The accompanyBot

Team D7: Aden Fiol, Rahul Khandelwal, Nora Wan
18-500 Capstone Design, Spring 2023
Electrical and Computer Engineering Department
Carnegie Mellon University

Piano icon created by Freepik = Flaticon at https://www.flaticon.com/free-icons/piano

Product Pitch

The accompanyBot is a piano playing robot that aims to enhance the playing experience for musicians in need of piano accompaniment for practicing and performing. From a streamlined GUI on a local application, users can upload their sheet music and control piano player actions with **low latency**. Our design ensures that notes are played **accurately** and at the **correct tempo** while also offering users the freedom of **tempo variability.**

Our product provides the appeal of live acoustics compared to MIDI recordings while offering a cheaper alternative to hiring professional accompanists or purchasing high-end player pianos on the market today.

System Architecture

The Local Application provides an interface that allows users to upload their sheet music. The sheet music is then parsed with Optical Music Recognition software into an XML file that is sent to the Raspberry Pi using SCP. Additional user features such as pausing, playing, changing the tempo, jumping measures etc. are sent to the Raspberry Pi serially through an Arduino Uno.

The Raspberry Pi 4 takes the XML file and parses it with the help of the music21 Python library. The main control function then converts the parsed music into scheduled GPIO signals that play at specific time instances. It also takes in byte commands sent via **The Arduino Uno** acting as a USB-to-UART serial converter.

The Physical Interface consists of 12 solenoids, a chassis to suspend the solenoids over the keyboard, and a circuit that controls the solenoids. The circuit is made of switching transistors that are turned on and off by the Raspberry Pi and diodes used to prevent voltage flyback caused by the solenoids from damaging our other systems.

