



Projekt 1001 – Abschlussprojekt

HBFIT21

Dokumentation zum Projekt (Menschlicher Höhenmesser inklusive BMI-Rechner)

Abschlussprojekt zur Erreichung des Abschlusses staatlich geprüfte Assistentin/staatlich geprüfter Assistent Fachrichtung Informationstechnik

Abgabetermin: 02.02.2023

Schüler:

Name: Loran Demiroglu

E-Mail: lorandemiroglu04@gmail.com

Tel. Nummer: 0176 477 707 03

Betreuer:

Name: Herr Gemmel

E-Mail: stefan.gemmel@drsneuwied.de

Hiermit erkläre ich, dass ich die vorliegende Dokumentation selbstständig verfasst und keine anderen als die angegebenen Hilfsmittel benutzt habe. Teile einer Hausarbeit, die wörtlich oder sinngemäß aus anderen Quellen übernommen wurden, werden durch Angaben zu ihrer Herkunft gekennzeichnet. Dies gilt auch für Zeichnungen, Skizzen, Bilder und Quellen aus dem Internet.

Inhaltsverzeichnis

1	Beschreibung	1-10
1.1	Allgemeine Erläuterung des Projektes	1
1.2	Stand der Technik.....	1-10
1.2.1	Gesamtprojekte	1-2
1.2.2	Teilprojekte	3-10
1.2.3	Schlussfolgerung	10
1.3	Vision des Produktes	11
1.3.1	Dimensionierung.....	11
1.3.2	Komponenten und spezielle Funktionen	11
2	Detaillierte Planung.....	12-15
2.1	Technologieschema.....	12
2.2	UML Flowchart.....	13
2.3	Materialliste	14
2.4	Schaltplan.....	15
3	Projektziele & Meilensteine.....	15-17
3.1	MUSS Ziele	15
3.2	KANN Ziele	16
3.3	NICHT Ziele	16
3.4	Meilensteinplan.....	16
3.4.1	Meilensteine fürs Projektergebnis	16
3.4.2	Meilensteine der Projektdurchführung.....	16
3.4.3	Checkliste Prototyp.....	17
4	Durchführung	17-21
4.1	Realisierung.....	17-20
4.2	Testfälle	20-21
5	Feedback.....	21-22
5.1	Projektabschluss.....	21-22
5.1.1	Vergleich: ursprüngliche Planung zu erreichten Zielen.....	21
5.1.2	Ursachenanalyse der Abweichungen.....	21
5.1.3	Selbstreflexion: Lerntransfer für zukünftige Projekte	21
5.1.4	Begründeter Notenvorschlag	22
6	Anhang.....	22-26

1. Beschreibung

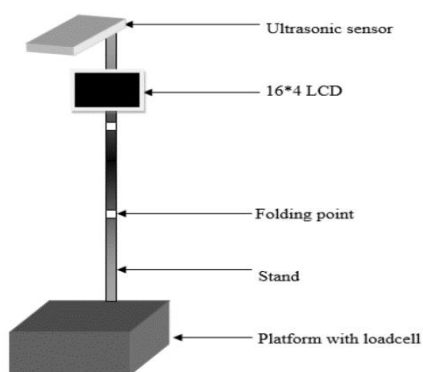
1.1 Allgemeine Erläuterung des Projektes

Jeder kennt es. Man will herausfinden, wie groß man eigentlich ist oder, ob man unter- oder übergewichtig für seine Größe ist. Normalerweise würde der Mensch direkt auf ein Maßband zugreifen, um seine Größe zu messen, was jedoch sehr ungenau und zeitaufwendig sein kann. Sinn und Zweck des Projektes ist es daher, die Größe eines Menschen genauestens zu bestimmen und daraufhin seinen Body-Mass-Index (BMI) zu ermitteln. Meine Motivation dieses Projekt zu konstruieren war es eine Lösung zu finden seine Größe schneller und genauer zu ermitteln als mit herkömmlichen Mitteln, wie das Maßband. Des Weiteren fand ich es wichtig den BMI-Rechner einzubauen, weil es ein wichtiger Indikator für den Gesundheitszustand einer Person ist.

Die Bedienung des Projektes ist simpel. Um seine Größe zu bestimmen, muss man erst einmal vor einer Tür stehen, weil sich der Ultraschallsensor, der dafür zuständig ist Abstände zu messen über der Tür befindet. Dieser Sensor wird in einer Höhe von 226 Zentimetern in einer kleinen selbst angefertigten Box mit Hilfe von Magneten aufgestellt und im Programmcode auch angegeben. Wenn sich eine Person nun unter den Sensor aufstellt, piept ein Piezo Buzzer einen hohen Ton und dann misst der Sensor die Entfernung zwischen ihm und der Person. Daraufhin wird die Höhe des aufgestellten Sensors (226 Zentimeter) mit dem Abstand zwischen ihm und dem Menschen subtrahiert. Dieser Wert ist dann der genaue Wert wie groß die Person ist und der Wert wird schlussendlich auf einem 20 x 4 LCD-Bildschirm angegeben. Daraufhin hat man die Möglichkeit sein Gewicht mit Hilfe von Buttons anzugeben. Der LCD-Bildschirm und die Buttons sind in einer Box auf Augenhöhe aufgestellt. In der Box befindet sich außerdem der Arduino nano. Nachdem das Gewicht angegeben wurde, wird der BMI (indem man das Körpergewicht in Kilogramm durch das Quadrat der Körpergröße in Metern) berechnet. Falls man möchte, kann sich ebenfalls mit dem drücken eines weiteren Buttons eine BMI-Skala ansehen, um zu gucken, ob die Person zu viel für seine Größe wiegt.

1.2 Stand der Technik

1.2.1 Gesamtprojekte



Tragbarer BMI-Rechner mit Gewichtssensor

Dieses Projekt ist ebenfalls ein BMI-Rechner mit dem Zusatz, dass das Gewicht durch einen Gewichtssensor ermittelt wird. Um seinen BMI zu rechnen, muss sich die Person auf die Platte (unten) hinstellen, da sich dort der Gewichtssensor befindet. Durch eine Aluminiumstange befindet sich oberhalb der Person ein Ultraschallsensor, welcher die Körpergröße ermittelt. Diese beiden Werte werden daraufhin mit der BMI-Formel berechnet und auf dem LCD-Bildschirm, welcher an der Stange befestigt wurde, angezeigt.

(Link: [TRAGBARER BMI-RECHNER BASIEREND AUF MIKROCONTROLLER-| Elektronik-Werkstatt \(electronicworkshops.com\)](http://www.electronicworkshops.com))

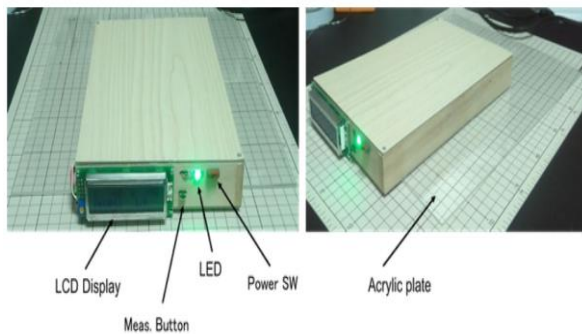


DIY Arduino Höhenmessgerät

Dieses Projekt ist ein Höhenmessgerät, was für Menschen bestimmt ist. Dafür wird die Box, wo Ultraschallsensor, LCD-Bildschirm, Wippschalter, Batterie, Buzzer und Arduino enthalten sind auf eine bestimmte Höhe mit doppelseitigem Klebestoff an der Wand befestigt. Der Ultraschallsensor guckt dabei nach unten, um die Entfernung zwischen ihm und der Person zu ermitteln. Um die Höhe der Person schlussendlich herauszufinden, wird

(Link: [DIY Arduino Höhenmessgerät - Arduino Project Hub](#))

die Höhe, in der die Box aufgestellt wurde mit der Entfernung zwischen Sensor und Person subtrahiert und auf dem 16 x 2 LCD-Bildschirm angezeigt. Zudem ermöglicht es der integrierte Buzzer ein Piepsenden Ton hörbar zu machen, wenn sich eine Person unter den Sensor hinstellt.



Höhenmessgerät mit VL53L0X Sensor

Dieses abgebildete Projekt misst die Höhe eines Menschen mit Hilfe eines VL53L0X Abstandssensors, anstatt eines Ultraschallsensors. Dieser Sensor ermittelt die Entfernung, indem es Infrarotimpulse verwendet. Zudem verfügt es über ein 3-Achsen-Gyroskop mit 3-Achsen-Beschleunigungsmesser, welcher die

(Link: [Using VL53L0X With Arduino to Measure Height - Electronics-Lab.com](#))

horizontale Richtung misst, und die X- und Y-Achse angibt. Um die Höhe eines Menschen zu erfassen, muss man in diesem Projekt nur die Acrylplatte, die unter der Box befestigt ist auf den Kopf der Person stellen.

BMI-Rechner mit der Programmiersprache JavaScript

Dieses Projekt ist im Gegensatz zu den anderen vorgestellten Projekten kein Arduino Projekt. Der Unterschied dabei ist, dass dieses Projekt ein reines Programmierprojekt ist und kein Mikrocontroller oder sonstige Elektronik, wie Sensoren benötigt. Um seinen BMI zu erfahren, muss man seine Höhe (in cm) und sein Gewicht (in kg) angeben und daraufhin auf „rechnen“ drücken. Schlussendlich wird der BMI angezeigt.

