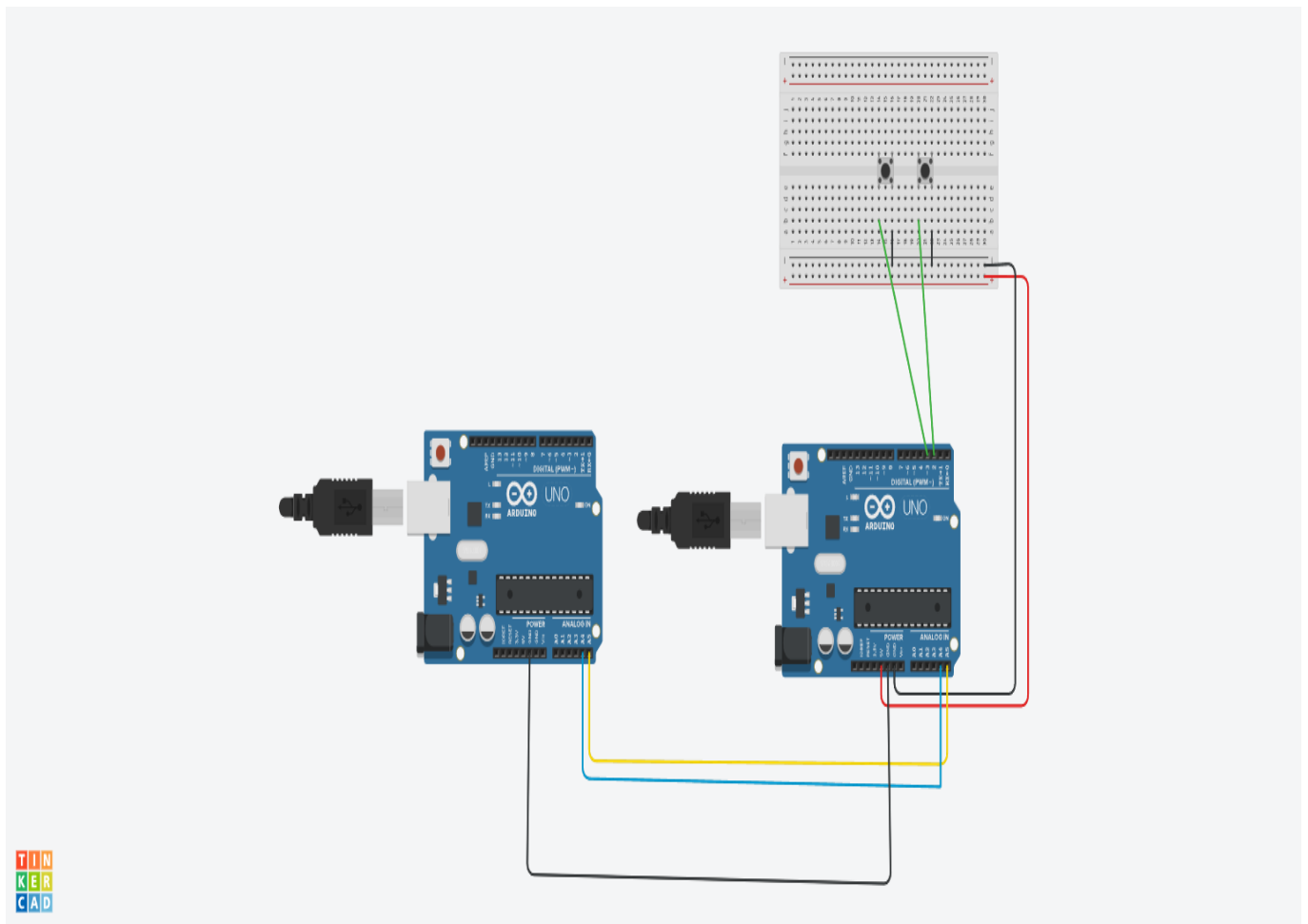
