**Anlagen**

**I. Projekttagebuch**

**I.I. Organisatorisches**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Datum** | **Tätigkeit** | **Erreicht** | **Nächstes Ziel** | **gelernt** | **Dauer in Stunden** |
| 03.01.2021 | Arduino IDE installiert | Kann anfangen zu Programmieren | -Arduino erwerben | -Arduino IDE ist Grundlage für Programmierung mit Arduino | 0,1 |
| 21.01.2021 | Zeitplan erstellt und abgegeben | Zeitplan erstellt | -Inhaltsverzeichnis erstellen | - | 0,5 |
| 18.02.2021 | Inhaltsverzeichnis angefertigt und abgegeben | Inhaltsverzeichnis erstellt | -Einleitung erstellen | - | 0,5 |
| 05.03.2021 | Einleitung angefertigt und abgegeben | Einleitung erstellt und abgegeben | -Hardwarekonzept erstellen | - | 0,3 |
| 26.03.2021 | Hardwarekonzept angefertigt und abgegeben | Hardwarekonzept erstellt und abgegeben | -Softwarekonzept anfertigen | - | 1 |
| 29.04.2021 | Softwarekonzept angefertigt | Softwarekonzept erstellen | - | - | 1 |

**Gesamtzeit 3,4 Stunden**

**I.II. Recherche**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Datum** | **Tätigkeit** | **Erreicht** | **Nächstes Ziel** | **gelernt** | **Dauer in Stunden** |
| 01.03.2021 | Über Verkabelung recherchiert | Jumperkabel erworben | -Über Arduino recherchieren | -Komponenten werden mit Kabeln verbunden | 1 |
| 08.03.2021 | Über Arduino recherchiert | Arduino Starter Kit erworben | -Über benötigte Komponenten recherchiert | -Arduino ist wichtig für | 1 |
| 01.04.20201 | Über Speichermöglichkeiten recherchiert | SD Shield Erworben | -Ein- und Ausgabegeräte recherchieren, -Speichermöglichkeit durchgehen | -Arduino hat nur RAM | 2 |
| 01.04.2021 | Über Ein- und Ausgabe recherchiert | Taster sowie I2C-Display Erworben | -Lötkolben für LCD Pins erwerben | -Ein- und Ausgabe ist nicht im Arduino integriert | 2 |
| 01.04.2021 | Über Lötkolben recherchiert | Lötkolben erworben | -LCD Display löten | -Nicht alle Komponenten sind Einbaufertig | 1 |
| 14.04.2021 | Über alternative Eingabemöglichkeiten recherchiert | TCRT5000 Sensoren erworben | -TRCT5000 Sensoren einbauen | -Sensoren sind für das Projekt besser geeignet | 2 |
| 24.04.2021 | Über Stromversorgung recherchiert | Netzteil erworben | -Über Backup recherchieren | -Falsches Netzteil führt zu Schäden | 1 |
| 21.05.2021 | Über Fall eines Spontanen Ausfalls recherchiert | Backupplan erstellt | - | -Immer auf Nummer sicher gehen | 2 |

**Gesamt 12 Stunden**

**I.III. Tätigkeit**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Datum** | **Tätigkeit** | **Erreicht** | **Nächstes Ziel** | **gelernt** | **Dauer in Stunden** |
| 01.04.2021 | Programmiert | Mit Taster funktionierender Personenzähler | LCD löten | - | 4 |
| 01.04.2021 | gelötet | LCD verlötet und verbaut | Programmierung auf LCD anpassen | Löten | 1 |
| 02.04.2021 | Programmierung angepasst (LCD) | Ausgabe via LCD | Über alternative Programmierweisen recherchieren | Werteausgabe per LCD | 2 |
| 14.04.2021 | Programmierung angepasst (Sensoren) | Personenzähler mit Sensoren | Testen | Umgang mit Sensoren | 4 |
| 14.04.2021 | Programm getestet | Programm getestet | Programm auf drei Sensor Taktik umformen | Programmtestung | 1 |
| 18.04.2021 | Programmierung angepasst (Drei Sensor Taktik) | Personenzähler mit drei Sensor Taktik | Testen auf verschiedene Szenarien | Umgang mit Sensoren | 8 |
| 20.04.2021 | Programm getestet | Programm auf Herz und Nieren geprüft | Dokumentation vollenden | Programmtestung | 3 |

**Gesamt 23 Stunden**

**I.IV. Dokumentation**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Datum Anfang** | **Datum Ende** | **Kapitel** | **Geplante Zeit** | **Tatsächliche Zeit** |
| 05.03.2021 | 17.06.2021 | 1. Umfeld | 1 | 1 |
| 05.03.2021 | 17.06.2021 | 2. SOLL-Zustand | 2 | 1 |
| 21.01.2021 | 17.06.2021 | 3. Planung | 1 | 1 |
| 20.04.2021 | 17.06.2021 | 4. IST-Zustand | 1 | 2 |
| 01.03.2021 | 17.06.2021 | 5. Kostenkalkulation | 3 | 3 |
| 01.04.2021 | 17.06.2021 | 6. Implementierung | 2 | 3 |
| 20.04.2021 | 17.06.2021 | 7. SOLL/IST Vergleich | 1 | 1 |
| 20.04.2021 | 17.06.2021 | 8. Nutzungsanalyse | 4 | 3 |
| 24.04.2021 | 17.06.2021 | 9. Fazit | 1 | 1 |
| 17.06.2021 | 17.06.2021 | I. Projekttagebuch | 1 | 2 |
| 17.06.2021 | 17.06.2021 | II. Glossar | 1 | 1 |
| 17.06.2021 | 17.06.2021 | III. Verwendete Werkzeuge | 1 | 1 |
| 26.03.2021 | 17.06.2021 | IV. Konzept | 2 | 2 |
| 16.06.2021 | 17.06.2021 | V. Kundendokumentation | 1 | 1 |
| 05.03.2021 | 17.06.2021 | VI. Quelltextdokumentation | 4 | 3 |