(Link: [Design a BMI Calculator using JavaScript - GeeksforGeeks](#))

1.2.2 Teilprojekte

In diesem Abschnitt geht es um die Teilkomponenten, die für das Projekt benötigt werden und dazu werden Alternativen aufgelistet.

Bildschirme:

LCD 20 x 4 Zeichen



Dieses LCD verfügt über 20 Zeichen in der Breite und 4 Zeilen in der Höhe, was bedeutet, dass es insgesamt 80 Zeichen auf einmal anzeigen kann. Die Hintergrundbeleuchtung ist frei wählbar zwischen Grün und Blau. Durch die mitgelieferte I2C-Schnittstelle, die an der Rückseite des Displays gelötet werden muss, spart man sich den Kabelsalat und spart I/O-Ports für den Arduino. Der Preis für das Display beträgt inklusive I2C-Schnittstelle 11.29 Euro. Falls man keine Lust hat die Schnittstelle zu löten kann man es vorgelötet für den gleichen Preis kaufen.

(Link: [HD44780 2004 LCD Display Bundle 4x20 Zeichen mit I2C Schnittstelle – AZ-Delivery](#))

2,4 Zoll TFT LCD Touch Display Shield



Dieses spezielle Display ermöglicht es Bilder und Zeichen darzustellen. Zudem ist es mit einem Touchscreen ausgestattet, womit der Benutzer mit seinen Fingern oder mit dem beigelieferten Stift direkte Befehle ausführen kann. Das Display hat eine Größe von 2,4 Zoll und hat eine weiße Hintergrundbeleuchtung. Geeignet ist dieses Touch Display, um Grafiken auszugeben. Das Display benötigt mindestens eine Spannung von 3,3V und verträgt es bis 5V. Insgesamt nimmt es 12 Pins vom Arduino ein. Der Preis beträgt inklusive dem Stift 13.99 Euro.

(Link: [2,4 Zoll TFT LCD Touch Display Shield 5V / 3,3V kompatibel mit Arduino – AZ-Delivery](#))



LCD 16 x 2 Zeichen

Eine weitere Variante für ein Bildschirm ist dieses LCD-Display mit 16 Zeichen in der Breite und 2 Zeilen in der Höhe. Genau wie beim ersten vorgestellten Bildschirm kann man sich die Hintergrundbeleuchtung zwischen Grün und Blau aussuchen. Eine I2C-Schnittstelle ist mitgeliefert damit man sich In und Output Ports am Arduino spart. Um es zu betreiben, benötigt man 5V Spannung. Der Preis beträgt 6.29 Euro.

(Link: [HD44780 1602 LCD Modul Display Bundle mit I2C Schnittstelle 2x16 Zeich – AZ-Delivery](#))



1.3 Zoll OLED Display

Dieses Display verfügt über OLED-Technologie und hat somit eine hohe Auflösung von 128 x 64 Pixeln. Dieser Bildschirm kann mit seinen 1,3 Zoll klar Sachen anzeigen und benötigt nur eine Spannung von 3,3V ist jedoch auch kompatibel mit 5V. Der Vorteil dieses Bildschirms ist, dass es nur 4 Ports vom Arduino einnimmt, weil eine I2C-Schnittstelle bereits integriert ist. Der Preis dieses Displays beträgt 7,99 Euro.

(Link: [AZDelivery 3 x 1,3 Zoll OLED Display I2C SSH1106 Chip 128 x 64 Pixel I2C Bildschirm Anzeigemodul mit weißen Zeichen kompatibel mit Arduino und Raspberry Pi inklusive E-Book! : Amazon.de: Computer & Zubehör](#))

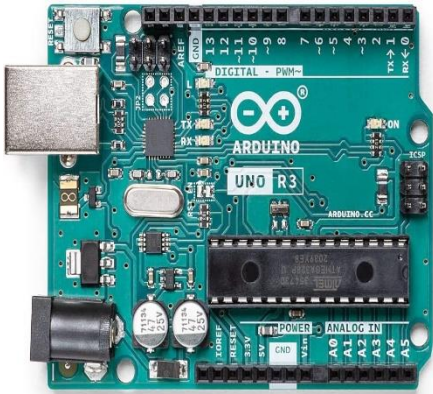
Mikrocontroller



Nano V3.0 Atmega328 CH340

Der Arduino Nano ist eine der kleinsten Arduinos und ist speziell für kompakte Projekte entwickelt worden. Der Nano ist etwa 45 mm x 18 mm groß und verfügt über insgesamt 22 Digitale In- und Output Pins. Zudem hat der Nano 8 Analoge-Input-Pins und kann mit 6 – 12 V betrieben werden. Der Nano wird mit USB mit Strom versorgt kann aber auch extern über VIN und GND mit Strom versorgt werden. Dieser Mikrocontroller kostet rund 17,99 Euro.

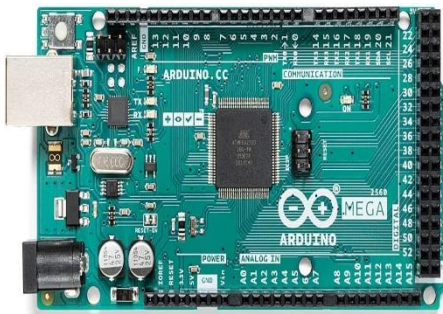
([Nano V3.0 Atmega328 CH340 fertig verlötete, verbesserte Version mit US – AZ-Delivery](#))



Arduino Uno R3

Der Arduino Uno ist bekannt dafür, dass es zum Einstieg in elektronischen Projekten zum Programmieren verwendet wird. Der Uno ist in etwa 68,6 mm x 53,4 mm groß und somit ein gutes Stück größer als der Nano. Mit seinen 14 digitalen Ein- und Ausgängen und 6 analogen Eingängen hat er weniger als der Nano. Der Uno hat einen Spannungsbereich von 7- 12V und wird durch USB mit Strom versorgt. Der Uno hat als extra einen ICSP-Anschluss womit man direkte Programmierungen ohne das Board zu öffnen vornehmen kann. Der Preis für diesen Arduino beträgt UVP: 29,28 Euro

(Link: [Arduino UNO Rev3 \[A000066\] : Amazon.de: Computer & Zubehör](#))



Arduino Mega 2560

Der Arduino Mega ist bekannt für größere Projekte, weil es der Arduino mit den meisten digitalen Ein- und Ausgängen (54) ist. Außerdem verfügt er über 16 analoge Eingänge und hat damit genug Platz für große Projekte. Der Spannungsbereich liegt bei dem Mega bei 7-12V und hat eine Größe von 101,5 mm x 53,3 mm. Der Arduino Mega hat ebenfalls, wie der Uno als extra einen ICSP- Anschluss.

(Link: [Arduino Mega 2560: Amazon.de: Games](#))



Raspberry Pi 4

Der Raspberry Pi 4 ist ein Mikrocontroller der aufgrund des leistungsfähigen eingebauten Prozessors leistungsfähiger als die meisten Arduinos ist. Der Raspberry Pi wird in komplexen Projekten/Systemen eher genutzt als der Arduino. Der Raspberry hat einen RAM von 1 GB und kann Inhalte von bis zu 4 GB speichern. Dieser Mikrocontroller kann mit WLAN, Bluetooth und USB mit dem PC oder Laptop

(Link: [Raspberry Pi 4 Model B Barebone 1GB: Amazon.de: Computer & Zubehör](#))

verbunden werden. Außerdem hat der Raspberry PI reichlich an Anschlüssen, wie zum Beispiel 5 USB-Anschlüsse, einen Gigabit Ethernet (LAN) Anschluss, HDMI, einen Audioausgang und einen Kamera- und Display-Anschluss. Der Preis von diesem Raspberry beträgt 119,99 Euro und ist somit viel teurer als Arduino Mikrocontroller.

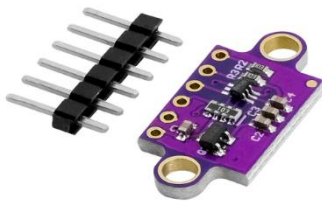
Sensoren zum Messen von Entfernungen



HC-SR04 Ultraschallsensor

Der HC-SR04 misst Entfernungen, indem er Ultraschallwellen sendet und dann wieder empfängt. Dadurch kann der Sensor ermitteln, wie weit entfernt ein Objekt ist. Diese 2 runden Teile, die wie Lautsprecher aussehen heißen Transducer, wobei eine von den die Ultraschallwellen sendet und der andere sie empfängt. Dieser Sensor kann Entfernungen von 3cm – 4m erfassen. Der Sensor wird mit 5V betrieben und nimmt neben den 5V und Ground Ports nur 2 digitale Ports ein. Der Sensor kostet UVP 4,99 Euro.