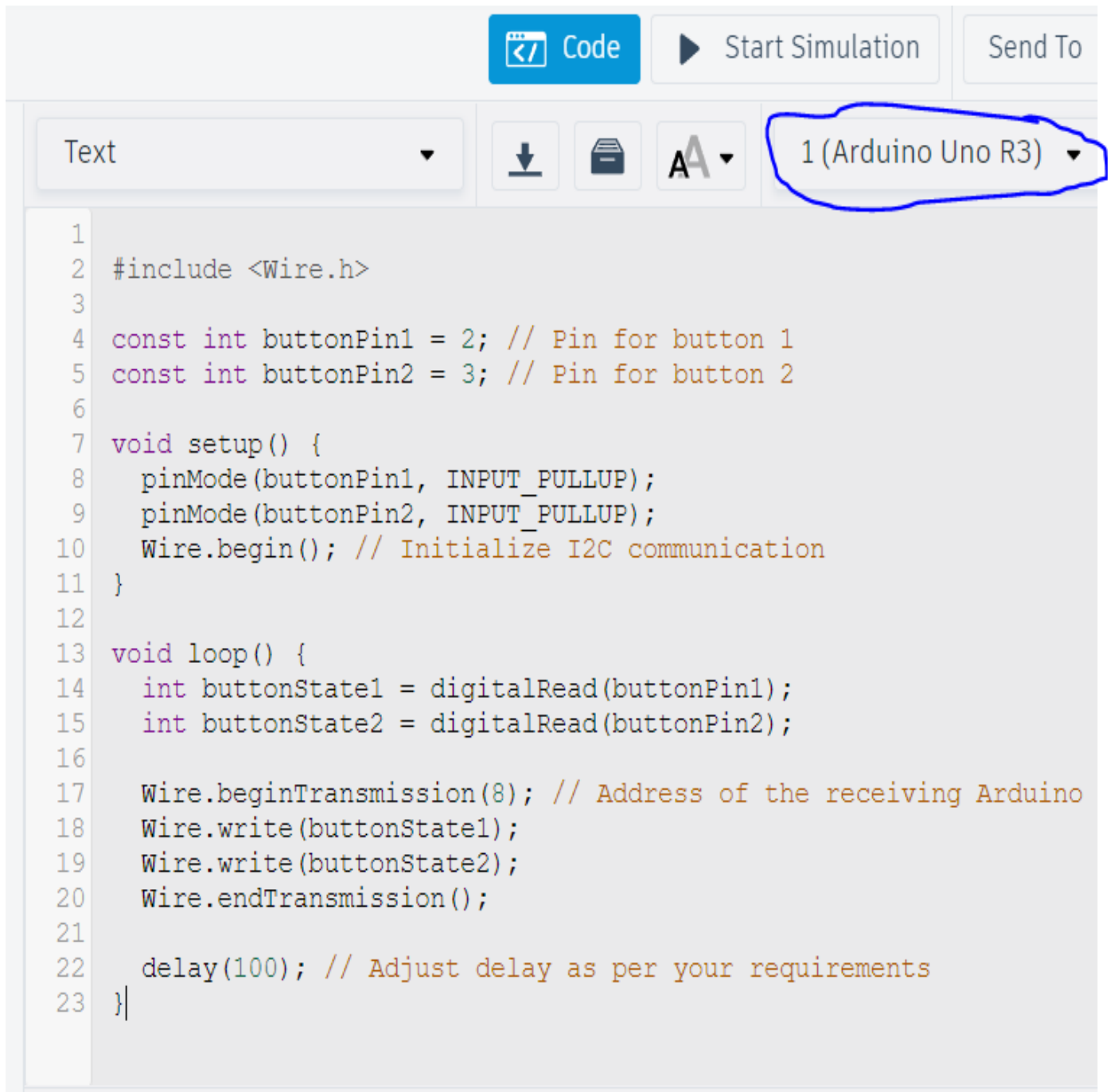


