

Optische Alarmanlage

Im Video haben wir eine Schaltung aufgebaut, die eine LED einschaltet, wenn es dunkel wird.

Du sollst das jetzt zu einer Alarmanlage umbauen. Für den Alarm beschränken wir uns aber auf eine LED, damit unsere Nachbarn nicht unter unseren Versuchen leiden müssen. Die Aufgabe enthält drei Stufen, jede etwas schwieriger als die Vorangehende.

Alle drei Stufen lassen sich mit den Befehlen realisieren, die wir schon besprochen haben. Falls du bereits bessere Befehle kennst, darfst du sie selbstverständlich anwenden.

Stufe 1

Ein Einbrecher kommt nachts in den Raum mit einer Taschenlampe. Deine Schaltung muss das Licht bemerken und den Alarmzustand durch das Einschalten einer LED anzeigen. Wenn der Alarm einmal ausgelöst ist, bleibt der Alarmzustand erhalten.

Stufe 2

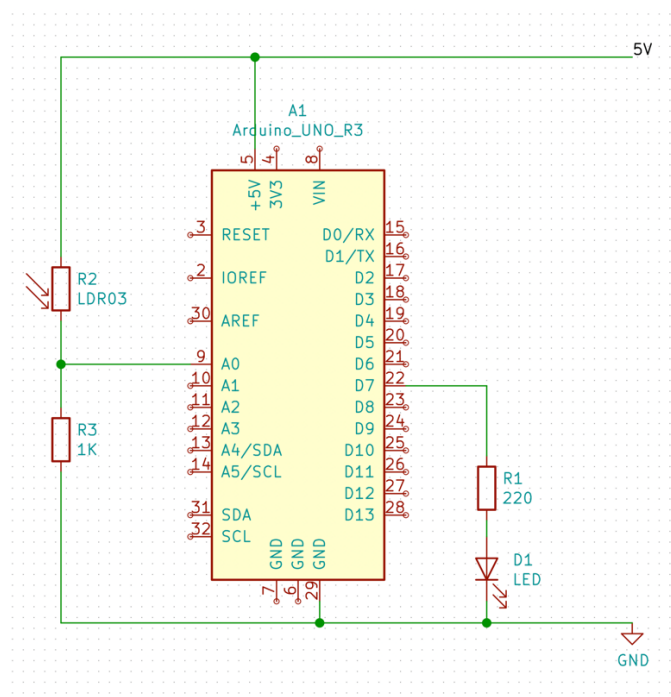
Der Alarm soll nicht durch eine ständig leuchtende LED angezeigt werden. Die LED soll nach der Auslösung des Alarms dauerhaft blinken.

Stufe 3

Unser Raum hat ein Fenster. Jedes Mal, wenn ein Auto vorbei fährt und seine Scheinwerfer in Richtung Fenster zeigen, gibt es einen Fehlalarm. Überlege, was du tun kannst. Mit unseren Möglichkeiten findest du keinen absolut zuverlässige Lösung, aber du kannst die Fehlalarme doch sehr stark reduzieren.

Wenn du alle drei Stufen erfolgreich bewältigst, dann hast schon viel gelernt und kannst dein Wissen auch in neuen Situationen einsetzen und neu kombinieren. Oder du hast schon Vorkenntnisse, dann wird diese Aufgabe für dich keine grosse Herausforderung darstellen.

Falls du Probleme hast, lass dir Zeit. Mache eine kleine Pause und versuche es nochmals. Wenn es dann immer noch nicht klappt, kein Problem! Die Lösung kommt in einem separaten Video, das gleichzeitig freigeschaltet wird. Ich werde dort sehr ausführlich eine mögliche Lösung vorstellen.



Lösung

Stufe 1

Das Programm ist in zwei Teile aufgeteilt. Wir testen, ob ein Alarmzustand vorliegt und zeigen diesen über die LED an.

Wenn ein Alarmzustand vorliegt, wird dies in der globalen Variablen **alarmzustand** gespeichert. Der Test erfolgt in der Funktion **alarmTest()**. **alarmzustand** wird hier nie zurückgesetzt, sondern nur gesetzt, falls ein Licht detektiert wird.