(Link: [HC-SR04 Ultraschall Modul Entfernungsmesser Sensor für Raspberry Pi – AZ-Delivery](#))



VL53L0X Laser Abstandssensor

Dieser Sensor misst Entfernungen, indem er Infrarot-Lichtimpulse ähnlich wie der HC-SR04 sendet und dann wieder empfängt. Der Sensor misst dabei, wie lange die Impulse brauchen, um wieder den Sensor zu erreichen. Die Reichweite des Sensors liegt je nach Modus bei 1-2 Metern und hat je nach größerem Abstand mehr Abweichungen. Der Sensor wird mit 2,6-5V betrieben und kostet 6,99 Euro.

(Link: [VL53L0X Time-of-Flight \(ToF\) Laser Abstandssensor kompatibel mit Ardui – AZ-Delivery](#))



TFmini Plus Lidar-Entfernungsmesser-Sensor

Dieser abgebildete Sensor ermittelt Entfernung mit Hilfe eines Lasers. Dadurch entstehen genauere Werte als die vorher vorgestellten Sensoren. Zudem kann dieser Entfernungsmesser Entfernungen im Bereich von 10cm – 12 Metern messen. Der Sensor kann mit einer I2C-Schnittstelle mit dem Arduino verbunden werden. Der Preis beträgt 73,90 Euro und ist somit ein weites Stück teurer als die anderen vorgestellten Entfernungsmesser.

(Link: [TFmini Plus Lidar-Entfernungsmesser-Sensor für Arduino IP65 1~1000Hz: Amazon.de: Sport & Freizeit](#))

TF-Luna LiDAR Module Radar Distance Meter



Dieser Sensor misst ebenfalls Entfernungen mit seinem Laser und kann Entfernungen im Bereich von 0,3 – 12 Metern messen. Für Projekte, wo genaue Werte erforderlich sind, ist dieser Sensor eine gute Wahl. Der Sensor kann mit einer I2C-Schnittstelle mit dem Arduino verbunden werden. Der Preis beträgt 39,90 Euro.

(Link: [Stemedu Benewake TF-Luna LiDAR Module Radar Distance Meter Sensor Single Point Micro Ranging Module measurement for Arduino Pixhawk pedestrian vehicle detection, UART IIC 0.2 m ~ 8 m : Amazon.de: Gewerbe, Industrie & Wissenschaft](#))

Buttons



4 Pin-Drucktastenschalter

Dieser Button ist ein 4 Poliger Taster, der dafür gedacht ist, Funktionen durch das drücken des Tasters auszulösen. Damit der Button funktioniert müssen die Pole mit GND, einem digitalen Eingang, einem Widerstand und dann das andere Ende des Widerstands mit 5V verkabelt werden. 40 von diesen Tastern kosten mit Kappen insgesamt 7,99 Euro. (Auf Amazon steht 80 Stück, jedoch wurden da die Kappen mitgezählt)

(Link: [RUNCCI-YUN Momentary Tactile Push Button, 12 x 12 x 7,3 mm, Micro Momentary Drucktastenschalter, 4 Pin Mikroschalter, oberflächenmontierte Schaltplatte, mit Kappe für Arduino, 80 Stück : Amazon.de: Gewerbe, Industrie & Wissenschaft](#))

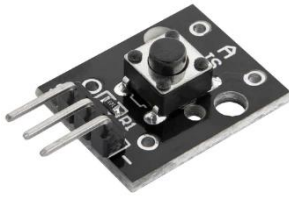
PS2 Joystick Shield Game Pad



Dieses Shield Pad, ist eher für Spiele gedacht kann aber auch für andere zwecke benutzt werden. Mit dem integrierten Joystick kann man perfekt Bewegungen eingeben. Um das Pad zu benutzen, muss man nur den Shield Pad in das Arduino aufstecken. Neben dem Joystick verfügt er über 4 Buttons und 2 Taster. Der Joystick funktioniert nur über 2 Achsen, was bedeutet, dass der Joystick nur reagiert, wenn man nach links oder rechts geht. Der Joystick kann mit 3,3 und 5V betrieben werden und kostet 8,99 Euro.

(Link: [PS2 Joystick Shield Game Pad Keypad V2.0 – AZ-Delivery](#))

KY-004 Taster



Dieser Taster hat 3 Pole und ist dafür gedacht Funktionen auszulösen. Um diesen Taster zu benutzen, muss man die Pole mit GND, einem digitalen Eingang und einer 5V Spannung verkabeln. Einer von diesen Tastern kostet 3,49 Euro und ist somit um einiges teurer als die 4 Poligen Buttons

(Link: [KY-004 Taster Modul Sensor Taste Kopf Schalter Schlüsselschalter – AZ-Delivery](#))

Alarm Lautsprecher

KY-006 Passiver Piezo Buzzer Alarm Modul



Dieser Buzzer ist extra dafür gedacht Töne auszuspielen, die sich wie ein Alarm anhören. Der Buzzer benötigt von den insgesamt 3 Polen nur 2. Eins für GND (Ground) und das andere für einen digitalen Ausgang. Der Preis für so einen Buzzer beträgt 3,79 Euro.

(Link: [KY-006 Passiver Piezo Buzzer Alarm Modul – AZ-Delivery](#))

KY-012 Buzzer Modul aktiv



Dieser Buzzer ist die Erweiterung des KY-006. Theoretisch macht er das gleiche, wie der Vorgänger nur mit dem Unterschied das der Ton anders erzeugt wird. Der KY-012 ist nämlich ein aktiver Buzzer und der KY-006 ein passiver. Dieser Buzzer wird genauso, wie der KY-006 verkabelt. Der Preis für diesen Sensor beträgt 3,49 Euro.

(Link: [KY-012 Buzzer Modul aktiv – AZ-Delivery](#))

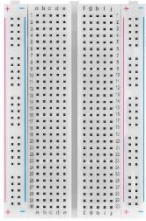
VISSQH Alarm Summer für Arduino



Dieser Lautsprecher nicht nur für Arduino Projekte verwendbar, sondern für vieles mehr. Die Verkabelung ist gleich wie die vorher vorgestellten Buzzer. Der Ton erfolgt durch einen integrierten Lautsprecher und 30 von diesen Lautsprechern kosten 9,56 Euro.

(Link: [VISSQH 30pcs PC Motherboard Interner Piezolautsprecher Computer Mainboard Buzzer ,Elektronischer Summer, BIOS Alarm Summer für Arduino Computer Case DIY : Amazon.de: Computer & Zubehör](#))

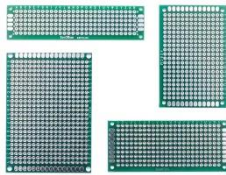
Aufbau von elektrischen Schaltungen



Breadboard

Breadboards sind perfekt für Projekte, die sich noch in der Prototyp Phase befinden. Die einzelnen Pins sind in Reihen angeordnet und mit Brücken verbunden. Oft werden sie bei Einsteigern verwendet, weil es damit einfacher ist Schaltungen (Projekte) aufzubauen. Der Preis für ein Board beträgt UVP 4,49 Euro.

(Link: [Mini Breadboard 400 Pin mit 4 Stromschienen für Jumper Kabel – AZ-Delivery](#))



Lochrasterplatine

Lochrasterplatten sind eine Alternative zu dem Breadboard, mit dem Unterschied, dass die Komponenten an die Platine angelötet werden müssen. Lochrasterplatten sind eher weniger für Projekte in einer Prototypenphase, sondern eher für das Endprodukt. Der Preis beträgt für 4 Platinen UVP 7,29 Euro.

(Link: [PCB Board Set Lochrasterplatte Lochrasterplatine Leiterplatte – AZ-Delivery](#))




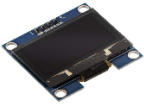







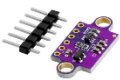


Löten

Eine weitere Möglichkeit ist das normale Löten ohne Lochrasterplatine oder Breadboard. Der Preis für einen Lötkolben beträgt 16,99 Euro

(Link: [LUX Lötkolben 15 W kaufen bei OBI](#))

Zusammengefasst:

Bildschirme	 LCD 20 x 4 Zeichen	 2,4 Zoll TFT LCD Touch Display Shield	 LCD 16 x 2 Zeichen	 1,3 Zoll OLED Display
Mikrocontroller	 Nano V3.0 Atmega328 CH340	 Arduino Uno R3	 Arduino Mega 2560	 Raspberry Pi 4

Entfernungsmesser Sensoren	 HC-SR04 Ultraschallsensor	 VL53L0X Laser Abstandssensor	 TFmini Plus Lidar- Entfernungsmesser-Sensor	 TF-Luna LiDAR Module Radar Distance Meter
----------------------------	---	--	---	--

Buttons	 4 Pin- Drucktastenschalter	 PS2 Joystick Shield Game Pad	 KY-004 Taster
Alarm Lautsprecher	 KY-006	 KY-012	 VISSQH Alarm Summer
Aufbau elektrischer Schaltungen	 Breadboard	 Lochrasterplatine	 Löten

