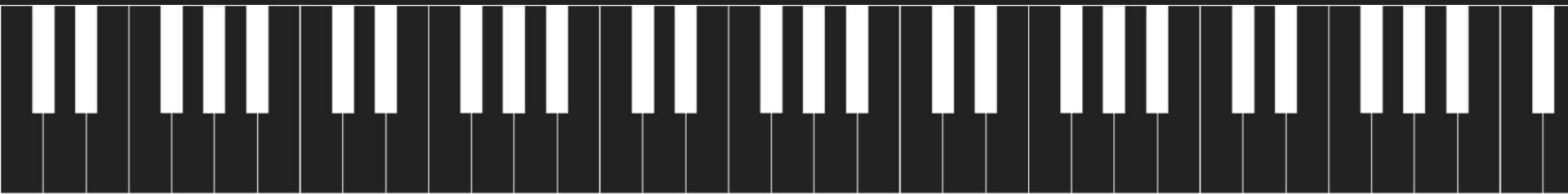
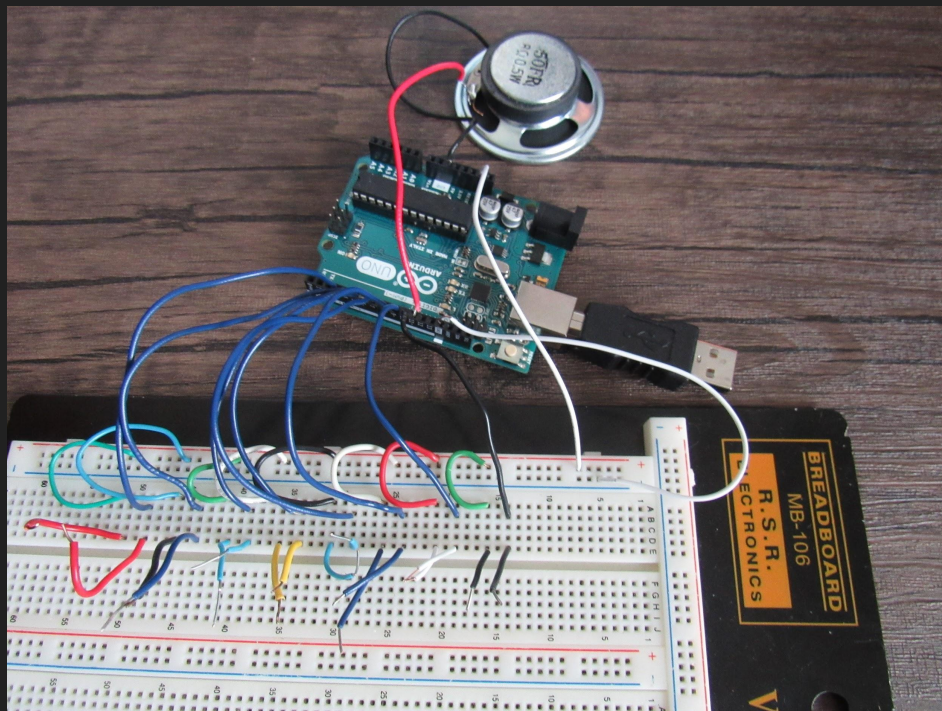


Music Keyboard

Yusuf Morsi, Foster Stein-White



The Project



- On the left is a picture of our project (the Arduino Uno Musical Keyboard)
- On the right is the code for the musical keyboard

