

Teil 2

Lektion

7

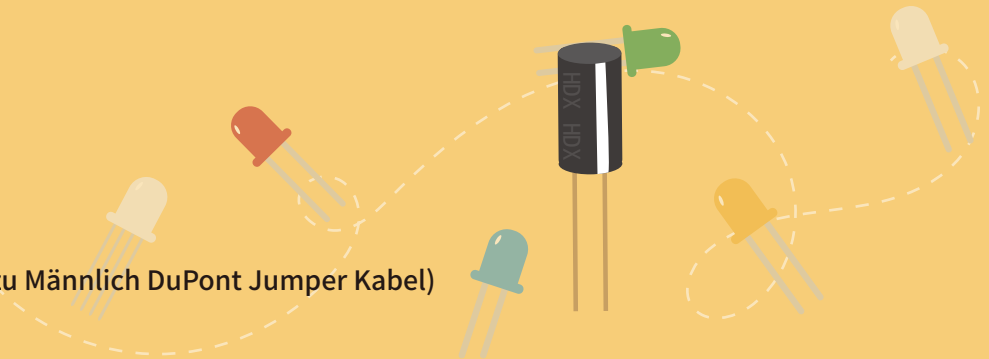
Neigungssensor

Übersicht

In dieser Lektion lernen sie, wie man einen Neigungssensor benutzt, um den Grad der Neigung herauszufinden.

Benötigte Bauteile:

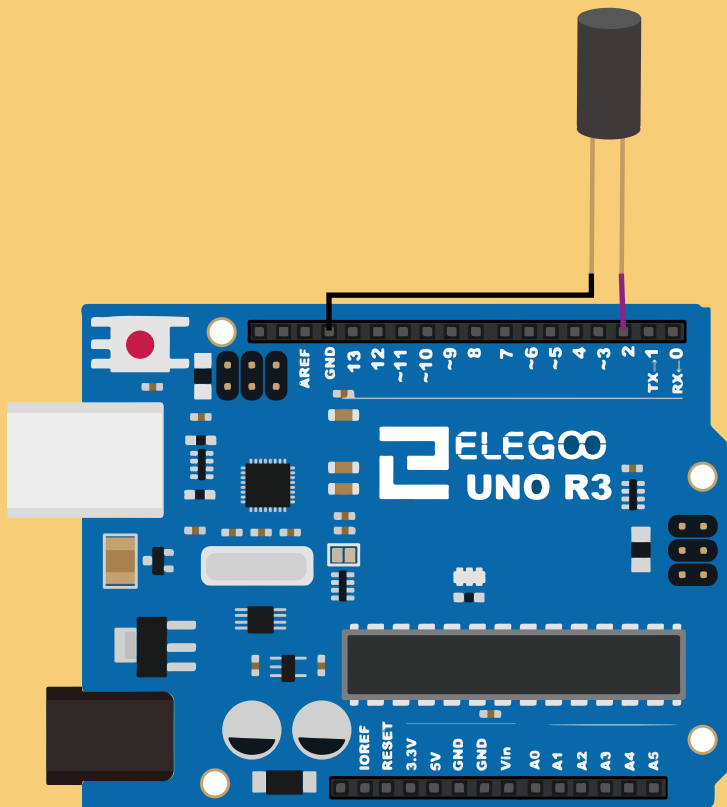
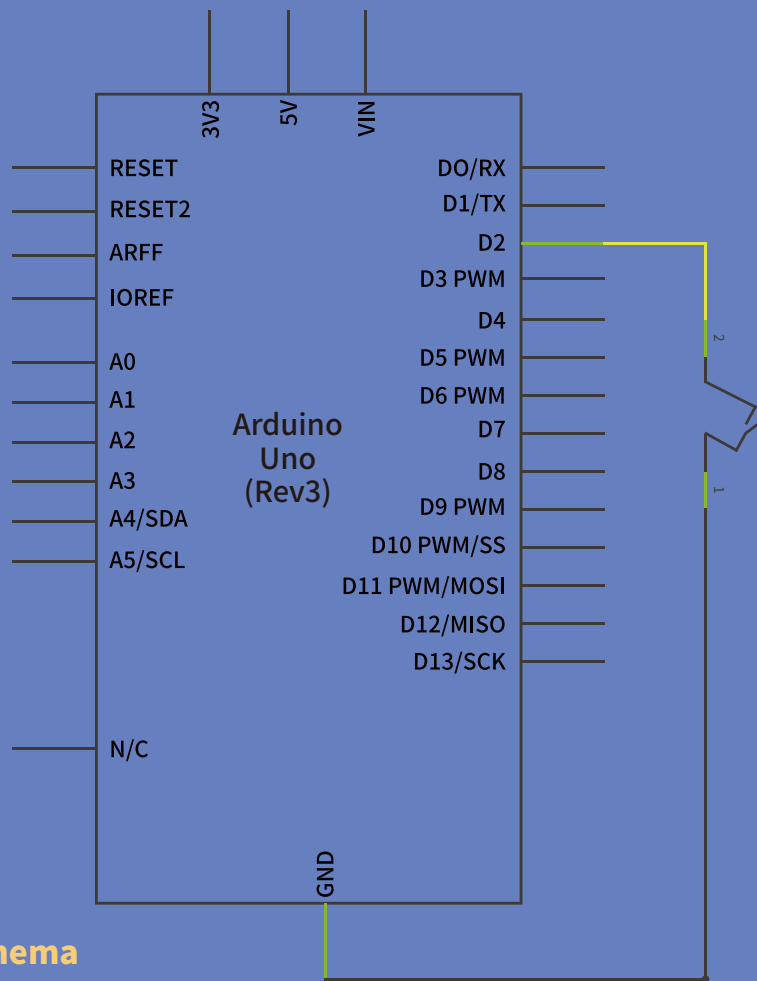
- (1) x Elegoo Uno R3
- (1) x Neigungssensor
- (2) x W-M Kabel (Weiblich zu Männlich DuPont Jumper Kabel)