**Gesamt 26 Stunden 26 Stunden   
Zeit Gesamt 64,4 Stunden**

**II. Glossar**

|  |  |  |
| --- | --- | --- |
| **Nummer** | **Begriff** | **Bedeutung** |
| 1 | BBS Verden | Schule in Verden |
| 2 | Personenzähler | Meist mechanisches Gerät zur Messung auf Knopfdruck |
| 3 | Datensatz | Eine Gruppe inhaltlich zusammenhängender Daten |
| 4 | Sensorik | Anwendung von Sensoren zur Messung und Kontrolle von Veränderungen |
| 5 | Corona | Meldepflichtige Infektionskrankheit |
| 6 | Präposition | Zeigt, wie etwas sich zum anderen verhält |
| 7 | Budget | Gegenüberstellung von Einnahmen und Ausgaben |
| 8 | Integriert | Zusammenschluss einzelner Elemente zu einem System |
| 9 | delay | Zeitintervall, um welchen ein Ereignis verzögert wird |
| 10 | RTC | Uhr, welche die Physikalische Zeit misst |
| 11 | CSV | Format mit Aufbau einer Textdatei zur Speicherung einfach strukturierter Daten |
| 12 | XLSX | Open Office XML basiertes Dateiformat |
| 13 | LCD | Auf Flüssigkristallen basierte Anzeige |
| 14 | int | Ganzzahliger Datentyp |
| 15 | float | Datentyp zur angenäherten Darstellung Reeler Zahlen |
| 16 | string | Datentyp für endlichen Reihenfolge von Zeichen |
| 17 | char | Datentyp zur Wiedergabe einzelner Zeichen |
| 18 | file | Objekt der Arduino SD.h Bibliothek zum beschreiben und Lesen von Dateien |
| 19 | void | Datentyp ohne Rückgabewert |
| 20 | Struktogramm | Diagrammtyp zur Darstellung von Programmentwürfen |
| 21 | NSD | Dateiendung für Dateien vom Structorizer |
| 22 | Infrarot | Elektromagnetische Strahlung im Spektralbereich zwischen sichtbarem Lich und Terahertzstrahlung |

Vorschau. Siehe „Dateien/Quellen/Quellenverzeichnis“ für Quellen.

**III. Verwendete Werkzeuge**

**III.I. Beschreibung**  
Für die Umsetzung des Projektes sind verschiedene Tools eingesetzt worden.   
Im Folgenden eine Liste der verwendeten Werkzeuge nach Kategorie.   
Nach Gesprächen mit Herr Düren am 17.06.2021 bleibt die Nennung derer Quellen aus.

**III.II. Software**

|  |  |
| --- | --- |
| **Zweck** | **Programm** |
| Programmierumgebung | Arduino DIE |
| Struktogrammerstellung | Structorizer |
| Anfertigen der Dokumentation | Microsoft Office Suite |
| Erstellen der Schaltskizze | Fritzing |

**III.III. Hardware**

|  |  |
| --- | --- |
| **Zweck** | **Programm** |
| Festigen der LCD Pins | Lötkolben |
| Konfigurieren der Potentiometer mehrerer Komponenten | Kreuzschraubenzieher |

**IV. KONZEPT**

**IV.I. Bibliotheken**

|  |  |
| --- | --- |
| **Name** | **Funktion** |
| Wire.h | Für die Kommunikation mit I2C Geräten |
| LiquidCrystal\_I2C.h | Um über i2c mit dazugehörigem LCD zu kommunizieren |
| SPI.h | Um mit SPI Geräten zu kommunizieren |
| SD.h | Um auf die SD zu schreiben bzw. von dieser zu lesen |

Sämtliche Bibliotheken sind, vorausgesetzt sie sind nicht in der IDE integriert, im zugehörigen Ordner unter „Dateien/Programm/Bibliotheken/“ im digitalen Anhang zu finden.

**IV.II. Variablen**

|  |  |  |
| --- | --- | --- |
| Name | Typ | Funktion |
| maximaleAnzahlPersonen | int15 | Maximal im Raum erlaubte Anzahl an Personen |
| aktuelleAnzahlPersonen | Int | Aktuelle Personenzahl im Raum |
| zeitMesser | float16 | Wird als Zeitstempel genutzt |
| zeitSprung | int | Zeitabstand zwischen Durchläufen. In Millisekunden |
| raumNummer | string17 | Gibt Raumnummer wieder |
| Status | string | Gibt aus, ob noch Platz ist oder sich jemand vor den mittleren Sensoren stellt. |
| anzahlAenderung | char18 | Gibt im Datensatz an, ob jemand hinzugekommen ist oder rausgegangen ist |
| personenZahlGesamt | int | Insgesamt vorbeigegangene Personen |
| personenZahlRein | int | Insgesamt reingegangene Personen |
| personenZahlRaus | int | Insgesamt rausgegangene Personen |
| modulSDPin | int | Variable für SD Datenpin |
| myFile | file19 | Für Zugriff auf Dateien |
| Dateiname | string | Um auf bestimmte Datei zuzugreifen |
| pinSensorRein | int | Pin für Sensor fürs reingehen |
| pinSensorRaus | int | Pin für Sensor fürs rausgehen |
| pinSensorMitte | int | Pin für Sensor in der Mitte |
| fertig | int | Variable um Änderung der Personenzahl zu bestätigen |

|  |  |
| --- | --- |
| **Typ** | **Beschreibung** |
| int | Dient zur Speicherung ganzer Zahlen. |
| float | Dient zur Speicherung von Gleitkommazahlen |
| string | Dient zur Speicherung von endlichen Zeichenketten |
| char | Dient zur Speicherung einzelner Zeichen |
| file | Dient zur Speicherung einer Datei als Variable |
| void | Funktionstyp ohne Rückgabewert |

**IV.III. Unterfunktionen**

**IV.III.I. Beschreibung**

Neben den Setup- sowie Loopteil Funktionen hat das Programm insgesamt zwei Unterfunktionen. Die erste, vom Typ void20 und mit dem Namen Speichern(), dient der Speicherung der Datensätze. Eine zweite Unterfunktion, vom Typ void und mit dem Namen zeitMesserErhoehung(), dient der Erhöhung des Zeitstempels.

**IV.III.II. Speichern**  
Die Funktion dient zum Speichern der Datensätze und wird bei jeder Betätigung des Bestätigers aufgerufen. Für eine Erklärung der Variablenfunktion siehe Kapitel „IV.II. Variablen“.

void Speichern() //Speicherfunktion  
{  
 myFile = SD.open(Dateiname, FILE\_WRITE); //SD objekt zuweisen  
 myFile.print(zeitMesser, 3);   
 myFile.print(";");   
 myFile.print(maximaleAnzahlPersonen);  
 myFile.print(";");   
 myFile.print(aktuelleAnzahlPersonen);  
 myFile.print(";");   
 myFile.print(anzahlAenderung);   
 myFile.print(";");   
 myFile.print(personenZahlGesamt);   
 myFile.print(";");   
 myFile.print(personenZahlRaus);   
 myFile.print(";");   
 myFile.print(personenZahlRein);   
 myFile.print(";");   
 myFile.print(Status);   
 myFile.println(";");   
 myFile.close();   
}

Siehe „Dateien/Programmcode/Personenzaehler/Personenzaehler.ino“ für Vollständigen Programmcode

**IV.III.III. Zeitstempel erhöhen**Die Funktion dient zum erhöhen des Zeitstempels um die Dauer des Delays  
Für eine Erklärung der Variablenfunktion siehe Kapitel „IV.II. Variablen“.

