
BACHELORARBEIT

Herr
Yuyao Wang

**Konzeption und Entwicklung
eines Abstands- und Ge-
schwindigkeitsmesssystems
auf Basis von mehreren qua-
si-parallel arbeitenden
Entfernungs-Sensoren**

Mittweida, 2020

Fakultät: Elektro- und Informationstechnik

BACHELORARBEIT

Konzeption und Entwicklung eines Abstands- und Geschwindigkeitsmesssys- tems auf Basis von mehreren quasiparallel arbeitenden Entfernungs-Sensoren

Autor:

Herr

Yuyao Wang

Studiengang:

Elektro- und Informationstechnik

Seminargruppe:

EI16wA2-BC

Erstprüfer:

Prof. Dr.-Ing. Michael Kuhl

Zweitprüfer:

Blümel Kevin

Einreichung:

Mittweida, 10, 08, 2020

Verteidigung/Bewertung:

Mittweida, 20, 08, 2020

BACHELOR THESIS

**Concept and development
a distance and
Speed measuring system
based on several
quasi parallel working
Distance sensors**

author:

Mr.

Yuyao Wang

course of studies:

Electrical and Computer Engineering

seminar group:

EI16wA2-BC

first examiner:

Prof. Dr.-Ing. Michael Kuhl

second examiner:

Blümel Kevin

submission:

Mittweida, 10, 08, 2020

defence/ evaluation:

Mittweida, 20, 08, 2020

Bibliografische Beschreibung:

Nachname, Vorname: Wang Yuyao

Thema der Bachelorarbeit: Konzeption und Entwicklung eines Abstands- und Geschwindigkeitsmesssystems auf Basis von mehreren quasiparallel arbeitenden Entfernungs-Sensoren

Topic of thesis: Concept and development a distance and Speed measuring system based on several quasi parallel working Distance sensors

59 Seiten, Hochschule Mittweida, University of Applied Sciences,
Fakultät Ingenieurwissenschaften, Bachelorarbeit, 2020

Die Bachelorarbeit wurde im Wesentlichen während der Lock-Down-Zeit der COVID19-Pandemie erstellt. Die Hochschule war in der Zeit geschlossen bzw. es war nur ein sehr begrenzter Zugang möglich. Praktische Aufbauten und Experimente konnten damit nur in sehr eingeschränkter, häuslicher Umgebung mit minimalen Mitteln durchgeführt werden.

Dies ist bei der Bewertung der Arbeit zu berücksichtigen.

gezeichnet: Prof. Dr.-Ing. Michael Kuhl

Referat:

Die vorliegende Arbeit befasst sich mit der Kombination aller drei Sensoren bei der Entfernungsmessung und Geschwindigkeitsmessung. Das Hauptziel ist aber, dass man die Vorteile und Nachteile von den Sensoren kennt und nach ihre Vor- und Nachteile wählt man den geeigneten Sensor, die Arbeitsaufgaben zu erreichen.

Inhaltsverzeichnis

Inhaltsverzeichnis	I-II
Abkürzungsverzeichnis	III
Abbildungsverzeichnis	V
Tabellenverzeichnis	V
1 Einleitung.....	1
2 Aufgabenstellung.....	3
3 These.....	4
4 Stand der Technik... ..	5
5 Vorversuch... ..	6
5.1 <i>Entfernungsmessung</i>	6
5.2 <i>Geschwindigkeitsmessung.....</i>	8
5.3 <i>Winkelmessung.....</i>	8
6 Umsetzung.....	13
6.1 <i>Gesamtkonzept.....</i>	13
6.2 <i>Hardware</i>	14
6.2.1 <i>Anforderungen und Systemkonzept / Systementwurf.....</i>	14
6.2.2 <i>Hardware Umsetzung</i>	15
6.3 <i>Software.....</i>	17
6.3.1 <i>Softwarekonzept</i>	17
6.3.2 <i>Umsetzung des Softwarekonzepts</i>	17
7 Versuchsdurchführung.....	20
7.1 <i>Versuchskonzept.....</i>	20
7.2 <i>Darstellung von möglichen Fehlereinflüssen</i>	21
7.3 <i>Versuchsdurchführung</i>	22
7.4 <i>Kombination aller Sensoren und Auswertung.....</i>	32
8 Zusammenfassung, Schlussfolgerung.....	36
9 Ausblick.....	37
Literaturverzeichnis	VII

Anlagen	IXI-XV
<i>Arduino Mega2560</i>	<i>XI</i>
<i>Arduino Uno r3</i>	<i>IXII</i>
<i>Brennen</i>	<i>XIII</i>
Eigenständigkeitserklärung	XVI

Abkürzungsverzeichnis

Alu	das Aluminium
C	die Geschwindigkeit von Licht
GND	Ground
I2C (I²C- Bus)	Inter-Integrated Circuit
LCD	Liquid Crystal Display
LED	Light-emitting diode
PWM	Pulse Width Modulation
SCL	Serial Clock line
SDA	Serial Data line
t	die Zeit
TOF	Time-of-flight
US	Ultraschall-Sensor
USB	Universal Serial Bus
V_s	die Geschwindigkeit von Stimme

Abbildungsverzeichnis

Abbildung 1: Autonome Fahrtechnik	1
Abbildung 2: Tof und Ultraschall für die Winkelmessung	9
Abbildung 3: Tof und Ultraschall für die Winkelmessung	10
Abbildung 4: reale Situation für die Winkelmessung	11
Abbildung 5: Skizze des Gesamtsystems	13
Abbildung 6: Arduino Mega2560	14
Abbildung 7: HC-SR04	14
Abbildung 8: VL53L1X	15
Abbildung 9: ipm-165	15
Abbildung 10: 1602-Display	15
Abbildung 11: Schaltplan	16
Abbildung 12: Steckplatine.....	16
Abbildung 13: Parameter und Setup	17
Abbildung 14: wichtige Teile vom Programm	18
Abbildung 15: Anzeige-Teile vom Programm	19
Abbildung 16: Programmplan.....	19
Abbildung 17: Auflauf des Messgeräts	20
Abbildung 18: Versuchsablauf.....	21
Abbildung 19: kleiner Fehler.....	23
Abbildung 20: ferngesteuertes Fahrzeug	24
Abbildung 21: L298N.....	24
Abbildung 22: Abweichung vom Ultraschall-Sensor.....	28
Abbildung 23: Abweichung vom Tof-Sensor	28
Abbildung 24: Abweichung vom Mikrowellenradar-Sensor	29
Abbildung 25: Lego-Auto.....	30
Abbildung 26: Schematische Darstellung mehrerer Sensoren	37
Abbildung 27: alle Pinout vom Mega2560	XI
Abbildung 28 alle Pinout vom Uno r3	XII
Abbildung 29: Schaltplan von Brennen	XIII
Abbildung 30: Wählen von Sketch	XIV
Abbildung 31: Setzen und Brennen.....	XIV

Tabellenverzeichnis

Tabelle 1: die vom Ultraschall-Sensor und ToF-Sensor gemessenen Entfernungen beim Winkel 0°	6
Tabelle 2: der Einfluss von Objektmaterial auf die Entfernungsmessung beim Winkel 0°	7
Tabelle 3: die vom Ultraschall-Sensor und Mikrowellenradar-Sensor gemessenen Geschwindigkeiten beim Winkel 0°	8
Tabelle 4: Winkel und Entfernung bei Ultraschall-Sensor	11
Tabelle 5: Test beim Winkel 0-20 Grad	22
Tabelle 6: Geschwindigkeit der unterschiedlichen Entfernung	25
Tabelle 7: Geschwindigkeit der unterschiedlichen Entfernung	26
Tabelle 8: Geschwindigkeit der unterschiedlichen Entfernung	26
Tabelle 9: Geschwindigkeit der unterschiedlichen Entfernung	27
Tabelle 10: Lego-Auto mit niedriger Geschwindigkeit	30
Tabelle 11: Lego-Auto mit hoher Geschwindigkeit	31
Tabelle 12: Lego-Auto mit Unterschiedlichen Geschwindigkeiten	31
Tabelle 13: Lego-Auto mit niedriger Geschwindigkeit	31
Tabelle 14: Lego-Auto mit hoher Geschwindigkeit	32
Tabelle 15: Lego-Auto mit Unterschiedlichen Geschwindigkeiten	32
Tabelle 16: Vor- und Nachteile beim Ultraschall-Sensor	34
Tabelle 17: Vor- und Nachteile beim ToF-Sensor	34
Tabelle 18: Vor- und Nachteile beim Mikrowellenradar-Sensor	34

1 Einleitung

Heutzutage ist die Gesellschaft schrittweise in die Intelligenz eingetreten und nehmen wir das Auto als Beispiel. Jetzt hat jeder Haushalt ein Auto, sodass Verkehrssicherheitsprobleme nicht ignoriert werden können. Man kann mehrere Sensoren verwenden, um ein intelligentes System zu implementieren, das den vorderen Bereich überwacht und die Entfernung, die relative Geschwindigkeit und den entsprechenden Winkel zum vorderen Objekt in Echtzeit anzeigt. Wenn die Geschwindigkeit zu hoch oder zu nahe am vorausfahrenden Fahrzeug oder Fußgänger ist, wird der Fahrer gewarnt oder bremst das Auto selbst aktiv.

Für die Fahrzeuge mit autonomen Fahrfunktionen wird dieses Sicherheitssystem bestimmt angewendet werden. Das Auto mit autonomen Fahrfunktionen verwendet dieses Sicherheitssystem, die Geschwindigkeit oder Richtung anzupassen, um gefährliche Situationen zu vermeiden, die auftreten können. Ich denke, die Forschung an diesem Sicherheitssystem zur Zeit eine relativ neue und bedeutende Richtung ist.



Bilder 1: Autonome Fahrtechnik [1]

Für die Arbeit kann man mit drei Sensoren ein einfaches und kleines Modell machen, also Ultraschall-Sensor, Tof-Sensor und Mikrowellenradar-Sensor. Mithilfe des Ultraschall-Sensors und des Tof-Sensors kann die Entfernung gemessen werden. Die Geschwindigkeit des gemessenen Objekts kann über den Radar-Sensor und den Ultraschall-Sensor erhalten werden. Damit die gesamte Hardware ordnungsgemäß funktioniert, muss man gleichzeitig mit der Software arbeiten, und das erforderliche Programm zu schreiben.

2 Aufgabenstellung

Das Ziel der Arbeit ist die prototypische Entwicklung eines Abstands- und Geschwindigkeitsmesssystems auf der Basis von mehreren quasi parallel arbeitenden Sensoren, und der Arbeitsbereich soll 1 m bis 4 m betragen.

Man kann eine Möglichkeit der Kombination von Ultraschall-Sensor, Tof-Sensor und Mirowellenradar-Sensor realisieren, um die Arbeit zu schaffen. Dabei soll die Funktionsfähigkeit der Sensoren für Objekte aus verschiedenen Materialien untersucht und ein Vergleich der Sensoren untereinander vorgenommen werden.

3 These

Man kann mit drei Sensoren, die basiert auf Arduino-Board verwendet werden, das kleine Modell erstellen und nach dem Anschließen der Leitungsdraht von aller Elemente soll man das Programm gemäß der Verwendungsmethode jedes Sensors schreiben und dann ladet man das kompilierte Programm auf Arduino-Broad hoch.

Nach den unterschiedlichen Eigenschaften von den Sensoren kann man die Geschwindigkeit und den Abstand bei unterschiedlichem Winkel messen. Die drei Sensoren sollten sich in derselben Ebene befinden und das Zielobjekt gleichzeitig messen.

4 Stand der Technik

Der Ultraschall-Sensor kann zur Entfernungsmessung und Geschwindigkeitsmessung eingesetzt werden. Mithilfe vom ToF-Sensor kann man die Entfernung auch bestimmen. Mikrowellenradar-Sensor ist auch eine gute Wahl für die Geschwindigkeitsmessung.

Der Ultraschall-Sensor (*HC-SR04*) kann die Ultraschallwellen emittieren, und sie werden zurück reflektiert, wenn die Ultraschallwellen auf ein Objekt treffen. Dann wird der Ultraschall-Sensor das Echo empfangen. Deshalb kann man den Wert der Entfernung bekommen. Dieser Grundsatz gilt auch für ToF-Sensor (*VL53L1X*), aber der Unterschied besteht darin, dass die Geschwindigkeit von Licht- und Schallwellen unterschiedlich ist. Die Geschwindigkeit von den Schallwellen ist 340m/s und von den Lichtwellen ist $3 \cdot 10^8$ m/s.

Der Mikrowellenradar-Sensor (*ipm-165*) misst die Geschwindigkeit mit Hilfe des **Doppler-Effekts**, also Wenn die Wellenquelle und der Beobachter eine Relativbewegung haben, unterscheidet sich die Frequenz der vom Beobachter empfangenen Welle von der Frequenz der Wellenquelle, Z.B. Durch die Reflektion des Radarstrahls an dem sich bewegenden Fahrzeug tritt aufgrund des Doppler-Effekts eine Frequenzänderung der Strahlung ein, deren Größe der Geschwindigkeit des Fahrzeugs proportional ist. [2] Man kann die Geschwindigkeit nicht nur mit dem Mikrowellenradar-Sensor aber auch mit dem Ultraschall-Sensor (*HC-SR04*) messen. Man braucht zwei Entfernung s_1 , s_2 (Vor und nach der Bewegung) und einem Delaytime, die Geschwindigkeit einfach zu berechnen, also v ist gleich $(s_1 - s_2) / t$.

Im **Kapitel 5** wird ein paar Versuchen für die Grundmessung durchführt.

5 Vorversuch

5.1 Entfernungsmessung

Bei der Entfernungsmessung benutzt man den Ultraschall-Sensor (*HC-SR04*) und den ToF-Sensor (*VL53L1X*). Für die **Entfernungsmessung** kann man von Physikalischer Formel die Hilfe bekommen. Man kann die alle Informationen bekommen, Z.B. die Ultraschallflugzeit t , die Schallgeschwindigkeit v , also 340m/s bei der Temperatur: 25° C. Mit einfacher physikalischer Formel, also die Entfernung s ist gleich v mal Ultraschallflugzeit t^* durch 2(Hin und her ist doppelt so weit).

Formeln: $V = S / t$

$$\Rightarrow S = V * t \quad (\text{Achtung: hier } t = t^* / 2)$$

Das gilt gleich auch für den ToF-Sensor (*VL53L1X*), der ein VCEL (Vertical-Cavity Surface-Emitting Laser) ist, aber es gibt sehr groß anders bei der Geschwindigkeit von Laser, also $3 \cdot 10^8$ m/s. Es wird über den Einfluss des Winkels auf den Ultraschall-Sensor (*HC-SR04*) in 5.3 diskutiert.

