Reaktionszeit und Präzessions Testgerät

```cpp
const int Tasterpin = 2;
const int LEDpin = 5;
const int FotoRpin = A0;
const int Relaispin = 6;
int FotoSchwelle;


void Kalibieren() {
  Serial.println("Kalbrierung für Fotoresistor starten?\nZum starten bitte mit
\'OK\' antworten");
  String txt = "";
  while (!txt.startsWith("OK")) {
    if (Serial.available()) {
      txt = Serial.readString();
      Serial.println(txt);
    }
  }
  digitalWrite(LEDpin, LOW);
  digitalWrite(Relaispin, LOW);
  Serial.println("Starte Kalibartion\n LED aus Werte:");
  long sum = 0;
  for (int i = 0; i < 100; i++) {
    int x = analogRead(FotoRpin);
    sum += x;
    Serial.println(x);
  }
  int US = sum / 100;
  digitalWrite(LEDpin, HIGH);
  delay(100);
  Serial.println("LED an Werte:");
  sum = 0;
  for (int i = 0; i < 100; i++) {
    int x = analogRead(FotoRpin);
    sum += x;
    Serial.println(x);
  }
  digitalWrite(LEDpin, LOW);
  FotoSchwelle = (((sum / 100) - US) / 2) + US;
  Serial.print("Schwellenwert = ");
  Serial.println(FotoSchwelle);
  delay(100);
  digitalWrite(LEDpin, LOW);
  delay(100);
}
```

```cpp
void setup() {
  // put your setup code here, to run once:
  pinMode(Tasterpin, INPUT_PULLUP);
  pinMode(LEDpin, OUTPUT);
  pinMode(FotoRpin, INPUT);
  pinMode(Relaispin, OUTPUT);
  Serial.begin(115200);
  //Kalibieren();
}

void loop() {
  // put your main code here, to run repeatedly:
  if (Serial.available()) {
    String txt = Serial.readString();
    Serial.println(txt);

    if (txt.startsWith("start")) {
      Serial.println("Random delay gestartet");
      delay(random(500, 400));
      digitalWrite(LEDpin, HIGH);
      int t = millis();

      while (digitalRead(Tasterpin) == HIGH) {
      }

      t = millis() - t;
      Serial.println(t);
      digitalWrite(LEDpin, LOW);
    } else if (txt.startsWith("EMSstart")) {
      if (FotoSchwelle == 0) {
        Kalibieren();
      }
      Serial.println("Random delay gestartet");
      delay(random(500, 4000));
      digitalWrite(LEDpin, HIGH);
      long t = millis();
      bool realisiert = false;
      while (digitalRead(Tasterpin) == HIGH) {
        if (analogRead(FotoRpin) > FotoSchwelle && !realisiert) {
          digitalWrite(Relaispin, HIGH);
          Serial.print("Realisationszeit: ");
          Serial.println(millis() - t);
          realisiert = true;
        }
      }

      t = millis() - t;
      Serial.print("Reaktionszeit: ");
      Serial.println(t);
      digitalWrite(LEDpin, LOW);
```

```
      digitalWrite(Relaispin, LOW);
    } else if (txt.startsWith("Kalibrieren")) {
      Kalibieren();
    } else if (txt.startsWith("EMSon")) {
      digitalWrite(Relaispin, HIGH);
    } else if (txt.startsWith("EMSoff")) {
      digitalWrite(Relaispin, LOW);
    } else if (txt.startsWith("LEDon")) {
      digitalWrite(LEDpin, HIGH);
    } else if (txt.startsWith("LEDoff")) {
      digitalWrite(LEDpin, LOW);
    } else if (txt.startsWith("readTaster")) {
      Serial.print("TasterState: ");
      Serial.println(digitalRead(Tasterpin));
    } else if (txt.startsWith("readFotoR")) {
      Serial.print("FotoR: ");
      Serial.println(analogRead(FotoRpin));
    } else if (txt.startsWith("FotoSchwelle")) {
      Serial.print("FotoSchwelle: ");
      Serial.println(FotoSchwelle);
    } else if (txt.startsWith("FingerSteuern")) {
      Serial.print("Fingersteuerung aktiv (knopf zum Steuern drücken): ");
      while (true) {
        if (digitalRead(Tasterpin) == LOW) {
          digitalWrite(Relaispin, HIGH);
        } else {
          digitalWrite(Relaispin, LOW);
        }
        if (Serial.available()) {
          String txt = Serial.readString();
          Serial.println(txt);

          if (txt.startsWith("stop")) {
            break;
          }
        }
      }
    }
  }
}
}
```