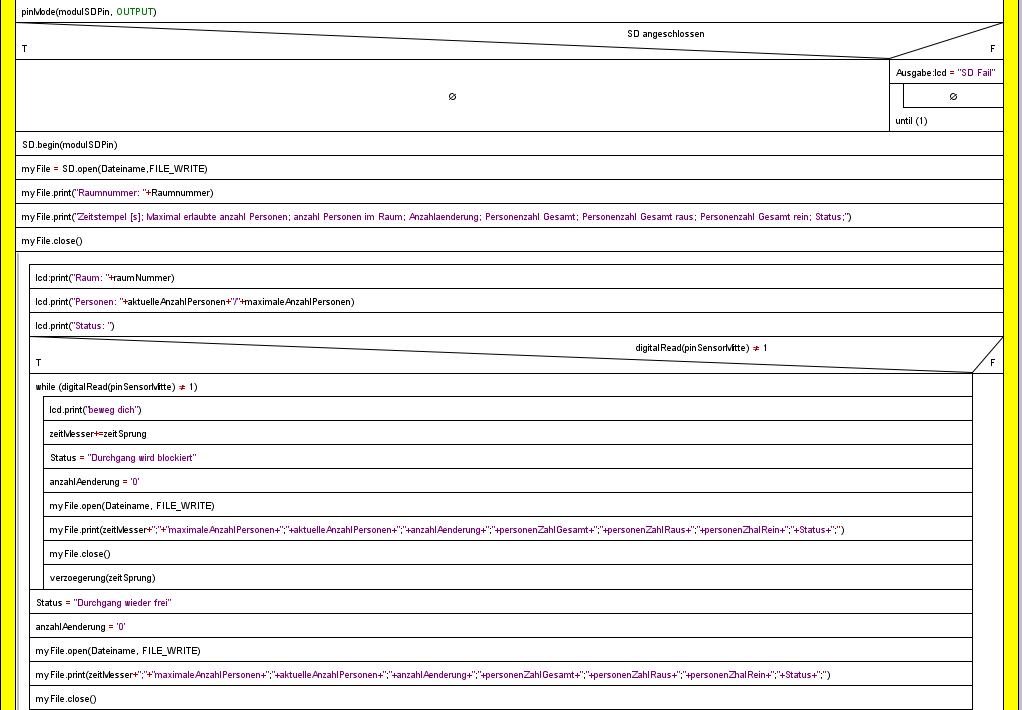
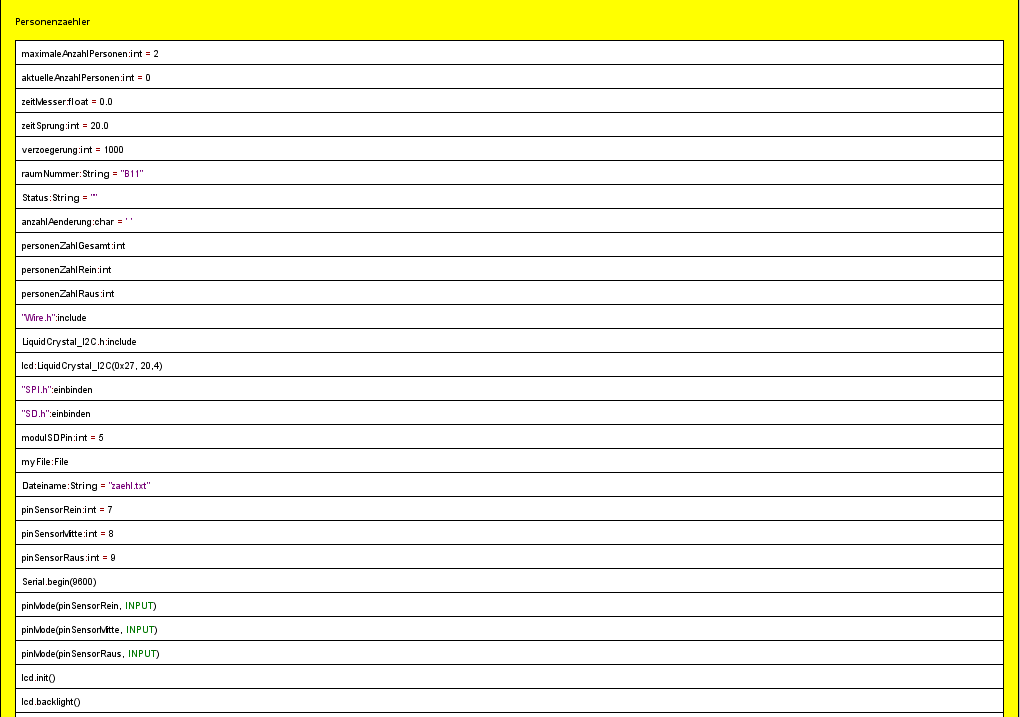
void zeitMesserErhoehung() //Funktion um zeitMesser um zeitSprung Sekunden erhoehen

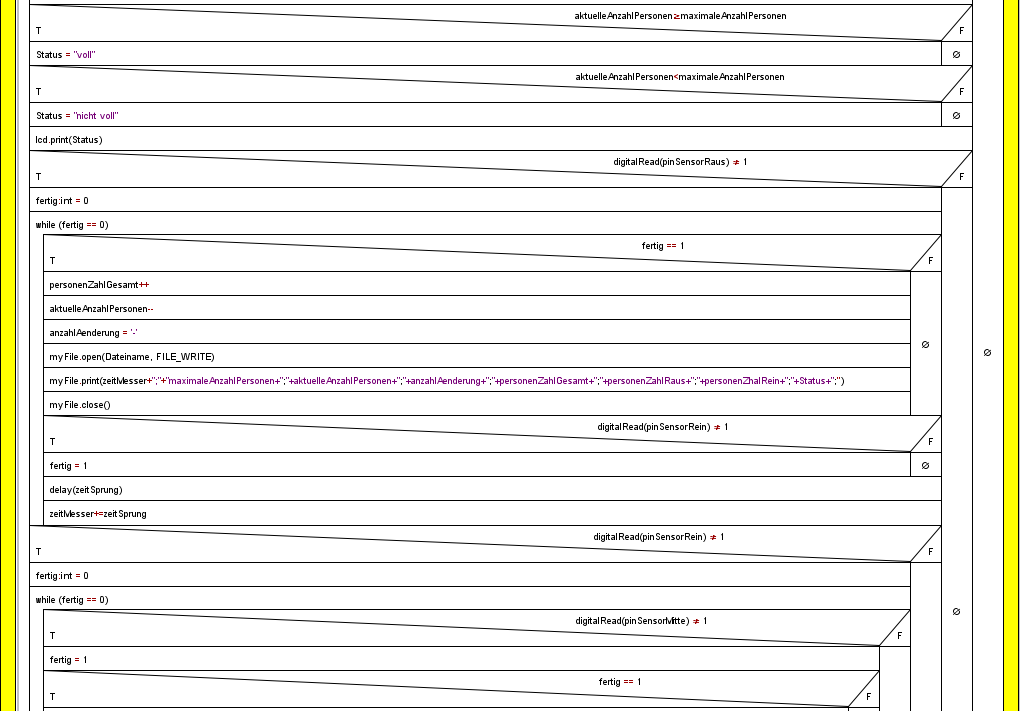
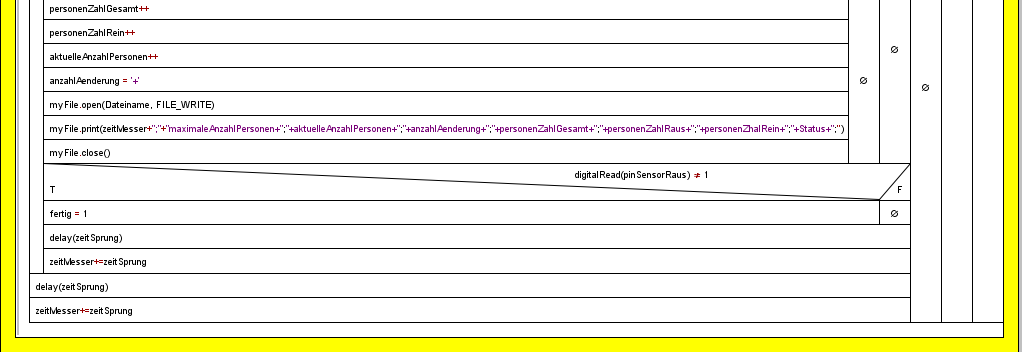
{