Ultraschall	Tof	Mittelwert
0.09m	0.08m	0.08m
0.10m	0.10m	0.10m
0.11m	0.11m	0.11m
0.12m	0.12m	0.12m
0.13m	0.13m	0.13m
0.16m	0.16m	0.16m
0.18m	0.18m	0.18m

Tabelle 1: die vom Ultraschall-Sensor und ToF-Sensor gemessenen Entfernungen beim Winkel 0

Anschließend wird über den Einfluss von Objektmaterial auf die Entfernungsmessung diskutiert. Es wird mal gezeigt, ob der Sensor angesichts der Objekte aus verschiedenen Materialien die Entfernung messen kann.

Der Ultraschallsensor und der Tof-Sensor werden zusammen befestigt, und verschiedene Materialien werden in einem Abstand von 0.5 m platziert. Wenn der vom Sensor abgelesene Abstand 0,5 Meter beträgt, bedeutet dies, dass dieses Material keinen Einfluss auf ihn hat und **Ja** in der Tabelle angezeigt wird, andernfalls wird **Nein** angezeigt. (Nein / Ja bedeutet, dass der Wert vom Sensor nicht stabil ist.)

Material <i>(größer als 200mm*200mm)</i>	Ultraschall-Sensor <i>(HC-SR04)</i>	Tof-Sensor <i>(VL53L1X)</i>
Einschichtige transparente Plastiktüte	Ja	Nein / Ja
Glattes Papier	Ja	Ja
Raues Papier	Nein	Ja
leere Flasche (Plastik /PET)	Ja	Ja
kurzsichtige Brille (Glas/ Plexiglas)	Ja	Nein
Sonnenbrille (Glas, Harz)	Ja	Nein
Medizinische Schutzbrille (Stark beschichtetes Polycarbonat)	Ja	Nein / Ja
Rot/grün/schwarz Pappkarton	Ja	Ja
Pullover/Sweatshirt	Nein	Ja

Tabelle 2: der Einfluss von Objektmaterial auf die Entfernungsmessung beim Winkel 0°

5.2 Geschwindigkeitsmessung

Für die **Geschwindigkeitsmessung** kann man mithilfe des Mikrowellenradar-Sensor (*ipm-165*) und Ultraschall-Sensor (*HC-SR04*) einfach schaffen.

Aber bei niedriger Geschwindigkeit (vielleicht <1km/h oder 0.278m/s) und kurzer Distanz ist der Mikrowellenradar-Sensor (*ipm-165*) von etwas anderes sehr leicht zu beeinflussen, Z.B.: Wenn man ein leicht zitterndes Objekt (z.B. ein Finger) vor das Objekt hält, liest der Sensor die Geschwindigkeit des Fingers aus. Eine Tabelle wird verwendet, um das zu zeigen. In den übrigen Fällen ist der Mikrowellenradar-Sensor (*ipm-165*) genauer als Ultraschall-Sensor (*HC-SR04*).

Ultraschall	Mikrowellenradar	Mittelwert
0.02m/s	0.36m/s	0.19m/s
0.04m/s	0.31m/s	0.18m/s
0.12m/s	0.29m/s	0.21m/s
0.85m/s	0.44m/s	0.64m/s
0.20m/s	0.24m/s	0.22m/s
0.71m/s	0.78m/s	0.74m/s
0.95m/s	0.67m/s	0.81m/s

Tabelle 3: die vom Ultraschall-Sensor und Mikrowellenradar-Sensor gemessenen Geschwindigkeiten beim Winkel 0°

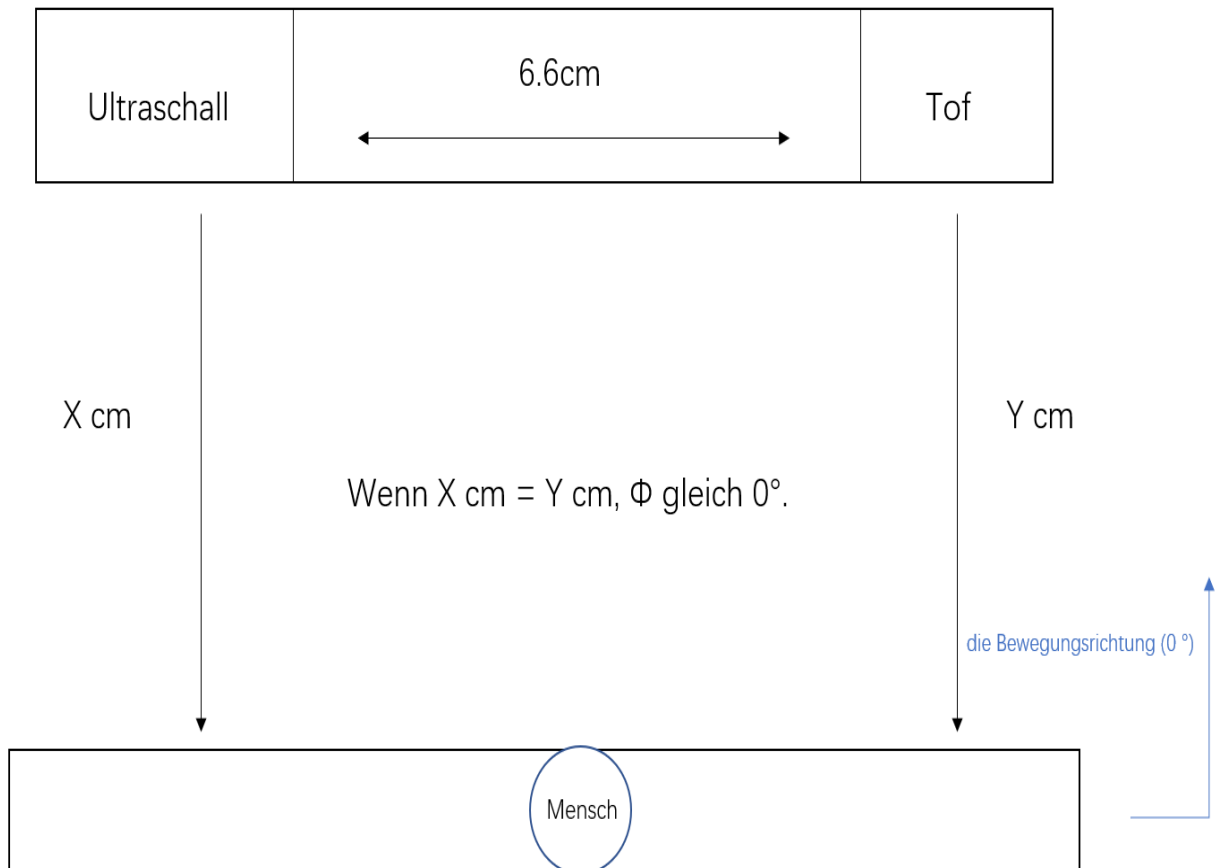
5.3 Winkelmessung

Für die **Winkelmessung** ist der Versuch ursprünglich geplant worden, mit Ultraschall-Sensor (*HC-SR04*) und ToF-Sensor (*VL53L1X*) zu machen. Der anfängliche Plan wird mit den Bildern gezeigt.

Man kann vom ersten Bild sehen, wenn eine Person direkt auf die Sensoren zugeht, beträgt der Winkel zwischen dieser Person und der Ebene von der Sensoren 0° . Vom zweiten Bild kann man sehen, Wenn eine Person diagonal (mit einem Winkel) auf die Sensoren zugeht, kann man auch mithilfe der beiden (von ToF-Sensor und Ultra-

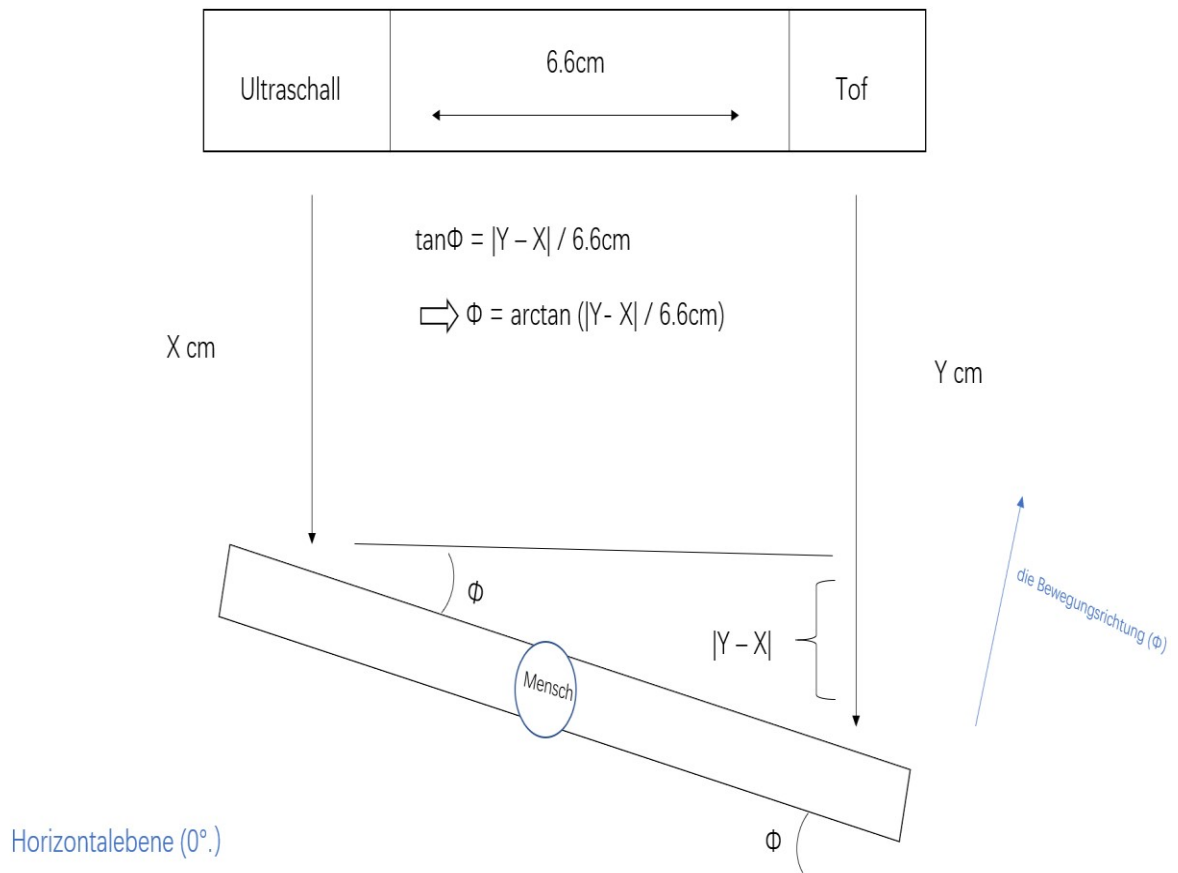
schall-Sensor gemessenen) Entfernung und der Trigonometrische Funktion den Winkel einfach berechnen.

Aber der Plan ist total falsch. Der Grund wird nach den Bildern erklärt.



Bilder 2: ToF und Ultraschall für die Winkelmessung

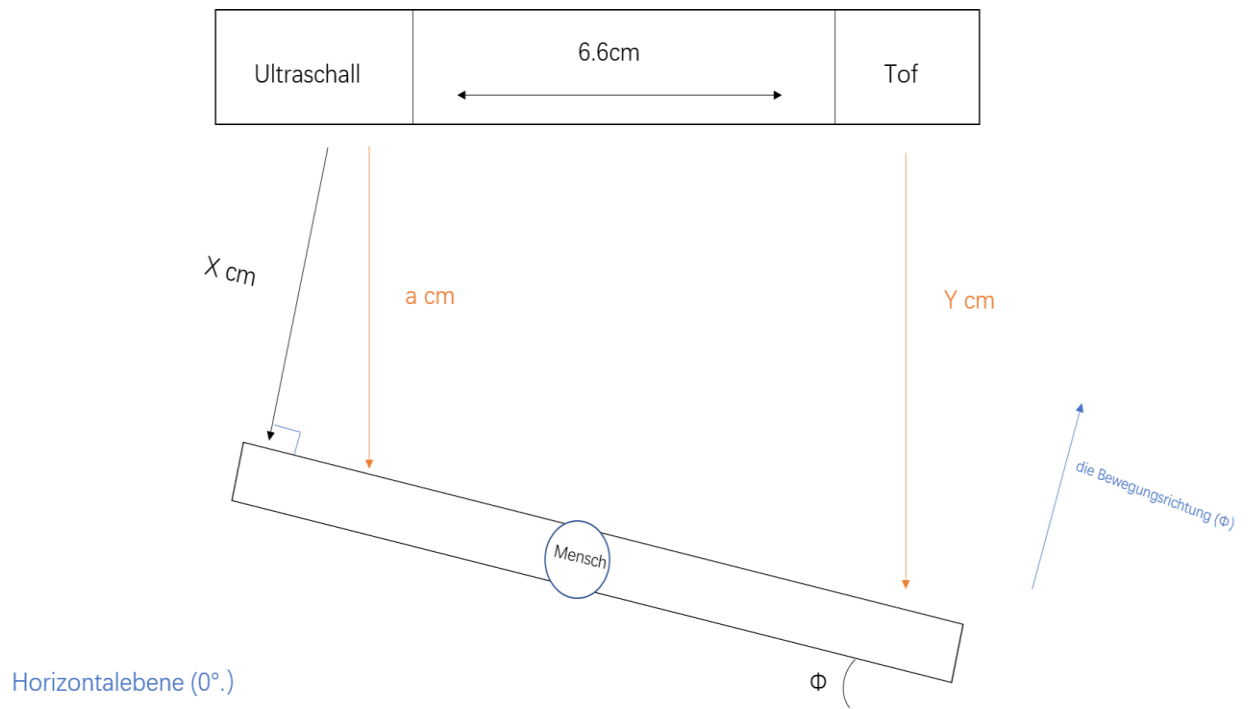
Die vom Ultraschall-Sensor (*HC-SR04*) gemessenen Entfernung ist X cm. Die vom ToF-Sensor (*VL53L1X*) gemessenen Entfernung ist Y cm. Falls X cm gleich Y cm ist, dann ist der Winkel Φ gleich 0° .



Bilder 3: ToF und Ultraschall für die Winkelmessung

Die vom Ultraschall-Sensor (*HC-SR04*) gemessenen Entfernung ist X cm. Die vom ToF-Sensor (*VL53L1X*) gemessenen Entfernung ist Y cm. Falls X cm nicht gleich Y cm ist, dann ist der Winkel Φ gleich $\arctan(|Y \text{ cm} - X \text{ cm}| / 6,6 \text{ cm})^\circ$.

Dieser Plan ist am Ende gescheitert, weil der Ultraschall-Sensor von Winkeländerungen sehr groß betroffen ist (ToF-Sensor ist nicht vom Winkel betroffen), und Ich kann Bilder verwenden, um dies zu zeigen, und Tabelle verwenden, um dies zu veranschaulichen. In der Tabelle wird der Mittelpunkt des Objekts als Drehachse genommen.



