

Selbstschießende Hand (SsH)

Niklas Keischgens

Aaron Philipzen

Henri Kühn

Jugend forscht 2017

Kurzfassung:

Wir kamen auf die Idee eine selbst schließende Roboterhand zu planen und zu bauen. Der Gedanke dazu kam uns, als wir an einem Heim für körperlich eingeschränkte Menschen vorbeikamen. Unsere Grundidee war, eine Lichtschranke zu nutzen. Diesen Mechanismus kann man vor allem in Kaufhäusern beobachten. Diesen Mechanismus wollen wir in unsere Hand einbauen. Wir planen die Lichtschranke und den Schließmechanismus der Tür mit einem Arduino zu steuern. Natürlich geht es uns um ein praktisches Design, damit die Hand die Person nicht behindert, sondern hilft. Gebraucht werden kann die Hand in der Industrie aber vor allem soll sie im Alltag körperlich eingeschränkten Menschen helfen.

Idee

Die Idee der SsH kam uns im Supermarkt beim Beobachten einer automatischen Tür. Dieses Prinzip des automatischen Öffnens und Schließens per Leuchtschranke, war die Grundlage der SsH. Auch in Deutschland lebt ein Teil der Bevölkerung mit nur einer Hand. Das Prinzip der Tür wollten wir für diese Menschen umtransformieren, damit diese Menschen die Möglichkeit einer zweiten künstlichen Hand bekommen. Die Idee der Ssh war entstanden. In der Handfläche sollte eine Leuchtschranke eingebaut werden, die bei Erkennung von Gegenständen, an kleine Schrittmotoren Signale sendet, um die Hand zu schließen. Die Programmierung soll per Arduino ausgeführt werden.

Protokoll

Nachdem wir die Idee hatten, haben wir nach einem Bewegungsmelder gesucht, welcher eine kleine, dafür aber sichere Reichweite hat. Diesen brauchen wir, weil die Roboterhand auf kurze Distanzen eine sichere Funktion benötigt, aber nicht auf eine längere Distanz funktionieren muss. Danach haben wir nach einem Mikrocontroller gesucht, mit dem wir den Bewegungsmelder ansteuern können. Wir haben uns schließlich für den Arduino entschieden, da dieser einfach und verständlich zu bedienen ist und es keine Schwierigkeiten gibt einen „Sketch“ zu programmieren.

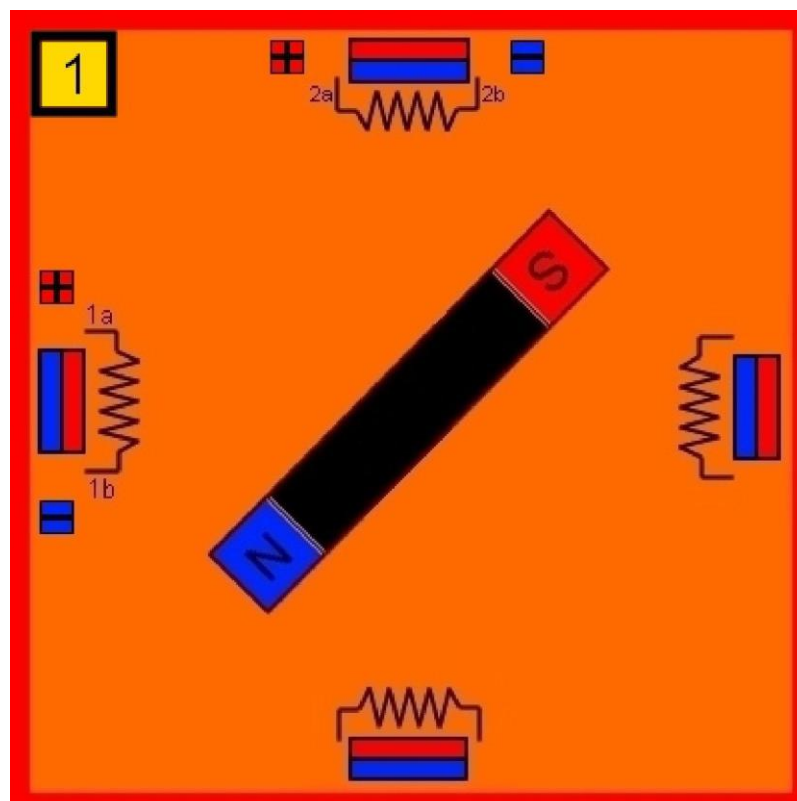
Unsere ersten Versuche führten wir mit einer einfachen LED durch, welche wir zum Leuchten oder Blinken bringen wollten. In den nächsten Versuchen haben wir dann zusätzlich den Bewegungsmelder eingebaut. Die „Sketche“ mit

Bewegungsmelder waren ähnlich, wie die ohne, nur das man an den Anfang einen Auslöser setzen musste.

Als nächstes müssen wir noch eine einfache und möglichst kompakte Hand konstruieren, welche im Alltag verwendet werden kann. In diese Hand sollen dann noch die technischen Geräte (siehe Absatz „Aufbau und technische Daten“) eingebaut werden.

