

# DJ PULT with Arduino

## Tools:

- 1 x Arduino Mega 2560
- 1 x Breadboard
- 4 x Button
- 4 x LED
- 1 x RGB LED
- 1 x Passive Buzzer
- 1 x Potentiometer
- 1 x 8-segment display
- Cables
- Resistors (110 ohm, 220ohm, 1kohm)

## Principle:

In my project I wanted to create a DJ Pult where we can mix different melodies, and also make on the breadboard some light show.

My Arduino was not so reliable so when I read the analog values from the potentiometer the 8-segment display has some anomalies.

### Theory:

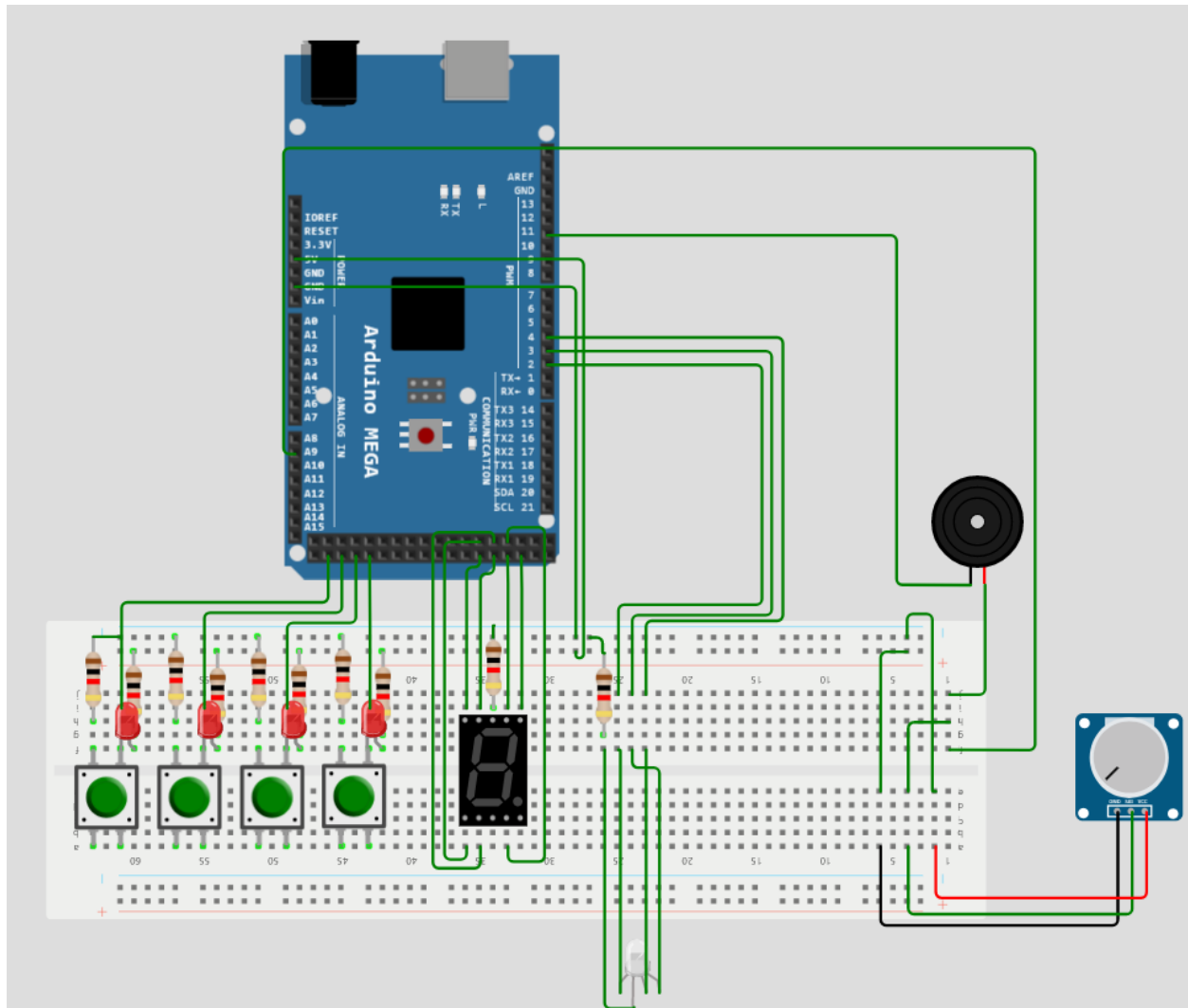
With the help of the 4 buttons we are able to generate melodies depending on what frequencies are used at each button. Furthermore we are able to mix these or just play them individually. 4 LEDs are showing us which buttons are pressed at a given moment.

