

Eine sensorgesteuerte Ampelanlage

Stellt euch vor, eure Idee zur Errichtung einer Ampelanlage, die sich selbsttätig den jeweils herrschen Verkehrsbedingungen anpassen kann, sei im Verkehrsausschuss auf Interesse gestoßen.

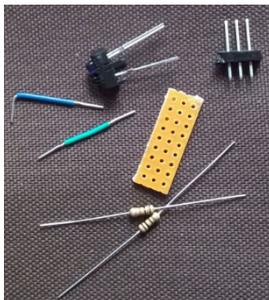
Allerdings wurde der Gedanke nicht recht verstanden, dass statt der Taster Sensoren eingesetzt werden sollten. Viele im Ausschuss haben das Modell eins zu eins auf die Straßensituation übertragen und fanden es gefährlich, wenn man auf einer viel befahrenen Straße aussteigen müsse, um Grün anzufordern.

Um dieses Missverständnis auszuräumen, erklärt ihr euch bereit, euer Modell der Ampelanlage entsprechend umzurüsten und es bei der nächsten Sitzung erneut vorzustellen.

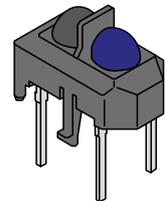
Die Aufgabe

Das Ziel besteht darin, in dem bestehenden Modell einer Dreifarben-Verkehrsampel die Taster gegen Sensoren auszutauschen, die das Annähern eines Fahrzeugs erkennen können. Sie sollen die gleichen Vorgänge wie die mechanischen Schalter auslösen.

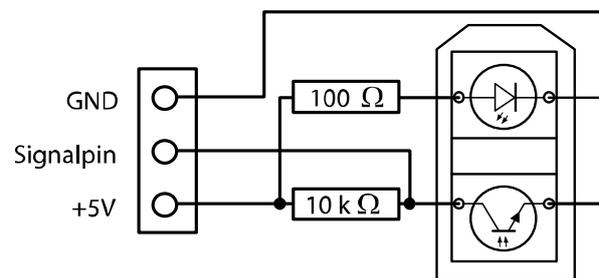
Hinweise



Als Sensor empfiehlt sich eine Kombination aus Infrarot-Transistor (Empfänger) und Infrarot-LED (Sender). Besonders preisgünstig ist der TCRT5000, der mit Widerständen von 100 Ohm und 10 Kiloohm zu einem einfach zu handhabenden Modul verbunden werden kann.



Der Vorteil bei einer Verwendung solcher Module liegt in der Übersichtlichkeit, die etwas leidet, wenn alle Teile der Ampelanlage



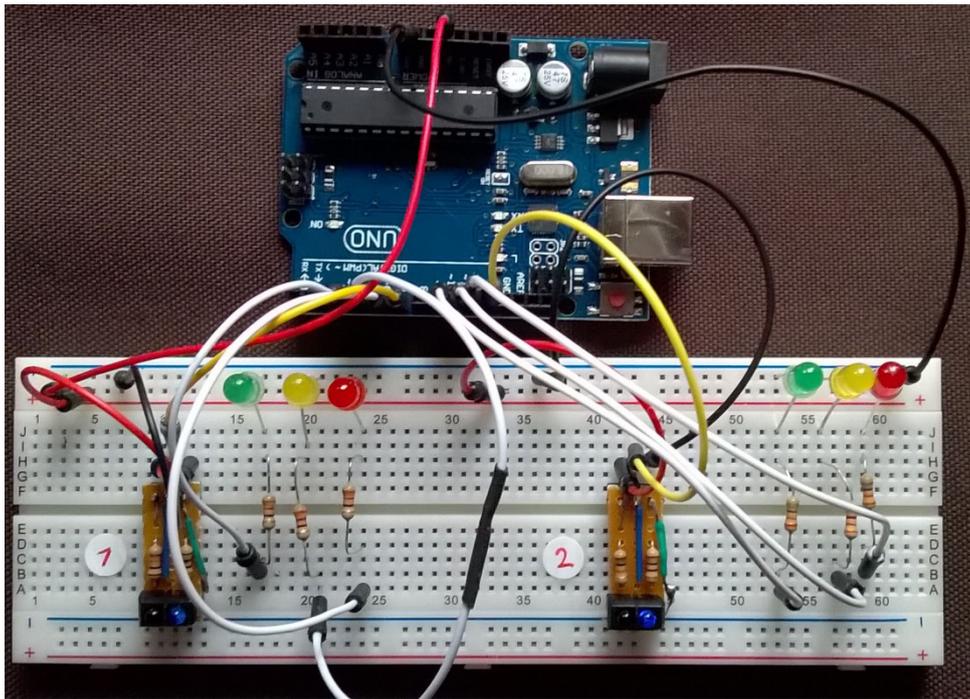
einzeln über Steckverbindungen zwischen dem Arduino und einem

Breadboard realisiert werden.