Aufbau und technische Daten

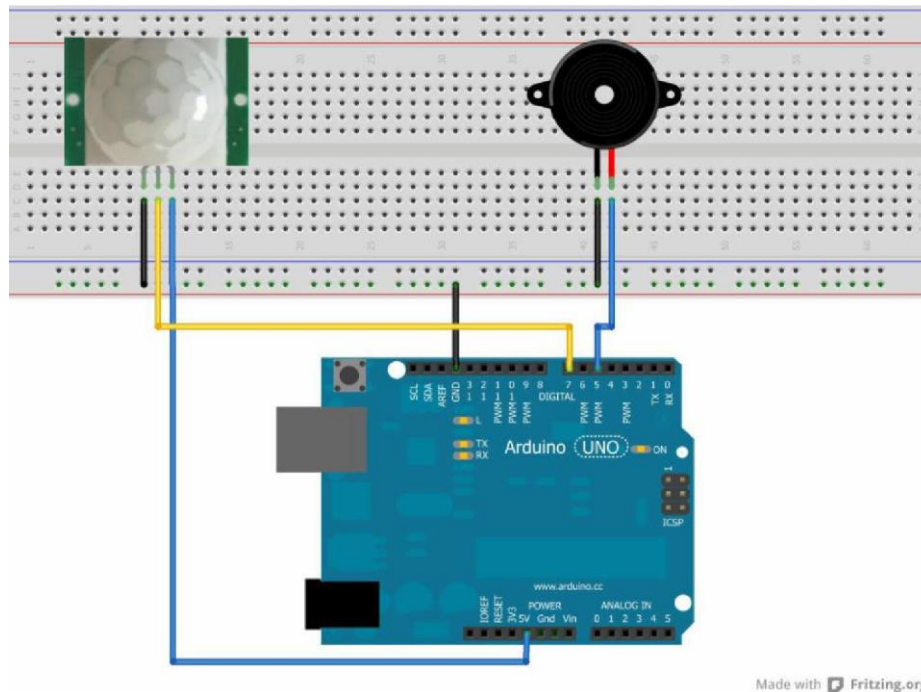
Die Roboterhand soll voraussichtlich aus zwei Winkeln bestehen, welche an ein Scharniergelenk befestigt werden. Dieses Scharniergelenk soll der Hand mehr Flexibilität im Alltag geben. Das Ganze soll von einem Schrittmotor angetrieben werden. Dieser funktioniert ähnlich wie ein Elektromotor. Er besitzt vier Spulen und einen Magneten. Sobald die Spulen unter Spannung gesetzt werden, entsteht ein Magnetfeld und der Magnet beginnt sich zu drehen. So entsteht die Bewegung der Scharniergelenke.



Funktionsweise eines Schrittmotors

In die Hand soll ein Bewegungssensor eingebaut werden, welcher mit passiver Infrarotstrahlung arbeitet und auf Wärmestrahlung in der Nähe reagiert.

Statische bzw. sich langsam verändernde Wärmewerte werden nicht erfasst. Also reagiert der Sensor lediglich bei einer plötzlichen Wärmeveränderung. Dieser Bewegungssensor wird von einem ArduinoUno (siehe Abschnitt Arduino) angesteuert.



Schaltplan Arduino Uno, verbunden mit einem Bewegungsmelder und einem Schrittmotor

Der Bewegungssensor soll möglichst unkompliziert im Zwischenstück der Hand und dem Arm untergebracht werden. Etwas weiter oben soll der Schrittmotor mit Energiezufuhr in Form einer Batterie eingesetzt werden.

Arduino

Ein Arduino ist ein Mikrocontroller, welcher z. B. zur Programmierung von Robotern genutzt wird.

Bei dem Arduino gibt es zwei wichtige Dinge:

den Sketch und das Board.

Das Board nimmt Befehle vom Sketch auf, welchen man in zwei Teile unterteilen kann:

1. Teil:

An dem Arduino Board kann man an der oberen Hälfte die sogenannten Pins erkennen, welche mit Zahlen gekennzeichnet sind. Diese Zahlen oder Pins sind für den Arduino sehr wichtig, da mit ihnen Befehle in Form eines In- und Outputs gegeben und empfangen werden. Diese beiden Befehle sind für die Programmierung extrem wichtig. Die Unterschiede sind relativ einfach. Bei einem Input möchte man den Zustand von einem Gerät wissen, z. B. bei einem Widerstand die Ohmzahl. Der Output gibt eine Spannung von plus 5 Volt ab, um eine Lampe zum Beispiel zum Leuchten zu bringen.

Ein Beispiel für einen Anfang wäre also zum Beispiel:

```
pinMode(5, Output);
```

Das PinMode ist der Befehl, die Fünf steht für den Pin 5 und Output für eine Spannung von 5 Volt. Heißt, der Pin 5 gibt eine Spannung von 5 Volt ab, um z. B. eine Lampe zum Leuchten zu bringen.

