**Laser Harmony**

**Overview**

LaserHarmony is an innovative and captivating project that merges the realms of music and technology, turning conventional musical instruments into mesmerising laser generators. This repository hosts the code, resources, and documentation necessary to create your very own LaserHarmony setup.

**Table of Contents**

1. Introduction

2. Hardware Components

3. Software Setup

4. Circuit Diagram

5. Usage Instructions

6. Future Enhancements

7. Conclusion

**1.Introduction**

LaserHarmony is born from the imagination of combining the sonic world of music with the visual allure of lasers.

This project is about creating a string musical instrument and replacing the strings with lasers. We aim to save resources and decrease the waste of nylon and metal strings that are used to play on those instruments.

**2.Hardware components**

* Arduino micro
* Photoresistors -> 7
* Laser -> 7
* Bluetooth module HC-05 -> 1
* Wire -> 3 metres
* Wooden board -> 1, 20x60cm

**3. Software Setup**

**PySerial**

**Requirements**

* Python 2.7 or Python 3.4 and newer
* If running on Windows: Windows 7 or newer
* If running on Jython: “Java Communications” (JavaComm) or compatible extension for Java

**Installation**

**https://lh3.googleusercontent.com/7vRaicY-7gWfgTCtHPxFQAGnKfwbrwh2U8JdyFynoc2WzYMvAp6ymhID8TB-X6-vhAoG2gwKbhmXlTkn6gVTm1rZ6NiiI23kaYLGN_koq-Wq9EiDaeFA-qm4Q7IPqrZ2vx8AglaQ6gqQ6n-Jq92q5-Q**

**\*if throws error, try running the command prompt as administrator**

**NumPy**

**Requirements**

* Python 3.8 and newer
* If running on Windows: Windows 7 or newer

**Installation**

**https://lh4.googleusercontent.com/S6flvCwDWoqwD89CpD989c6PO8KZhXrhbNK582v5g0pbEt-av-waENe-vAJ9IRCvI11xp3vTrEPNHwbhjJoIZfuEi3ErcfUMIDRJlIF4hGz8MdmXNF02rvcQ-h4c98lcF_DNaetOKfA-bsfIq9VdYks**

**PyAudio**

**Requirements**

* Python 2.6,2.7 or Python 3.2 and newer
* If running on Windows: Windows 7 or newer, you may also need to have a C/C++ compiler toolchain installed
* You might need to install PortAudio, NumPy separately.

**Installation**

**https://lh5.googleusercontent.com/saXuJmGkbaPuaHs7fP7D1jxZbo3BvajLmDTb4SkNNJbZcEPmmi65d6RnMAyUD68UNwpZjQGywCVF0rzy2ZljJ4aE6k4T_hSCK4Wa11BmJUYZBQJoZN7IpeqNbapH1_PWNeoT5Wur8psN6W-rmtwc2Wk**

**Tkinter**

**Requirements**

* Python 3.7  and newer

**Installation**

**https://lh4.googleusercontent.com/GxVFwgIU7seEbrYqT7fi5efS5ZB-WkQzzdqivcxxzLiwnE5SBDwugEd14EFoN8brigxxJTvlqljBq4BNF-tFSuYc1mv73X1SbHMzk9LhFLfT_oOTgNCUmA3AbtjkpZ415vVb8zijzmUPbuXNL-O8PwE**

**4. Bluetooth Connection and PC Audio Output**

In our musical instrument project, the Arduino Nano establishes a wireless Bluetooth connection with a computer, allowing users to control the instrument remotely. Additionally, the project leverages the computer's speakers to output sound generated by the instrument.

Bluetooth Connection: The Bluetooth module integrated into the Arduino Nano facilitates the wireless connection to the computer. This module communicates with the computer's Bluetooth system, enabling data exchange between the Arduino and the computer.

PC Interaction:

Bluetooth Pairing: The first step involves pairing the Bluetooth module of the Arduino with the computer. This is typically done through the computer's Bluetooth settings, where the Bluetooth module is recognized and paired with a unique identifier.

Serial Communication: Once paired, the Arduino establishes a serial communication channel with the computer over Bluetooth. This allows data to be sent and received between the Arduino and the computer in a structured manner.

Data Exchange: The Arduino sends control signals, sensor data, and other relevant information to the computer. The computer interprets these signals using a script or software written in a programming language like Python. This script interfaces with the Arduino through the PySerial library, allowing bidirectional communication.

PC Speakers for Audio Output: While the Arduino Nano itself does not have built-in audio output capabilities, we can utilize the computer's audio system to produce sound generated by the musical instrument.

Sound Generation: The Arduino generates sound signals based on user input, sensor readings, or any other control mechanism. These sound signals can represent musical notes, tones, or any desired audio output.

Data Transmission: The Arduino sends these sound signals to the computer through the established Bluetooth communication. The computer receives the data and processes it.

Audio Output: Using a script or software running on the computer, the received sound signals are converted into audio output. This audio output is then played through the computer's speakers, producing the audible results of the musical instrument.

Benefits:

Wireless Control: Bluetooth connectivity enables users to control the musical instrument from a distance, enhancing flexibility and creativity.

High-Quality Sound: Utilising the computer's speakers allows for high-quality audio output, providing a richer and more immersive experience.

By combining the Arduino's Bluetooth communication capabilities with the computer's audio system, our musical instrument project achieves wireless control and high-quality sound output, contributing to an engaging and versatile user experience.

This section should help your readers understand how the Arduino connects to a computer via Bluetooth and utilises the computer's speakers for audio output, enhancing the overall functionality of your musical instrument project.

**5. Usage Instructions**

1. Turn on the computer that you are going to use
2. Connect the bluetooth module to your computer
3. Play by sliding your fingers over the lasers
4. You can change the octaves in the interface
5. Enjoy

**6. Future Enhancements**

In the future we plan to add features like:

1. More options for sounds
2. Listening to the music by headphones
3. Recording the music you play

**7. Conclusion**

In the world of wires and whispers, our Arduino Nano musical instrument has sung its digital melodies. Through the magic of Bluetooth, we connected with computers, shaping soundscapes that danced from our fingertips to the speakers.

In this symphony of technology and art, we discovered:

* Wireless Enchantment: Bluetooth bridged the gap between Arduino and computer, giving us the freedom to play our instrument from afar.
* Expressive Harmonies: By uniting Arduino's ingenuity with computer audio, we painted vibrant sonic canvases that resonated with emotion.
* Innovation's Call: Our journey doesn't end here; it's an invitation to fellow creators to explore, refine, and compose anew.

With every key pressed and every note whispered, our musical instrument tells a tale of curiosity, tinkering, and boundless possibilities. As the last echo fades, may it inspire future artisans to craft their own digital crescendos.

From pixels to melodies, The LaserHarmony Team