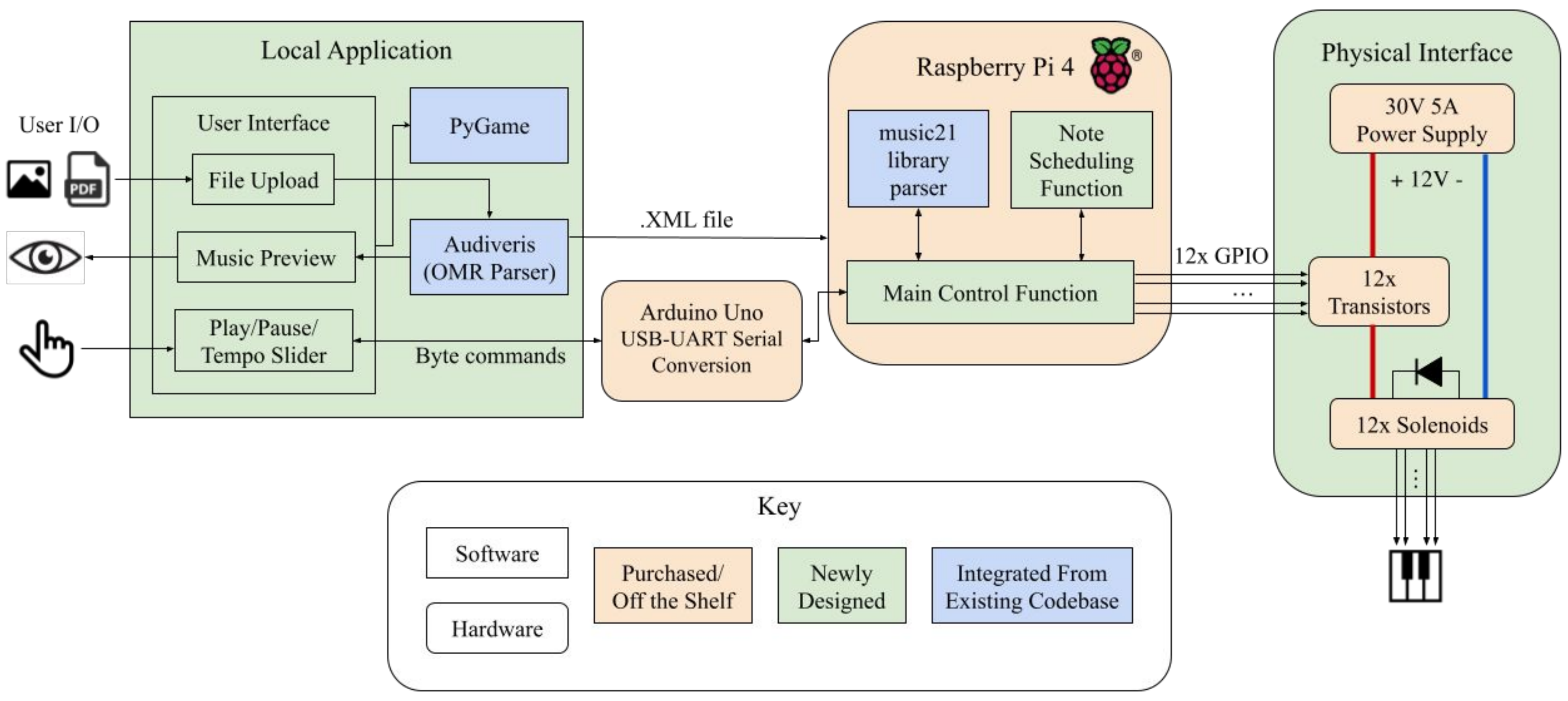


Fig. 4. System Block Diagram

Conclusions & Additional Information

Scan the QR code to check out our blog!

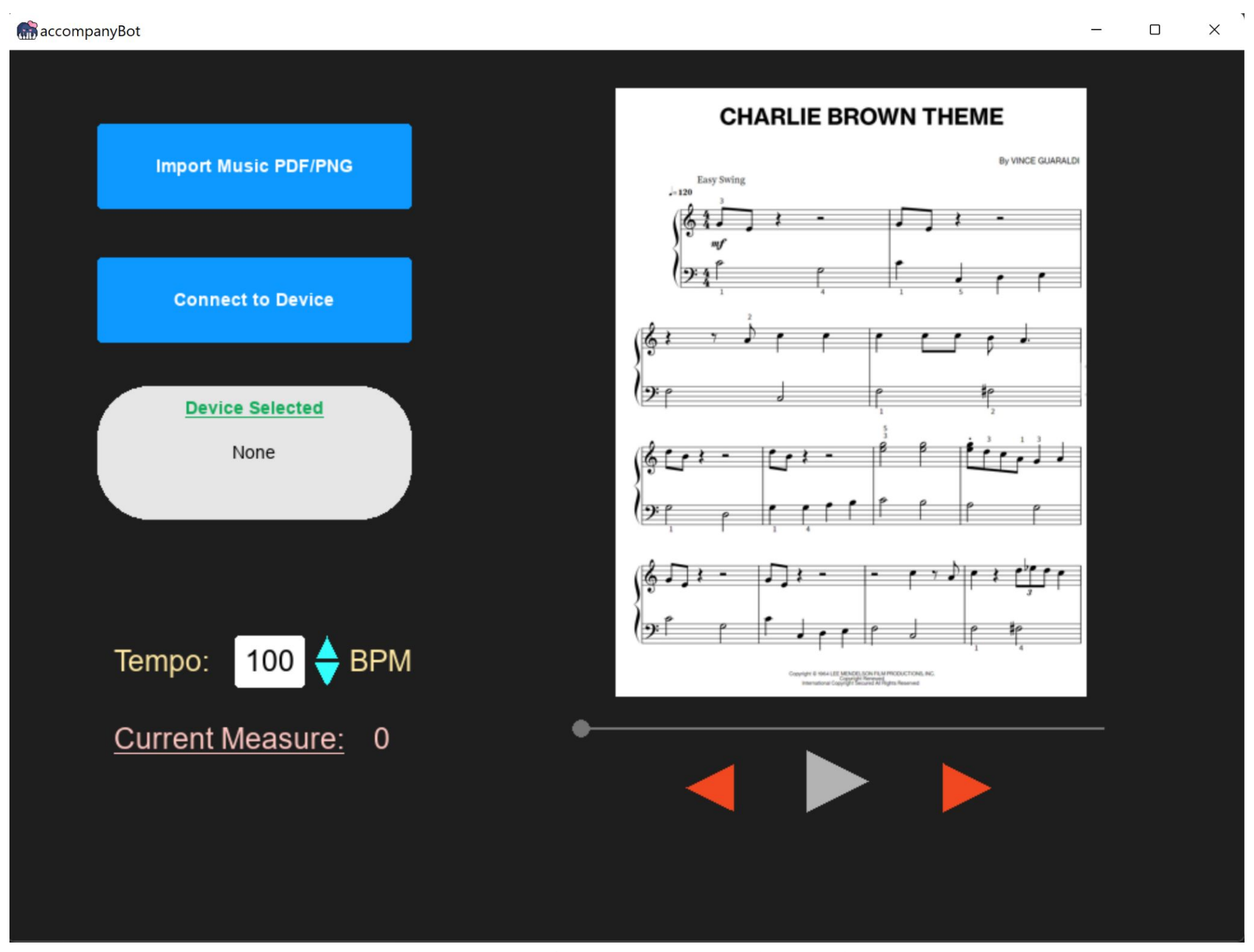
http://course.ece.cmu.edu/~ece500/projects/s23-teamd7/