Task3

. Simulation on tinkercad:

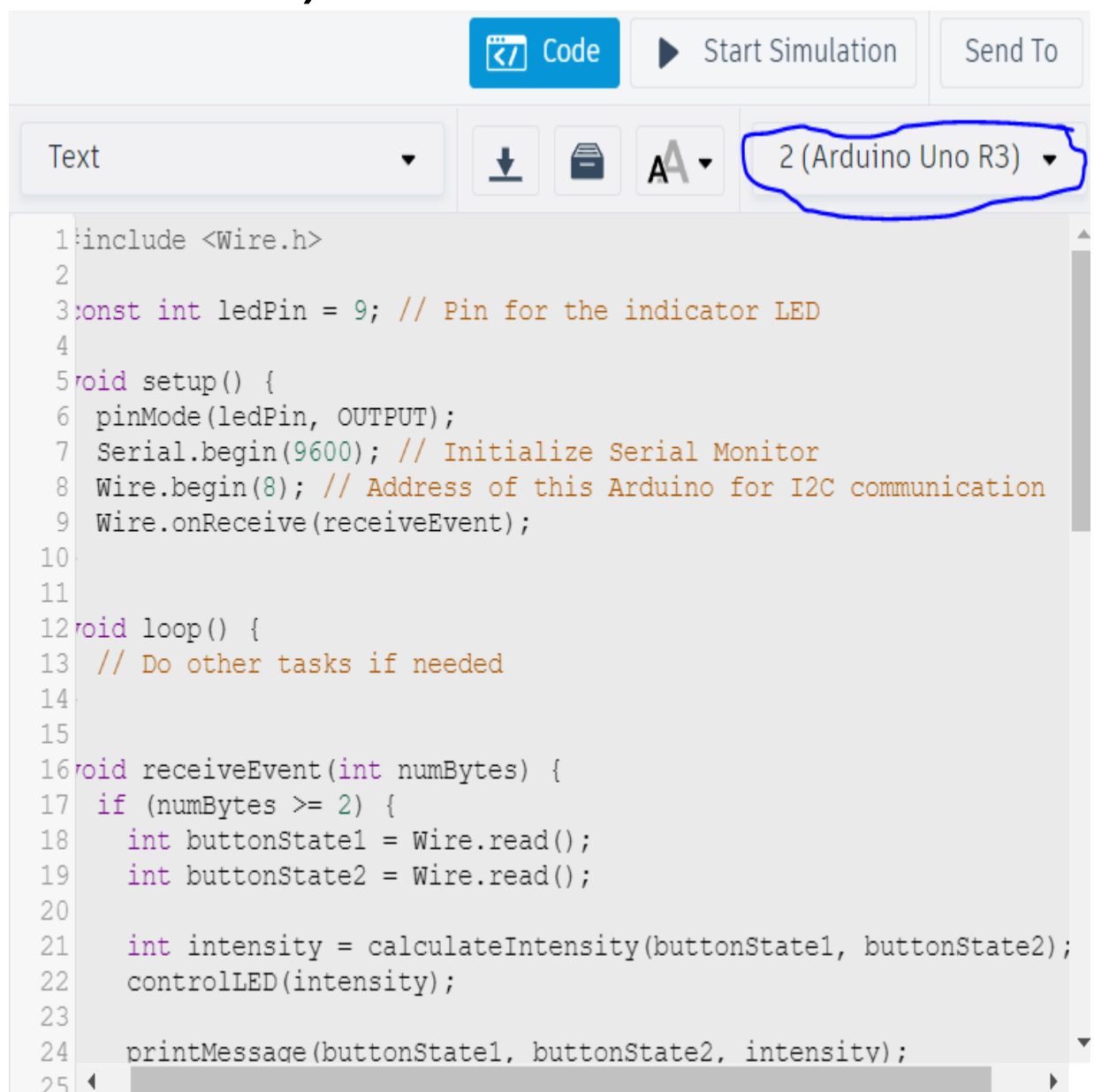


Code for the push button interface (Sender Arduino):



```
1
2 #include <Wire.h>
3
4 const int buttonPin1 = 2; // Pin for button 1
5 const int buttonPin2 = 3; // Pin for button 2
6
7 void setup() {
8     pinMode(buttonPin1, INPUT_PULLUP);
9     pinMode(buttonPin2, INPUT_PULLUP);
10    Wire.begin(); // Initialize I2C communication
11 }
12
13 void loop() {
14     int buttonState1 = digitalRead(buttonPin1);
15     int buttonState2 = digitalRead(buttonPin2);
16
17     Wire.beginTransmission(8); // Address of the receiving Arduino
18     Wire.write(buttonState1);
19     Wire.write(buttonState2);
20     Wire.endTransmission();
21
22     delay(100); // Adjust delay as per your requirements
23 }
```

- Code for receiving data and controlling LED intensity (Receiver Arduino):



The screenshot shows the Arduino IDE interface. At the top, there are buttons for 'Code', 'Start Simulation', and 'Send To'. Below these, there is a toolbar with a 'Text' dropdown, a download icon, a file icon, a font size icon, and a target board dropdown menu. The target board dropdown menu is highlighted with a blue circle and shows '2 (Arduino Uno R3)'. The code editor displays the following code:

```
1#include <Wire.h>
2
3const int ledPin = 9; // Pin for the indicator LED
4
5void setup() {
6  pinMode(ledPin, OUTPUT);
7  Serial.begin(9600); // Initialize Serial Monitor
8  Wire.begin(8); // Address of this Arduino for I2C communication
9  Wire.onReceive(receiveEvent);
10
11
12void loop() {
13  // Do other tasks if needed
14
15
16void receiveEvent(int numBytes) {
17  if (numBytes >= 2) {
18    int buttonState1 = Wire.read();
19    int buttonState2 = Wire.read();
