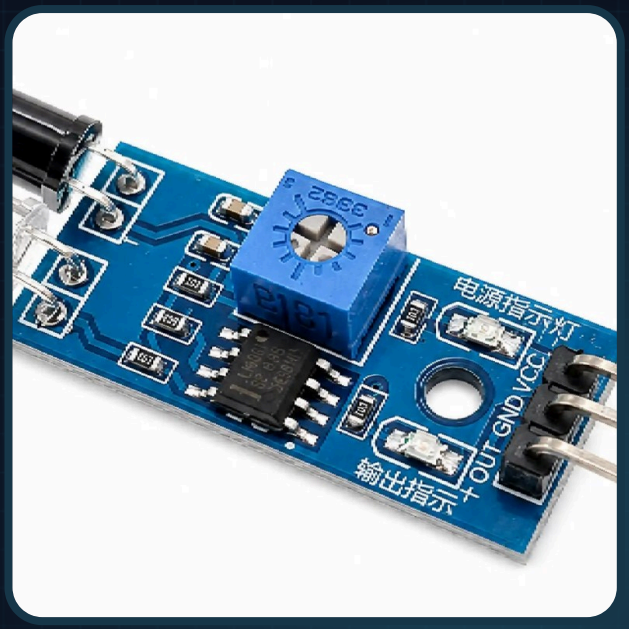
INFRAROT-HINDERNISERKENNUNG

FC-51
IR-Hindernis

Berührungslose Hinderniserkennung mit einstellbarer Reichweite. Erste Erkennung in rund 15 Minuten – mit UNO R3 und drei Kabeln.



REICHWEITE
ca. 2 … 30 cm

AUSGANG
LOW = Hindernis

VERSORGUNG
3,3 … 5 V DC

EINSTELLUNG
per Poti

01
Verkabeln
3 Leitungen zum UNO

02
Einrichten
IDE und Treiber

03
Hochladen
Beispiel-Sketch

04
Messen
Serieller Monitor

01 Auf einen Blick

Der FC-51 strahlt Infrarotlicht ab und registriert die Reflexion. Kommt etwas in Reichweite, schaltet der Ausgang auf LOW – ohne Bibliothek, direkt an einem Digitalpin auswertbar. Die Auslösedistanz stellst du am Poti ein.

Technische Daten

Prinzip	**IR-Sender + IR-Empfänger**
Reichweite	**ca. 2 bis 30 cm, einstellbar**
Ausgang	**digital, LOW bei Hindernis**
Komparator	**LM393**
Versorgungsspannung	**3,3 bis 5 V DC**
Anzeigen	**Power-LED, Hindernis-LED**

Lieferumfang

1 × **FC-51 IR-Hindernissensor-Modul**

Kein Entfernungsmesser
Der FC-51 meldet nur „da ist etwas“. Wie weit er auslöst, hängt stark von Farbe und Oberfläche ab.

Schwarz und Sonne
Mattes Schwarz schluckt Infrarot und wird schlecht erkannt. Direktes Sonnenlicht kann Fehlauslösungen verursachen.

Pinbelegung

PIN	FUNKTION	SPANNUNG	HINWEIS
OUT	Digitalausgang DIGITAL	LOW bei Hindernis	Ruhepegel HIGH.
GND	Masse GND	0 V	Gemeinsame Masse.
VCC	Versorgung POWER	3,3–5 V DC	Am UNO an 5V.

Die Stiftleiste ist mit OUT · GND · VCC beschriftet.

