



INTERACTIVE MEDIA

DOCUMENTATION WEEK 10 MUSIC INSTRUMENT

VIDEO LINK IN CODE

SYED FAHAD RIZWAN
AAKARSH SINGH



IDEAS

While brainstorming, **Syed Fahad Rizwan** wished to make an all-in-one instrument, changing instruments via a pre-defined input but that would have required playing .wav files on our Arduino. However, all requisites for that were beyond our knowledge. Therefore, after email consultation with you, we chose to wait until we connected **Arduino with p5js**.

Aakarsh wished to incorporate an assortment of sensory inputs to maximize the amount of creative control available for the users, one of his inspirations being the **modular synthesizer** - see image below.

In a meandering journey, we explored the **MEAS** sensor. We also considered the **photoresistor**. However, they did not feel too user-friendly and reliable, which was an essential factor in our design choices.

While we respected intuition, we did not wish to recreate an existing instrument but pioneer one of our own.

Eventually, we chose to use the following components:

- 2 Speakers
- 2 Potentiometers
- 1 Ultrasonic Sensor
- 1 Force Sensor
- 1 Switch
- 1 LED
- 3 Resistors
- Jumper Wires



MODULAR SYNTHESIZER



FINAL CONCEPT

SWITCH

To turn our musical instrument **ON** or **OFF**.

POTENTIOMETERS

In synchrony with Aakarsh's idea of multiple control points, we used both **potentiometers** for two different purposes - duration of notes and frequency between the notes.

LED

We added an indicative **LED** that alternated between **HIGH** and **LOW** depending on the frequency and duration to help the user keep track of both parameters, essential to successful music production.

SPEAKERS

We used **2 speakers**. Speaker 1.0 took input from the **force sensor** while Speaker 2.0 took it from the **ultrasonic sensor**.

FORCE SENSOR SPEAKER

The **frequency** of the tone is directly proportional to force applied.

ULTRASONIC SENSOR SPEAKER

The **frequency** of the tone is directly proportional to change in distance per unit time between successive notes, set by **potentiometer 1**.

RESISTORS

To regulate flow of current.



CODE SNIPPETS

INPUTS & OUTPUTS

```
//inputs  
const int POT = A0;  
const int FORCE = A1;  
const int POT2 = A2;  
#define echoPin 2  
#define trigPin 3
```

```
//Outputs  
const int buzzer = 6;  
const int buzzer2 = 9;  
const int switcher = 7;  
const int led = 11;
```

VARIABLE DECLARATION

```
// defines variables  
int switcherState;  
long duration; // variable for the duration of sound wave travel  
int distanceNew; // variable for the new distance measurement  
int distanceOld = 0; // variable for the old distance measurement  
int distSpeed; // variable for speed  
unsigned long previousMillis = 0; // variable for setting up the interval code  
long tracker = 0; // the tracker basically keeps track of which iteration we are in of the loop  
long interval; // interval being set up by POT  
int forceValue;  
int potValue;  
int pot2Value;  
int note;
```