Bilder 4: reale Situation für die Winkelmessung

Aufgrund der Reflexionseigenschaften von Schallwellen beträgt der vom Ultraschallsensor gelesene Wert X cm, nicht a cm. Daher ist es schwierig, den Winkelwert Φ durch das Programm zu berechnen.

Winkel	Entfernung
0°	0.17m
10°	0.17m
20°	0.16m
22°	0.42m
25°	0.61m

30°	0.91m
40°	0.95m
50°	-1.06m (Error)

Tabelle 4: Winkel und Entfernung beim Ultraschall-Sensor

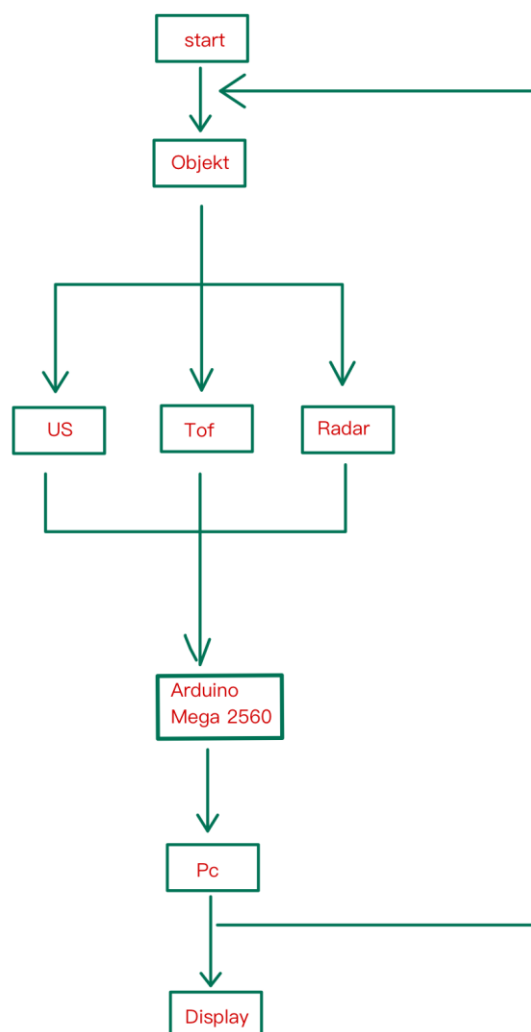
Wie aus der obigen Tabelle ersichtlich ist, ist ca. 20 Grad die Grenze des Ultraschallsensors. Über diesen Wert (ca. 20 Grad) empfängt der Ultraschall-Sensor das Echo nicht oder er kann die Entfernung nicht genau messen.

6 Umsetzung

6.1 Gesamtkonzept

Man verwendet den Ultraschall-Sensor und den Tof-Sensor, um eine Entfernungsmessung zu erzielen, und die Geschwindigkeitsmessung wird von dem Ultraschall-Sensor und dem Mikrowellenradar-Sensor realisiert.

Die entsprechende Programmierung erfolgt über die Software, und das Programm wird über USB-Kabel auf Arduino-Board hochgeladen. Schließlich werden die gemessene Entfernung und Geschwindigkeit auf dem Display angezeigt.



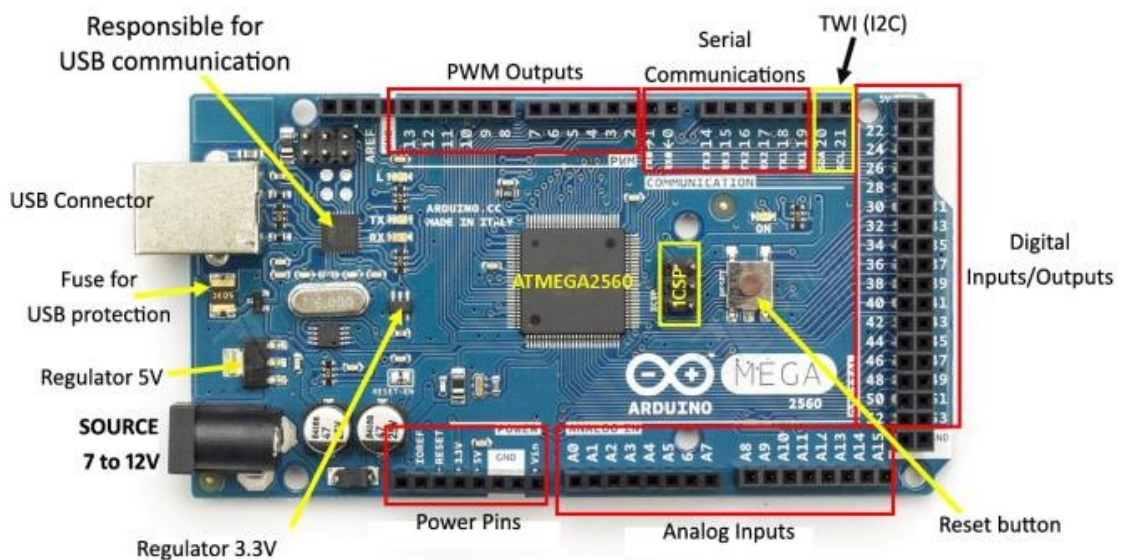
Bilder 5: Skizze des Gesamtsystems

6.2 Hardware

6.2.1 Anforderungen und Systemkonzept / Systementwurf

Dieses Gerät muss die Funktion zum Messen der Entfernung und Geschwindigkeit von Objekten realisieren. Der Messbereich muss 0.02 bis 4 Meter betragen. Gemäß Abschnitt 5.3 sollte der Winkelbereich 0 bis 20 Grad betragen.

Das gesamte Gerät basiert auf den Arduino Mega2560, deshalb ist die Betriebsspannung 5V und ist die Eingangsspannung (empfohlen) 7V bis 12V. Der Gleichstrom pro E/A-Pin (I/O) ist 20mA und der Gleichstrom für 3.3V-Pin ist 50mA.



Bilder 6: Arduino Mega2560 [3]

Für Ultraschall-Sensor (HC-SR04):

Betriebsspannung: 5V

Strom: 15mA

Arbeitsfrequenz: 40kHz



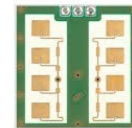
Bilder 7: HC-SR04 [4]

Für Tof-Sensor (VL53L1X):
Betriebsspannung: 3.3V / 5V
Strom: 15mA
Messfrequenz: 50Hz



Bilder 8: VL53L1X [5]

Für die Mikrowellenradar-Sensor (ipm-165):
Versorgungsspannung: 4.75 – 5.25V
Strom: 30 - 40mA



Bilder 9: ipm-165 [6]

Für das Display (1602 LCD-Display):
Betriebsspannung: 4.5 – 5.5V
Strom: 2.0mA

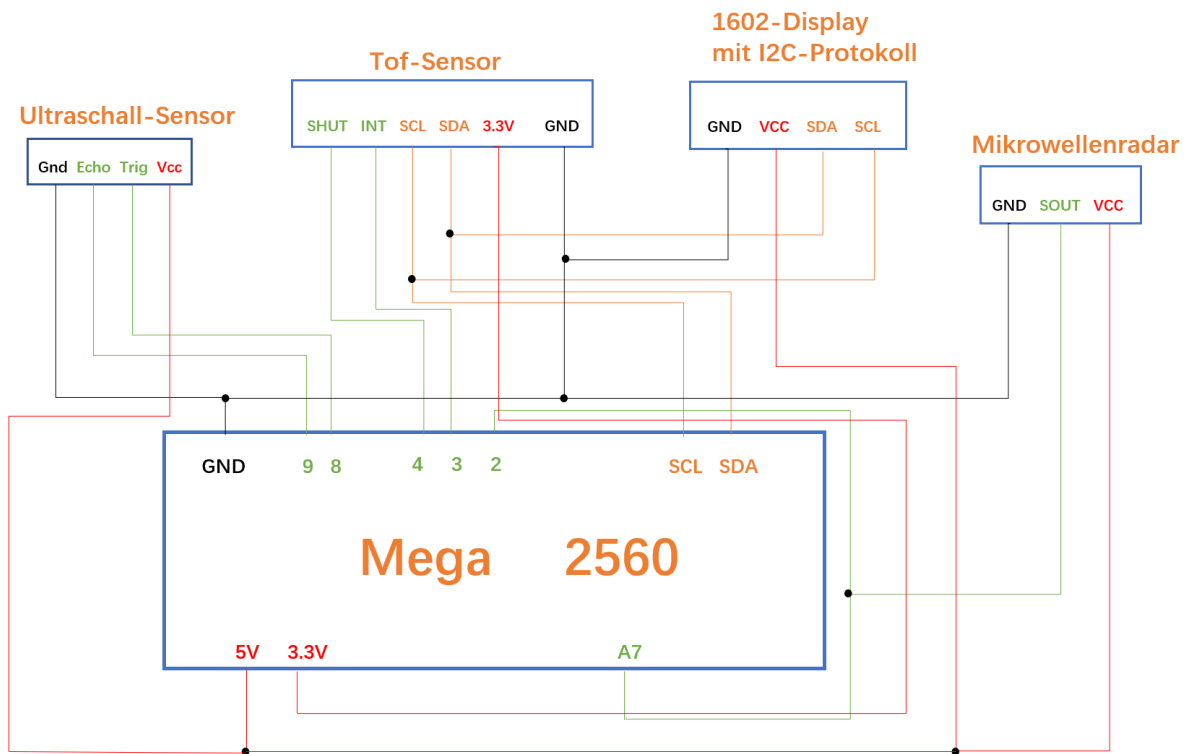


Bilder 10: 1602-Display [7]

6.2.2 Hardware Umsetzung

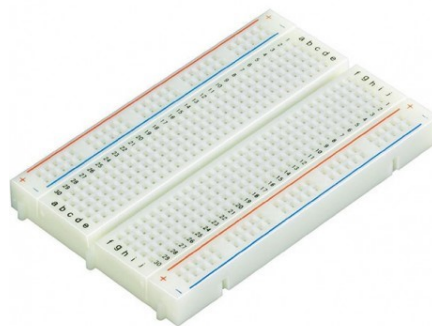
Die gesamte Hardware sollte gemäß dem Anschlussplan angeschlossen werden. Der Arduino Mega2560 muss mit einer 9-V-Batterie betrieben werden oder man kann auch mit USB-Kabel den Computer anschließen, um Arduino Mega2560 mit Strom zu versorgen.

Ein einfacher Schaltplan ist jetzt unten dargestellt.



Bilder 11: Schaltplan

Während des Verbindungsvorgangs müssen einige Stifte auf der Platine gleichzeitig mit mehreren Drähten verbunden werden. Jetzt ist es sehr wichtig, eine Steckplatine (*englisch breadboard*) zu verwenden. (Versuchsaufbau wird auf den Aufbau auf einer Steckplatine beschränkt.)



Bilder 12: Steckplatine [8]

6.3 Software

6.3.1 Softwarekonzept

Das Programm wird in der Software Arduino IDE (1.8.12) kompiliert.

Die Arduino Integrated Development Environment (IDE) ist eine plattformübergreifende Anwendung (für Windows, MacOS, Linux), die in Funktionen von C und C ++ geschrieben ist. Es wird verwendet, um Programme zu schreiben und auf Arduino-kompatible Boards hochzuladen, aber auch mit Hilfe von Kernen von Drittanbietern andere Entwicklungsboards von Anbietern. [9]

Damit die Anweisungen einfach angerufen werden, wird im Programm auf viele Bibliotheksdateien verwiesen, Z.B. <Wire.h>, <LiquidCrystal_I2C.h>,[10] <SparkFun_VL53L1X.h> [11]

6.3.2 Umsetzung des Softwarekonzepts

Ich schreibe das Programm gemäß den Verwendungsregeln der einzelnen Sensoren und Display und auch dem Pin-Port, an den sie angeschlossen im **Kapitel 5.2.2** sind. Nachfolgend kombiniere ich die alle Teile des Programms ohne Konflikte. Mein Programm wird in der folgenden Abbildung gezeigt.

```
#include <Wire.h>
#include <SparkFun_VL53L1X.h>
#include <LiquidCrystal_I2C.h>
#define TRIGGER_LEVEL 30 // ipm 165 level für Triggern
LiquidCrystal_I2C lcd(0x3F,16,2);
const int SHUTDOWN_PIN = 4;
const int INTERRUPT_PIN = 3; // für tof
const int TrigPin = 8;
const int EchoPin = 9; // für Ultraschall
const int TRIGGER_IN = A7;
const int RADAR_IN = 2;
const int RADAR_OUT = 13;
int trigger; // für ipm165
SFEVL53L1X distanceSensor;
float distance;
float cm;
float vultra;
double v,vm; // Variablentyp
void setup()
{ lcd.init(); // Initialisierung für LCD
  lcd.backlight();
  pinMode(TrigPin, OUTPUT); // pinMode definieren
  pinMode(EchoPin, INPUT);
  Wire.begin(); // die Wire-Bibliothek Initiiieren und mit dem I2C-Bus sich verbinden
  Serial.begin(9600); // stellt die Datenrate auf 9600 Bit/s ein
  Serial.println("Test");
  if (distanceSensor.begin() == 0) //Begin returns 0 (on a good init)
  {
    Serial.println("Sensor online!");
  }
  pinMode(RADAR_IN, INPUT); // pinMode definieren
  pinMode(RADAR_OUT, OUTPUT);
  pinMode(TRIGGER_IN, INPUT);
}
```

