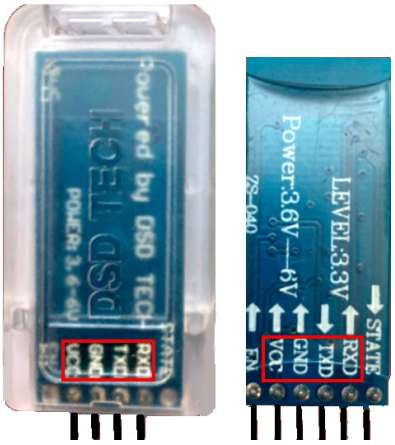
Eine Ampel soll mit Hilfe eines Bluetooth-Moduls "per Hand" geschaltet werden.

Beispiele für Bluetooth-Module HM-10

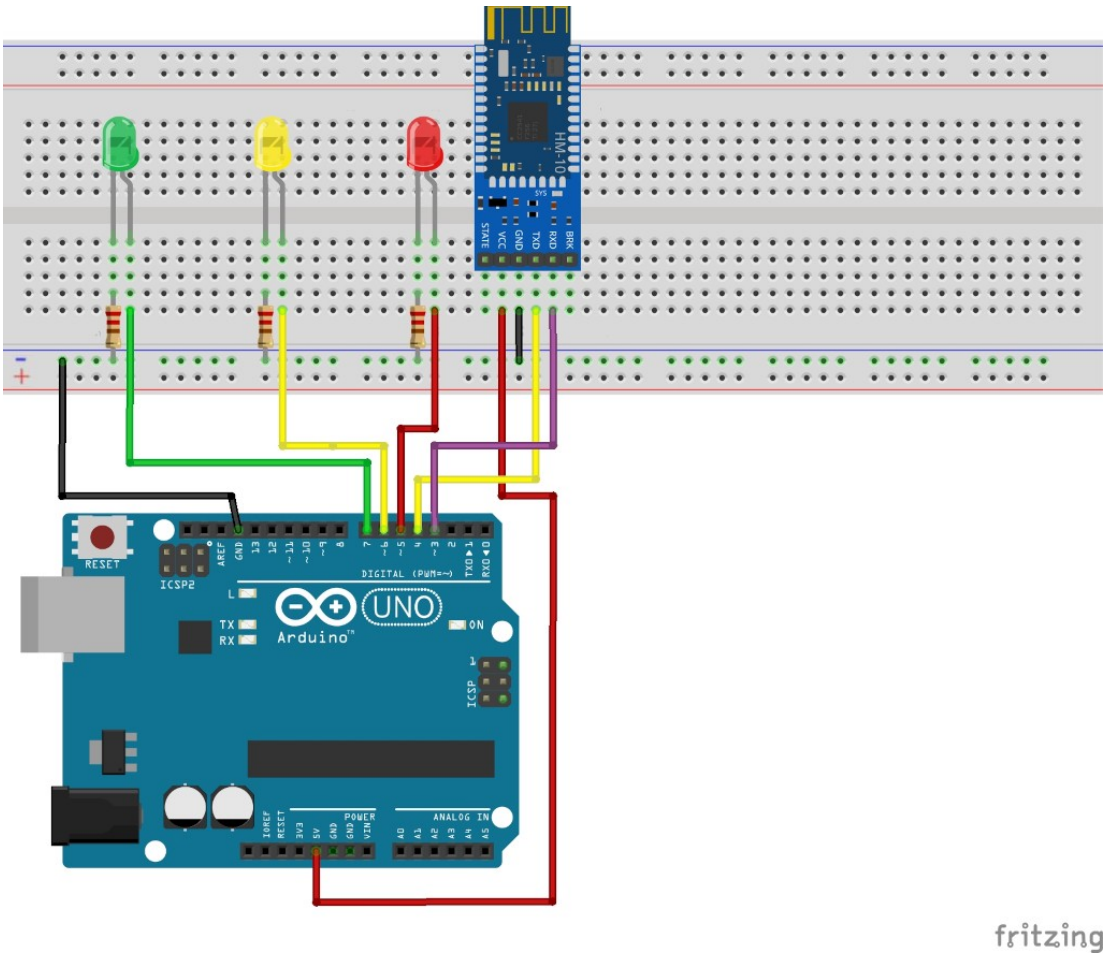


Das Bluetooth-Modul HM-10 arbeitet mit BLE (Bluetooth Low Energy) und kann mit Android-Betriebssystemen und iOS umgehen.