2. Teil:

Wenn man das Setup durchgeführt hat, kommt man zum loop, der Ausführung. Hier gibt es zwei Befehle, den digitalWrite und den delay.

Das digitalWrite gibt an, ob eine Spannung von plus fünf Volt abgegeben werden soll. Dieses erreicht er mit den Nebengebieten: High und Low.

```
High: plus  
5V  
Low:  
0V
```

Dazu kommt dann noch der zugehörige Pin, z. B.:

```
digitalWrite(4, High);
```

Der Pin 4 gibt eine Menge von plus 5 Volt ab. Der andere Befehl „delay“ gibt in Millisekunden eine Pause an.

```
Delay(500);
```

Es liegt eine Pause von 0,5 Sekunden vor.

Ein kompletter Sketch würde dann so aussehen:

```
void setup
```

```
Pinmode(4, Output);
```

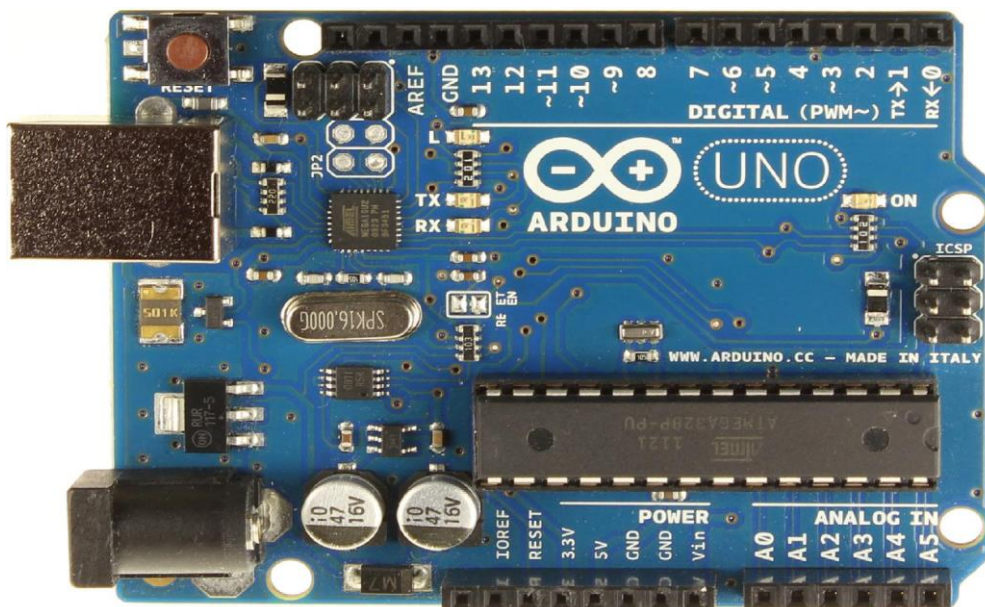
```
void loop
```

```
digitalWrite(4, Low);
```

```
delay(5000);
```

```
digitalWrite(4, High);
```

Der Sketch sagt aus, dass die Lampe am Anfang aus ist und nach einer Pause von fünf Sekunden leuchtet.



Arduino Uno von oben

(Quelle: http://images.google.de/imgres?imgurl=http%3A%2F%2Fdanielscheidler.de%2Fwp-content%2Fuploads%2F2013%2F02%2FArduinoUno_R3_Front.jpg)

SsH (Sketch)

Unser Sketch ist dagegen etwas komplexer, da wir nicht nur ein Gerät, sondern zwei Geräte einbringen müssen. Die beiden Geräte müssen aufeinander eingestellt werden, so dass keine Missverständnisse entstehen können. Bei den beiden Geräten handelt es sich um eine Lichtschranke und einen Schrittmotor. Da der Schrittmotor auf die Lichtschranke angewiesen ist, müssen hier noch zusätzliche Befehle hinzugefügt werden.

Unser Sketch lautet

dann: void setup

```
pinMode(13,  
Output);  
pinMode(12,  
Input);
```

void loop

```
bewegungsstatus=digitalRead(beweg  
ung);if (bewegungsstatus == HIGH)
```

```
digitalWrite(13,  
High);  
delay(5000);  
digitalWrite(13, Low)
```

else

```
digitalWrite(13, Low);
```

**Erläuterung der Pins und deren
Bedeutung:(13=Schrittmotor)
(12=Bewegung)
(11=Bewegungsstatus)**

Am Anfang wird ein normales Setup durchgeführt: „Was muss hier wie gesteuert werden.“

Im loop-Teil passiert Folgendes:

Sobald eine Bewegung wahrgenommen wird, erfolgt eine Abgabe von 5 Volt an den Schrittmotor und dieser schließt das Scharniergelenk. Nach einer noch unbestimmten Zeit schaltet sich dieser wieder aus und das Scharniergelenk öffnet sich wieder. Wird keine Bewegung wahrgenommen, bleibt die Hand im geöffneten Zustand.

Bad Münstereifel, Januar 2017

Arbeit von Niklas Keischgens, Aaron Philipzen und Henri Kühn.