Bilder 13: Parameter und Setup

```

void loop()
{ // Tofteil
  distanceSensor.startRanging(); // Konfigurationsbytes Schreiben, um die Messung zu starten
  while (!distanceSensor.checkForDataReady()) {
    delay(1);
  }
  float distance = distanceSensor.getDistance(); // das Ergebnis der Messung vom Sensor bekommen (mm)
  distanceSensor.clearInterrupt();
  distanceSensor.stopRanging();
  distance = distance / 1000.0; // die Einheit Millimeter in Meter ändern
  Serial.print("ToFDistance: "); // Anzeige auf dem seriellen Monitor
  Serial.print(distance);
  Serial.print("m ");
  delay(20);
// Ultraschallteil
digitalWrite(TrigPin, LOW); // einen kurzen Zeitimpuls(Low High Low) an TrigPin senden
delayMicroseconds(2);
digitalWrite(TrigPin, HIGH);
delayMicroseconds(2);
digitalWrite(TrigPin, LOW);

/*pulseIn() Einheit ist Mikrosekunde,
die Geschwindigkeit = 344m/s, cm=344*100/1000000*pulseIn()/2,
ungefähr gleich pulseIn()/58.0*/
cm = pulseIn(EchoPin, HIGH) / 58.0; // die Echozeit in cm Konvertieren(die Dauer des Rückimpulses in ms.
// Wenn das Timeout nicht gelesen wurde, wird 0 zurückgegeben)*/

cm = (int(cm * 100.0)) / 100.0; // Zwei Dezimalstellen halten
float s1 = cm / 100; // die Einheit Zentimeter in Meter ändern
Serial.print("UltraschallDistance:"); // Anzeige auf dem seriellen Monitor
Serial.print(s1);
Serial.print("m");
Serial.println();
float realdistance = (distance + s1) / 2.0; // den Mittelwert vom Tof-Messwert und vom Ultraschall-Messwert berechnen
Serial.print("Realdistance: "); // Anzeige auf dem seriellen Monitor
Serial.print(realdistance);
Serial.print("m");
Serial.println();
float diff = abs(distance * 100 - cm); // nur für die Erklärung vom Wineklmessung im Kapitel 5.3(Nur als Referenz)
float Winkel =atan(diff / 6.6);
Winkel = Winkel * 180.0 / 3.1415926;
Serial.print("Winkel: "); // Anzeige auf dem seriellen Monitor
Serial.print(Winkel);
Serial.print("°");
Serial.println();
delay(300);
digitalWrite(TrigPin, LOW); // einen kurzen Zeitimpuls(Low High Low) an TrigPin senden
delayMicroseconds(2);
digitalWrite(TrigPin, HIGH);
delayMicroseconds(2);
digitalWrite(TrigPin, LOW);
cm = pulseIn(EchoPin, HIGH) / 58.0; // wie oben
cm = (int(cm * 100.0)) / 100.0;
float s2 = cm / 100;
vultra = (s2 - s1) / 0.3; // die Geschwindigkeit berechnen (mit zwei Entfernungen und eine Delayzeit)
vultra = abs(vultra); // den Absolutwert nehmen
Serial.print("ultraspeed:");
Serial.print(vultra);
Serial.print("m/s ");
//ipm 165teil
unsigned long T; // Periodendauer in us
double f; // Frequenz in MHz
vm=0;
trigger=analogRead(TRIGGER_IN);
analogWrite(RADAR_OUT, map(trigger, 0, 1023, 0, 255));
//while(trigger>TRIGGER_LEVEL);
analogWrite(RADAR_OUT,0);
for(int i=0;i<5;i++) // 5 Messungen nehmen
{
  T = pulseIn(RADAR_IN, HIGH) + pulseIn(RADAR_IN, LOW);
  f=1/(double)T; // f=1/T
  v=((f*1e6)/44.0); // 24 GHz Radar
  vm=max(v, vm); // Schleppzeiger,Maximalwert nehmen
}
vm=vm / 3.6; // die Einheit km/h in m/s ändern
Serial.print("RadarSpeed: ");
Serial.print(vm);
Serial.print("m/s");
Serial.println();
delay(10);
float realspeed = (vm + vultra) / 2.0; // den Mittelwert vom Radar-Messwert und vom Ultraschall-Messwert berechnen
Serial.print("Realspeed: ");
Serial.print(realspeed);
Serial.print("m/s");
Serial.println();
Serial.println();
delay(10);
}

```

Bilder 14: wichtige Teile vom Programm

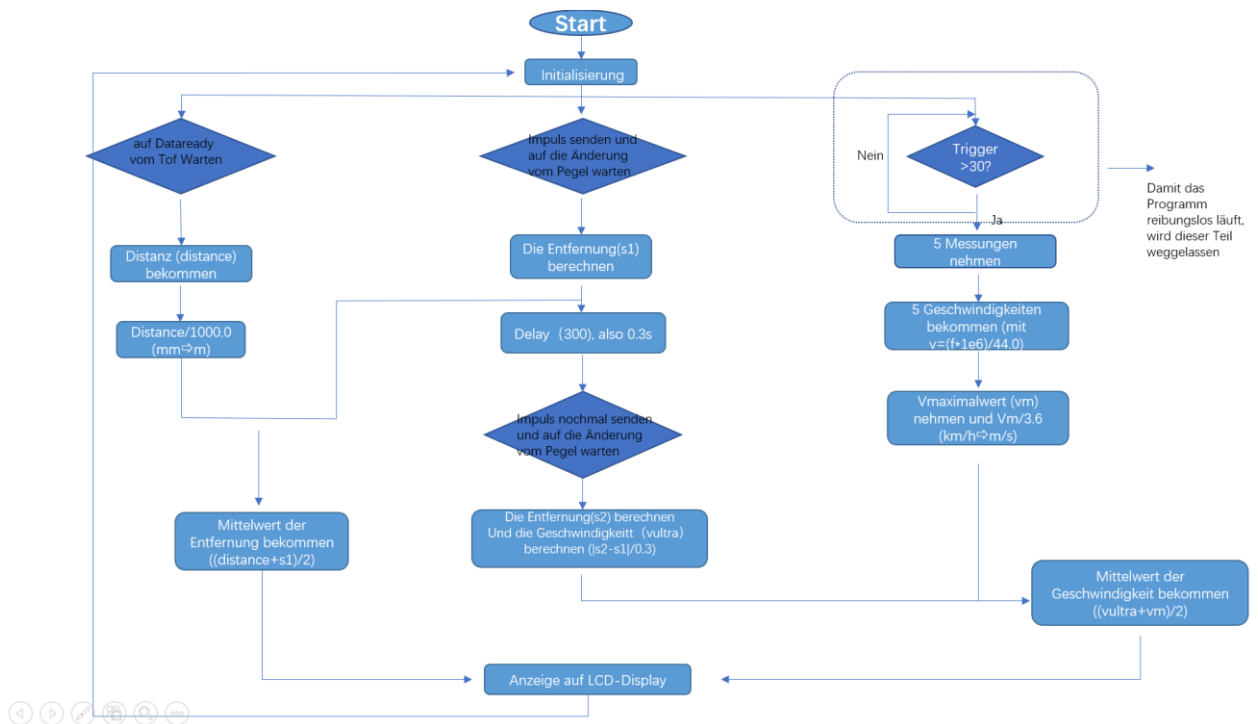
```

if (realdistance<0) // Anzeige auf LCD-Display
{
  lcd.setCursor(0, 0);
  lcd.print("Dis:");
  lcd.setCursor(5, 0);
  lcd.print(0);
  lcd.setCursor(10, 0);
  lcd.print("m");
  //lcd.setCursor(12, 0);
  //lcd.print(Winkel);
}
else
{
  lcd.setCursor(0, 0);
  lcd.print("Dis:");
  lcd.setCursor(5, 0);
  lcd.print(realdistance);
  lcd.setCursor(10, 0);
  lcd.print("m");
  //lcd.setCursor(12, 0);
  //lcd.print(Winkel);
}

lcd.setCursor(0, 1);
lcd.print("Spe:");
lcd.setCursor(5, 1);
lcd.print(realspeed);
lcd.setCursor(10, 1);
lcd.print("m/s");
delay(60);
}
    
```

Bilder 15: Anzeige-Teil vom Programm

Weil drei Sensoren gleichzeitig arbeiten müssen, ist es sehr kompliziert, dass man einen sehr klaren Programmablaufpläne schreibt.



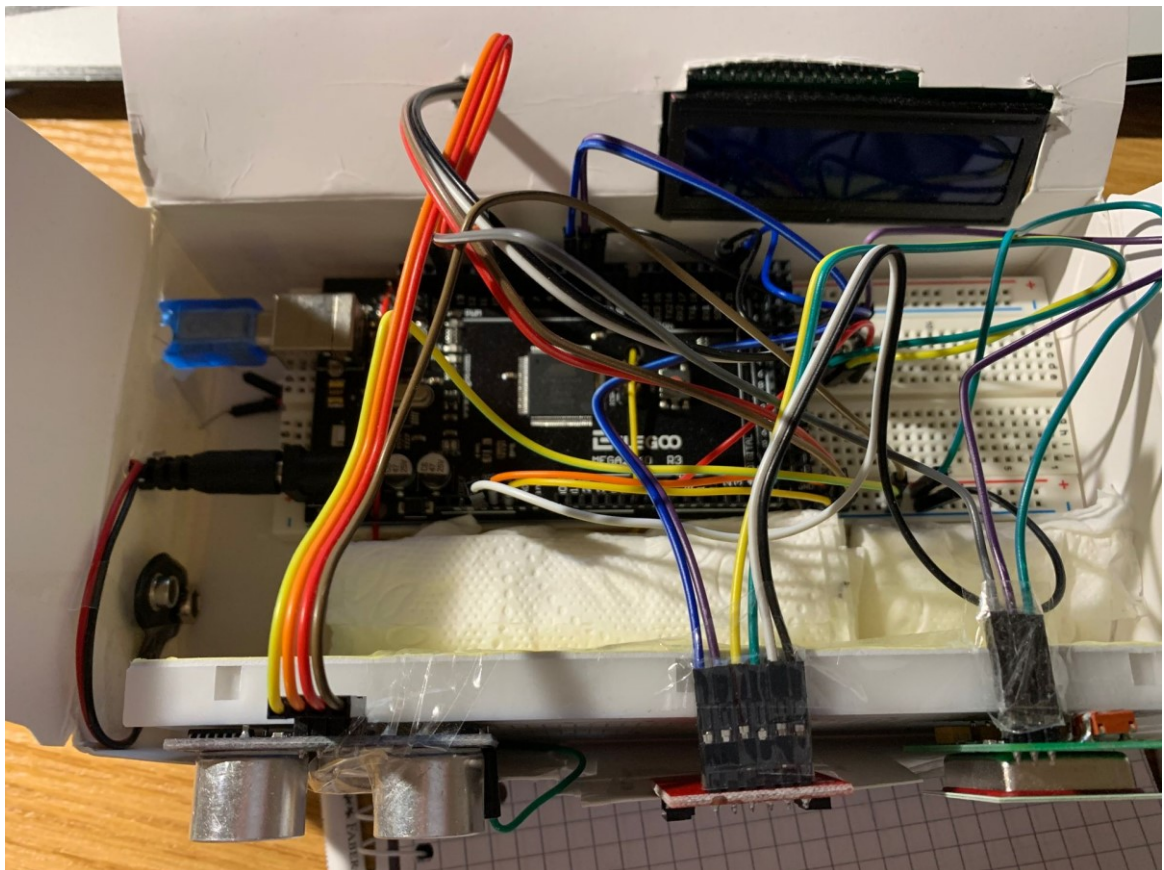
Bilder 16: Programmplan

7 Versuchsdurchführung

7.1 Versuchskonzept

Die Sensoren werden parallel angeordnet, die Messungen werden im Abstand von 0.02m bis 4m bei einem Winkel von 0 bis 20 Grad gemacht. Dadurch kann wie möglich sichergestellt werden, dass sich die drei Sensoren in derselben Ebene befinden.

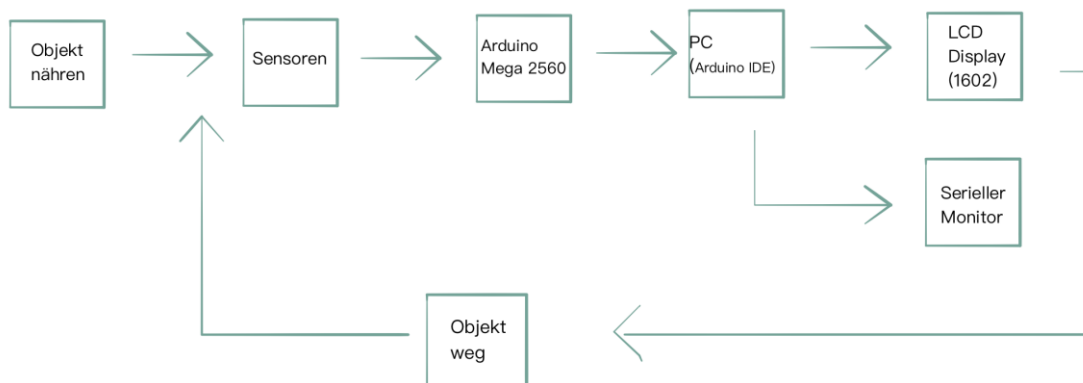
Alle Verbindungen wurden gemäß dem Schaltplan vom **Kapitel 6.2.2** abgeschlossen, und das physische Bild ist unten dargestellt



Bilder 17: Auflauf des Messgeräts

Der kleinen Kasten, der die glatte Oberfläche hat, wird mit unterschiedlichen Geschwindigkeiten näher senkrecht an den Sensor oder das Messgerät geschoben, dann erhält man alle Messwerte über den seriellen Monitor und das 1602-Display. Anschlie-

ßend zieht man die Kasten auf die gleiche Weise vom Sensor weg, dann wird ein paar Messwert erneut aufgenommen.



Bilder 18: Versuchsablauf

Wie erwartet sollten alle Sensoren normal funktionieren, das Display und der serielle Monitor zeigen die Werte an, und der von der trigonometrischen Funktion berechnete Winkel sollte zwischen 0 und 20 Grad liegen.

7.2 Darstellung von möglichen Fehlereinflüssen

1. Wenn sich das Objekt zu nahe am Messgerät befindet oder sich das Objekt mit niedriger Geschwindigkeit bewegt, kann der **Mikrowellenradar-Sensor** möglicherweise Geschwindigkeiten von etwas anderes falsch messen. In diesem Fall sollte zuerst der vom Ultraschall-Sensor gemessene Wert verwendet werden.
2. Man kann nicht garantieren, dass die Kasten beim Schieben von Hand zu hundertprozentig senkrecht zum Messgerät stehen kann und beim Kippen kann der Winkel sehr leicht 20 Grad zu überschreiten sein. Die zu diesem Zeitpunkt gemessene Entfernung und Geschwindigkeit von dem **Ultraschall-Sensor** sind äußerst ungenau. In diesem Fall sollte der vom Tof-Sensor und Mikrowellenradar-Sensor auf dem seriellen Monitor gemessene Wert Vorrang haben, sondern nicht der Wert im 1602-Display oder der Wert vom Ultraschall-Sensor.

3. Stellt man wie möglich sicher, dass der Versuch in Innenräumen oder im Wind durchgeführt wird. Übermäßiges Sonnenlicht im Freien wirkt sich stark auf den **Tof-Sensor** aus und übermäßige Windgeschwindigkeit beeinträchtigt den **Ultraschall-Sensor**.

7.3 Versuchsdurchführung

Arduino Meag-2560 ist schon mit einem USB-Kabel an den Computer gut angeschlossen worden, und der kompilierte Sketch wird auf Mega-2560 hochgeladen. Dann wird der Versuch die ganze Zeit drinnen durchgeführt. Die Ergebnisse werden unten in Tabellenform gezeigt.

