

Raumsonar

Ziel

Einen Bereich per Ultraschallsensor abtasten. Es geht dabei um drei Dinge:

- Die Ansteuerung eines Servos
- Das Auslesen eines Ultraschall-Abstandssensors
- Die Visualisierung von Messwerten mit Processing

Ein Ultraschallsensor wird durch einen Servo über einen Winkelbereich von 180° bewegt. Die Abstandsdaten werden periodisch ausgelesen und grafisch innerhalb der Processing Umgebung visualisiert.

Vorgehensweise

Beschäftigen Sie sich zuerst mit dem Abstandssensor anhand des Beispiels für den Ping Sensor. Verbinden Sie den Sensor, studieren Sie den Quelltext und probieren Sie das Beispiel aus. Beschäftigen Sie sich dann dem Servo. Verbinden Sie den Servo, sowie das Potentiometer und probieren Sie das „Knob“ Beispiel aus.

Entfernen Sie nun das Potentiometer und setzen Sie den Abstandssensor auf den Servo. Schreiben Sie ein Programm, welches den Servo ständig von 0° bis 180° schwenkt und dabei periodisch den Abstand ausliest. Geben Sie den aktuellen Winkel und den gemessenen Abstand auf der seriellen Schnittstelle als Tupel **Winkel;Abstand** aus.

Starten Sie nun die Processing IDE und probieren Sie das „Continuous Lines“ Beispiel. Schreiben Sie innerhalb von Processing nun ein Programm, welches die Datentupel des Arduino von der seriellen Schnittstelle entgegennimmt und möglichst aussagekräftig grafisch visualisiert.

Vorbereitung

Beschäftigen Sie sich mit der Dokumentation und den Funktionen zur seriellen Schnittstelle. Arbeiten Sie sich in Processing ein: Probieren Sie Beispiele aus und implementieren Sie eine GUI zum Anzeigen der Daten.

Achtung

Die Processing IDE sieht der Arduino IDE sehr ähnlich – sie kompiliert aber nach Java (!) – hier gibt es also ein paar subtile Unterschiede.

Auf die serielle Schnittstelle kann mit verschiedensten Strategien (z.B. blockend / nicht blockend oder byteweise / pufferweise) zugegriffen werden. Eine Möglichkeit komplette Zeilen von der seriellen Schnittstelle entgegenzunehmen besteht beispielsweise im Aufruf von `readBytesUntil("\n")`, einer Funktion der `Serial` Bibliothek bei Processing.

Wichtige Funktionen

- `pulseIn()`
- `map()`
- Die Servo Bibliothek
- Processing:
 - `trim()`
 - `split()`
 - `parseInt()`

Sie brauchen

- Das Arduino Board mit Servo, USB Kabel, Drahtbrücken, Potentiometer, Abstandssensor
- Das Ping Sensor Beispiel (Im Menu Beispiele -> 06.Sensors -> Ping) und unter <http://arduino.cc/en/Tutorial/ping>
- Das Servo Beispiel (Im Menu Beispiele -> Servo -> Knob) und unter <http://arduino.cc/en/Tutorial/Knob>
- Das „Continuous Lines“ Beispiel (In der Processing IDE, im Menu unter Examples -> Topics -> Drawing -> Continuous Lines) und im Web unter <https://processing.org/examples/continuouslines.html>
- Die Processing Referenz: <https://processing.org/reference/>

Notengebung

4,0 (Anwesend); 3,0 (Servo bewegt sich, Ping gibt Werte); 2,3 (Korrekte Ausgabe der Tupel); 2,0 (+ Ausgabe in Processing); 1,7 (+Arduino Code strukturiert und dokumentiert); 1,3 (+Processing Code strukturiert und dokumentiert); 1,0 (+Besonders schöne Visualisierung)