zeitMesser = zeitMesser+((float)zeitSprung/1000); //zeitMesser um zeitSprung Sekunden

}

Siehe „Dateien/Programmcode/Personenzaehler/Personenzaehler.ino“ für Vollständigen Programmcode

**IV.IV. Struktogramm21**  


  
  
Siehe „Dateien/Struktogramm/struktogrammVoll.png“ sowie  
„Dateien/Struktogramm/arduino projekt.nsd“ im digitalen Anhang für Vollständiges Struktogramm.  
Dabei ist zu beachten, dass alle Einträge drin sind, aber in einer anderen Reihenfolge

Für .nsd22 Dateien ist der Structorizer notwendig.   
Verfügbar unter https://www.chip.de/downloads/Structorizer\_64884440.html

**IV.V. Pinbelegung**

|  |  |  |  |
| --- | --- | --- | --- |
| **Zweck** | **Bauteilname** | **Shield Pin** | **Arduino Pin** |
| Sensor Innen | TCRT5000 | + | 5V |
|  |  | - | GND |
|  |  | s | D9 |
| Sensor Mitte | TCRT5000 | + | 5V |
|  |  | - | GND |
|  |  | s | D8 |
| Sensor Außen | TCRT5000 | + | 5V |
|  |  | - | GND |
|  |  | s | D7 |
| Speicherung | MicroSD | GND | GND |
|  |  | VCC | 5V |
|  |  | CS | D5 |
|  |  | SCK | D13 |
|  |  | MOSI | D11 |
|  |  | MISO | D12 |
| Ausgabe | 20x4 2004 LCD | GND | GND |
|  |  | VCC | 5V |
|  |  | SDA | A4 |
|  |  | SCL | A5 |
| Stromversorgung | Netzteil 9V 2A | - | Netzteilanschluss |

Siehe „Dateien/Hardware/Pinbelegung.xlsx“ im digitalen Anhang für Vollständiges Auflistung.

Da der Arduino nicht über ausreichende Pins für die Stromversorgung der Komponenten besitzt wird diese Funktionalität über das Breadboard erweitert. VCC wird mit der + Reihe und GND mit der – Reihe verbunden.   
Für eine Grafische Darstellung siehe Kapitel „IV.VII. Skizze“

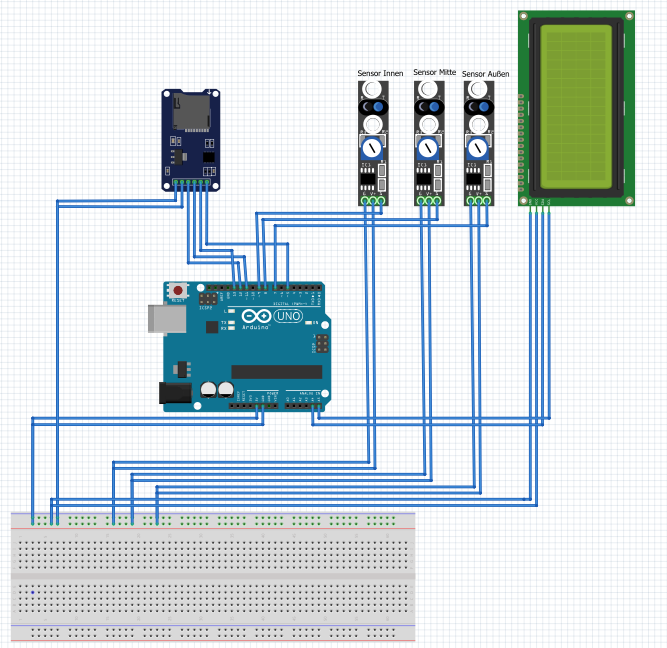
**IV.VI. Bauteilliste**Für das Projekt wird eine Infrarot23 Sensorik bestehend aus drei Sensoren für die Eingabe genutzt, ein SD Kartenmodul für die Speicherung der Datensätze und ein LCD-Display für die Ausgabe der aktuellen Personenzahl verwendet.

Bauteilliste

|  |  |  |
| --- | --- | --- |
| **Anzahl** | **Bauteilbezeichnung** | **Funktion** |
| 3 | TCRT5000 | Sollen dem Arduino ein Signal abschicken, um je nach Sensorkombination den Zählwert zu erhöhen oder zu verringern. |
| 1 | Micro SD-Reader | Erhält vom Arduino die Datensätze, um diese dann zu speichern. |
| 1 | 20x4 2004 LCD-Display | Gibt die errechnete Personenzahl aus |
| 1 | Arduino Uno Revision 3 | Erhält von der Sensorik die Signale zur Veränderung des Zählwertes und gibt die Veränderung mitsamt eines Zeitstempels als Datensatz an die SD-Karte bzw. das Display. |
|  | Jumper Kabel | Zur Verbindung des Arduinos mit den Komponenten |

Siehe „Dateien/Hardware/Bauteilliste.xlsx“ im digitalen Anhang für Vollständiges Auflistung.

**IV.VII. Skizze**

****Vorschaubild. Siehe „Dateien/skizze/Steckplatine.png“ sowie „Steckplatine.fzz“ im selben Verzeichnis im digitalen Anhang.

**V. Kundendokumentation**

**V.I. Bedienungsanleitung**

Benötigtes Werkzeug: Kreuzschraubenzieher

1. Bauen Sie den Arduino entsprechend der Pinbelegung Kapitel „IV.V. Pinbelegung“ auf.  
Eine grafische Hilfestellung wird durch Kapitel „IV.VII. Skizze“ gestellt.

2. Befestigen Sie die Sensoren Horizontal in Reihe an einer Oberfläche. Achten Sie dabei darauf, dass der innere und der äußere Sensor jeweils rechts und links vom mittleren Sensor sind.   
Beachten Sie dabei, dass der Datenpin des inneren Sensors mit dem Digitalen Pin D9, der des mittleren Sensors mit dem Digitalen Pin D8 und der des äußeren Sensors mit dem Digitalen Pin D7 verbunden sind.