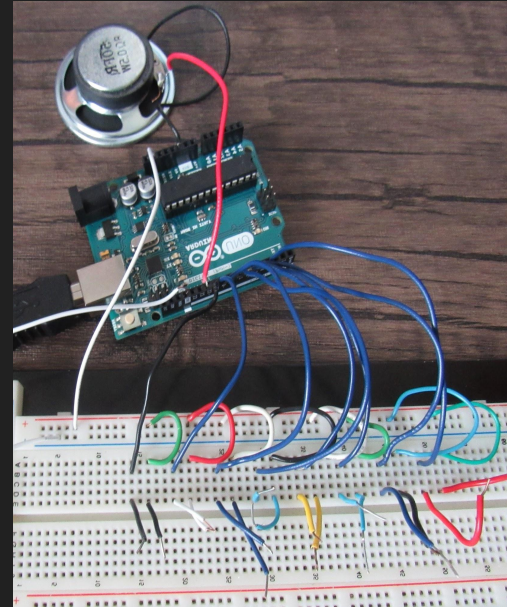
```
#define NOTE_C 262
#define NOTE_D 294
#define NOTE_E 330
#define NOTE_F 349
#define NOTE_G 392
#define NOTE_A 440
#define NOTE_B 493
#define ACTIVATED LOW
const int PIEZO = 11; // SPEAKER IS PIEZO
const int LED = 13;
const int BUTTON_C = 10;
const int BUTTON_D = 9;
const int BUTTON_E = 8;
const int BUTTON_F = 7;
const int BUTTON_G = 6;
const int BUTTON_A = 5;
const int BUTTON_B = 4;
void setup()
{
  pinMode(LED, OUTPUT);
  pinMode(PIEZO, OUTPUT);
  pinMode(BUTTON_C, INPUT);
  pinMode(BUTTON_D, INPUT);
  pinMode(BUTTON_E, INPUT);
  pinMode(BUTTON_F, INPUT);
  pinMode(BUTTON_G, INPUT);
  pinMode(BUTTON_A, INPUT);
  pinMode(BUTTON_B, INPUT);
}
void loop()
{
  while(digitalRead(BUTTON_C) == ACTIVATED)
  {
    tone(PIEZO, NOTE_C);
    digitalWrite(LED, HIGH);
  }
  while(digitalRead(BUTTON_D) == ACTIVATED)
  {
    tone(PIEZO, NOTE_D);
    digitalWrite(LED, HIGH);
  }
  while(digitalRead(BUTTON_E) == ACTIVATED)
  {
    tone(PIEZO, NOTE_E);
    digitalWrite(LED, HIGH);
  }
  while(digitalRead(BUTTON_F) == ACTIVATED)
  {
    tone(PIEZO, NOTE_F);
    digitalWrite(LED, HIGH);
  }
  while(digitalRead(BUTTON_G) == ACTIVATED)
  {
    tone(PIEZO, NOTE_G);
    digitalWrite(LED, HIGH);
  }
  while(digitalRead(BUTTON_A) == ACTIVATED)
  {
    tone(PIEZO, NOTE_A);
    digitalWrite(LED, HIGH);
  }
  while(digitalRead(BUTTON_B) == ACTIVATED)
  {
    tone(PIEZO, NOTE_B);
    digitalWrite(LED, HIGH);
  }
  noTone(PIEZO);
  digitalWrite(LED, LOW);
}
```

Project Function

- In this project, we created a keyboard that is connected to a speaker, and using different wires, we have created a variety of keys (C, D, E, F, G, A, B, C).
- This keyboard is a fully functional musical device thanks to our code and access to a breadboard, arduino, and wires.

How it Works

First, we connected the speaker to the Arduino Uno (Pin 11). Then, on the breadboard, instead of using buttons (which weren't available), we stuck wires in the board (and those wires touching cause the speaker to make noise) and connected them to pins on the Arduino (Pins 4-10) with more wires. In order to assure that each key emitted the specific sound we needed it to, we made sure that our code knew what frequency number each note has, and established the connections of each note to its assigned key.



Problems and Difficulties

- Not having any buttons to work with
- Having time as a restriction
- The arduino application not working at first
- The speaker breaking
- Not having enough wires

Solutions to Our Problems

- Instead of using buttons, we used wires
- One of us took the project home during the weekend to test the code and fix the wiring to have a little bit of extra time
- When the application did not work at first, we used a different computer while we waited for it to be downloaded
- At first when the speaker broke, we replaced it. Then, when we found a problem with our new speaker, we got a wire, wrapped it around the speaker's dysfunctional wire, and added tape.
- When we realized that we didn't have enough wires, one of us brought a wire spool from home and used some of them

Thank You