Benötigte Bauteile:

- 3 LEDs (rot, gelb, grün)
- 3 Widerstände 220 Ω
- Bluetooth-Modul HM-10
- Leitungsdrähte

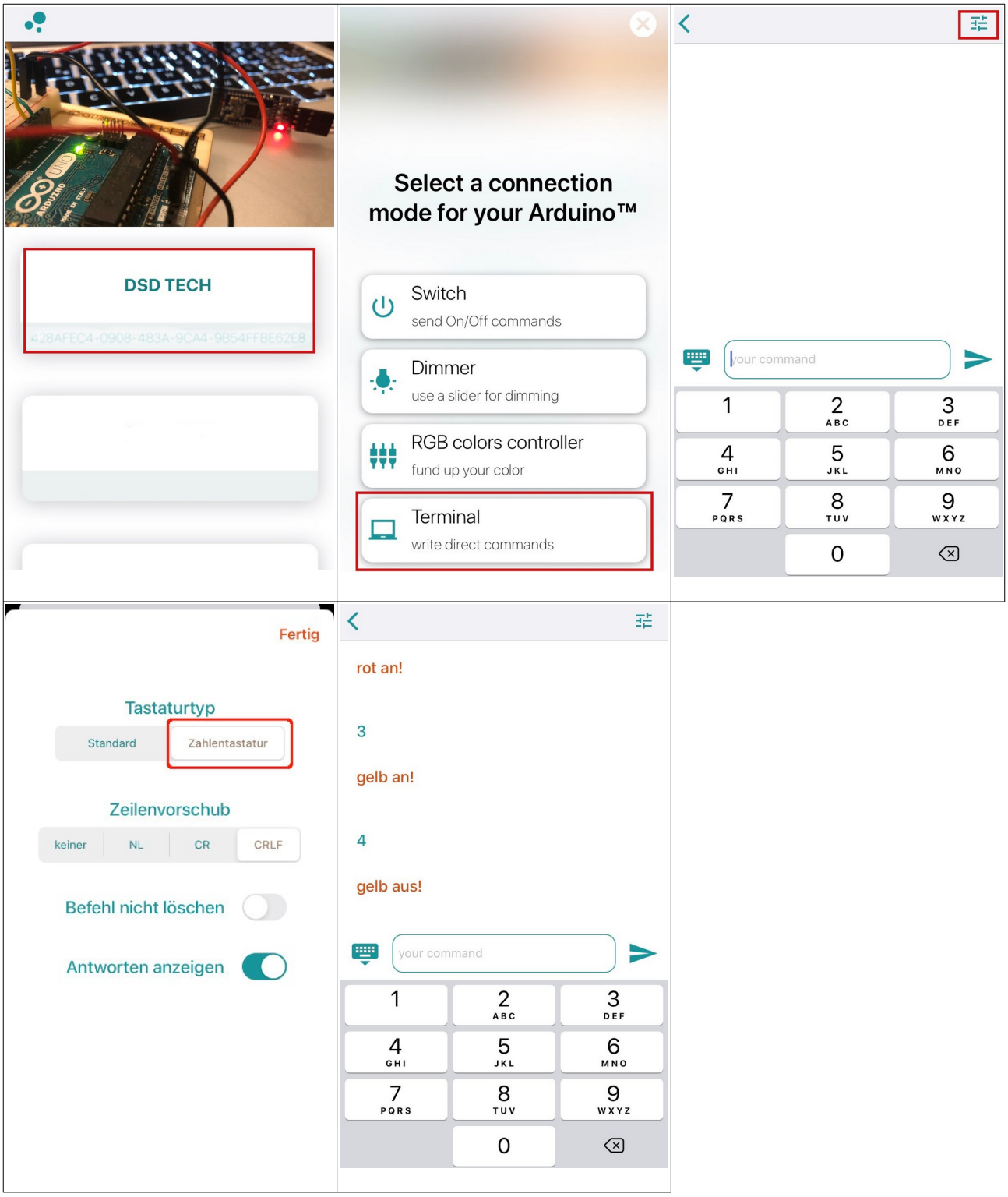
Baue die Schaltung auf.



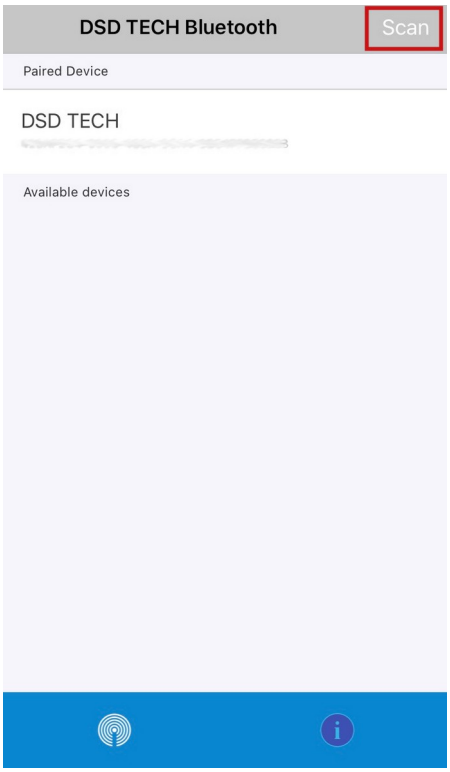
Wenn das Bluetooth-Modul korrekt angeschlossen ist, beginnt die Status-LED zu blinken. Um das Modul nutzen zu können, musst du eine App auf dem Smartphone installieren.