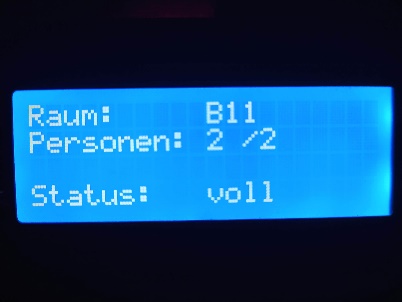
3. Legen Sie eine Mikro SD-Karte ein

4. Schließend Sie den Arduino per USB-A auf USB-B Stecker an einen Computer. Vorteilhafter ist ein entsprechendes Netzteil.   
Empfohlen wird ein 9V 2A Netzteil.

5. Der LCD leuchtet auf und sollte in etwa folgendermaßen aussehen

6. Sobald Sie am äußeren und anschließend am mittleren Sensor vorbeigehen wird die Personenzahl, links im Bild 0, um eins erhöht.

Solange weniger als die maximale Anzahl an Personen, in diesem Fall zwei, im Raum sind, wird in der vierten Zeile der Status „nicht voll“ ausgegeben. Zu erwähnen ist, dass die Personenzahl im Raum kann nicht unter 0 sinken

  
Wenn die Aktuelle Anzahl an Personen so groß wie die Maximal erlaubte Anzahl ist oder diese übersteigt, so wird der Status auf Voll gesetzt, und bleibt solange so, bis die Personenzahl im Raum unter die maximal erlaubte Anzahl sinkt.

7. Um den Personenzähler auszuschalten trennen Sie diesen von der Stromversorgung.

Datensätze

Sobald der mittlere Sensor betätigt wird, egal ob alleine oder nach Betätigung einer der äußeren Sensoren, wird ein Datensatz erstellt. Der Datensatz enthält:

-Raumnummer: Gibt wieder, wo der Personenzähler aufgebaut ist. Dieser Wert kann vom Nutzer im Konfigurationsteil der Programmierung verändert werden

-Zeitstempel: Standardmäßig 20 Millisekunden genauer Zeitstempel. Gibt an, wann der Datensatz nach Start eingetragen worden ist. Die Angabe erfolgt In Sekunden.

-Maximale Anzahl an Personen: Maximal erlaubte Anzahl an Personen im Raum. Dieser Wert kann vom Nutzer im Konfigurationsteil der Programmierung

-Aktuelle Anzahl an Personen: Aktuell im Raum befindliche Anzahl an Personen.

-Anzahländerung: Gibt wieder, ob jemand rein bzw. rausgegangen ist oder ob der Bestätiger blockiert wird.

-Gesamtzahl: Gibt an, wie viele Personen insgesamt vorbeigegangen sind.

-Gesamt raus: Insgesamt rausgegangene Anzahl an Personen.

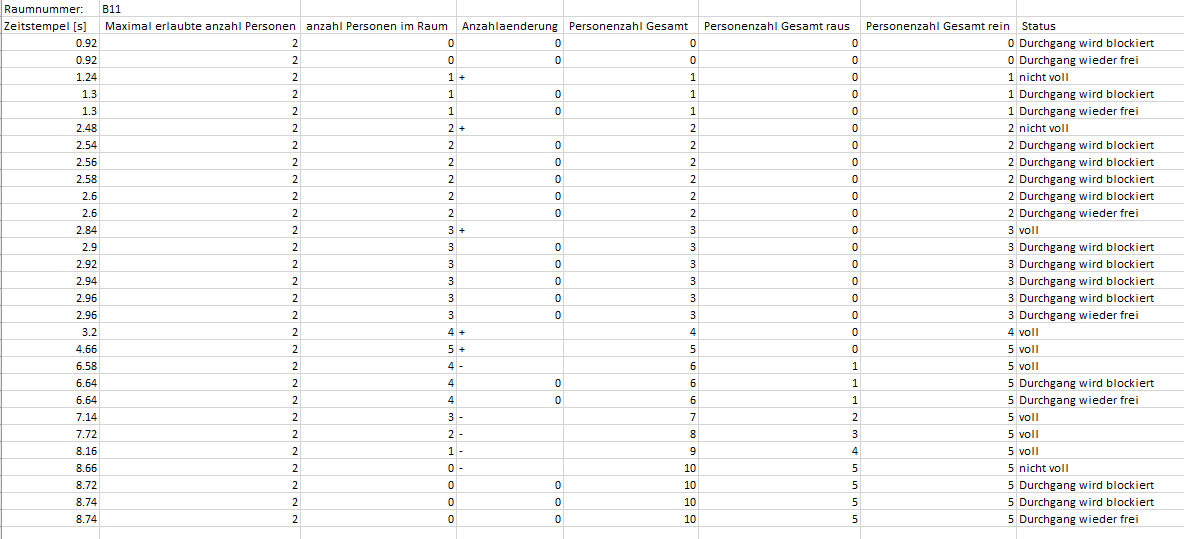
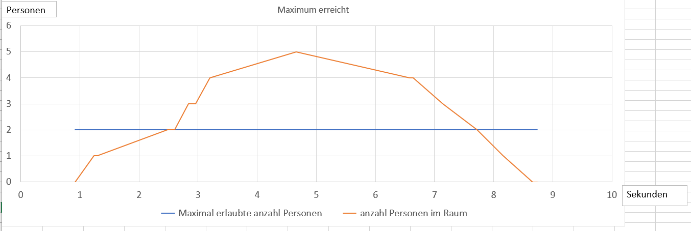
-Gesamt rein: Insgesamt reingegangene Anzahl an Personen.

-Status: Gibt wieder, ob der Raum voll ist, nicht voll ist oder ob der Bestätiger blockiert oder wieder frei ist.

Weiterhin kann der Zeitsprung, die „Sensibilität“ der Sensoren auf Bewegung, vom Nutzer im Konfigurationsteil der Programmierung verändert werden. Da es sich beim Personenzähler um Sensible Hardware handelt, ist es besonders wichtig, auf eine entsprechende Kühlung zu achten.

Der Arduino legt die Datensätze im CSV1 Format ab. Da er dieses aber nicht kennt, speichert er diese mit der Endung einer Textdatei ab. Der Vorteil besteht darin, dass Excel durch diese eine Möglichkeit zur Darstellung mit Diagrammen hat.

Folgend eine Darstellung einer ins Excel Format umgespeicherten Tabelle aus Datensätzen:

  
Vorschaubild. Siehe „Dateien/Testdurchlauf/maximum erreicht/Tabelle.xlsx“ auf Folie 1 sowie 2 im digitalen Anhang.

Konfigurationsteil.

  
Vorschaubild. Siehe „Dateien/Programm/Konfigurierbare Variablen.png“ sowie   
„Dateien/Programm/Personenzähler/Personenzaehler.ino“ im digitalen Anhang.

