is turned off and generates voltage across its terminals. This voltage can damage the MOSFETs, which control the motor, and must be dealt with via a flyback/free-wheeling diode. The flyback/free-wheeling diode will be placed across the motor in order to clamp any stray voltage generated by the motor when being pulsed.

2.4.2 Motor and solenoid control accuracy

Accurately playing a song on a harmonica requires precise timings of pushing and pulling air through the harmonica. Therefore, it is highly important for the accuracy of the music that our device controls when to play notes and stop playing notes with high precision. This is a challenge because the compressibility of air makes it so that there will be some delay between the pump starting to push air and the note starting to play. This limits the speed at which we can switch from playing to not playing, and vice versa. Given that the pump has a maximum output of 2.5 LPM and the tubing is ¼ inch, we can calculate that the air speed through this tubing is 4.3 feet per second. Then, estimating that we will have about 3 feet of tubing from the pump to the harmonica, that means that the pump to harmonica delay will be approximately 0.7 seconds. This is useful to know because we can factor this into our controls and change our input timings to adjust for this delay.

2.4.3 Even air distribution for the harmonica

Even air distribution is vital for chords on the harmonica. However, because the airflow through the harmonica only affects the volume of the note being played, the chord being recognizable will be considered acceptable. This is because it is difficult to design a 10-port air manifold that has both even distribution and low delay, given our low air pressure of ~1 psi.

3. Ethics, safety, and societal impact

3.1 Ethics

Creating a self-playing harmonica brings up questions about the ethical considerations surrounding accessibility as well as the role of automation in the arts. On one hand, this device could significantly expand access to music for individuals with physical impairments. On the other hand, there could possibly be concerns raised by the possibility of replacing humans with automation. As a team, we think that the ethical concerns are addressed by the intention of the design as well as its capabilities. The device we are proposing to create is not meant to replace human musicians, nor will it have the same capabilities as a musician. Rather, the self-playing harmonica is meant to increase the spread and awareness of harmonica music. The IEEE code of ethics section I.2. states that members should strive to improve the understanding of the public of intelligent systems and emerging technologies. By introducing this device, we aim to increase the awareness of smart systems. Therefore, we believe that we are operating in accordance with this code of ethics.

3.2 Safety

According to our analysis, our device can draw upwards of 5 amps. In order to have substantial overhead, we choose to utilize a 5V 10A power supply. This could be dangerous, as 10A is a very high current. We will follow proper electrical safety procedures in the lab safety training to ensure the safety of our team members as we proceed with this project.

References

IEEE. “IEEE Code of Ethics | IEEE.” *Ieee.org*, 2020, www.ieee.org/about/corporate/governance/p7-8.

Datasheets:
DFR0866 2-Position 3-Way Air Valvet
ESP32-S3 Technical Reference
Sunfounder 1080SF Servo
ZR370-02P Pump