1.2.3 Schlussfolgerung

Bei den Bildschirmen habe ich mich für den LCD mit 20 x 4 Zeichen entschieden, weil ich finde das in diesem Projekt kein Touch Display benötige. Zudem benötige ich 4 Zeilen in der Höhe (mehr Platz), womit man den LCD 16 x 2 ausschließen kann. Bei dem Mikrocontroller habe ich mich für den Nano entschieden aus dem einfachen Grund das ich mehr digitale IN- und Outputs benötige und auch mehr analoge Eingänge. Dadurch das der Arduino Uno wenige hat kann man ihn ausschließen. Den Mega habe ich nicht genommen, weil er zu viele Ports hat und ich so viele nicht benötige. Den Raspberry habe ich aus dem Grund ausgelassen, weil er erstens zu teuer ist und zweitens ich mich mit dem ganzen System vom Raspberry nicht auskenne und die Einführung viel Zeit kosten würde. Bei den Entfernungsmesser Sensoren habe ich mich für den HC-SR04 Ultraschallsensor entschieden, weil ich mich mit dem Sensor bestens auskenne, da ich in der Vergangenheit ein Arduino Projekt hatte, wo dieser Sensor mit als Komponente enthalten war. Bei den Buttons habe ich mich für die 4 Pin-Drucktastenschalter entschieden. Ich habe mich für die entschieden, weil die Buttons zu meinem Projekt gut passen. Den PS2 Joystick Shield habe ich ausgeschlossen, weil ich es überflüssig fand für solch ein Projekt und es eher für Spiele sinnvoll wäre. Bei den Alarm Lautsprechern habe ich den KY-006 genommen, weil ich mich mit ihm gut auskenne, weil ich diesem bereits privat für Projekte benutze. Als letztes habe ich mich für das Breadboard entschieden, weil ich finde das das Verkabeln durch das Breadboard einem einfacher fällt und nicht sonderlich zeitaufwendig ist. Die Lochrasterplatine ist jedoch zeitaufwendiger, weil man alles anlöten muss.

1.3 Vision des Produktes

1.3.1 Dimensionierung

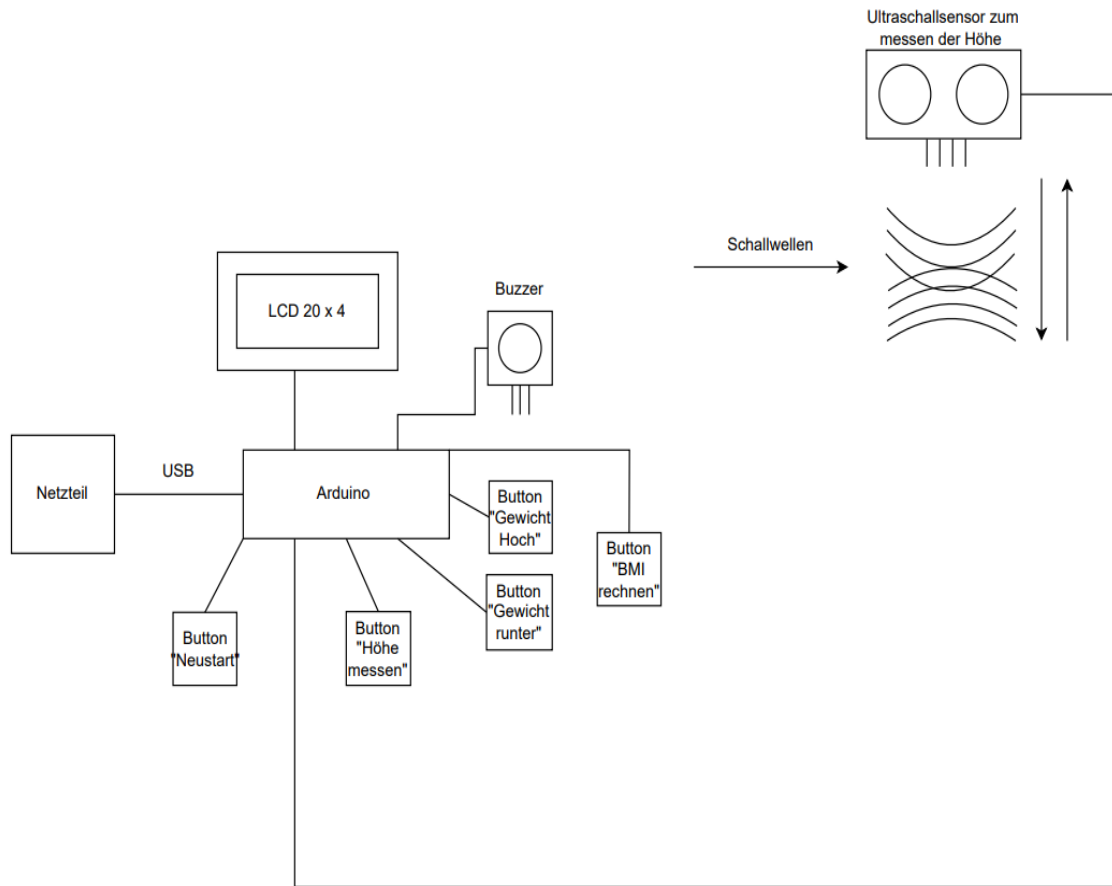
Das Projekt wird aus 2 Boxen bestehen. Die eine Box wird über der Tür mit Hilfe eines Magneten hängen. In dieser Box befindet sich nur der Ultraschallsensor, der nach unten guckt. Dafür wird in der Unterseite der Box ein Loch angefertigt, damit der Sensor perfekt durchpasst. Die Länge der Box wird in etwa 13 cm betragen und die Höhe in etwa 9 cm. Beide Boxen werden aus 15 mm dicken HPL-Platten gebaut. Die andere Box mit LCD-Bildschirm, dem Arduino, dem Buzzer und den ganzen Buttons werden sich neben der Tür befinden. Diese Box wird in etwa 23 cm in der Länge und 18 cm in der Breite sein. Auch diese Box wird aus dünnen Spanplatten gebaut. Die 4 Kabel, die für den Ultraschallsensor benötigt werden, werden mit Leisten gedeckt, damit man keinen Kabelsalat erkennt. Die Genauigkeit des HC-SR04 Ultraschallsensors ist sehr wichtig, weil es essenziell für die Rechnung des BMIs ist. Die Messung der Höhe kann nur von einer Person gleichzeitig durchgeführt werden. Durch eine Aufforderung am LCD wird man aufgefordert sich unter den Ultraschallsensor zu stellen. Daraufhin kann man im LCD seine Größe in cm sehen. Der Programmcode wird geschätzt nicht länger als 150 Zeilen betragen. Das Projekt wird 5 Buttons haben. Einer, um das Projekt zu starten, der zweite und dritte, um sein Gewicht hoch oder runterzustellen, der 4, um die BMI-Skala anzuzeigen und der letzte, um seinen BMI rechnen zu lassen. Die benötigte Spannung beträgt 5V.

1.3.2 Komponenten und spezielle Funktionen

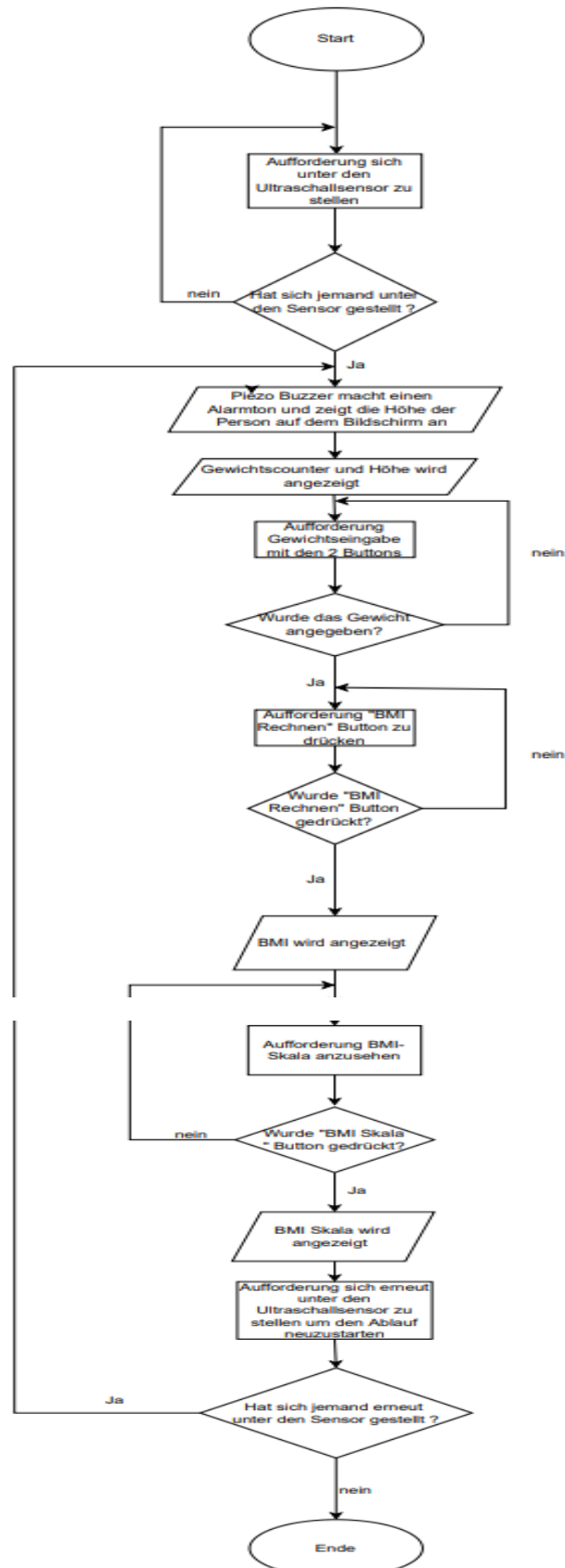
Die Stromversorgung für das Projekt wird per USB stattfinden. Jedoch wird ein extra Netzteil benutzt, damit das Projekt durch die Steckdose mit Strom versorgt werden kann. Darauf wurde sehr geachtet, weil man es immer benutzen kann und nicht zum Beispiel wie bei einer Powerbank neu aufladen muss.