The RGB LED is synchronized with the sound frequencies and as the frequencies are changing the color of the RGB LED is changing, as well.

With the help of a potentiometer we are able to change the level of the input voltage of the buzzer, this way we are able to turn down or turn up the intensity of sound.

The intensity of the sound is displayed on the 8-segment display (in discrete values that correlate with the analogue values).

## Schematic:



# Code:

sketch\_feb09a | Arduino 1.8.19

File Edit Sketch Tools Help



```
int value;
```

```
void setup () {
```

```
  pinMode (2,OUTPUT); //RGB led pins
```

```
  pinMode (3,OUTPUT);
```

```
  pinMode (4,OUTPUT);
```

```
  pinMode (11,OUTPUT); //Buzzer pin
```

```
  pinMode (25,OUTPUT); //8 segment display pins (no need for the decimal point led)
```

```
  pinMode (26,OUTPUT);
```

```
  pinMode (27,OUTPUT);
```

```
  pinMode (28,OUTPUT);
```

```
  pinMode (29,OUTPUT);
```

```
  pinMode (30,OUTPUT);
```

```
  pinMode (31,OUTPUT);
```

```
  pinMode (47,INPUT); //Button pins
```

```
  pinMode (49,INPUT);
```

```
  pinMode (51,INPUT);
```

```
  pinMode (53,INPUT);
```

```
}
```

```
void loop ()
{

value = analogRead(9);

if(value >= 0 and value <= 100) {
digitalWrite(25, HIGH);
digitalWrite(26, HIGH);
digitalWrite(27, HIGH);
digitalWrite(28, HIGH);
digitalWrite(29, HIGH);
digitalWrite(30, HIGH);
digitalWrite(31, LOW);
}
else if(value > 100 and value <= 200) {
digitalWrite(25, HIGH);
digitalWrite(26, HIGH);
digitalWrite(27, LOW);
digitalWrite(28, LOW);
digitalWrite(29, LOW);
digitalWrite(30, LOW);
digitalWrite(31, LOW);
}
else if(value > 200 and value <= 300) {
digitalWrite(25, HIGH);
digitalWrite(26, LOW);
digitalWrite(27, HIGH);
digitalWrite(28, HIGH);
digitalWrite(29, LOW);
digitalWrite(30, HIGH);
digitalWrite(31, HIGH);
}
else if(value > 300 and value <= 400) {
```

---

```
digitalWrite(25, HIGH);
digitalWrite(26, HIGH);
digitalWrite(27, HIGH);
digitalWrite(28, HIGH);
digitalWrite(29, LOW);
digitalWrite(30, LOW);
digitalWrite(31, HIGH);
}
else if(value > 400 and value <= 500) {
digitalWrite(25, HIGH);
digitalWrite(26, HIGH);
digitalWrite(27, LOW);
digitalWrite(28, LOW);
digitalWrite(29, HIGH);
digitalWrite(30, LOW);
digitalWrite(31, HIGH);
}
else if(value > 500 and value <= 600) {
digitalWrite(25, LOW);
digitalWrite(26, HIGH);
digitalWrite(27, HIGH);
digitalWrite(28, HIGH);
digitalWrite(29, HIGH);
digitalWrite(30, LOW);
digitalWrite(31, HIGH);
}
else if(value > 600 and value <= 700) {
digitalWrite(25, LOW);
digitalWrite(26, HIGH);
digitalWrite(27, HIGH);
digitalWrite(28, HIGH);
digitalWrite(29, HIGH);
digitalWrite(30, HIGH);
digitalWrite(31, HIGH);
}
}
```