CODE SNIPPETS

CRUX

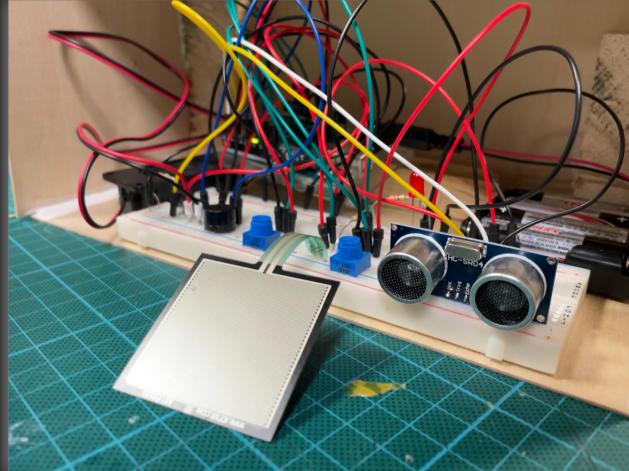
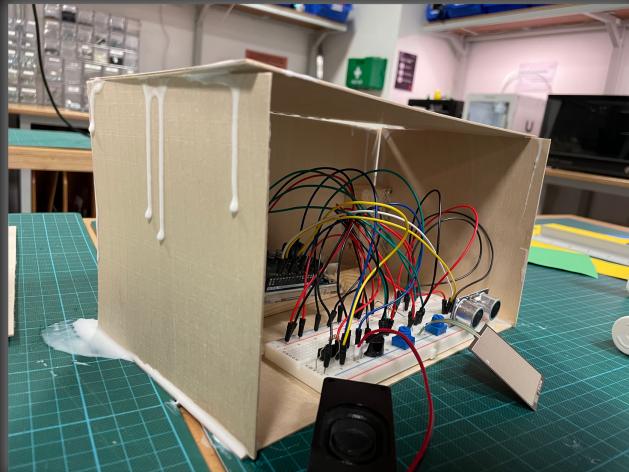
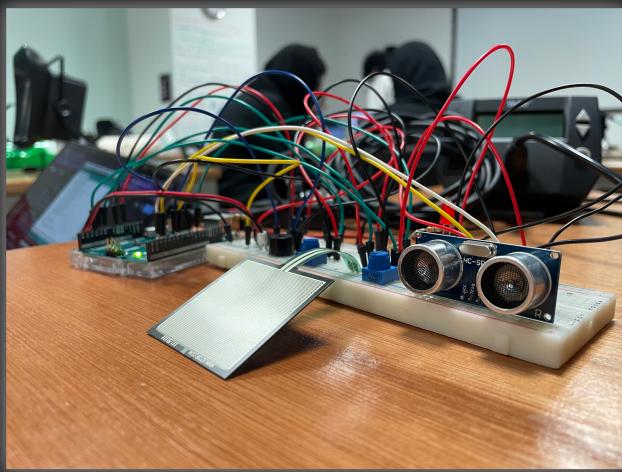
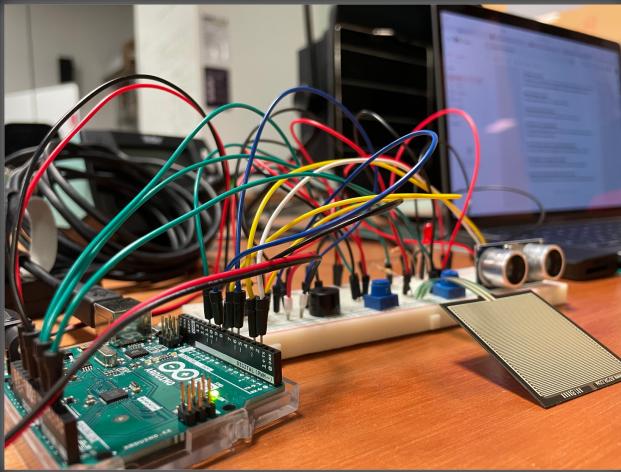
```
if (tracker % 4 == 0) {  
    digitalWrite(led, HIGH);  
    distSpeed = abs(distanceNew - distanceOld);  
    distanceOld = distanceNew;  
  
    if (distSpeed < 1) {  
        note = NOTE_C7;  
    } else if (distSpeed < 2) {  
        note = NOTE_D7;  
    } else if (distSpeed < 3) {  
        note = NOTE_E7;  
    } else if (distSpeed < 4) {  
        note = NOTE_F7;  
    } else if (distSpeed < 5) {  
        note = NOTE_G7;  
    } else if (distSpeed < 7) {  
        note = NOTE_A7;  
    } else if (distSpeed < 8) {  
        note = NOTE_B7;  
    } else if (distSpeed < 9) {  
        note = NOTE_C8;  
    } else if (distSpeed < 10) {  
        note = NOTE_D8;  
    }  
    else {  
        note = NOTE_G6;  
    }  
    tone(buzzer, note, duration);
```

```
if (tracker % 4 == 2) {  
    // the note here is determined by pressure  
    digitalWrite(led, LOW);  
    if (forceValue < 100) {  
        note = NOTE_C5;  
    } else if (forceValue < 200) {  
        note = NOTE_D5;  
    } else if (forceValue < 300) {  
        note = NOTE_E5;  
    } else if (forceValue < 400) {  
        note = NOTE_F5;  
    } else if (forceValue < 500) {  
        note = NOTE_G5;  
    } else if (forceValue < 600) {  
        note = NOTE_A5;  
    } else if (forceValue < 700) {  
        note = NOTE_B5;  
    } else if (forceValue < 800) {  
        note = NOTE_C6;  
    } else if (forceValue < 900) {  
        note = NOTE_D6;  
    }  
    else if (forceValue < 1000) {  
        note = NOTE_E7;  
    }  
    else if (forceValue < 1100) {  
        note = NOTE_F7;  
    }  
    tone(buzzer2, note, duration);
```

These tracker conditions play different notes depending on change in distance and force applied. If `(tracker % 4) == 1` or `(tracker % 4) == 3`, the result is a `noTone()` to ensure gap between successive notes. The tracker increases by 1 after each note.



PHOTO GALLERY





GROWTH AND CHALLENGES

Challenge - Finding appropriate sensors

Solution - Experiment as much as you can to find what works

Challenge - Debugging the hardware and software

Solution - The use of print statements in different blocks of code

Challenge - Adding creative sophistication and control in limited space

Solution - Using a bigger breadboard for more space

Challenge - Playing notes from 2 speakers

Solution - Using noTone() after playing from 1 speaker



LEARNINGS

Aakarsh

- How to use multiple sensors simultaneously to take in different kinds of data
- How to set the durations of notes
- I understood the importance of **noTone()**

Syed Fahad Rizwan

- Using print statements in the serial monitor to adequately label sensor input
- Woodcutting thin sheets via a cutter instead of a handsaw is an efficient yet physically and temporally less taxing, especially for lightweight projects
- The importance of beginning with an idea. I believe that technical aspects are secondary. If one masquerades with a worthwhile idea, the technical aspects manifest themselves out of respect for the idea

We both learnt how to create a wooden box to present the instrument