While our product cannot outperform a virtuoso pianist, the accompanyBot is capable of performing simple pieces for accompaniment. Through the process, we discovered reading music is a hard task and a complete automated solution is yet to exist. Another challenge we faced was underestimating the time for integration and prototyping. Our system is a solid proof of concept for playing one octave and has the potential to be expanded to play the entire range of keys on a piano, with cost being the limiting factor.

System Description

Software:

- ➢ **pyGame Application**
  - ○ Dynamically handles user inputs and actions
  - ○ Interfaces with note scheduling and OMR process
- ➢ **Audiveris Optical Music Recognition (OMR)**
  - ○ Converts notated sheet music into organized, descriptive, text logs, readable by the notes scheduler


accompanyBot

Import Music PDF/PNG

Connect to Device

Device Selected
None

Tempo: 100 BPM

Current Measure: 0

Fig. 1. GUI application

Hardware:

- ➢ **Arduino Uno**
  - ○ Serves as a proxy between GUI and Raspberry Pi
- ➢ **Raspberry Pi4**
  - ○ Hosts notes scheduling algorithm
  - ○ Communicates metrics for software application
  - ○ Sends digital signals to switching MOSFETS
- ➢ **Robotic "Hand"**
  - ○ Adafruit 25N Push/Pull Solenoids to press on piano keys
  - ○ Adafruit N-channel MOSFET switches for solenoid "fingers"