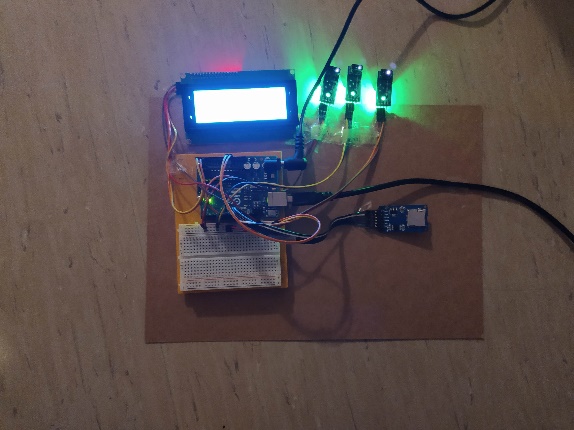
Bei den Farblich markierten Feldern handelt es sich um durch den Benutzer konfigurierbare Variablen.

Die rot markierte Variable, maximaleAnzahlPersonen, dient zur Festlegung der maximal erlaubten Anzahl an Personen.

Die grün markierte Variable, zeitSprung, dient zur Festlegung der Sensibilität des Arduinos.

Die blau markierte Variable, raumNummer, dient zur Festlegung des Raumes, in welchem der Personenzähler aufgebaut worden ist.

Die lila markierte Variable, Dateiname, dient zur Festlegung des Dateinamens mitsamt der Dateiendung, unter welchem die Datei abgespeichert wird.  
Empfohlen wird, den Dateinamen samt Dateiendung nicht Länger als zehn Zeichen zu machen.

Hier ein Beispiel eines fertig aufgebauten

Vorschaubild. Siehe „Dateien/Fotos/Personenzähler.jpg“ für vergrößerte Ansicht

Hinweise

-Nutzen Sie den Schraubenzieher, um die Potentiometer an den Sensoren sowie dem LCD-Display einzurichten.  
Dieser Schritt ist zur Einstellung der Empfindlichkeit der Sensoren sowie des Kontrastes des LCD-Displays notwendig.

-Ziehen Sie den Stecker, um die Stromversorgung zu unterbrechen.

-Sollte die SD-Karte nicht eingebaut sein, so wird eine Fehlermeldung über den LCD-Display ausgegeben.

-Sollten Sie den Personenzähler ohne SD-Karte betreiben wollen, so ist es dennoch nötig, diesen beim Start einzufügen. Nach Ausgabe der Raumnummer, Personenzahl sowie des Status kann die SD-Karte jederzeit entfernt werden.

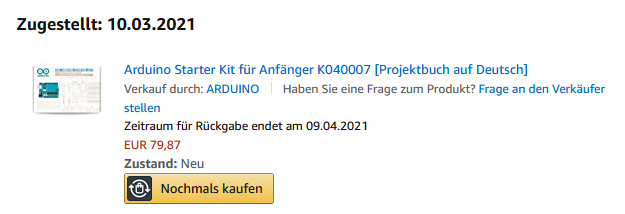
-Sollte der Potentiometer des Displays nicht korrekt eingestellt sein, so wird dieser nichts ausgeben.

-Der Potentiometer der Sensoren ist je nach Lichtbestrahlung zu drehen, um durch Sonnenstrahlen verursachte Fehlmesswerte zu minimieren.

**V.II. Rechnungen der verbauten Komponenten**

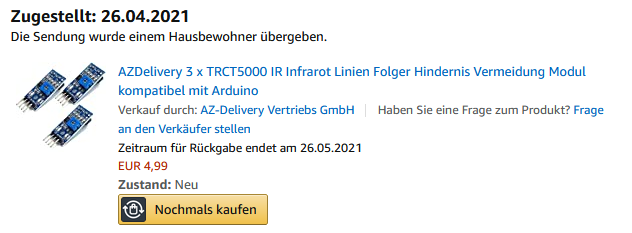
Auflistung der Rechnungen der im Projekt verwendeten Komponenten.

**V.II.I. Arduino**



Vorschaubild. Siehe „Dateien/Rechnungen/Arduino Uno/rechnungAmazonArduino.png“ für vergrößerte Ansicht.

**V.II.II. Infrarotsensoren**



Vorschaubild. Siehe „Dateien/Rechnungen/Sensoren/rechnungAmazonSensor.png“ für vergrößerte Ansicht.

**V.II.III. Netzteil und Mikro SD-Karte**

****Vorschaubild. Siehe „Dateien/Rechnungen/Netzteil und SD/rechnungAmazonNetzteilSD.png“ für vergrößerte Ansicht. Die SD-Karte wird Preislich nicht mit zum Projekt gezählt da sie nur nebenbei für das Projekt genutzt wird.

**V.II.IV. SD Shield**

Vorschaubild. Siehe „Dateien/Rechnungen/SD Shield/rechnungAmazonNetzteilSDShield.png“ für vergrößerte Ansicht.

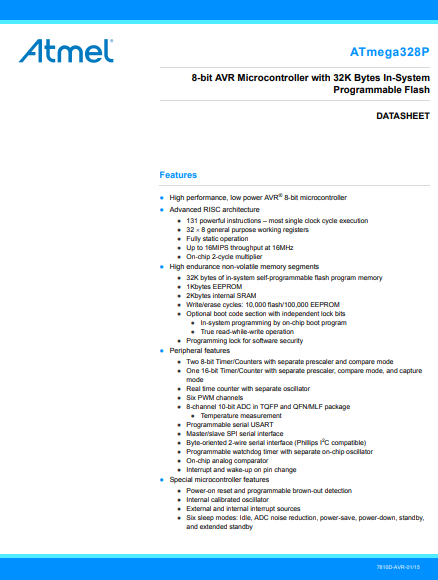
**V.II.V. LCD-Display  
**Vorschaubild. Siehe „Dateien/Rechnungen/LCD und Taster/rechnungAmazonLCD.png“ für vergrößerte Ansicht.

**V.II.VI. Jumperkabel**

****

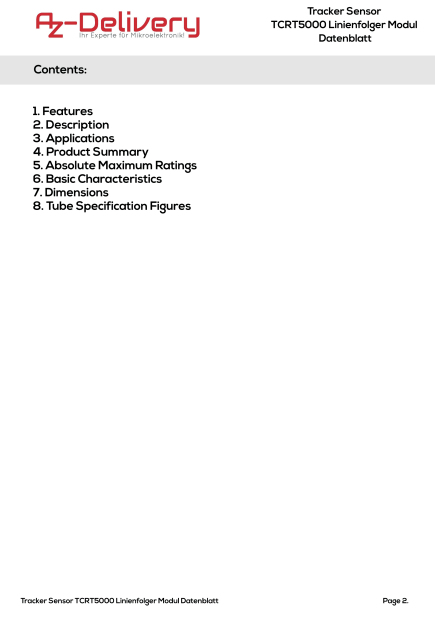
Vorschaubild. Siehe „Dateien/Rechnungen/Jumperkabel/rechnungAmazonJumperkabel.png“ für vergrößerte Ansicht.