2. Detaillierte Planung

2.1 Technologieschema



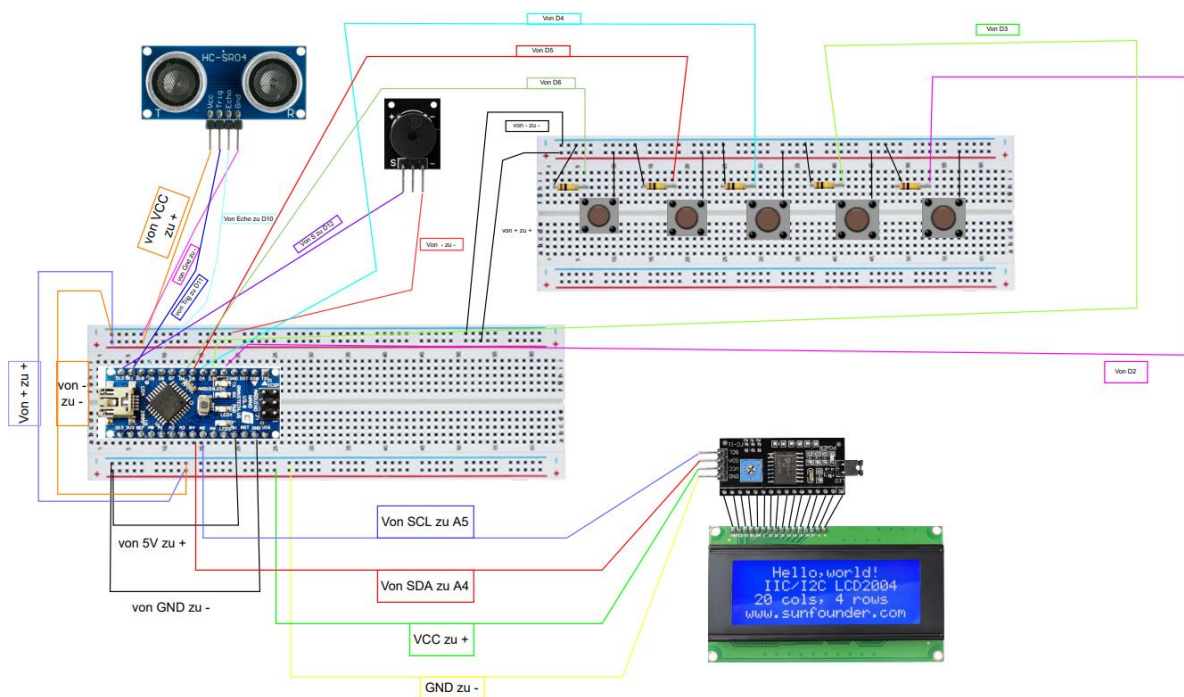
2.2 UML Flowchart



2.3 Materialliste

Artikel	Anzahl	Preis	Lieferzeiten
 <p>KY-006 Passiver Piezo Buzzer Alarm Modul</p> <p>KY-006 Passiver Piezo Buzzer Alarm Modul – AZ-Delivery</p>	1	3,79 Euro	1-2 Werktage
 <p>Breadboard Kit</p> <p>Breadboard Kit – 3x Jumper Wire m2m/f2m/f2f + 3er Set MB102 Breadboard – AZ-Delivery</p>	1	11,99 Euro	1-2 Werktage
 <p>HC-SR04 Ultraschall Modul</p> <p>HC-SR04 Ultraschall Modul Entfernungsmesser Sensor für Raspberry Pi – AZ-Delivery</p>	1	4,79 Euro	1-2 Werktage
 <p>2004 LCD-Display Bundle 4x20 Zeichen mit I2C Schnittstelle</p> <p>HD44780 2004 LCD Display Bundle 4x20 Zeichen mit I2C Schnittstelle – AZ-Delivery</p>	1	11,29 Euro	1-2 Werktage
 <p>Nano V3.0 Atmega328 CH340 fertig verlötete</p> <p>Nano V3.0 Atmega328 CH340 fertig verlötete, verbesserte Version mit USB – AZ-Delivery</p>	1	17,99 Euro	1-2 Werktage
 <p>40 Buttons mit 40 Kappen</p> <p>BUIBIO 40x Membrane Push-Buttons, 12 x 12 x 7,4 mm, Micro-Momentary Drucktastenschalter, 4 Pin Mikroschalter, oberflächenmontierte Schaltplatte, mit Kappen für Arduino, 80 Stück - Amazon.de: Gewerbe, Industrie & Wissenschaft</p>	1	7,99 Euro	1-2 Werktage
 <p>Widerstände</p> <p>BOJACK Widerstände Sortiment Kit 1 Ohm - 1M Ohm 1W Metallfilm Widerstand Resistor Kit (30 Werte 350 Stück) - Amazon.de: Gewerbe, Industrie & Wissenschaft</p>	1	5,73 Euro	3-4 Werktage

2.4 Schaltplan



3 Projektziele und Meilensteine

3.1 MUSS Ziele

- Das Projekt muss beim messen der Höhe eine hohe Genauigkeit aufweisen, damit der BMI, der abhängig von der Höhe des Menschen ist einen genauen Wert liefern kann. Ziel des Projektes ist es, bei der Messung der Höhe eine maximale Abweichung von +/-1 cm zu erreichen
- Das Projekt muss eine Qualität aufweisen, dass es für eine längere Zeit beständig bleibt. Ziel ist es, dass das Projekt mindestens 1 Jahr bei täglicher Benutzung fehlerfrei funktioniert.
- Das Projekt muss übersichtlich aufgebaut sein (Bedienung des Projektes soll einfach zu verstehen sein). Ziel ist, dass Benutzer ihren BMI in weniger als 1,5 Minuten erfahren
- Das Projekt muss nach dem erfahren des BMIs erfolgreich den gleichen Ablauf immer wieder abspielen können (Neustart muss flüssig laufen) Ziel ist es das man nach jeder neuen Messung und BMI-Berechnung nicht den Stecker rausziehen muss, um es neu zu starten.
- Das Projekt muss ästhetisch aussehen. Ziel ist es, dass man keinen Kabelsalat mit dem bloßen Auge erkennt und das die Boxen auch gut aussehen

3.2 KANN Ziele

- Das Projekt kann erweiterte Funktionen beinhalten. Zum Beispiel könnte man ein Gewichtssensor einbauen, sodass man sein Gewicht nicht eingeben muss. Zudem ist es dann genauer, weil nicht jeder Mensch weiß wie viel genau man in dem Moment wiegt. Ziel ist es somit die Genauigkeit höher zu stellen
- Das Projekt kann außerdem noch sicherer (robuster) aufgebaut werden. Ziel ist es damit das Projekt länger anhaltend zu bauen. Statt 1 Jahr bei täglicher Nutzung könnte es somit 5 Jahre halten
- Das Projekt kann die Messung der Höhe noch genauer erfassen. Zum Beispiel könnte man einen anderen Entfernungsmesser Sensor einbauen, der technisch genauere Werte aufweisen kann als der vorherige. Ziel ist es also bei der Messung der Höhe eine maximale Abweichung von +/-0,5 cm zu erreichen

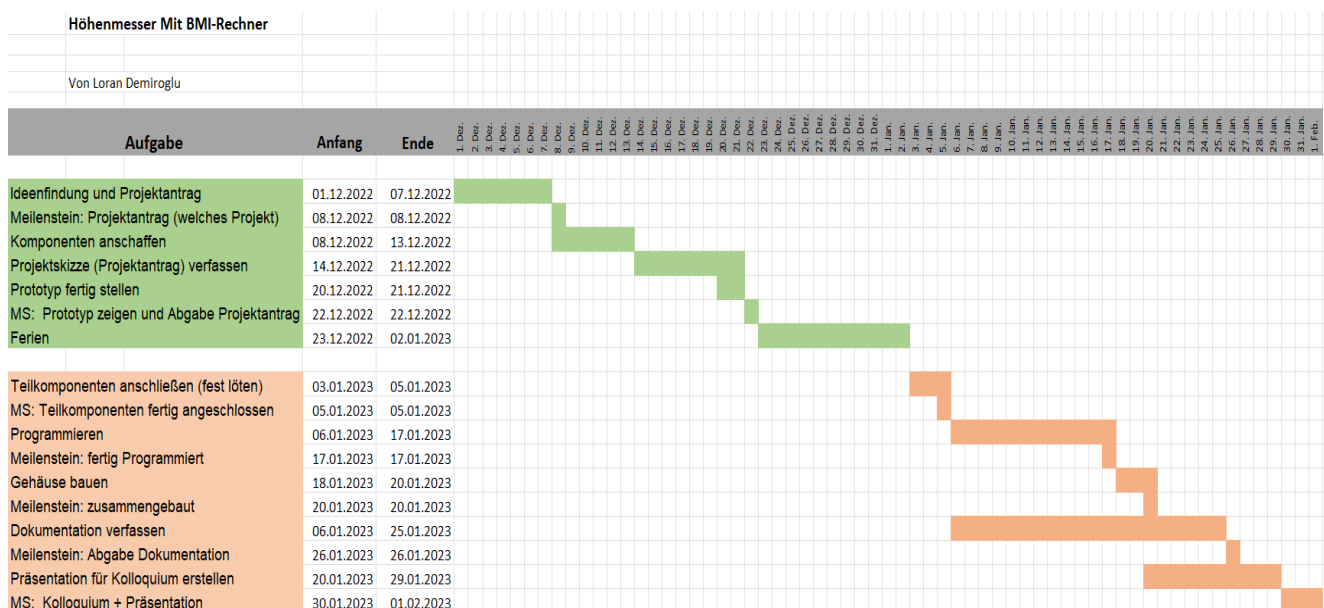
3.3 NICHT Ziele

- Das Projekt soll nicht andere Körperteile messen, sondern nur die Höhe des Menschen
- Das Projekt soll nicht den Körperfettanteil oder die Muskelmasse des Menschen angeben