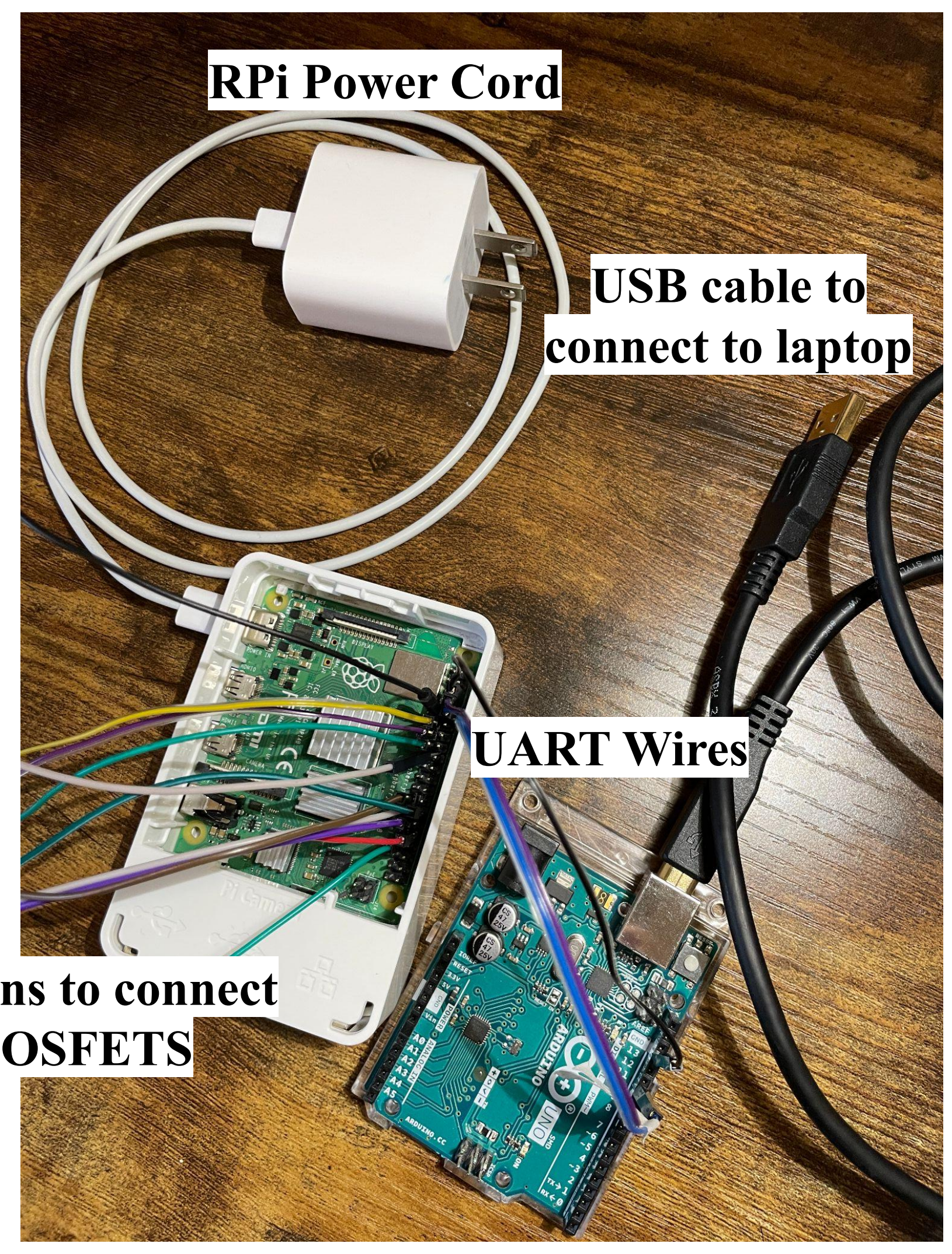

RPi Power Cord
USB cable to connect to laptop
UART Wires
GPIO Pins to connect to MOSFETS