Abstandswert vom Tof-Sensor (m)	Abstandswert vom Ultraschall-Sensor (m)	Mittelwert (m)	Geschwindigkeitswert vom Ultraschall-Sensor (m/s)	Geschwindigkeitswert vom Mikrowellenradar (m/s)	Mittelwert (m/s)
0.58	0.58	0.58	0.12	0.24	0.18
0.40	0.39	0.40	0.26	0.30	0.28
0.17	0.16	0.17	0.16	0.32	0.24
0.10	0.11	0.11	0.00	0.31	0.16
0.10	0.11	0.11	0.00	0.33	0.17
0.09	0.11	0.10	0.03	0.21	0.13
0.26	0.28	0.27	0.28	0.29	0.28
0.50	0.51	0.51	0.17	0.34	0.26
0.62	0.62	0.62	0.01	0.36	0.19
0.62	0.64	0.63	0.05	0.36	0.21
0.54	0.53	0.54	0.18	0.23	0.21

Tabelle 5: Test beim Winkel 0-20 Grad

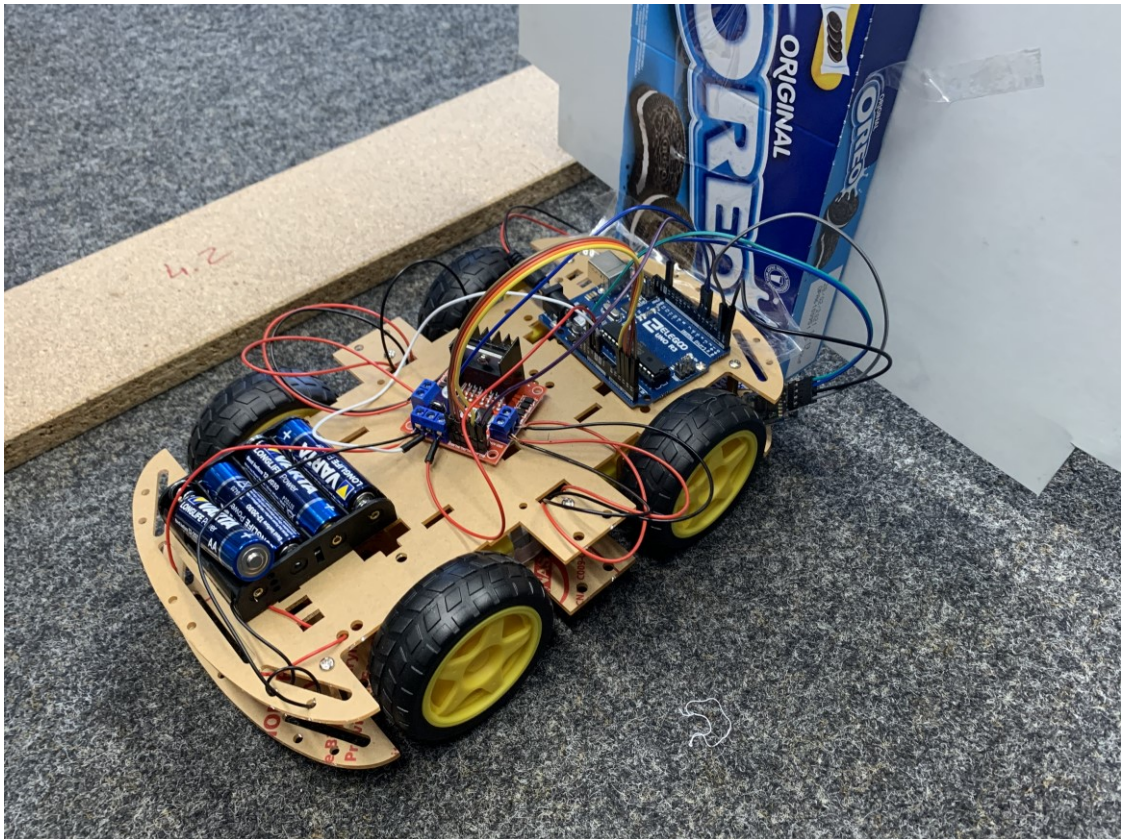
Zehn Versuchsreihen sind durchgeführt worden. (Z.B. Gleichmäßige Geschwindigkeit oder plötzliche Geschwindigkeitsänderung), die Ergebnisse ähneln den beiden obigen zwei Bildern. Die Daten zu diesen beiden Zahlen entsprechen in etwa meinen Erwartungen, also, der Winkel ist schon innerhalb von 20 Grad gesteuert worden. Aber es gibt aber noch einige kleine Probleme.

Abstandswert vom Tof-Sensor (m) [↵]	Abstandswert vom Ultraschall-Sensor (m) [↵]	Mittelwert (m) [↵]	Geschwindigkeitswert vom Ultraschall-Sensor (m/s) [↵]	Geschwindigkeitswert vom Mikrowellenradar (m/s) [↵]	Mittelwert (m/s) [↵]
0.58 [↵]	0.58 [↵]	0.58 [↵]	0.12 [↵]	0.24 [↵]	0.18 [↵]
0.40 [↵]	0.39 [↵]	0.40 [↵]	0.26 [↵]	0.30 [↵]	0.28 [↵]
0.17 [↵]	0.16 [↵]	0.17 [↵]	0.16 [↵]	0.32 [↵]	0.24 [↵]
0.10 [↵]	0.11 [↵]	0.11 [↵]	0.00 [↵]	0.31 [↵]	0.16 [↵]
0.10 [↵]	0.11 [↵]	0.11 [↵]	0.00 [↵]	0.33 [↵]	0.17 [↵]
0.09 [↵]	0.11 [↵]	0.10 [↵]	0.03 [↵]	0.21 [↵]	0.13 [↵]
0.26 [↵]	0.28 [↵]	0.27 [↵]	0.28 [↵]	0.29 [↵]	0.28 [↵]
0.50 [↵]	0.51 [↵]	0.51 [↵]	0.17 [↵]	0.34 [↵]	0.26 [↵]

Bilder 19: kleiner Fehler

Wenn der Kasten zum Stillstand neigt, misst der Mikrowellenradar-Sensor meinen Finger oder Arm, der versehentlich zittert, sondern nicht die Geschwindigkeit des Kastens, also 0 km/h.

Nach diesem Versuch verwendeten wir im Labor ein ferngesteuertes Fahrzeug und Lego-Auto, um ein paar neue Versuche durchzuführen.



Bilder 20: ferngesteuertes Fahrzeug

Das ferngesteuerte Fahrzeug kann per Mobiltelefon oder Computer über das Bluetooth-Modul *HC-05* gesteuert werden. Er kann die Geschwindigkeit weitgehend konstant halten. Da er jedoch gleichzeitig von vier Motoren durch Motor-driver-Module *L298N* angetrieben wird, kann er manchmal nicht die ideale Gerade fahren. Und wenn man die Geschwindigkeit ändern möchte, muss die Spannung über PWM geändert werden, dann muss man etwas im Programm ändern und es erneut hochladen, was nicht so einfach oder praktisch ist.



Bilder 21: L298N [12]

Die Höchstgeschwindigkeit des ferngesteuerten Fahrzeugs beträgt ca. 0,4 m / s und Die Geschwindigkeit kann durch PWM auf 0,3m/s und 0,4m/s eingestellt werden. Das Oberflächenmaterial des Messobjekts ist Aluminium (Alufolie) und Schwarz Schleifpapier. Alle drei Sensoren werden verwendet, und mehrere Reihen von Geschwindigkeitsdaten gleichzeitig zu messen. Eine Reihe von Daten wird ausgewählt, um die folgende Tabelle zu erstellen. Je nach Tabelleninhalt (0,3m/s) kann die Diagramme der Abweichung verschiedener Sensoren in verschiedenen Entfernungen erstellt werden.

(Achtung: In jeder Tabelle haben die alle drei Sensoren unterschiedliche Geschwindigkeitsdaten bei den neun unterschiedlichen Entfernungen. Beim Ultraschall-Sensor und ToF-Sensor haben die alle Daten einen Startpunkt (Z.B. S1, S3, S5...) und einen Endpunkt (Z.B. S2, S4, S6...), um dann die momentane Geschwindigkeit des gewünschten Punkts abzuschätzen. Wenn dieses Symbol / angezeigt wird, bedeutet es, dass der Sensor gerade keine Daten liest oder der Datenfehler zu groß ist oder die Daten falsch sind.)

1. Die Geschwindigkeit vom ferngesteuerten Fahrzeug wird auf **0.3m/s** gesetzt und das Oberflächenmaterial des Messobjekts ist **Schwarz Schleifpapier**.

	S1 (m)	S2 (m)	V1 (m/s)	S3 (m)	S4 (m)	V2 (m/s)	S5 (m)	S6 (m)	V3 (m/s)	S7 (m)	S8 (m)	V4 (m/s)
Ultra	0.45	0.57	0.28	0.99	1.13	0.28	1.41	1.55	0.29	1.95	2.11	0.32
Tof	0.42	0.45	0.32	0.92	0.98	0.28	1.48	1.56	0.31	2.03	2.12	0.34
Radar	0.5		0.22	1.0		0.30	1.5		0.38	2.0		0.39

S9 (m)	S10 (m)	V5 (m/s)	S11 (m)	S12 (m)	V6 (m/s)	S13 (m)	S14 (m)	V7 (m/s)	S15 (m)	S16 (m)	V9 (m/s)
/	/	/	/	/	/	/	/	/	/	/	/
/	/	/	/	/	/	/	/	/	/	/	/
2.5		0.34	3.0		0.39	3.5		0.38	4.0		0.31

Tabelle 6: Geschwindigkeit der unterschiedlichen Entfernung

2. Die Geschwindigkeit vom ferngesteuerten Fahrzeug wird auf **0.3m/s** gesetzt und das Oberflächenmaterial des Messobjekts ist **Aluminium (Alufolie)**.

	S1 (m)	S2 (m)	V1 (m/s)	S3 (m)	S4 (m)	V2 (m/s)	S5 (m)	S6 (m)	V3 (m/s)	S7 (m)	S8 (m)	V4 (m/s)
Ultra	0.49	0.62	0.24	0.98	1.12	0.26	1.43	1.59	0.28	1.87	2.02	0.27
Tof	0.42	0.49	0.29	1.02	1.08	0.27	1.46	1.52	0.26	1.99	2.05	0.28
Radar	0.5		0.33	1.0		0.32	1.5		0.33	2.0		0.34

S9 (m)	S10 (m)	V5 (m/s)	S11 (m)	S12 (m)	V6 (m/s)	S13 (m)	S14 (m)	V7 (m/s)	S15 (m)	S16 (m)	V9 (m/s)
/	/	/	/	/	/	/	/	/	/	/	/
2.52	2.58	0.31	2.93	3.05	0.8	/	/	/	/	/	/
2.5		0.33	3.0		0.29	3.5		0.28	4.0		0.29

Tabelle 7: Geschwindigkeit der unterschlichen Entfernung

3. Die Geschwindigkeit vom ferngesteuerten Fahrzeug wird auf **0.4m/s** gesetzt und das Oberflächenmaterial des Messobjekts ist **Schwarz Schleifpapier**.

	S1 (m)	S2 (m)	V1 (m/s)	S3 (m)	S4 (m)	V2 (m/s)	S5 (m)	S6 (m)	V3 (m/s)	S7 (m)	S8 (m)	V4 (m/s)
Ultra	0.33	0.57	0.38	0.73	0.98	0.39	1.41	1.65	0.38	2.03	2.28	0.31
Tof	0.46	0.56	0.43	0.92	1	0.33	1.51	1.60	0.36	1.99	2.05	0.26
Radar	0.5		0.38	1.0		0.44	1.5		0.49	2.0		0.47

S9 (m)	S10 (m)	V5 (m/s)	S11 (m)	S12 (m)	V6 (m/s)	S13 (m)	S14 (m)	V7 (m/s)	S15 (m)	S16 (m)	V9 (m/s)
/	/	/	/	/	/	/	/	/	/	/	/
/	/	0.28	/	/	/	/	/	/	/	/	/
2.5		0.62	3.0		0.52	3.5		0.38	4.0		0.4

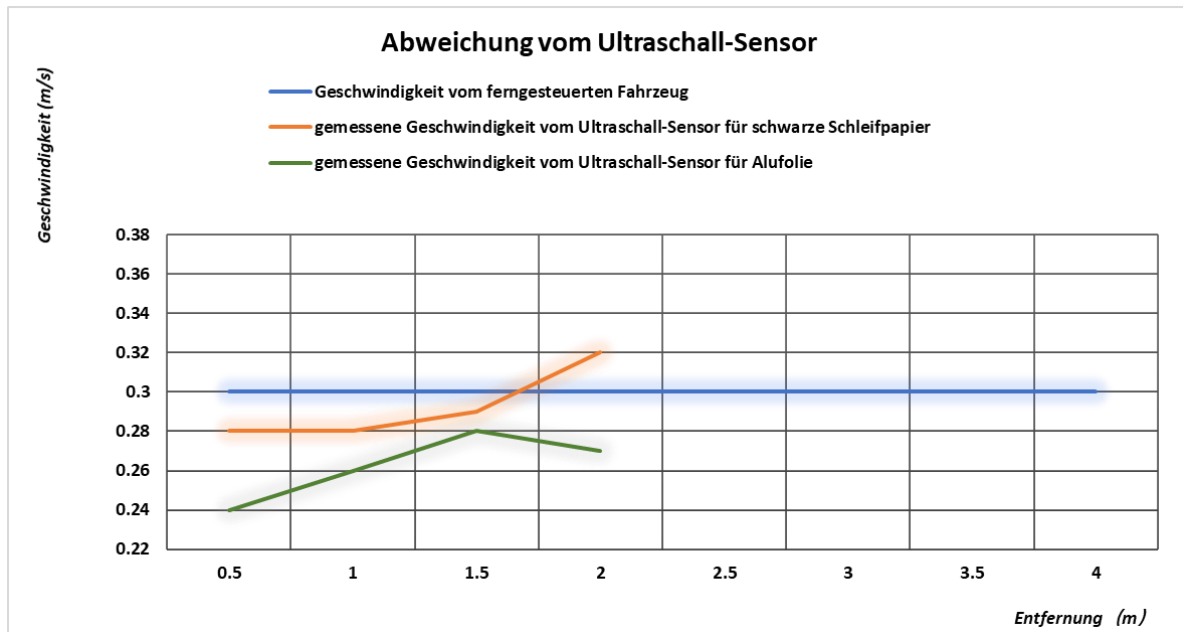
Tabelle 8: Geschwindigkeit der unterschlichen Entfernung

4. Die Geschwindigkeit vom ferngesteuerten Fahrzeug wird auf **0.4m/s** gesetzt und das Oberflächenmaterial des Messobjekts ist **Aluminium (Alufolie)**.