3.4 Meilensteinplan

3.4.1 Meilensteine fürs Projektergebnis

3.4.2 Meilensteine der Projektdurchführung



3.4.3 Checkliste Prototyp

- ✓ Ultraschallsensor anschließen und so programmieren, dass die Entfernungen, die der Ultraschallsensors misst auf dem seriellen Monitor angezeigt wird
- ✓ Button (Taster) und LCD-Bildschirm anschließen und so programmieren, dass wenn man auf den Button drückt, etwas auf dem Bildschirm ausgegeben wird. Zum Beispiel Hallo oder Willkommen
- ✓ Piezo Buzzer anschließen und so programmieren, dass wenn auf den gleichen Button (Taster) gedrückt wird ein Ton ausgegeben wird.

4.0 Durchführung

4.1 Realisierung

HC-SR04 Ultraschallsensor

Der HC-SR04 Ultraschallsensor, welcher in diesem Projekt für die Höhenmessung der Person zuständig ist, arbeitet auf Basis des Ultraschalls. Dieser sendet Ultraschallwellen aus und misst die Zeit, die sie braucht, um ein Objekt (im Fall dieses Projekts ein Kopf) zu erreichen und von dort zurückzukehren. Dadurch kann die Entfernung zu dem bestimmten Objekt berechnet werden. Der Sensor verwendet dafür einen Sender und Empfänger. Der Sender sendet einen Impuls (Ultraschallwelle) aus und der Empfänger nimmt das Echo auf. Wenn man dann die Zeit misst von dem senden und empfangen und es daraufhin mit der Schallgeschwindigkeit multipliziert kommt man auf die Entfernung von dem Sensor zum Objekt.

Der Sensor arbeitet in diesem Projekt mit dem Piezo Buzzer und dem LCD zusammen, weil die Höhe der Person auf dem LCD angezeigt wird. Wenn sich eine Person unter den Sensor stellt, sendet der Piezo Buzzer einen Alarmton, dass die Messung erfolgreich war, was darauf hinweist, dass der Piezo Buzzer und der Ultraschallsensor ebenfalls zusammenarbeiten.

```
5   int echo = 10;
6   int trigger = 11;
7   long dauer;
8   int entfernung;
9   int entfernung_hoehe;
```

Bei diesem Abschnitt wurden die Variablen, die für den Ultraschallsensor von Bedeutung sind, deklariert und initialisiert. Der Echo Pin vom Ultraschallsensor ist am Arduino Nano in dem Digitalen Pin (D10) aufgesteckt und der Trigger Pin im (D11).

```

25 void setup() {
26   pinMode(trigger, OUTPUT);
27   pinMode(echo, INPUT);

```

Im Setup werden nun der Eingang und Ausgang Pins des Sensors konfiguriert. Zeile 26 setzt den Trigger-Pin des Sensors in den Ausgabemodus. Das bedeutet, dass der Pin verwendet wird, um Signale an andere Geräte zu senden. Zeile 27

versetzt den Echo-Pin des Sensors in den Eingabemodus. Das bedeutet, dass der Pin verwendet wird, um Signale von anderen Geräten zu empfangen.

```

67   digitalWrite(trigger, LOW);
68   delayMicroseconds(2);
69   digitalWrite(trigger, HIGH);
70   delayMicroseconds(10);
71   digitalWrite(trigger, LOW);
72   dauer = pulseIn(echo, HIGH);
73   delay(1000);
74   entfernung = dauer * 0.034/2;
75   entfernung_hoehe = 226 - entfernung;

```

Im Loop Teil wird jetzt der Code geschrieben, welcher die Entfernung misst. Zeile 67 setzt den Trigger Pin des Ultraschallsensors auf LOW also 0V. Zeile 68 verzögert nun das Programm um 2 Mikrosekunden. In Zeile 69 ist der Trigger Pin HIGH also 5V und in der nächsten Zeile gibt es erneut eine Verzögerung von 10

Mikrosekunden. In Zeile 71 ist der Trigger Pin erneut auf 0V. Zeile 72 misst die Dauer (in Mikrosekunden), für die der Echo Pin des HC-SR04 Sensors auf 5V bleibt. Dadurch kann bestimmt werden wie lange es gedauert hat bis das Signal gesendet und Empfangen wurde. In Zeile 74 wird „Entfernung“ deklariert mit der Dauer multipliziert mit der Schallgeschwindigkeit. In Zeile 75 wird die Höhe des Sensors wo er später stehen soll mit der Entfernung zwischen dem Sensor und dem Kopf subtrahiert.

KY-006 Passiver Piezo Buzzer

Dieser Piezo Buzzer arbeitet mit Piezoelektrischen Elementen. Diese werden durch Spannung sozusagen „verformt“ und erzeugen dann den Schall. Dadurch dass dieser Piezo passiv ist braucht er keine Stromversorgung und wird von einer angelegten Spannung gespeist.

Der KY-006 arbeitet nur mit dem Ultraschallsensor zusammen, dass wenn sich eine Person unter den Ultraschallsensor stellt, der Piezo einen Alarmton ausgibt.

```

19 int buzzer = 2;

```

Hier wurde der Piezo Buzzer deklariert. Dieser ist am Arduino am Digitalen Pin 2 (D2) aufgesteckt.

```

81   tone(buzzer, 210, 200);

```

Mit dieser Zeile kommt letztendlich der Ton raus. Damit der Ton rauskommt benötigt der Buzzer zwei Informationen. Einmal die Frequenz in hz und die Dauer des Alarmtones. Diese kann man in den Klammern rechts neben Buzzer sehen und bei Wunsch ändern.

2004 LCD mit I2C Schnittstelle

Dieses LCD (Liquid Crystal Display) ist ein Anzeigemodul, welches 20 Zeichen in 4 Zeilen anzeigen kann. Es arbeitet mit einem Element des Liquid Crystals welches elektrisch gesteuert wird, um die Lichtbrechung zu steuern. (Sichtbarkeit der Zeichen auf dem Display) Das Display wird in diesem Projekt mit einer I2C Schnittstelle verbunden. Die Schnittstelle sendet mit den Pins SCL und SDA die Daten. Die Schnittstelle vereinfacht die Steuerung des LCD und spart eine Menge an Kabelsalat.

Der LCD arbeitet mit den 4-Poligen Tastern und dem Ultraschallsensor zusammen. Durch das drücken der Taster ändert sich im LCD immer wieder der Zustand, wie zum Beispiel, wenn man auf den Taster (Gewicht hoch) drückt ändert sich die Zahl im LCD und geht um 1 hoch. Die Abmessungen des Ultraschallsensors gelangen ebenfalls an den LCD indem die Daten an den LCD gesendet werden.

```
1  #include <LiquidCrystal_I2C.h> // Bibliothek zum initialisieren des LCD Bildschirms
2  LiquidCrystal_I2C lcd = LiquidCrystal_I2C(0x27, 20, 4);
```

Um den LCD mit der I2C Schnittstelle zu verwenden, benötigt man die dazugehörige Bibliothek, wie man in Zeile 1 erkennen kann. Diese Schnittstelle dient zum Initialisieren des LCD und wird zum ansteuern des LCDs verwendet. Wie zum Beispiel das Ausgeben von Texten oder die Hintergrundbeleuchtung. Zeile 2 wird verwendet, um eine „Realisierung“ der Klasse zu erstellen. 0x27 ist die I2C-Adresse des LCD. 20 steht für die Anzahl der Zeichen von links nach rechts und 4 für die Anzahl der Spalten.

```
34  lcd.init();
35  lcd.backlight();
```

Im Setup wird nun die Funktion `lcd.init()`; aufgerufen, um den LCD zu initialisieren. Zeile 35 ist dafür zuständig die Hintergrundbeleuchtung einzuschalten

```
45  if (tasterstatus1 == HIGH)
46  {
47      lcd.setCursor(0, 0);
48      lcd.print("Willkommen beim");
49      lcd.setCursor(0, 1);
50      lcd.print("Hoehenmesser und BMI");
51      lcd.setCursor(0, 2);
52      lcd.print("Rechner");
53      delay(4500);
54      lcd.clear();
55      lcd.setCursor(0, 0);
56      lcd.print("Stellen sie sich");
57      lcd.setCursor(0, 1);
58      lcd.print("unter den Sensor");
59      lcd.setCursor(0, 2);
60      lcd.print("um herauszufinden");
61      lcd.setCursor(0, 3);
62      lcd.print("wie gross sie sind");
63  }
```

Hier sieht man einen Teil meines Programm Codes im Loop. Hier sieht man das, wenn ein Button gedrückt wird etwas auf dem LCD stehen soll. Zeile 47 ist dafür zuständig wo zum Beispiel das Wort stehen soll. Die linke Zahl in der Klammer steht für die Positionierung von links nach rechts. Wenn dort jetzt beispielsweise steht `lcd.setCursor(1,0)` und im `lcd.print` zum Beispiel eine 1 wird die Zahl in der ersten Spalte ganz links ausgegeben. Die zweite Zahl am Cursor steht für die Positionierung der Zeilen. Diese gehen von 0-3. Wenn zum Beispiel da eine 3 steht ist die Ausgabe in der letzten Zeile. Die Zeile 54 entfernt oder reinigt alle

Wörter, die vorher dort standen. Zeile 48 ist für die Ausgabe zuständig. Diese kann Texte in Gänsefüßen ausgeben oder zum Beispiel deklarierte Variablen.