---

```
digitalWrite(31, HIGH);
}
else if(value > 700 and value <= 800) {
digitalWrite(25, HIGH);
digitalWrite(26, HIGH);
digitalWrite(27, HIGH);
digitalWrite(28, LOW);
digitalWrite(29, HIGH);
digitalWrite(30, LOW);
digitalWrite(31, LOW);
}
else if(value > 800 and value <= 900) {
digitalWrite(25, HIGH);
digitalWrite(26, HIGH);
digitalWrite(27, HIGH);
digitalWrite(28, HIGH);
digitalWrite(29, HIGH);
digitalWrite(30, HIGH);
digitalWrite(31, HIGH);
}
else {
digitalWrite(25, HIGH);
digitalWrite(26, HIGH);
digitalWrite(27, HIGH);
digitalWrite(28, HIGH);
digitalWrite(29, HIGH);
digitalWrite(30, LOW);
digitalWrite(31, HIGH);
}
```

```
if(digitalRead(47) == LOW) {
  tone(11,200);
  analogWrite(2,20);
  analogWrite(3,40);
  analogWrite(4,60);
  delay(200);
  tone(11,222);
  analogWrite(2,200);
  analogWrite(3,20);
  analogWrite(4,60);
  delay(200);
  tone(11,180);
  analogWrite(2,120);
  analogWrite(3,200);
  analogWrite(4,150);
  delay(200);
}
else {noTone(11); digitalWrite(2,HIGH); digitalWrite(3,HIGH); digitalWrite(4,HIGH);}
if(digitalRead(49) == LOW) {
  tone(11,400);
  analogWrite(2,256);
  analogWrite(3,3);
  analogWrite(4,70);
  delay(200);
  tone(11,470);
  analogWrite(2,2);
  analogWrite(3,2);
  analogWrite(4,6);
  delay(200);
  tone(11,330);
  analogWrite(2,70);
  analogWrite(3,72);
  analogWrite(4,250);
  delay(200);
}
else {noTone(11); digitalWrite(2,HIGH); digitalWrite(3,HIGH); digitalWrite(4,HIGH);}
```

```

if(digitalRead(51) == LOW) {
  tone(11,600);
  analogWrite(2,240);
  analogWrite(3,150);
  analogWrite(4,200);
  delay(200);
  tone(11,550);
  analogWrite(2,77);
  analogWrite(3,30);
  analogWrite(4,70);
  delay(200);
  tone(11,700);
  analogWrite(2,144);
  analogWrite(3,122);
  analogWrite(4,111);
  delay(200);
}
else {noTone(11); digitalWrite(2,HIGH); digitalWrite(3,HIGH); digitalWrite(4,HIGH);}

if(digitalRead(53) == LOW) {
  tone(11,800);
  analogWrite(2,100);
  analogWrite(3,200);
  analogWrite(4,96);
  delay(726);
  tone(11,200);
  analogWrite(2,80);
  analogWrite(3,80);
  analogWrite(4,123);
  delay(200);
  tone(11,1111);
  analogWrite(2,8);
  analogWrite(3,130);
  analogWrite(4,220);
  delay(200);
}
else {noTone(11); digitalWrite(2,HIGH); digitalWrite(3,HIGH); digitalWrite(4,HIGH);}

}

```

Done Saving.

Sketch uses 6458 bytes (2%) of program storage space. Maximum is 253952 bytes.  
Global variables use 222 bytes (2%) of dynamic memory, leaving 7970 bytes for local variables.



**Reality:**