In **alarm()** wird die LED eingeschaltet, falls ein Alarmzustand herrscht, sonst wird sie ausgeschaltet.

```
const int inputPin = A0;
const int ledPin = 7;

int alarmzustand; //0 = kein Alarm, 1 = Alarm

void setup() {
  pinMode(ledPin,OUTPUT);

  // Grundzustand
  alarmzustand = 0;           // Am Anfang haben wir keinen Alarm
}

void alarmTest() {
  if (analogRead(inputPin) > 500) alarmzustand = 1;
}

void alarm() {
  if (alarmzustand == 1) {    // wir haben einen Alarmzustand!
    digitalWrite(ledPin,HIGH);
  } else {                   // nein, alles ruhig
    digitalWrite(ledPin,LOW);
  }
}

void loop() {
  alarmTest();
  alarm();
}
```

Stufe 2

Falls ein Alarmzustand herrscht, wird die LED für 500 ms eingeschaltet und für weitere 500 ms ausgeschaltet. Dadurch blinkt sie. Dazu wird **alarm()** angepasst.

```
const int inputPin = A0;
const int ledPin = 7;

int alarmzustand; //0 = kein Alarm, 1 = Alarm

void setup() {
  pinMode(ledPin,OUTPUT);

  // Grundzustand
  alarmzustand = 0;          // Am Anfang haben wir keinen Alarm
}

void alarmTest() {
  if (analogRead(inputPin) > 500) alarmzustand = 1;
}

void alarm() {
  if (alarmzustand == 1) {    // wir haben einen Alarmzustand!
    digitalWrite(ledPin,HIGH);
    delay(500);
    digitalWrite(ledPin,LOW);
    delay(500);
  } else {                   // nein, alles ruhig
    digitalWrite(ledPin,LOW);
  }
}

void loop() {
  alarmTest();
  alarm();
}
```

Stufe 3

Wie lassen sich Fehlalarme verhindern? Ich gehe von der Überlegung aus, dass Einbrecher wohl längere Zeit im Raum sind und Autos recht schnell vorbeifahren. Deshalb wird der Alarm erst ausgelöst, wenn das Licht bei drei Messungen innerhalb von 15 Sekunden dauernd vorhanden ist.

Dazu wird die Bedeutung der Variablen **alarmzustand** etwas umdefiniert. Erst wenn sie den Wert 3 erreicht, soll Alarm ausgelöst werden. Also müssen wir in **alarm()** auf **alarmzustand == 3** testen.

Wenn ein Licht festgestellt wird, erhöhen wir **alarmzustand** um 1. Falls kein Licht festgestellt wird, wird **alarmzustand** wieder auf 0 gesetzt. Danach wird 5000 Millisekunden gewartet, damit die nächste Messung erst nach 5 Sekunden erfolgt. Das Ganze darf aber nur gemacht werden, wenn **alarmzustand** noch nicht auf 3 ist.

```
const int inputPin = A0;
const int ledPin = 7;

int alarmzustand; // <3 = kein Alarm, 3 = Alarm

void setup() {
    pinMode(ledPin, OUTPUT);

    // Grundzustand
    alarmzustand = 0;          // Am Anfang haben wir keinen Alarm
}

void alarmTest() {
    if (alarmzustand != 3) {
        if (analogRead(inputPin) > 500) {
            alarmzustand = alarmzustand + 1;
        } else {
            alarmzustand = 0;
        }
        delay(5000); // der nächste Test soll erst nach 5 Sekunden gemacht werden
    }
}

void alarm() {
    if (alarmzustand == 3) { // wir haben einen Alarmzustand!
        digitalWrite(ledPin, HIGH);
        delay(500);
        digitalWrite(ledPin, LOW);
        delay(500);
    } else { // nein, alles ruhig
        digitalWrite(ledPin, LOW);
    }
}

void loop() {
    alarmTest();
    alarm();
}
```