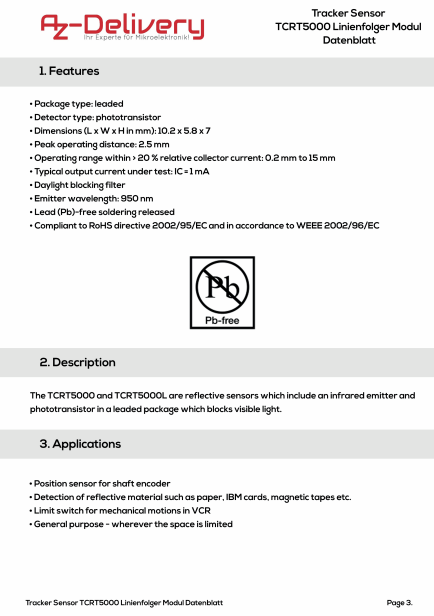
**V.III. Datenblätter**

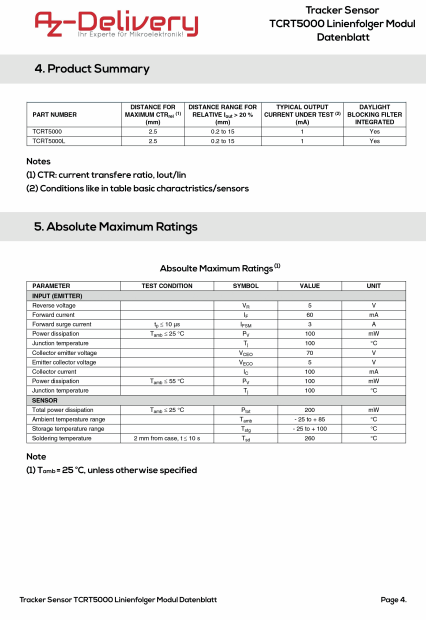
**V.III.I. Arduino Uno Rev 3**  
Vorschaubild. Siehe „Dateien/Datenblätter/Arduino/ATmega328P\_Datasheet.pdf“ für 294 Seiten langes Datenblatt.