Einführung in die Komponenten

Neigungssensor:

- Neigungssensoren** erlauben einem die momentane Ausrichtung oder Neigung festzustellen. Sie sind klein, günstig, einfach zu benutzen und verbrauchen wenig Strom. Wenn sie richtig benutzt werden, halten sie ewig. Durch ihren simplen Aufbau werden sie häufig in Spielzeugen, Gadgets und anderen Geräten eingesetzt.
- Sie** bestehen in der Regel aus einer Art Hohlraum (häufig in Form eines Zylinders) mit einer leitfähigen beweglichen Masse innerhalb. An einem Ende des Hohlraums befinden sich zwei leitende Elemente (Pole). Wenn der Neigungssensor senkrecht, wie im Bild oben, ausgerichtet ist, rollt die bewegliche Masse auf die beiden Pole und verbindet sie miteinander, wie bei einem Schalter.
- Obwohl** sie nicht so präzise und flexibel wie Beschleunigungssensoren sind, können Neigungssensoren Erschütterungen erkennen oder die grobe Neigung bestimmen. Ein weiterer Vorteil ist, dass große Neigungssensoren direkt die Stromversorgung schalten können. Beschleunigungssensoren geben dagegen digitale oder analoge Spannungswerte aus, die erst durch einen weiteren Schaltkreis oder Mikrocontroller analysiert werden müssen.



Code

Öffnen Sie nach der Verkabelung das Programm im Ordner Ball_Switch, in dem sich der Kurs befindet und klicken Sie auf HOCHLADEN innerhalb der Arduino IDE, um das Programm auf Ihr Device hochzuladen. Weitere Informationen zum Hochladen von Programmen finden Sie in Lektion 5 in Teil 1, wenn Fehler auftreten.

```
const int ledPin = 13;
```

const

[Variable Scope & Qualifiers] Beschreibung

Das Schlüsselwort `const` steht für Konstante. Es ist ein Variablenqualifizierer, welcher das Verhalten der Variablen ändert. Eine `const`-Variable ist "schreibgeschützt" und bleibt unverändert. Sie erhalten einen Compilerfehler wenn Sie versuchen, einer `const`-Variablen einen Wert zuzuweisen.

Mit dem Schlüsselwort "`const`" definierte Konstanten befolgen die Regeln des Variablenbereichs, die andere Variablen vorgeben. Dies und die Fallstricke bei der Verwendung von `#define` machen das Schlüsselwort "`const`" zu einer überlegenen Methode zum Definieren von Konstanten und wird der Verwendung von `#define` vorgezogen.

Example Code

```
const float pi = 3.14;  
float x;  
// ....  
x = pi * 2; // it's fine to use consts in math  
pi = 7;    // illegal - you can't write to (modify) a constant
```

Notes and Warnings #define or const

Sie können entweder `const` oder `#define` verwenden, um numerische oder String-Konstanten zu erstellen. Für Arrays müssen Sie `const` verwenden. Im Allgemeinen wird `const` gegenüber `#define` für die Definition von Konstanten bevorzugt.