4-Poliger Taster

Ein 4-Poliger Taster wird meist verwendet, um Zustände zu ändern oder um eine Eingabe von Benutzern zu empfangen. Es gibt verschiedene Wege einen Taster zu verkabeln. Es gibt einmal den Taster mit Pull down und Pull up Widerstand. Diese unterscheiden sich daher das beim Pull down der Taster HIGH (5V) ist, wenn man auf ihn drückt. Beim Pull up ist der Taster HIGH, wenn er nicht betätigt wird. In diesem Projekt wurden ausschließlich nur Taster mit Pull-down Widerständen benutzt. Der Unterschied an der Verkabelung liegt darin, dass 5V und GND vertauscht sind.

Die Taster arbeiten in diesem Projekt nur mit dem LCD zusammen, um die Zustände zu wechseln. Beispielsweise wird das Gewicht durch die Taster eingetragen und die Rechnung des BMIs erfolgt ebenfalls mit einem Knopfdruck.

```
11  int taster_start = 4;  
12  int tasterstatus1 = 0;
```

Zeile 11 ist eine Variable, welche dazu dient, den digitalen Pin des Arduinos zu definieren. In dem Fall ist der Button am digitalen Pin D4 angeschlossen. Zeile 12 ist da, um

den Zustand später zu speichern.

```
30  pinMode(taster_start, INPUT);
```

Im Setup wird nun der Eingang (Input) konfiguriert

```
44  tasterstatus1=digitalRead(taster_start);  
45  
46  if (tasterstatus1 == HIGH)  
47  {
```

Hier sieht man in Zeile 44 im Loop, wie der Taster definiert wird. digitalWrite() ist dafür zuständig um einen Wert des digitalen Pins (4) auszulesen. Das heißt, dass wenn tasterstatus1 LOW ist es 0 wird und wenn es HIGH wird

es zu 1 wird. Zudem kann man in Zeile 46 erkennen, wie ein Taster in Kombination mit einer If-Bedingung funktioniert.

4.2 Testfälle

Der HC-SR04 Ultraschallsensor wurde auf verschiedenen Entfernungen auf Genauigkeit geprüft. Dafür habe ich den Sensor auf verschiedene Höhen (50 cm; 1 m; 1,50 m; 2 m und 2,26 m) mit einem Zollstock genau gemessen aufgestellt und geguckt welche Werte der Ultraschallsensor ausspuckte. Es stellte sich heraus, dass der Sensor bei 50 cm und 1 m noch genau war. Jedoch stellte sich heraus, dass ab 1,50 m der Sensor anfang Abweichungen festzustellen. Bei 1,50 m kam beim Sensor 1,47 m raus. Das zeigt, dass der Sensor bereits bei 1,50 m eine Abweichung von 3 cm hat. Bei 2 Metern lag die Abweichung schon bei 5 cm und bei 2,26 m (Die Höhe, wo der Sensor in diesem Projekt stehen wird) gab es eine Abweichung von 6 cm. Das zeigt, dass der Ultraschallsensor auf geringe Entfernung ziemlich genau ist, jedoch ab einer Entfernung von 1,50 m stark nachlässt und Abweichungen vorzeigt.

Die Taster und der LCD wurden zusammen getestet, indem ich einen Programm Code geschrieben habe, welcher einen Counter hochzählt, wenn ich auf den Taster drücke (wenn er HIGH ist) Anfangs reagierten die Taster nur langsam und man musste den Taster etwas

gedrückt halten, bis er HIGH war. Jedoch habe ich es nach kurzer Zeit behoben indem ich ein delay(); eingebaut habe.

Den Piezo Buzzer habe ich so getestet, indem ich mehrere Programmcodes geschrieben habe mit verschiedenen Frequenzen und dauern beim Alarmton, um herauszufinden, welcher zu meinem Projekt passt. Dabei ist mir aufgefallen, dass es ab einer bestimmten Frequenz (hohen Frequenz) nicht mehr funktioniert oder zu hören ist.

5.0 Feedback

5.1 Projektabschluss

5.1.1 Vergleich: ursprüngliche Planung zu erreichten Zielen

Anfangs war es geplant die beiden Boxen aus dünnen Spanplatten zu bauen. Jedoch habe ich mich nach geraumer Zeit dafür entschieden die Boxen aus 15 mm dicken HPL-Platten zu erstellen. Zudem war geplant, die einzelnen Komponenten auf einer Lochrasterplatte zusammen zu löten. Eine weitere Abweichung ist, dass der Ablauf (wie das Projekt später laufen soll) geändert wurde. Beispielsweise war anfangs geplant einen Taster einzubauen, welcher wenn er gedrückt wird, die Höhe misst. Später habe ich mich dafür entschieden es ohne Taster zu machen. Des Weiteren war geplant einen Neustart Taster einzubringen, welcher den Ablauf neu startet. Als Ersatz wurden dafür die Taster (Start und BMI-Skala) verwendet. Taster (Start) startet den kompletten Ablauf und wenn der Taster (BMI-Skala) gedrückt wird, wird auf dem LCD angezeigt von wie viel bis wie viel (BMI) man untergewichtig, normalgewichtig oder übergewichtig ist.

5.1.2 Ursachenanalyse der Abweichungen

Der Grund, dass ich mich bei dem Material der beiden Boxen unentschieden habe und zu HPL gewechselt bin, ist der, dass HPL-Platten um Welten stärker und haltbarer sind. HPL-Platten sind bekannt für ihre Robustheit und Langlebigkeit. Ich fand diese Kriterien wichtig, da mein Projekt eine gewisse Qualität (Langlebigkeit und Belastbarkeit) aufweisen soll (wie es in den MUSS-Zielen steht. Außerdem sehen HPL-Platten besser aus als Spanplatten.

Der „Höhen messen“ Taster wurde ausgelassen, weil es ohne mehr Zeit spart und man sich direkt unter den Sensor stellen kann und nicht erst einen Taster drücken muss. Der Neustart Taster wurde ebenfalls ausgelassen, weil man keinen benötigt, um eine neue Messung und BMI-Rechnung zu machen, da es im Loop Teil ist.

5.1.3 Selbstreflexion: Lerntransfer für zukünftige Projekte

Folgende Punkte würde ich in folgenden Projekten besser gestalten:

- Zeitplanung einhalten (Zeitmanagement). In diesem Projekt habe ich nicht ganz meine Zeitplanung eingehalten, und habe dadurch mehr Stress gehabt.
- Hilfe suchen, wenn etwas nicht funktioniert.

5.1.4 Begründeter Notenvorschlag

Ich denke das ich die Note 2 verdient habe, weil ich alle Anforderungen erfüllt habe. Mein Projekt kann die Höhe eines Menschen messen und kann nach einer Gewichtseingabe erfolgreich den BMI ausrechnen. Ich denke, dass ich im Design meines Projektes auch eine gute Leistung erbracht habe. Jedoch könnte man noch verbessern den ganzen Ablauf noch schneller zu machen (Gewichtseingabe etc.)