Was du zusätzlich brauchst

UNO R3 mit USB-C
ATmega328P, CH340G-USB-Wandler
MKR-UNO-R3-013

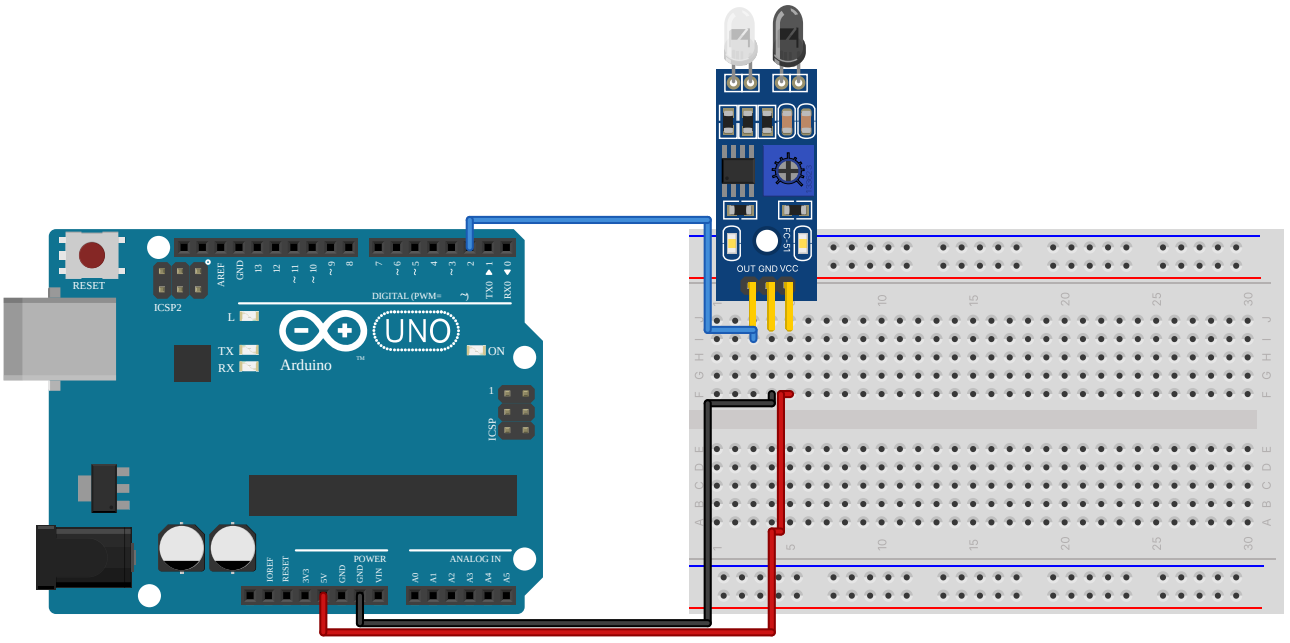
Breadboard + Jumper-Set
400 Kontakte, 40 Dupont-Kabel
MKR-SET-BB400-JUMPER

USB-C-Datenkabel
Kein reines Ladekabel – es muss Daten übertragen

02 Verkabeln

Drei Leitungen genügen. Zieh vorher das USB-Kabel ab – verkabelt wird immer stromlos.

STECKPLATINEN-ANSICHT



Verbindungen

SENSOR		UNO R3	LEITUNG
● OUT	→	D2	Hindernis
● GND	→	GND	Masse
● VCC	→	5V	Versorgung

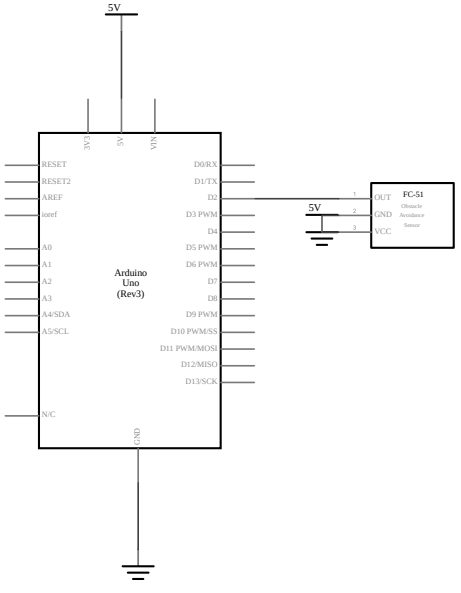
Reichweite einstellen

Hindernis im gewünschten Abstand halten und das Poti drehen, bis die zweite LED auf dem Modul gerade angeht.

Erst prüfen, dann einstecken

Vertauschte VCC und GND können das Modul beschädigen. Erst alle Leitungen prüfen, dann USB anstecken.

SCHALTPLAN



Verdrahtung erstellt mit Fritzing

03 Einrichten & Hochladen

Schritte 1 bis 3 machst du nur ein einziges Mal – danach ist dein UNO R3 für alle weiteren Projekte startklar.

1 Arduino IDE installieren

Lade die aktuelle Arduino IDE 2 von `arduino.cc/en/software` herunter und installiere sie.

2 Treiber für den CH340G prüfen

Der UNO R3 nutzt den CH340G als USB-Seriell-Wandler. Aktuelle Systeme erkennen ihn meist automatisch. Taucht in Schritt 3 kein Port auf, installiere den Treiber „CH341SER“ des Chipherstellers WCH (`wch-ic.com`).

3 Board und Port wählen

UNO per USB-C anschließen. Dann `Werkzeuge → Board → Arduino AVR Boards → Arduino Uno` wählen und unter `Werkzeuge → Port` den neu aufgetauchten COM-Port (Windows) bzw. `/dev/ttyUSB0` (Linux) / `/dev/cu.wchusbserial…` (macOS).

4 Keine Bibliothek nötig

Dieser Sketch kommt ohne Zusatzbibliothek aus – du kannst direkt mit Schritt 5 weitermachen.

5 Sketch hochladen

Neuen Sketch anlegen, den Code von Seite 5 einfügen und mit dem Pfeil-Symbol hochladen (`Strg` + `U`). Unten erscheint „Hochladen abgeschlossen“.