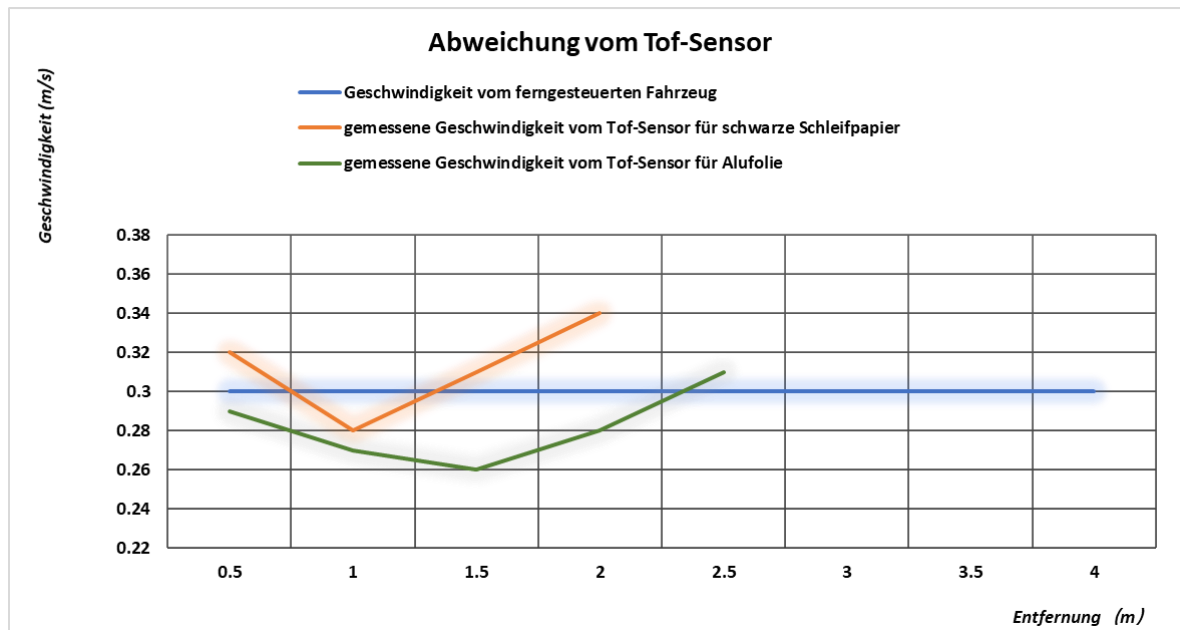
	S1 (m)	S2 (m)	V1 (m/s)	S3 (m)	S4 (m)	V2 (m/s)	S5 (m)	S6 (m)	V3 (m/s)	S7 (m)	S8 (m)	V4 (m/s)
Ultra	0.40	0.63	0.33	0.94	1.18	0.35	1.25	1.49	0.37	1.94	2.21	0.41
Tof	0.42	0.51	0.37	1.20	1.11	0.37	1.49	1.57	0.35	1.94	2.03	0.40
Radar	0.5		0.37	1.0		0.37	1.5		0.41	2.0		0.40

S9 (m)	S10 (m)	V5 (m/s)	S11 (m)	S12 (m)	V6 (m/s)	S13 (m)	S14 (m)	V7 (m/s)	S15 (m)	S16 (m)	V9 (m/s)
/	/	/	/	/	/	/	/	/	/	/	/
2.4	2.45	0.22	2.85	2.98	0.88	/	/	/	/	/	/
2.5		0.51	3.0		0.57	3.5		0.38	4.0		0.43

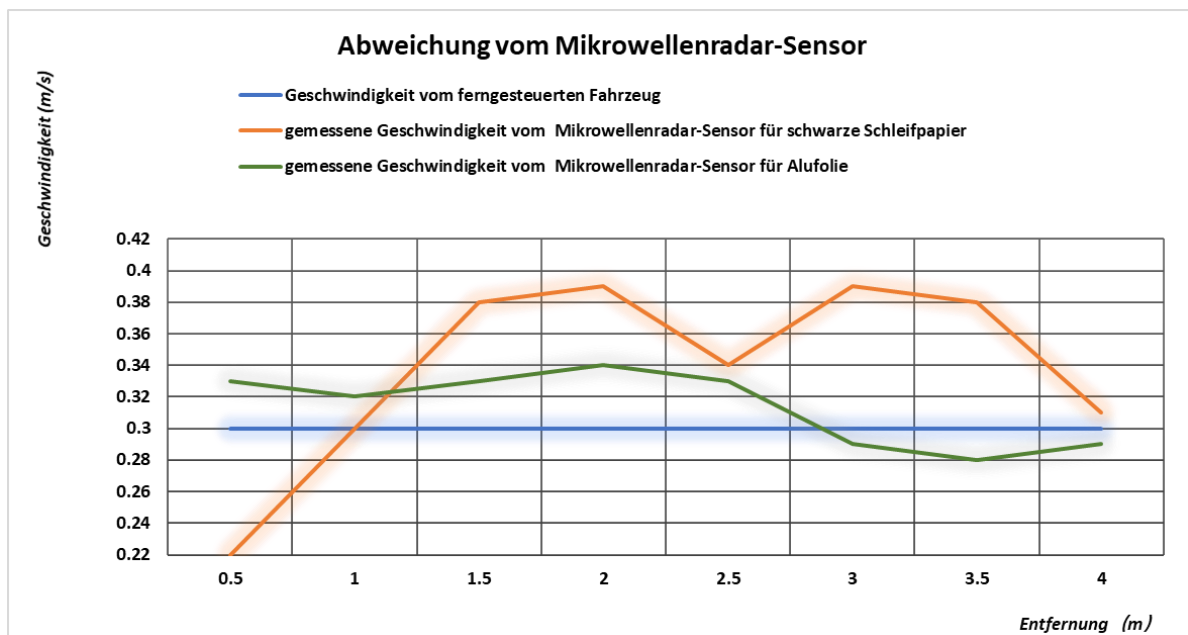
Tabelle 9: Geschwindigkeit der unterschlichen Entfernung



Bilder 22: Abweichung vom Ultraschall-Sensor



Bilder 23: Abweichung vom ToF-Sensor



Bilder 24: Abweichung vom Mikrowellenradar-Sensor

Das Lego-System kann mithilfe der Umlaufbahn das Auto geradeaus fahren lassen. Man kann auch die Geschwindigkeit des Autos mit dem gelben Dreh-Knopf einstellen. Das Lego-Auto wird von der Startgeschwindigkeit, der Spurreibung und anderen Faktoren beeinflusst. Es ist unmöglich sicherzustellen, dass das Auto vollständig gleiche Geschwindigkeit hält (schon einen Startabstand von etwa dreißig Zentimetern einstellen, d.h. 0,3 m). Aber man kann auch die alle drei Sensoren verwenden, um die Geschwindigkeit des Autos gleichzeitig zu messen und etwas Ergebnisse können durch Vergleich der Messungen gezogen werden.



Bilder 25: Lego-Auto

5. Der erste Versuch besteht darin, Alufolie zu verwenden, um die Oberfläche des gemessenen Objekts zu bedecken. Drei Versuchsreihen werden durchgeführt, nämlich mithilfe Lego-Auto mit niedriger Geschwindigkeit, hoher Geschwindigkeit und mit kontinuierlicher variabler Geschwindigkeit. Die Daten der drei Sensoren in unterschiedlichen Abständen werden in den folgenden Tabellen notiert. (Einheit ist m/s.)

	0.5 m	1 m	1.5 m	2 m	2.5 m
Ultra	0.26	0.25	0.25	0.21	0.20
Tof	0.22	0.25	0.24	0.20	0.20
Radar	0.22	0.22	0.29	0.30	0.23

Tabelle 10: Lego-Auto mit niedriger Geschwindigkeit

	0.5 m	1 m	1.5 m	2 m	2.5 m
Ultra	0.49	0.78	/	1.01	/
Tof	0.47	0.70	0.86	0.92	0.96
Radar	0.33	0.48	0.48	0.78	0.54

Tabelle 11: Lego-Auto mit hoher Geschwindigkeit

	0.5 m	1 m	1.5 m	2 m	2.5 m
Ultra	/	0.68	0.61	0.44	0.30
Tof	0.68	0.65	0.58	0.43	0.34
Radar	0.79	0.65	0.52	0.58	0.37

Tabelle 12: Lego-Auto mit unterschiedlichen Geschwindigkeiten

6. Beim zweiten Versuch wird die Alufolie von dem Kartonpapier ersetzt. Die andere drei Versuchsreihen werden durchgeführt, also auch mithilfe Lego-Auto mit niedriger Geschwindigkeit, hoher Geschwindigkeit und mit kontinuierlicher variabler Geschwindigkeit. Die Repräsentative Daten werden ausgewählt und in der folgenden Tabelle aufgezeichnet.

	0.5 m	1 m	1.5 m	2 m	2.5 m
Ultra	0.15	0.19	0.19	0.12	0.30
Tof	0.16	0.19	0.21	0.07	0.39
Radar	0.20	0.18	0.18	0.21	0.49

Tabelle 13: Lego-Auto mit niedriger Geschwindigkeit

	0.5 m	1 m	1.5 m	2 m	2.5 m
Ultra	0.59	0.81	0.81	1.00	—
Tof	0.42	0.64	0.80	0.86	0.65
Radar	0.41	0.58	0.82	1.05	1.11

Tabelle 14: Lego-Auto mit hoher Geschwindigkeit

	0.5 m	1 m	1.5 m	2 m	2.5 m
Ultra	0.47	0.33	0.37	0.46	0.39
Tof	0.53	0.41	0.43	0.37	0.84
Radar	0.63	0.66	0.40	0.45	0.48

Tabelle 15: Lego-Auto mit unterschiedlichen Geschwindigkeiten

Wenn die Geschwindigkeit groß ist oder sich die Geschwindigkeit stark ändert, sind die Fehler von Tof-Sensor und Ultraschall-Sensor unabhängig vom Material groß. Weil die beide Sensoren die Geschwindigkeit im Uhrzeigersinn nicht im wahrsten Sinne des Hardware-Niveaus angeben können, sondern nur den Abstand zwischen den beiden Messwerten, dann berechnen sie diese Geschwindigkeit. Aber Mikrowellenradar-Sensor kann eine stabile Leistung aufrechterhalten, wenn die Geschwindigkeit groß ist oder sich schnell ändert.

7.4 Kombination aller Sensoren und Auswertung

Nach Eigenschaften oder Vor- und Nachteile der Sensoren werden die alle drei Sensoren kombiniert.

Bei der Entfernungsmessung, Im Allgemeinen ist der Tof-Sensor (*VI53L1X*) genauer als Ultraschall-Sensor (*HC-SR04*). Die beide Sensoren haben gleich Reichweite aber der Tof-Sensor ist stabiler als der Ultraschall-Sensor. Der Tof-Sensor hat stärkere Anti-Interferenz-Fähigkeit, und die Winkeländerung hat keinen Einfluss auf ihn. Aber wenn man den Tof-Sensor verwendet, muss man das Material des Messobjekts berücksichtigen. Die meisten transparente Objekte können vom Tof-Sensor nicht genau gemessen. Unter der Voraussetzung, dass das Material des Objekts bekannt ist, werde ich

empfohlen, den Tof-Sensor zur Messung der Entfernung zu verwenden. Tof-Sensor ist schneller als der Ultraschall-Sensor. Zuerst ist die Geschwindigkeit von Schallwelle gleich nur 340m/s, beim Licht ist die $3 \cdot 10^8$ m/s. Außerdem ist die Schallwelle nicht ständig, sondern nur in Form von Impulsen zu emittieren, weil Klänge sich oft gegenseitig stören. Was die Sensoren selbst betreffen, muss der Tof-Sensor (*VI53L1X*) mit I2C-Kommunikationsschnittstelle verbunden werden, und nur für 3.3V Logikpegel (Ultraschall-Sensor für 5V Logikpegel) geeignet.

Bei der Geschwindigkeitsmessung ist es schon in 5.2 erwähnt worden, dass man den Ultraschall-Sensor (*HC-SR04*) aus nächster Nähe, niedriger Geschwindigkeit und vertikaler Richtung (beim Winkel 0°) verwenden kann. In den übrigen Fällen, werde ich empfohlen, den Mikrowellenradar-Sensor (*ipm-165*) zur Messung der Geschwindigkeit zu verwenden. Aber man kann nicht ignorieren, dass der Mikrowellenradar-Sensor nicht auf stationäre Objekte reagiert.

Die Winkelmessung kann nicht vollständig realisiert werden. Wie aus der Tabelle 4 ersichtlich ist, geeignet mein Plan theoretisch nur für 0~20 Grad. Eigentlich ist der Wert (also 20 Grad) sehr ideal. Unter realen Bedingungen, also Z.B. im Leben, nicht alle Objekte sind glatt und flach. In den meisten Fällen ist die Oberfläche des Objekts uneben, Z.B.: Selbst kleine Falten an der Kleidung können 20 Grad schon überschreiten. Das ist ein Problem, das für diesen Ultraschall-Sensor (*HC-SR04*) wegen seiner physikalischen Eigenschaften nicht optimiert werden kann.

Dieses Problem besteht nicht nur bei der Winkelmessung, sondern auch bei der Entfernungsmessung und der Geschwindigkeitsmessung. Man verwendet Ultraschall-Sensor und Tof-Sensor, Entfernung zu messen, Ultraschall-Sensor und Mikrowellenradar-Sensor, die Geschwindigkeit zu messen. Im Labor kann man versuchen, den Winkel des Objekts innerhalb von 20 Grad wie möglich zu halten. Aber im wirklichen Leben gibt es jedoch keine Möglichkeit, den Winkel des gemessenen Objekts zu garantieren. Bei der tatsächlichen Messung kann man nur sagen, dass das Messgerät (mit Ultraschall-Sensor, Tof-Sensor und Mikrowelle-Sensor) so senkrecht wie möglich zum Messobjekt stehen sollte. Natürlich ist dieses Messgerät nur ein unreifes Modell, und wenn man die Leistung und Genauigkeit der tatsächlichen Produkte verbessern möchte, müssen mehr Sensoroptionen vorhanden sein.

Schließlich werden Tabellen verwendet, um ihre Vor- und Nachteile intuitiver darzustellen.

Ultraschall-Sensor (HC-SR04)	
Vorteile	Nachteile
Einfach zu verwenden	Langsam messen
Energieeffizient	vom Winkel beeinflusst
kostengünstig	schlechte Anti-Interferenz-Fähigkeit
stabil	Ultraschall kann absorbiert werden
	Schallwellen können sich gegenseitig stören

Tabelle 16: Vor- und Nachteile beim Ultraschall-Sensor

Tof-Sensor (VL53L1X)	
Vorteile	Nachteile
Starke Anti-Interferenz-Fähigkeit	transparentes Objekt nicht messen kann
Nicht vom Winkel beeinflusst	Kann nicht im starken Sonnenlicht im Freien verwendet werden
Schnell und genau messen	
Unterstützt das I2C-Protokoll	
Einfach zu verwenden	

Tabelle 17: Vor- und Nachteile beim Tof-Sensor

Mikrowellenradar-Sensor (ipm-165)	
Vorteile	Nachteile


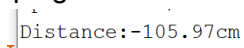
Sehr empfindlich gegenüber sich bewegenden Objekten	Stationäre Objekte können nicht erkannt werden
schnelle Reaktionsgeschwindigkeit	kein kontinuierliches Messergebnis
	Nicht für niedrige Geschwindigkeiten im Nahbereich geeignet.

