Master Code

```
#include <Wire.h>

#define BUTTON1_PIN 2
#define BUTTON2_PIN 3

int buttonState1 = LOW;
int buttonState2 = LOW;
int lastButton1State = LOW;
int lastButton2State = LOW;
unsigned long lastDebounceTime1 = 0;
unsigned long lastDebounceTime2 = 0;
unsigned long debounceDelay = 50;

void setup() {
  Wire.begin();
  pinMode(BUTTON1_PIN, INPUT_PULLUP);
  pinMode(BUTTON2_PIN, INPUT_PULLUP);
}

void loop() {
  int reading1 = digitalRead(BUTTON1_PIN);
  int reading2 = digitalRead(BUTTON2_PIN);

  if (reading1 != lastButton1State) {
    lastDebounceTime1 = millis();
  }

  if (reading2 != lastButton2State) {
```

```
  lastDebounceTime2 = millis();
}

if ((millis() - lastDebounceTime1) > debounceDelay) {
  if (reading1 != buttonState1) {
    buttonState1 = reading1;
    if (buttonState1 == LOW) {
      // Button 1 was clicked
      Wire.beginTransmission(9);
      Wire.write(1);
      Wire.write(0);
      Wire.endTransmission();
    }
    else {
          Wire.beginTransmission(9);
      Wire.write(1);
      Wire.write(1);
      Wire.endTransmission();
    }
  }
}

if ((millis() - lastDebounceTime2) > debounceDelay) {
  if (reading2 != buttonState2) {
    buttonState2 = reading2;
    if (buttonState2 == LOW) {
      // Button 2 was clicked
      Wire.beginTransmission(9);
      Wire.write(2);
```

```
      Wire.write(0);
      Wire.endTransmission();
    }
    else {
          Wire.beginTransmission(9);
      Wire.write(2);
      Wire.write(1);
      Wire.endTransmission();
    }
  }
 }

 lastButton1State = reading1;
 lastButton2State = reading2;
 delay(10);
}
```

Slave Code

```
#include <Wire.h>

#define LED_PIN 13
int buttonState1 = 0;
int buttonState2 = 0;
int ledIntensity = 0;

void setup() {
  Wire.begin(9);
  Serial.begin(9600);
  pinMode(LED_PIN, OUTPUT);
}

void loop() {
  Wire.onReceive(receiveEvent);
  analogWrite(LED_PIN, ledIntensity);

  if (buttonState1 == LOW && buttonState2 == LOW) {
    Serial.println("No message");
  } else if (buttonState1 == HIGH && buttonState2 == LOW) {
    Serial.println("Vector focused");
  } else if (buttonState1 == LOW && buttonState2 == HIGH) {
    Serial.println("Vector distracted");
  } else if (buttonState1 == HIGH && buttonState2 == HIGH) {
    Serial.println("Glitch");
  }
}
```

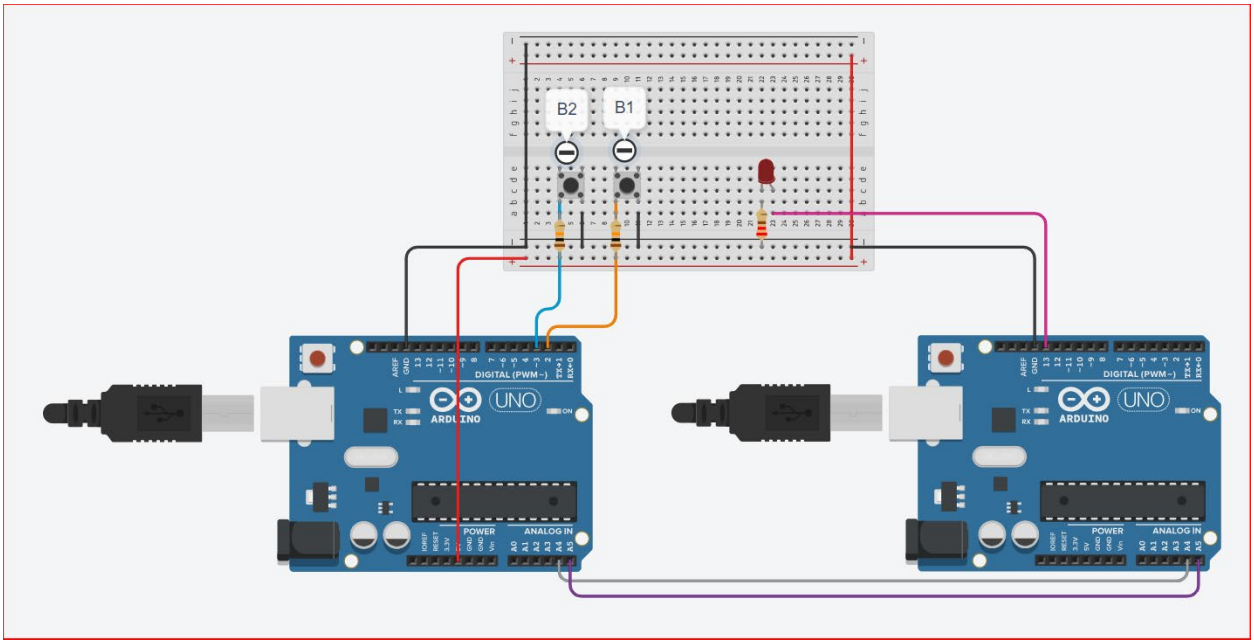
```
void receiveEvent(int bytes) {
  if (bytes >= 2) {
    int button = Wire.read();
    int state = Wire.read();

    if (button == 1) {
      buttonState1 = state;
    } else if (button == 2) {
      buttonState2 = state;
    }


    if (buttonState1 == LOW && buttonState2 == LOW) {
      ledIntensity = 0;
    } else if (buttonState1 == HIGH && buttonState2 == LOW) {
      ledIntensity = 128;
    } else if (buttonState1 == LOW && buttonState2 == HIGH) {
      ledIntensity = 192;
    } else if (buttonState1 == HIGH && buttonState2 == HIGH) {
      ledIntensity = 255;
    }

  }
}
```

Schematic


B2
B1
UNO
ARDUINO
DIGITAL (PWM~)
POWER
ANALOG IN
UNO
ARDUINO
DIGITAL (PWM~)
POWER
ANALOG IN