6.0 Anhang

Schaltplan

Der Schaltplan befindet sich in Seite 15 der Dokumentation

Quellenangaben:

(Link: [TRAGBARER BMI-RECHNER BASIEREND AUF MIKROCONTROLLER-I Elektronik-Werkstatt \(electronicsworkshops.com\)](https://www.electronicsworkshops.com/tragbarer-bmi-rechner-basierend-auf-mikrocontroller-i/))

(Link: [DIY Arduino Höhenmessgerät - Arduino Project Hub](https://www.arduino.cc/projects/diy-arduino-hoehenmessgeraet/))

(Link: [Using VL53L0X With Arduino to Measure Height - Electronics-Lab.com](https://www.electronic-lab.com/using-vl53l0x-with-arduino-to-measure-height/))

(Link: [Design a BMI Calculator using JavaScript - GeeksforGeeks](https://www.geeksforgeeks.org/design-a-bmi-calculator-using-javascript/))

(Link: [HD44780 2004 LCD Display Bundle 4x20 Zeichen mit I2C Schnittstelle – AZ-Delivery](https://www.amazon.de/dp/B07K9K9K9K))

(Link: [2,4 Zoll TFT LCD Touch Display Shield 5V / 3,3V kompatibel mit Arduino – AZ-Delivery](https://www.amazon.de/dp/B07K9K9K9K))

(Link: [HD44780 1602 LCD Modul Display Bundle mit I2C Schnittstelle 2x16 Zeich – AZ-Delivery](https://www.amazon.de/dp/B07K9K9K9K))

(Link: [AZDelivery 3 x 1,3 Zoll OLED Display I2C SSH1106 Chip 128 x 64 Pixel I2C Bildschirm Anzeigemodul mit weißen Zeichen kompatibel mit Arduino und Raspberry Pi inklusive E-Book! : Amazon.de: Computer & Zubehör](https://www.amazon.de/dp/B07K9K9K9K))

Link: [Nano V3.0 Atmega328 CH340 fertig verlötete, verbesserte Version mit US – AZ-Delivery](https://www.amazon.de/dp/B07K9K9K9K))

(Link: [Arduino UNO Rev3 \[A000066\] : Amazon.de: Computer & Zubehör](https://www.amazon.de/dp/B07K9K9K9K))

(Link: [Arduino Mega 2560: Amazon.de: Games](https://www.amazon.de/dp/B07K9K9K9K))

(Link: [Raspberry Pi 4 Model B Barebone 1GB: Amazon.de: Computer & Zubehör](https://www.amazon.de/dp/B07K9K9K9K))

(Link: [HC-SR04 Ultraschall Modul Entfernungsmesser Sensor für Raspberry Pi – AZ-Delivery](https://www.amazon.de/dp/B07K9K9K9K))

(Link: [VL53L0X Time-of-Flight \(ToF\) Laser Abstandssensor kompatibel mit Ardui – AZ-Delivery](https://www.amazon.de/dp/B07K9K9K9K))

(Link: [TFmini Plus Lidar-Entfernungsmesser-Sensor für Arduino IP65 1~1000Hz: Amazon.de: Sport & Freizeit](https://www.amazon.de/dp/B07K9K9K9K))

(Link: [Stemedu Benewake TF-Luna LiDAR Module Radar Distance Meter Sensor Single Point Micro Ranging Module measurement for Arduino Pixhawk pedestrian vehicle detection, UART IIC 0.2 m ~ 8 m : Amazon.de: Gewerbe, Industrie & Wissenschaft](https://www.amazon.de/dp/B07K9K9K9K))

(Link: [RUNCCI-YUN Momentary Tactile Push Button, 12 x 12 x 7,3 mm, Micro Momentary Drucktastenschalter, 4 Pin Mikroschalter, oberflächenmontierte Schaltplatte, mit Kappe für Arduino, 80 Stück : Amazon.de: Gewerbe, Industrie & Wissenschaft](https://www.amazon.de/dp/B07K9K9K9K))

(Link: [PS2 Joystick Shield Game Pad Keypad V2.0 – AZ-Delivery](https://www.amazon.de/dp/B07K9K9K9K))

(Link: [KY-004 Taster Modul Sensor Taste Kopf Schalter Schlüsselschalter – AZ-Delivery](https://www.amazon.de/dp/B07K9K9K9K))

(Link: [KY-006 Passiver Piezo Buzzer Alarm Modul – AZ-Delivery](https://www.amazon.de/dp/B07K9K9K9K))

(Link: [KY-012 Buzzer Modul aktiv – AZ-Delivery](#))

(Link: [VISSQH 30pcs PC Motherboard Interner Piezolautsprecher Computer Mainboard Buzzer ,Elektronischer Summer,BIOS Alarm Summer für Arduino Computer Case DIY : Amazon.de: Computer & Zubehör](#))

(Link: [Mini Breadboard 400 Pin mit 4 Stromschienen für Jumper Kabel – AZ-Delivery](#))

(Link: [PCB Board Set Lochrasterplatte Lochrasterplatine Leiterplatte – AZ-Delivery](#))

(Link: [LUX Lötkolben 15 W kaufen bei OBI](#))

(Link: [Body Mass Index Chart, Formula, How To Calculate for Men & Women \(healthjade.net\)](#))

(Link: [DIY Arduino Height Measuring Machine - Hackster.io](#))

(Link: <https://www.verival.de/bmi-rechner>)

Sourcecode:

```
1  #include <LiquidCrystal_I2C.h> // Bibliothek zum initialisieren des LCD Bildschirms
2  LiquidCrystal_I2C lcd = LiquidCrystal_I2C(0x27, 20, 4);
3
4  int echo = 10;
5  int trigger = 11;
6  long dauer;
7  double entfernung;
8  double entfernung_hoehe;
9
10 int taster_hoch = 9;
11 int tasterstatus_hoch = 0;
12
13 int taster_start = 4;
14 int tasterstatus_start = 0;
15
16 int taster_bmi = 12;
17 int tasterstatus_bmi = 0;
18
19 int taster_skala = 3;
20 int tasterstatus_skala = 0;
21
22 int taster_runter = 7;
23 int tasterstatus_runter = 0;
24
25 int buzzer = 2;
26
27 double gewicht = 60.00;
28
29 void setup()
30 {
31   pinMode(trigger, OUTPUT);
32   pinMode(echo, INPUT);
33   pinMode(taster_hoch, INPUT);
```

```

34     pinMode(taster_skala, INPUT);
35     pinMode(taster_start, INPUT);
36     pinMode(taster_bmi, INPUT);
37     pinMode(taster_runter, INPUT);
38     lcd.init();
39     lcd.backlight();
40 }
41
42 void loop()
43 {
44     tasterstatus_start = digitalRead(taster_start);
45
46     if (tasterstatus_start == HIGH)
47     {
48         lcd.setCursor(0, 0);
49         lcd.print("Willkommen beim");
50         lcd.setCursor(0, 1);
51         lcd.print("Hoehenmesser und BMI");
52         lcd.setCursor(0, 2);
53         lcd.print("Rechner");
54         delay(4500);
55         lcd.clear();
56         lcd.setCursor(0, 0);
57         lcd.print("Stellen sie sich");
58         lcd.setCursor(0, 1);
59         lcd.print("unter den Sensor");
60         lcd.setCursor(0, 2);
61         lcd.print("um herauszufinden");
62         lcd.setCursor(0, 3);
63         lcd.print("wie gross sie sind");
64     }
65
66     digitalWrite(trigger, LOW);

```

```

67     delayMicroseconds(2);
68     digitalWrite(trigger, HIGH);
69     delayMicroseconds(10);
70     digitalWrite(trigger, LOW);
71     dauer = pulseIn(echo, HIGH);
72     delay(1000);
73     entfernung = dauer * 0.034/2;
74     entfernung_hoehe = 226-entfernung;
75
76
77     if (entfernung_hoehe > 90)
78     {
79         lcd.clear();
80         tone(buzzer, 210, 200);
81         lcd.setCursor(0, 0);
82         lcd.print("Ihre Groesse");
83         lcd.setCursor(0, 1);
84         lcd.print("betraegt:");
85         lcd.setCursor(12, 1);
86         lcd.print(entfernung_hoehe);
87         lcd.setCursor(17, 1);
88         lcd.print("cm");
89         delay(4500);
90         lcd.clear();
91         lcd.setCursor(0,0);
92         lcd.print("Geben sie ihr");
93         lcd.setCursor(0,1);
94         lcd.print("Gewicht an");
95         lcd.setCursor(0,2);
96         lcd.print("Groesse:");
97         lcd.setCursor(12,2);
98         lcd.print(entfernung_hoehe);
99         lcd.setCursor(0,3);

```

```

100     lcd.print("Gewicht:");
101     lcd.setCursor(14,3);
102     lcd.print(gewicht);
103 }
104
105 tasterstatus_hoch=digitalRead(taster_hoch);
106
107 if (tasterstatus_hoch == HIGH)
108 {
109     gewicht= gewicht + 1.00;
110     lcd.setCursor(14,3);
111     lcd.print (gewicht);
112     delay(1000);
113 }
114
115 tasterstatus_runter=digitalRead(taster_runter);
116
117 if (tasterstatus_runter == HIGH)
118 {
119     gewicht= gewicht - 1.00;
120     lcd.setCursor(14,3);
121     lcd.print (gewicht);
122     delay(1000);
123 }
124
125 tasterstatus_bmi=digitalRead(taster_bmi);
126
127 if (tasterstatus_bmi == HIGH)
128 {
129     lcd.clear();
130     double bmi = gewicht / (entfernung_hoehe/100 * entfernung_hoehe/100);
131     lcd.setCursor(10,0);
132     lcd.print (bmi);
133 }
134
135 tasterstatus_skala=digitalRead(taster_skala);
136
137 if (tasterstatus_skala == HIGH)
138 {
139     lcd.clear();
140     lcd.setCursor(0, 0);
141     lcd.print("BMI Skala");
142     lcd.setCursor(0, 1);
143     lcd.print("unter 18,5: wenig");
144     lcd.setCursor(0, 2);
145     lcd.print("18,5 bis 24,9: gut");
146     lcd.setCursor(0, 3);
147     lcd.print("30 - 34,9: zu viel");
148 }
149
150
151 }

```