Tabelle 18: Vor- und Nachteile beim Mikrowellenradar-Sensor

8 Zusammenfassung, Schlussfolgerung

Im gesamten System wird die Leistung des Ultraschallsensors aufgrund der Reflexionseigenschaften von Schallwellen stark vom Winkel beeinflusst, ist jedoch nicht für Fernmessung geeignet (es ist nicht einfach, den Winkel zu steuern). Wenn man es auf das Produkt anwenden möchte, versuchen man, es im kurzen Abstand und senkrecht zum Objekt zu halten, z.B. beim Umkehradar (englisch *Parking sensor*).

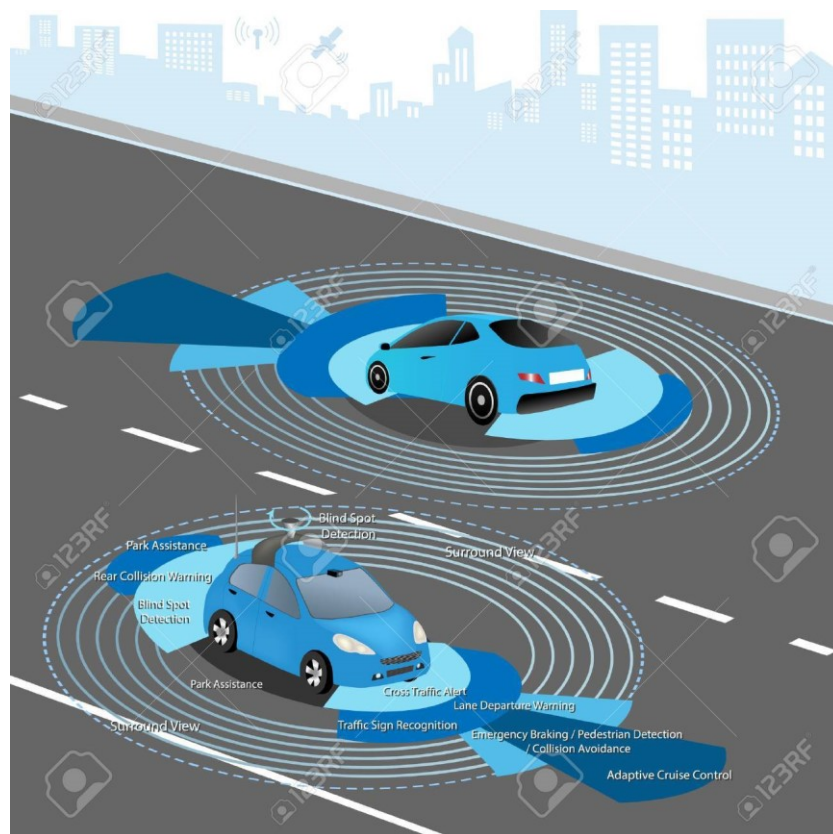
Ich denke jedoch, dass Ultraschall-Sensor als Werkzeug zur Winkelbestätigung verwendet werden kann, ohne an dem vom gesamten System angegebenen Entfernungswert teilzunehmen. Solange der von diesem Sensor angegebene Wert nicht -105 beträgt, kann davon ausgegangen werden, dass das Objekt vertikal zum Sensor zeigt. (Das *HC-SR04*, das ich erhalten habe, meldet einen Fehler, wenn der Winkel größer als 20 Grad ist. Der Fehler spiegelt sich in der Messwertanzeige um -105 wider. Siehe Tabelle 4, im **Kapitel 5.3**) 

Die beiden anderen Sensoren (ToF-Sensor Mikrowellenradar-Sensor) sind relativ stabil, wenn sie gewöhnlichen sich bewegenden Objekten zugewandt sind, und sie Grundsätzlich die erwarteten Anforderungen erfüllen. Jedoch emittiert ToF-Sensor (*VL53L1X*) immerhin das Licht. Wenn die Oberfläche des gemessenen Objekts zu glatt ist oder einen Spiegeleffekt zeigt, und wenn sein Winkel zu groß ist, wird ToF-Sensor nach den Eigenschaften des Lichts wahrscheinlich das gesamte reflektierte Licht nicht empfangen werden. Dies führt dazu, dass der von ToF-Sensor angegebene Messwert falsch ist oder sehr stark schwankt. Und natürlich ist ToF-Sensor nicht für vollständig transparente Objekte geeignet.

Ich bin sehr zufrieden mit der Leistung vom Mikrowellenradar-Sensor (*jpm-165*), er hat fast das professionelle Niveau erreichter, und er ist sehr stabil und genau, fast unabhängig vom Winkel. Das einzige Problem besteht darin, zu versuchen, andere sich bewegende Objekte nicht in der Nähe des gemessenen Objekts zu platzieren, da er wahrscheinlich die Geschwindigkeit dieses sich bewegenden Objekts anstelle des Zielobjekts liest. Natürlich ist der Sensor nur zum Bewegen von Objekten geeignet, kann aber die Richtung der Objektbewegung nicht bestimmen, er ist nicht empfindlich gegenüber stationären Objekten. Außerdem ist eine ordnungsgemäße, vorsichtig und ESD-konforme Verarbeitung des Moduls sehr wichtig, damit eine vorzeitige Beschädigung der empfindlichen Mischdiode vermieden werden.

9 Ausblick

Bei dieser Arbeit kann die Winkelmessung mit diesen drei Sensoren nicht erreicht werden, deshalb hoffe ich, mehrere Sensoren in alle Richtungen hinzuzufügen, damit dieses System in alle Richtungen oder um 360 Grad Objekte anstatt nur in eine Richtung überwachen kann. Oder eine kostengünstige Lösung besteht darin, einen Servomotor (Z.B. SG90) unter diesem System zu installieren, damit er sich um 360 Grad drehen kann.



Bilder 26: Schematische Darstellung mehrerer Sensoren [13]

Wenn man dieses System auf tatsächliche Produkte anwenden möchte, wäre es besser, ein paar genaueren Sensoren mit höherer Reichweite zu wählen. Ich hoffe auch, diesem System LED-Leuchten und Summer hinzuzufügen. Wenn das gemessene Objekt zu nahe ist oder die Annäherungsgeschwindigkeit zu hoch ist und das System bei der aktuellen Geschwindigkeit innerhalb von zwei Sekunden kollidiert, senden der Summer und das Licht einen Kollisionsalarm.

Literaturverzeichnis

- [1] „Abbildung1“ 26,05,2020. [Online]. Available:
<http://www.opticsjournal.net/Mobile/PostDetails/PT180109000201h0kQn?code=3>
- [2] „Doppler-Effekt“ 28,05,2020. [Online]. Available:
<https://verkehrslexikon.de/Texte/GSRadar01.php>
- [3] „Abbildung6“ 29,05,2020. [Online]. Available:
<https://www.3dzyk.cn/thread-40-1-1.html>
- [4] „Abbildung7“ 29,05,2020. [Online]. Available:
<https://nirtjo.com/product/ultrasonic-sensor-hc-sr04/>
- [5] „Abbildung8“ 29,05,2020. [Online]. Available:
<https://www.exp-tech.de/en/sensors/distanceapproximation/8907/sparkfun-distance-sensor-breakout-4-meter-vl53l1x-qwiic>
- [6] „Abbildung9“ 29,05,2020. [Online]. Available:
<https://shop.weidmann-elektronik.de/index.php?page=product&info=29>
- [7] „Abbildung10“ 29,05,2020. [Online]. Available:
<https://modulink.de/ML/d01d07/>
- [8] „Abbildung12“ 30,05,2020. [Online]. Available:
<https://christine30322.wordpress.com/2018/03/25/%E4%B8%80%E9%80%B1%E4%B8%80%E7%AF%87%E9%BA%B5%E5%8C%85%E6%9D%BF%E5%9F%BA%E6%9C%AC%E4%BB%8B%E7%B4%B9/>
- [9] „Arduino-IDE“ 05,06,2020. [Online]. Available:
https://en.wikipedia.org/wiki/Arduino_IDE
- [10] „<LiquidCrystal_I2C.h>“ 08,06,2020. [Online]. Available:
https://github.com/johnrickman/LiquidCrystal_I2C
- [11] „<SparkFun_VL53L1X.h>“ 08,06,2020. [Online]. Available:
https://github.com/sparkfun/SparkFun_VL53L1X_Arduino_Library
- [12] „Abbildung21“ 09,06,2020. [Online]. Available:
<https://www.smart-prototyping.com/L298N-Dual-H-bridge-Motor-Driver-Board>
- [13] „Abbildung26“ 11,06,2020. [Online]. Available:

https://www.123rf.com/photo_71265648_stock-vector-automobile-sensors-use-in-self-driving-cars-camera-data-with-pictures-radar-and-lidar-autonomous-dri.html

[14] „Mega2560(Abbildung27)“ 01,07,2020. [Online]. Available:
<https://www.roboternetz.de/community/threads/72004-MEGA-wie-viel-digitale-IO-Pins-sind-verf%C3%BCgbar>
<https://www.arrow.de/research-and-events/articles/arduino-mega-2560-overview>

[15] „Uno r3(Abbildung28)“ 01,07,2020. [Online]. Available:
<https://www.circuito.io/blog/arduino-uno-pinout/>
<https://www.jameco.com/jameco/workshop/circuitnotes/cn-arduino-uno.html>

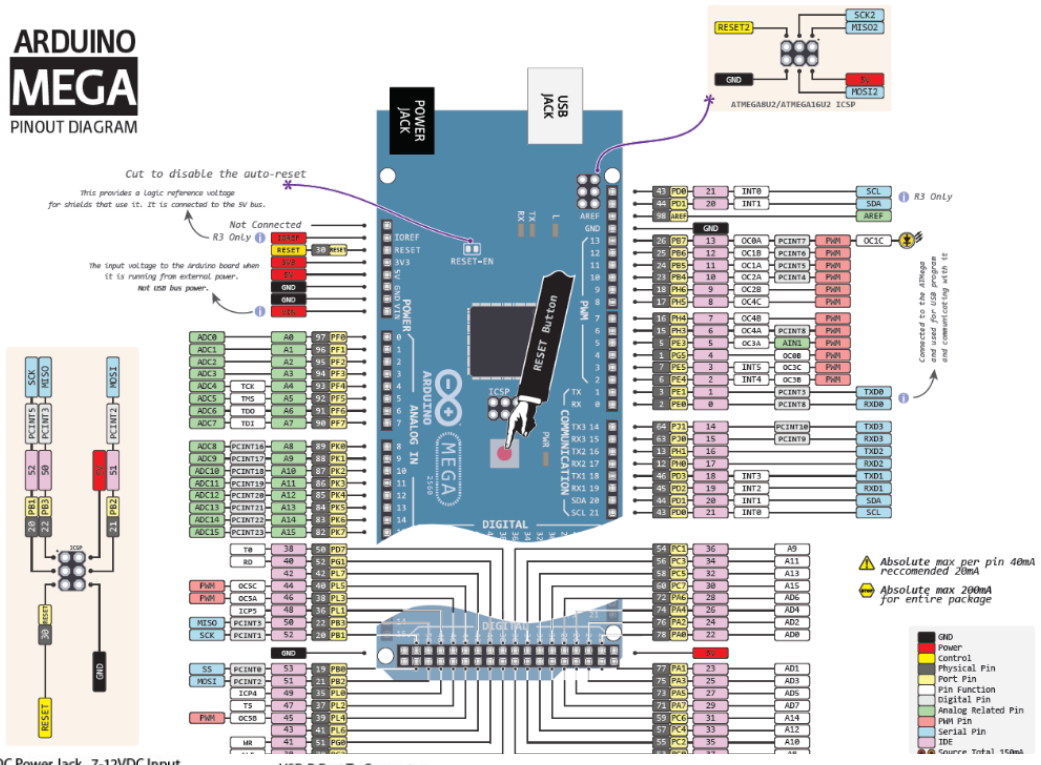
[16] „Abbildung29“ 01,07,2020. [Online]. Available:
<https://www.youtube.com/watch?v=X5achE10rCI>

Anlagen

Arduino Mega2560	IXI
Arduino Uno r3	IXII
Brennen	XIII

Arduino Mega 2560 PIN Diagramm

ARDUINO MEGA PINOUT DIAGRAM



DC Power Jack 7-12VDC Input 2.1mm x 5.5mm Male Center Positive USB-B Port To Computer