6 Seriellen Monitor öffnen

`Strg` + `Umschalt` + `M`, unten rechts **9600 Baud** einstellen. Bei jeder Änderung erscheint eine Zeile – die LED „L“ zeigt ein Hindernis an.

BEISPIELAUSGABE · SERIELLER MONITOR · 9600 BAUD

```
MAKEROO FC-51-Test startet ...
Hindernis erkannt!
Weg frei
Hindernis erkannt!
```

i Schnelltest

Halte ein weißes Blatt Papier vor die Dioden und zieh es langsam weg – die Ausgabe wechselt zwischen „Hindernis“ und „Weg frei“.

04 Der erste Sketch

Liest den Ausgang, zeigt ein Hindernis auf der Board-LED an und meldet jede Änderung.

MAKEROO_FC51_Start.ino

```
1  // MAKEROO Getting Started - FC-51 IR-Hindernissensor am UNO R3
2  // Meldet, sobald ein Hindernis in Reichweite ist.
3
4  const int SENSOR_PIN = 2;  // OUT an D2
5  bool letzterZustand = HIGH;
6
7  void setup() {
8    Serial.begin(9600);
9    pinMode(SENSOR_PIN, INPUT);
10   pinMode(LED_BUILTIN, OUTPUT);
11   Serial.println(F("MAKEROO FC-51-Test startet ..."));
12 }
13
14 void loop() {
15   bool zustand = digitalRead(SENSOR_PIN);  // LOW = Hindernis
16   digitalWrite(LED_BUILTIN, zustand == LOW);
17
18   if (zustand != letzterZustand) {
19     if (zustand == LOW) {
20       Serial.println(F("Hindernis erkannt!"));
21     } else {
22       Serial.println(F("Weg frei"));
23     }
24     letzterZustand = zustand;
25   }
26   delay(20);
27 }
```

Zeile 4
Pin
OUT hängt an D2. Anderer Pin? Nur diese Zahl ändern.

Zeile 15
Logik
Achtung: LOW bedeutet Hindernis, HIGH bedeutet freie Bahn.

Zeile 18
Nur Änderungen
Ausgabe nur beim Wechsel – so bleibt der Monitor lesbar.

Geprüft: kompiliert für Arduino Uno (2.2 kB Flash = 7 %, 188 Byte RAM).

05 Wenn es nicht klappt

SYMPTOM	LÖSUNG
Meldet immer Hindernis	Poti zurückdrehen. Nichts direkt vor die Dioden halten, keine helle Lampe in der Nähe.
Erkennt nichts	Poti weiter aufdrehen. Helle, glatte Flächen testen – Schwarz wird schlecht erkannt.
Logik vertauscht	Normal: LOW = Hindernis. Im Sketch ist das bereits berücksichtigt.
Kein Port in der IDE	Anderes USB-C-Kabel probieren – viele Kabel laden nur. Danach CH340-Treiber installieren (Schritt 2).
Nur wirre Zeichen im Monitor	Baudrate im seriellen Monitor auf 9600 stellen.

06 Wie geht's weiter?

Mit ESP32 & ESPHome

VCC an **3V3**, OUT an z. B. **GPIO27**. In ESPHome:

```
binary_sensor:
  - platform: gpio
    pin:
      number: GPIO27
      inverted: true
    name: "Hindernis"
```

Projektideen

Roboterauto
Hindernissen automatisch ausweichen

Stückzähler
Teile auf einem Förderband zählen

Berührungsloser Schalter
Hand davor schaltet das Licht

Endlagenschalter
Position ohne mechanischen Kontakt erkennen

Passt dazu

130er DC-Motor (3er)
Last für erste Messungen
`MKR-130-MOTOR-033`

2x SG90 Servo 360°
Bewegung per PWM
`MKR-SG90-360G-003`

WS2812 LED-Ring
12 RGB-LEDs als Anzeige
`MKR-WS2812-RING12-67`

ESP32 DevKit USB-C
WLAN, ESPHome, Home Assistant
`MKR-ESP32-005`

Sicherheitshinweise. Elektronikmodul für Ausbildungs-, Entwicklungs- und Experimentierzwecke. Nur mit Kleinspannung betreiben, niemals an Netzspannung. Nicht geeignet für Kinder unter 14 Jahren; Nutzung nur unter Aufsicht. Elektroaltgeräte gehören nicht in den Hausmüll – Rückgabe kostenlos bei kommunalen Sammelstellen.

Verantwortlicher Wirtschaftsakteur (GPSR): Alexander Prez · makeroo · Hinterm Bahnhof 4 · 90513 Zirndorf · support@makeroo.de
Fragen? Schreib uns mit einem Foto deiner Verkabelung.
Arduino ist eine Marke der Arduino SA und dient nur der Kompatibilitätsangabe.