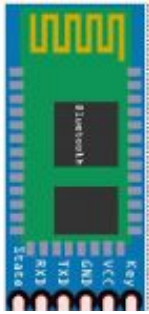


## Ampel mit Bluetooth schalten



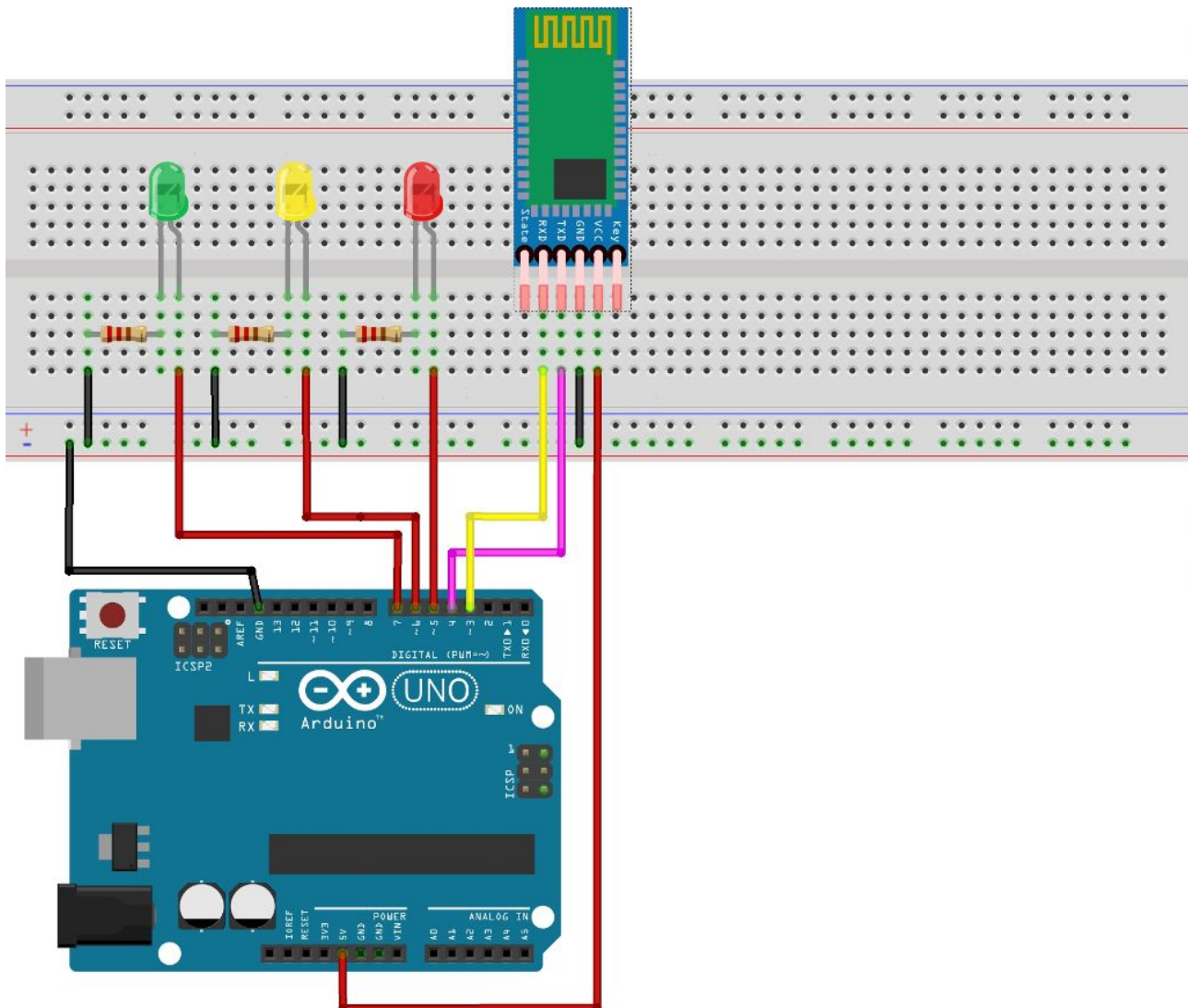
Die Bluetooth-Module HC-05 und HM-10 unterscheiden sich in der Verwendung von Apps für Android und iOS.  
Die Programmierung ist identisch.

**Der HC-05 und funktioniert nur mit Smartphones mit Android-Betriebssystem, der HM-10 kann mit dem Android-Betriebssystem und iOS umgehen. Apple-Geräte verwenden BLE (Bluetooth Low Energy), mit dem der HC-05 nicht arbeiten kann.**

### Benötigte Bauteile:

- 3 LEDs (rot, gelb, grün)
- 3 Widerstände > 100 Ω
- Bluetooth-Modul HC05/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.

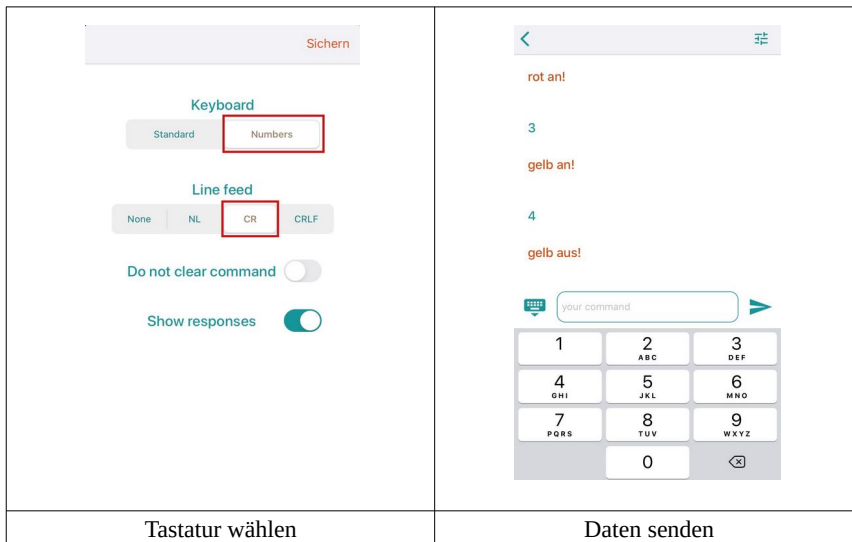
Für Android und iOS verfügbare Apps:

### DSD-Tech Bluetooth

<p>Bluetooth-Modul scannen</p>	<p>Daten senden</p>

### Arduino Bluetooth Controller

<p>Bluetooth-Modul scannen</p>	<p>Terminal auswählen</p>	<p>Einstellungen</p>



Tastatur wählen

Daten senden



**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 -> 3, RX -> 4
SoftwareSerial BTSerial(3, 4);
```

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);
      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')
    {
      . . .
    }
  }
}

```