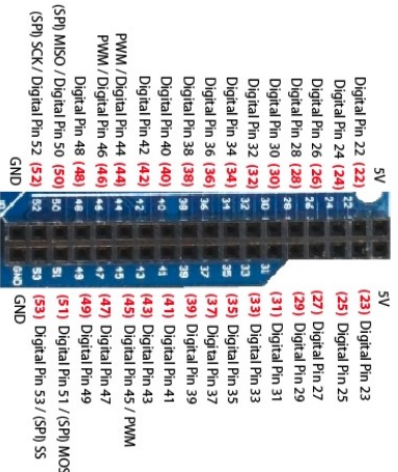
Mega 2560 Pinout



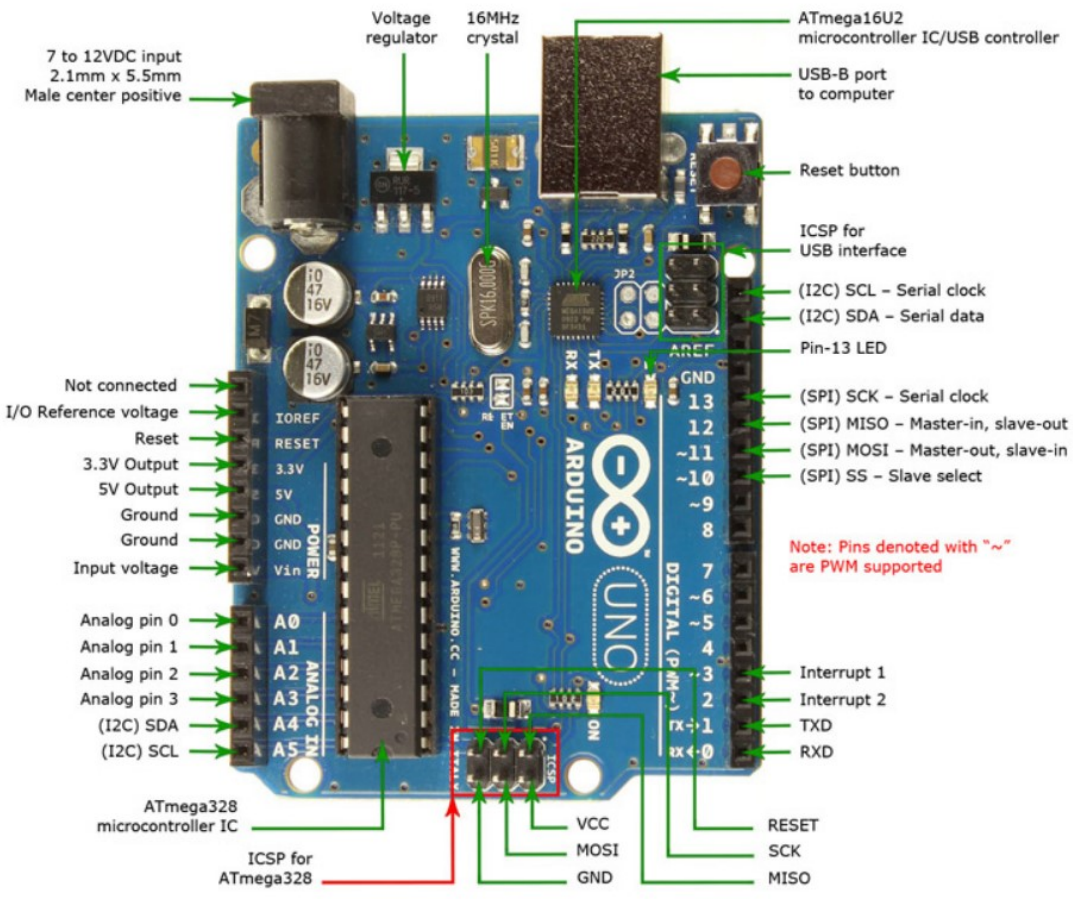
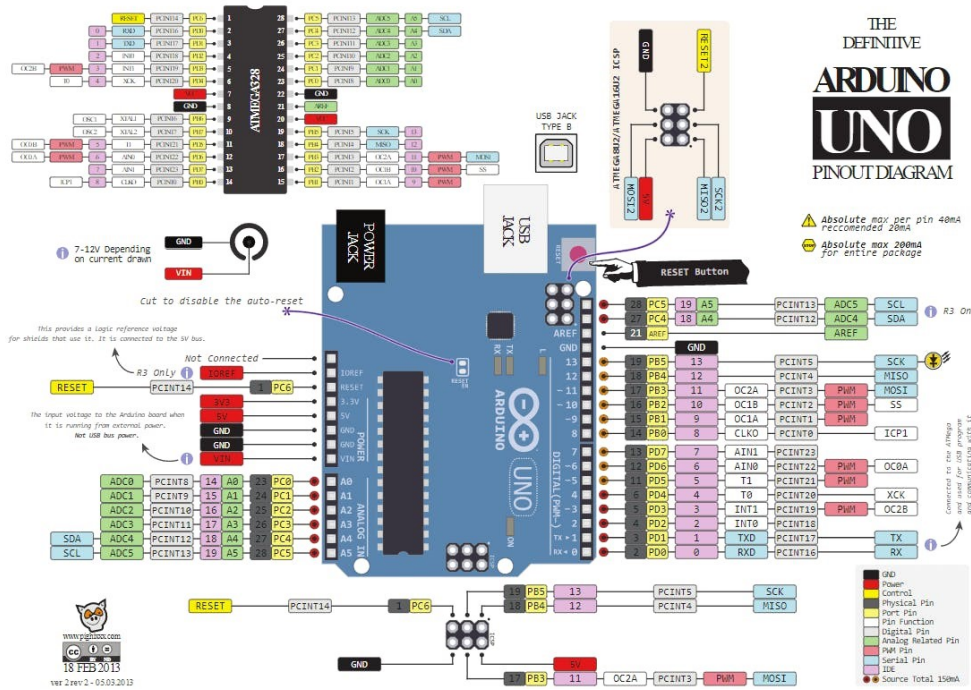
- No Connection
- I/O Reference Voltage for shields
- Reset Input
- 3.3V Output @ 50mA
- 5V Output or Input
- Ground
- Ground
- 7-12V Output or Input
- Analog Pin 0 / Digital Pin 54 (A0)
- Analog Pin 1 / Digital Pin 55 (A1)
- Analog Pin 2 / Digital Pin 56 (A2)
- Analog Pin 3 / Digital Pin 57 (A3)
- Analog Pin 4 / Digital Pin 58 (A4)
- Analog Pin 5 / Digital Pin 59 (A5)
- Analog Pin 6 / Digital Pin 60 (A6)
- Analog Pin 7 / Digital Pin 61 (A7)
- Analog Pin 8 / Digital Pin 62 (A8)
- Analog Pin 9 / Digital Pin 63 (A9)
- Analog Pin 10 / Digital Pin 64 (A10)
- Analog Pin 11 / Digital Pin 65 (A11)
- Analog Pin 12 / Digital Pin 66 (A12)
- Analog Pin 13 / Digital Pin 67 (A13)
- Analog Pin 14 / Digital Pin 68 (A14)
- Analog Pin 15 / Digital Pin 69 (A15)

Red numbers in parenthesis are the name to use when referencing that pin. Analog pins are referenced as A0 thru A15 even when using as digital I/O

- (I2C) SCL - Serial Clock
- (I2C) SDA - Serial Data
- Analog Reference Voltage
- Ground
- (13) Digital Pin 13 / PWM / Connected to on-board LED
- (12) Digital Pin 12 / PWM
- (11) Digital Pin 11 / PWM
- (10) Digital Pin 10 / PWM
- (9) Digital Pin 9 / PWM
- (8) Digital Pin 8 / PWM
- (7) Digital Pin 7 / PWM
- (6) Digital Pin 6 / PWM
- (5) Digital Pin 5 / PWM
- (4) Digital Pin 4 / PWM
- (3) Digital Pin 3 / PWM / Ext Int 5
- (2) Digital Pin 2 / PWM / Ext Int 4
- (1) Digital Pin 1 / Serial Port 0 TXD (Main Serial Port)
- (0) Digital Pin 0 / Serial Port 0 RXD (Main Serial Port)
- (14) Digital Pin 14 / Serial Port 3 TXD
- (15) Digital Pin 15 / Serial Port 3 RXD
- (16) Digital Pin 16 / Serial Port 2 TXD
- (17) Digital Pin 17 / Serial Port 2 RXD
- (18) Digital Pin 18 / Serial Port 1 TXD / Ext Int 3
- (19) Digital Pin 19 / Serial Port 1 RXD / Ext Int 2
- (20) Digital Pin 20 / (I2C) SDA / Ext Int 1
- (21) Digital Pin 21 / (I2C) SCL / Ext Int 0



Bilder 27: alle Pinout vom Mega2560 [14]

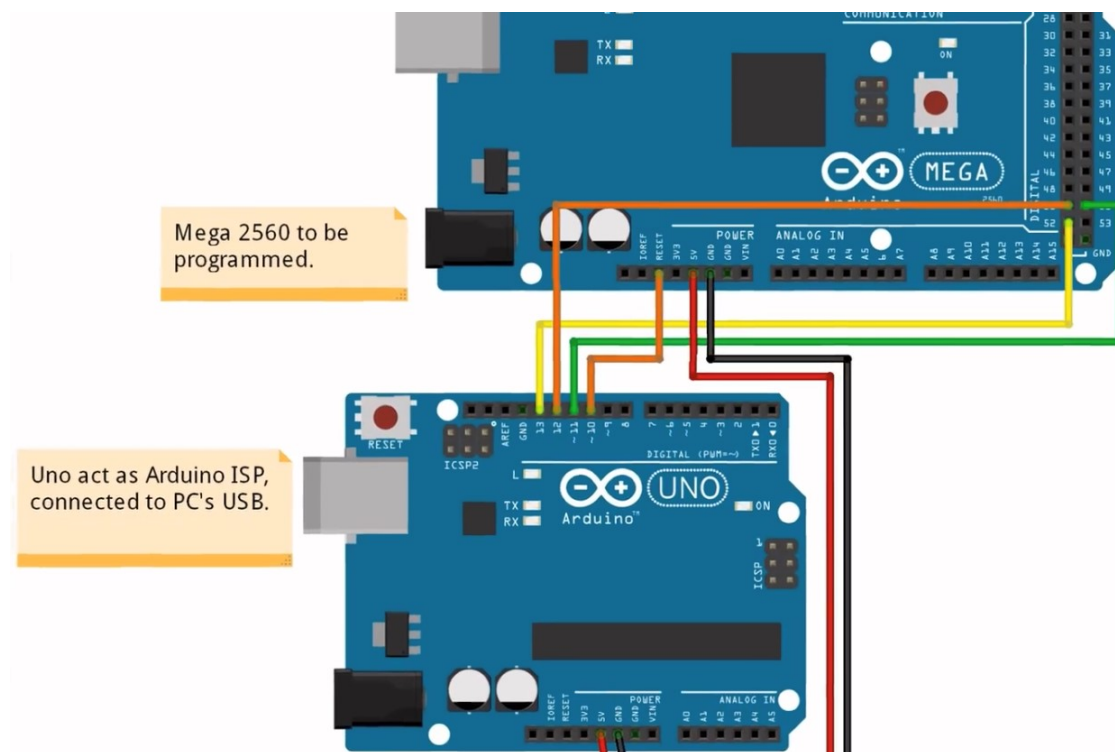


Bilder 28: alle Pinout vom Uno r3 [15]

Brennen

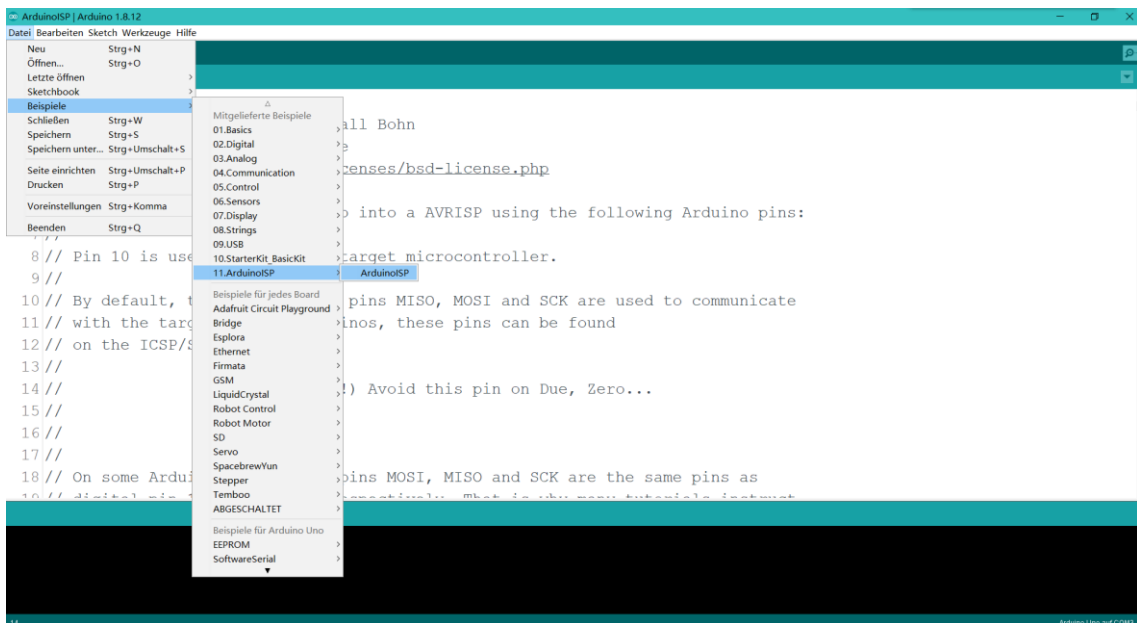
Nachdem ich die Leitung angeschlossen und das Programm kompiliert hatte, lud ich es zum ersten Mal auf Arduino Mega2560 hoch, aber ich finde, dass es beim Hochladen ganz nicht funktioniert. Es gab kein Problem im Programm deshalb kam ich zu dem Schluss, dass mit dem Arduino etwas schiefgelaufen ist. Ich fand im Internet den Informationen darüber, wie man ein Arduino Uno verwendet, um Arduino Mega2560 zu brennen, und versuchte es dann.

Zuerst muss den Mega2560 mit Uno verbunden werden.

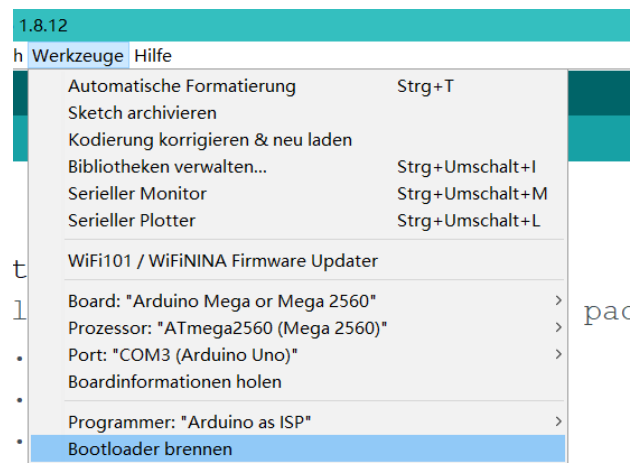
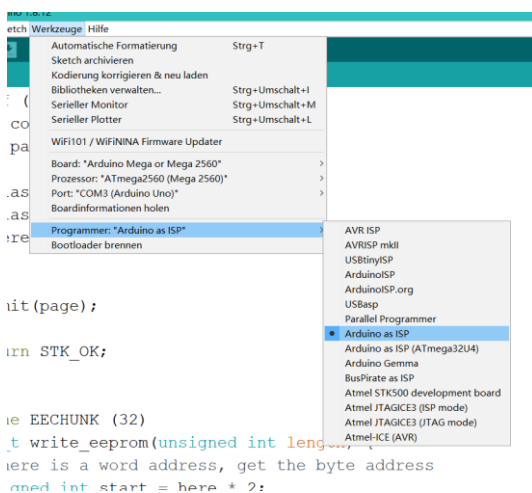


Bilder 29: Schaltplan von Brennen [16]

Dann muss man den korrekten Sketch in Arduino IDE wählen und Arduino auf isp setzen. Abschließend wartet man auf das erfolgreiche Brennen.



Bilder 30: Wählen von Sketch



Bilder 31: Setzen und Brennen

Nach erfolgreichem Brennen kann Arduino Mega2560 erneut hochgeladen werden, aber es funktioniert nur beim ersten Hochladen. Wenn man der andere Sketch auf diesem Board, also auf Mega2560 hochladen möchte, geht es wieder nicht und muss man wiederum das Board brennen. Deshalb kaufe ich ein neue Arduino Mega2560.

Nachfolgend müssen alle Leitungsdrähte verbunden werden. Der Schaltplan in 5.2.2 sieht sehr einfach aus, aber da zu viele Drähte vorhanden sind, können beim eigentlichen Verbindungsvorgang leicht Fehler gemacht werden, darauf muss man achten.

Eigenständigkeitserklärung

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbstständig und nur unter Verwendung der angegebenen Literatur und Hilfsmittel angefertigt habe. Stellen, die wörtlich oder sinngemäß aus Quellen entnommen wurden, sind als solche kenntlich gemacht. Diese Arbeit wurde in gleicher oder ähnlicher Form noch keiner anderen Prüfungsbehörde vorgelegt.

Ort, Datum

Vorname Nachname