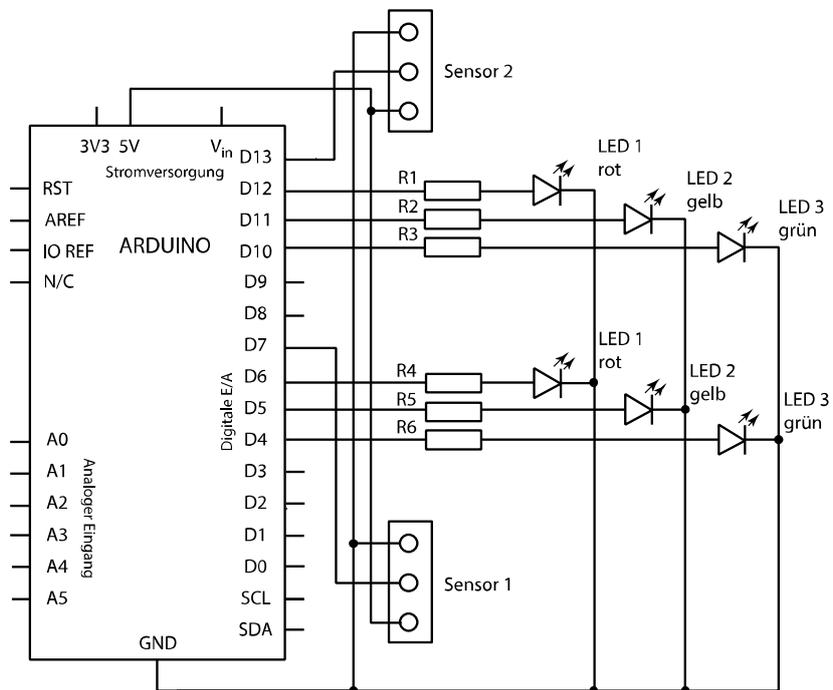
Benötigte Materialien

- Vorgänger-Ampel-Modell
- Computer mit der Arduino-Software
- Arduino UNO
- USB-Kabel
- Lötmaterialien
- zwei 100Ω Widerstände
- zwei 10kΩ Widerstände
- zwei TCRT5000
- zwei kurze Klingeldrahtstücke
- Abschnitte einer Lochrasterplatte

Lösung



Schaltplan



Programmcode

```
int IRsensor1 = 7;
int IRsensor2 = 13;
#define zweiRot 6
#define zweiGelb 5
#define zweiGruen 4
#define einsRot 12
#define einsGelb 11
#define einsGruen 10
#define pausenZeit 500
boolean richtungEins = true;

int flussZeit = 10000;
int wartezeit = 2000;
void setup() {

    pinMode(IRsensor1, INPUT);
    pinMode(IRsensor2, INPUT);
    pinMode(zweiRot, OUTPUT);
    pinMode(zweiGelb, OUTPUT);
    pinMode(zweiGruen, OUTPUT);
    pinMode(einsRot, OUTPUT);
    pinMode(einsGelb, OUTPUT);
    pinMode(einsGruen, OUTPUT);

    digitalWrite(zweiRot, LOW);
    digitalWrite(zweiGelb, LOW);
    digitalWrite(zweiGruen, HIGH);
    digitalWrite(einsRot, HIGH);
    digitalWrite(einsGelb, LOW);
    digitalWrite(einsGruen, LOW);
}
void loop() {

    if (digitalRead(IRsensor1) == LOW) {
        if (richtungEins != true) {
            richtungEins = true;
            delay(flussZeit);
            digitalWrite(einsGruen, LOW);
            digitalWrite(einsGelb, HIGH);
            delay(wartezeit);
            digitalWrite(einsGelb, LOW);
            digitalWrite(einsRot, HIGH);
            delay(wartezeit);
            for (int a = 0; a < 5; a++) {
                digitalWrite(zweiGelb, LOW);
                delay(pausenZeit);
                digitalWrite(zweiGelb, HIGH);
                delay(pausenZeit);
            }
            digitalWrite(zweiGelb, LOW);
            digitalWrite(zweiRot, LOW);
            digitalWrite(zweiGruen, HIGH);
        }
    }

    if (digitalRead(IRsensor2) == LOW) {
        if (richtungEins == true) {
            richtungEins = false;
            delay(flussZeit);
            digitalWrite(zweiGruen, LOW);
            digitalWrite(zweiGelb, HIGH);
        }
    }
}
```

```
    delay(Wartezeit);
    digitalWrite(zweiGelb, LOW);
    digitalWrite(zweiRot, HIGH);
    delay(Wartezeit);
    for (int a = 0; a < 5; a++) {
        digitalWrite(einsGelb, LOW);
        delay(pausenZeit);
        digitalWrite(einsGelb, HIGH);
        delay(pausenZeit);
    }
    digitalWrite(einsGelb, LOW);
    digitalWrite(einsRot, LOW);
    digitalWrite(einsGruen, HIGH);
}
}
```