Fig. 2. Arduino and RPi

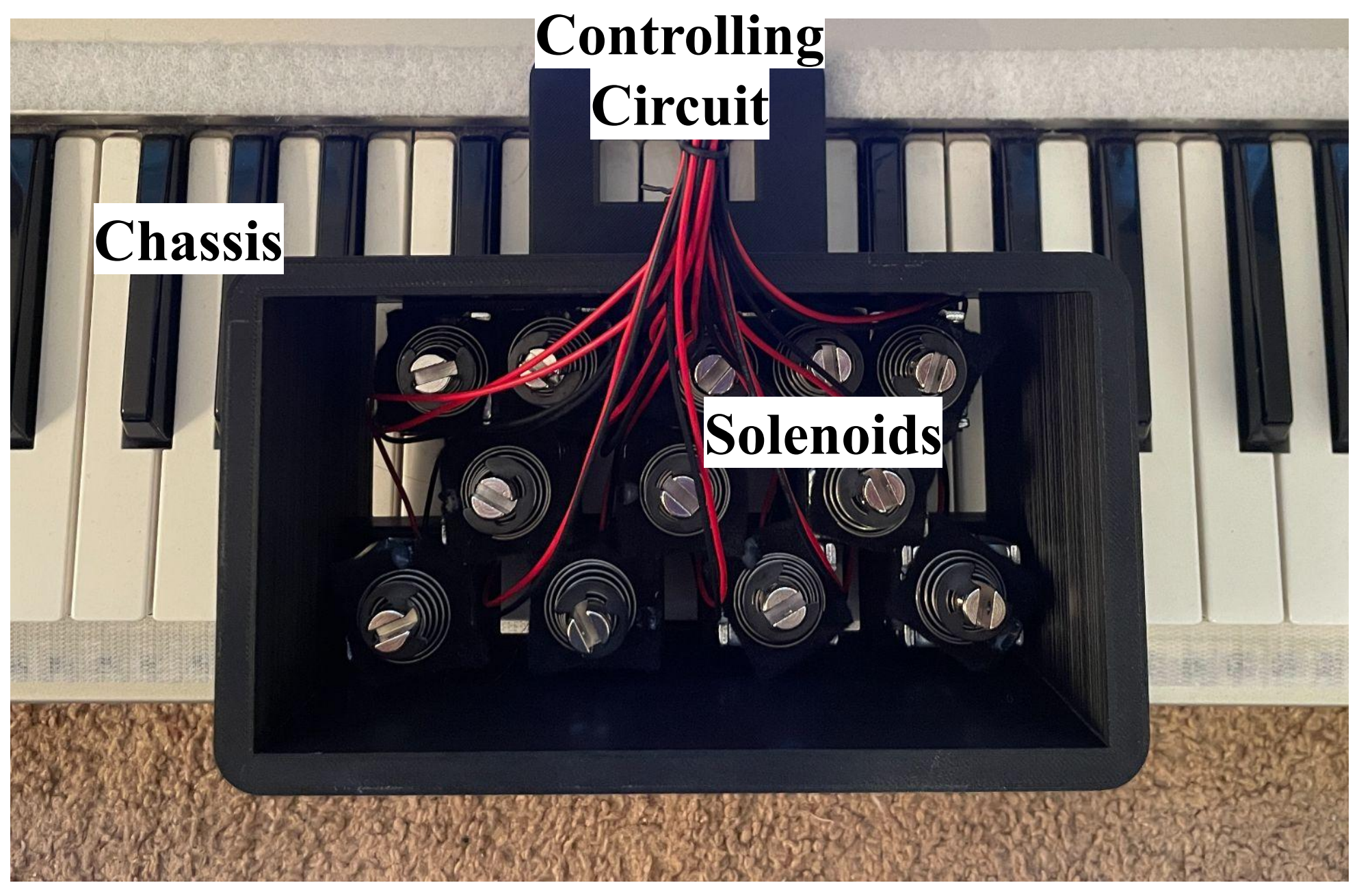


Fig. 3. Chassis and Solenoid

System Evaluation

- ➢ **OMR Accuracy -** Reconverted XML file back to pdf and visually compared output to the original
- ➢ **Tempo Accuracy -** Measured tempo with a metronome
- ➢ **Computer/RPi Latency -** Recorded system times when start/stop signals are sent and received and computed the difference
- ➢ **Power -** Inspected power and current readings on DC power supply

Design Requirement Metric	**System Performance**
> 95% OMR accuracy of note pitches and note values for 400+ DPI music scores	~99% accuracy for easy to moderate playing difficulty scores of 240 DPI
100% tempo accuracy	100% accurate between 30BPM-150BPM
< 150 ms latency between pressing the start/stop button and hearing a	Round trip communication latency of ~70ms
< 60W average power consumption	Max power of 46.8W

Table 1: Quantitative Testing Results

Electrical & Computer ENGINEERING
Carnegie Mellon