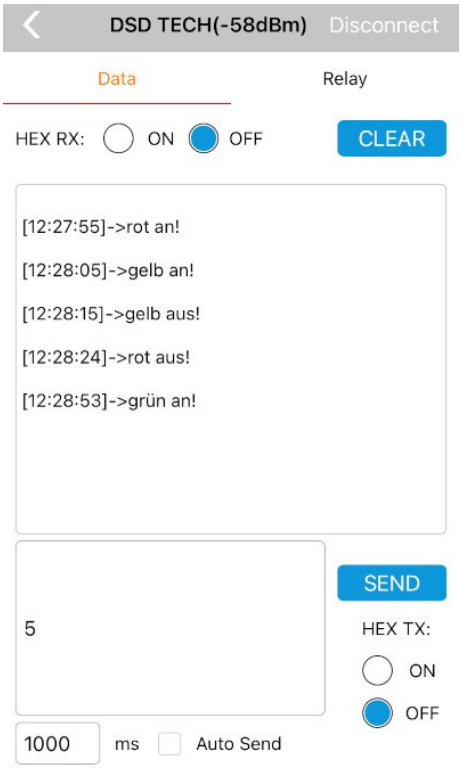
Für Android und iOS verfügbare Apps:

Arduino Bluetooth Controller



DSD-Tech Bluetooth





Die eigentlich vorgesehenen Pins RX (0) und TX (1) werden beim Hochladen verwendet. Deshalb müsste das Bluetooth-Modul beim Hochladen herausgezogen und anschließend wieder eingesteckt werden. Über die Bibliothek SoftwareSerial können die Pins RX und TX anderen digitalen Pins zugeordnet werden. Das Herausziehen und Wiedereinstecken entfällt dann.

Im Kopf des Programms werden die Variablen definiert. Die eingebundene Bibliothek SoftwareSerial ist eine Standardbibliothek, sie muss nicht installiert werden.

```
// über Bluetooth vom Smartphone gesendeter Wert
// dieser Wert wird im Programm ausgewertet
char btDaten;

int GRUEN = 7;
int GELB = 6;
int ROT = 5;

# include <SoftwareSerial.h>

// Zuordnung von RX und TX: TX -> 2, RX -> 3
SoftwareSerial BTSerial(2, 3);
```

Der setup-Teil definiert lediglich die pinModes für die LEDs und startet den über SoftwareSerial zugewiesenen Seriellen Monitor BTSerial:

```
void setup()
{
  BTSerial.begin(9600);
  pinMode(ROT, OUTPUT);
  pinMode(GELB , OUTPUT);
  pinMode(GRUEN, OUTPUT);
}
```

Den jeweiligen LEDs wurden in der App Werte zugewiesen, die im Programm mit BTSerial`.read()` gelesen werden:

Aktion	Taste
rot ein	1
rot aus	2
gelb ein	3
gelb aus	4
grün ein	5
grün aus	6

Im loop-Teil wird den Werten jeweils eine Aktion zugeordnet:

```
void loop()
{
  if (BTSerial.available())
  {
    // vom Smartphone gesendeten Wert lesen
    btDaten = BTSerial.read();

    // rot ein
    if (btDaten == '1')
    {
      digitalWrite(ROT, HIGH);

      // gelesenen Wert in der App anzeigen
      BTSerial.println("rot an!");
    }

    // rot aus
    if (btDaten == '2')
    {
      digitalWrite(ROT, LOW);
      BTSerial.println("rot aus!");
    }
```

```
    // gelb an
    if (btDaten == '3')
    {
      digitalWrite(GELB, HIGH);
      BTSerial.println("gelb an!");
    }

    // gelb aus
    if (btDaten == '4')
    {
      digitalWrite(GELB, LOW);
      BTSerial.println("gelb aus!");
    }
    // grün an
    if (btDaten == '5')
    {
      digitalWrite(GRUEN, HIGH);
      BTSerial.println("gr\u00fcn an!");
    }

    // grün aus
    if (btDaten == '6')
    {
      digitalWrite(GRUEN, LOW);
      BTSerial.println("gr\u00fcn aus!");
    }
  }
}
```