20
21    int intensity = calculateIntensity(buttonState1, buttonState2);
22    controlledLED(intensity);
23
24    printMessage(buttonState1, buttonState2, intensity);
25
```



Code



Start Simulation

Send To

Text



2 (Arduino Uno R3)

```
24   printMessage(buttonState1, buttonState2, intensity);
25 }
26
27
28 int calculateIntensity(int buttonState1, int buttonState2) {
29   if (buttonState1 == 0 && buttonState2 == 0) {
30     return 0;
31   } else if (buttonState1 == 1 && buttonState2 == 0) {
32     return 50;
33   } else if (buttonState1 == 0 && buttonState2 == 1) {
34     return 75;
35   } else if (buttonState1 == 1 && buttonState2 == 1) {
36     return 100;
37   }
38
39
40 void controlled(int intensity) {
41   int ledIntensity = map(intensity, 0, 100, 0, 255);
42   analogWrite(ledPin, ledIntensity);
43
44
45 void printMessage(int buttonState1, int buttonState2, int intensity
46   Serial.print("Push buttons: B1=");
47   Serial.print(buttonState1);
```

 Code

 Start Simulation

Send To

Text ▼



2 (Arduino Uno R3) ▼

```
43 }
44
45 void printMessage(int buttonState1, int buttonState2, int intensi
46   Serial.print("Push buttons: B1=");
47   Serial.print(buttonState1);
48   Serial.print(", B2=");
49   Serial.println(buttonState2);
50
51   Serial.print("LED Intensity: ");
52   Serial.print(intensity);
53   Serial.println("% of Maximum");
54
55   if (intensity == 0) {
56     Serial.println("No message");
57   } else if (intensity == 50) {
58     Serial.println("Vector focused");
59   } else if (intensity == 75) {
60     Serial.println("Vector distracted");
61   } else if (intensity == 100) {
62     Serial.println("Glitch");
63     delay(10000);
64   }
65 }
66
```

 Serial Monitor