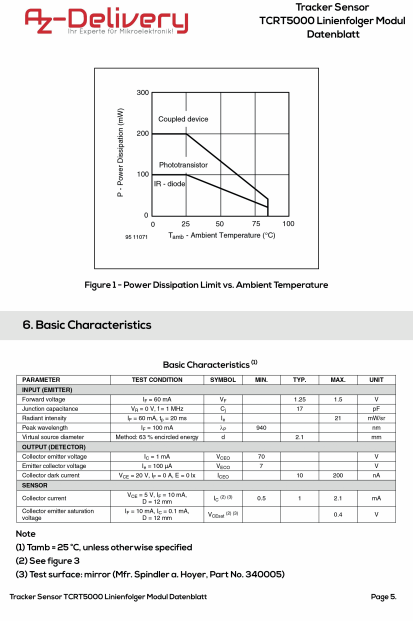
**VI.III.II. Infrarotsensor**

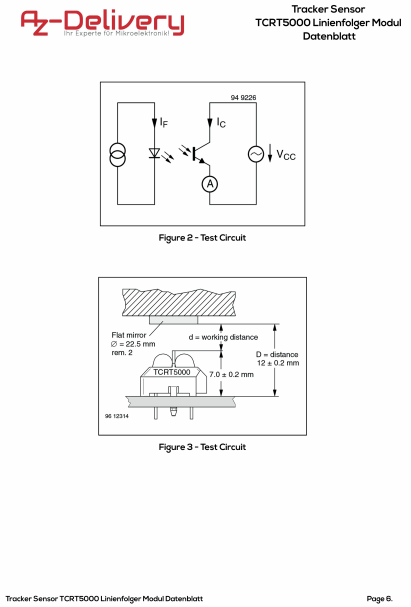


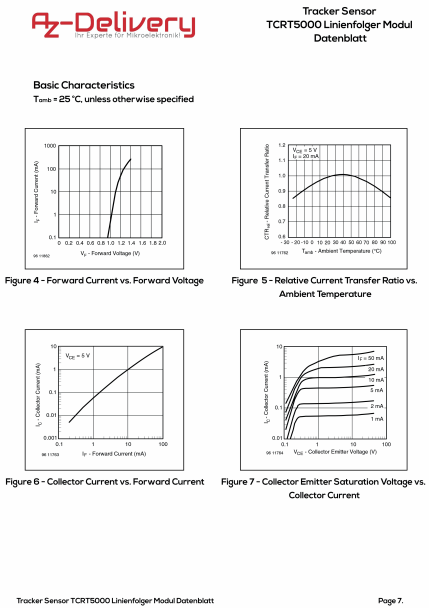


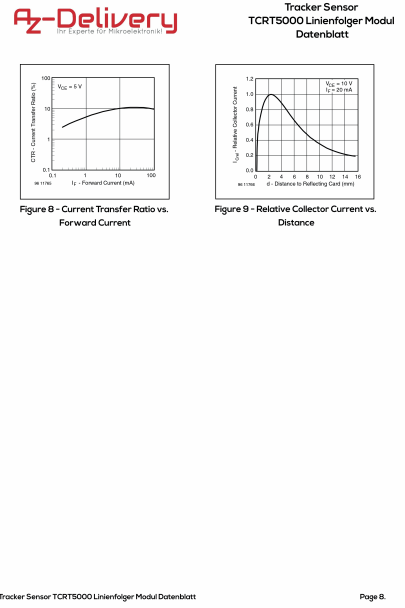


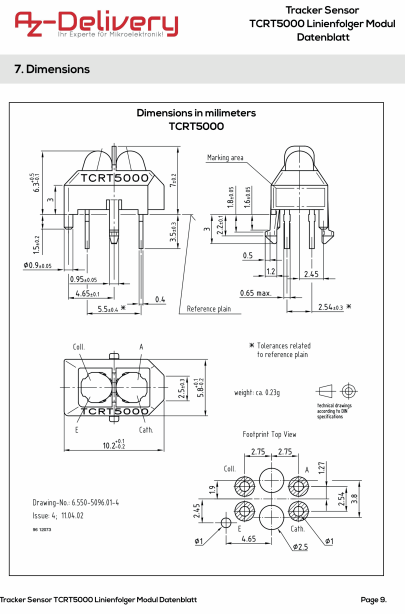


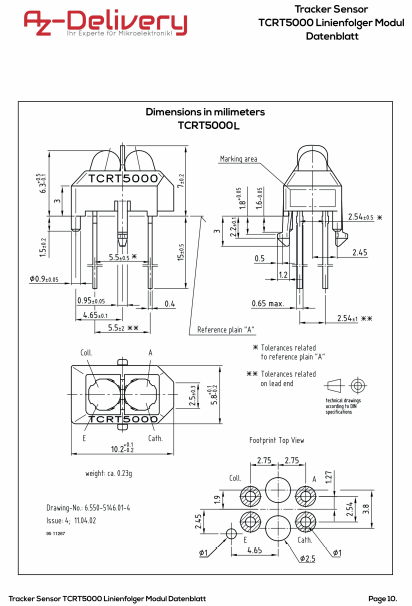


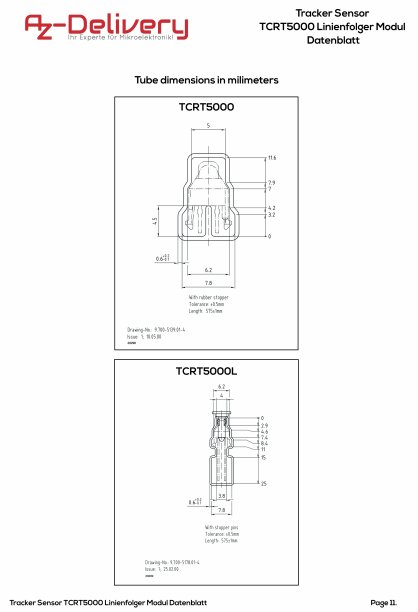


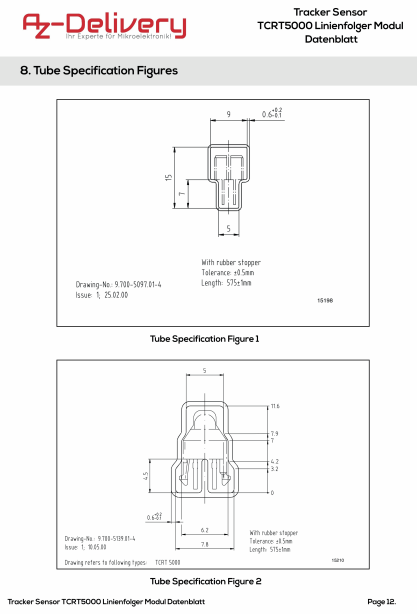


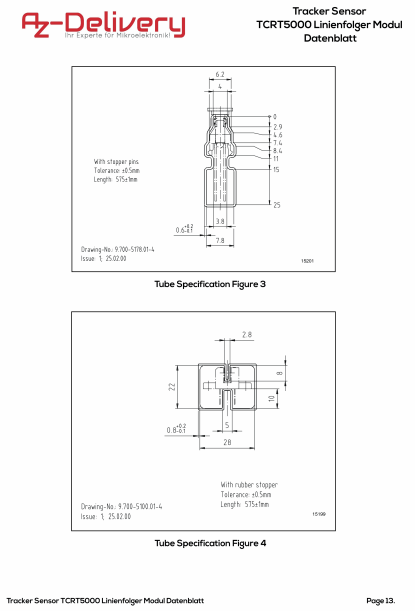


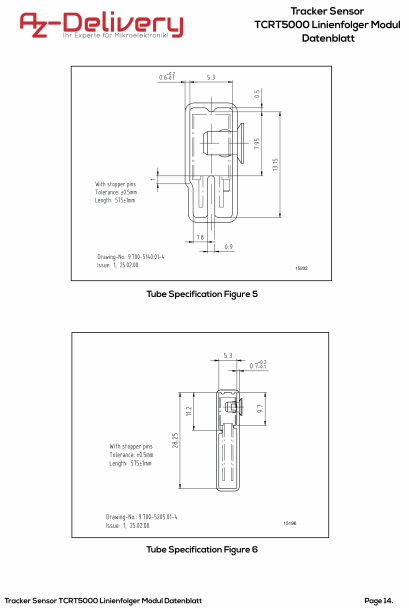


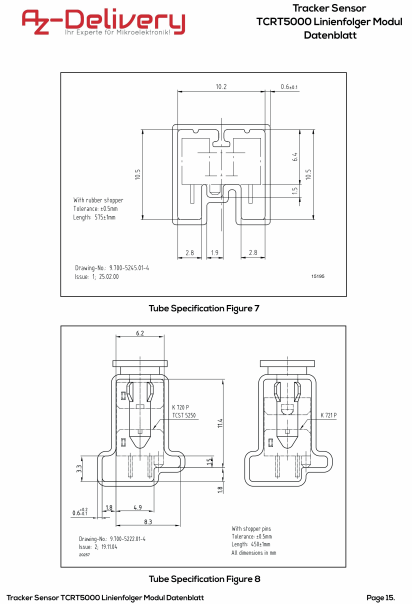




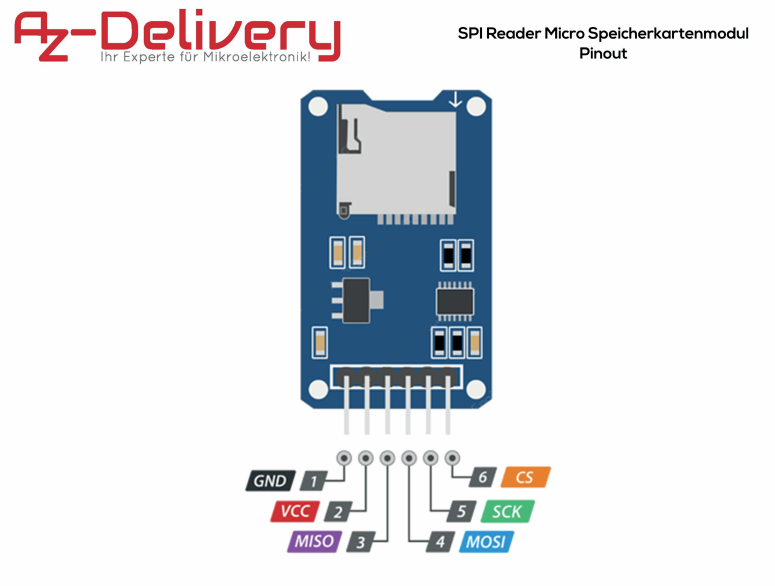


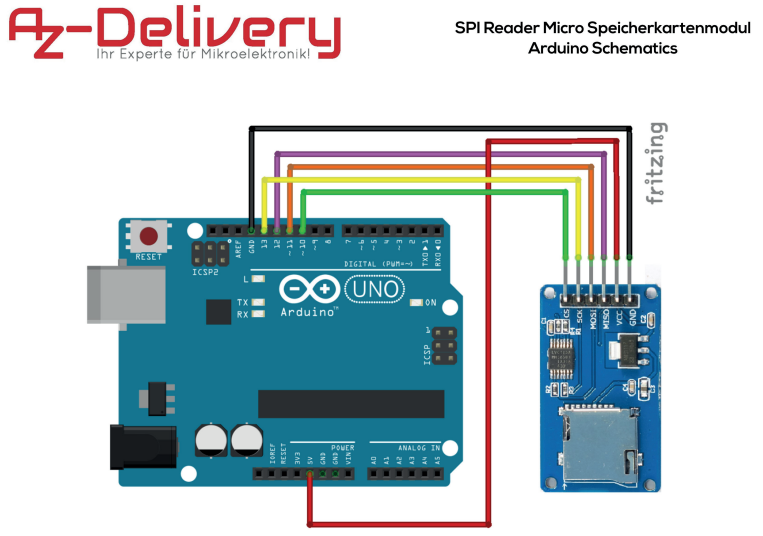


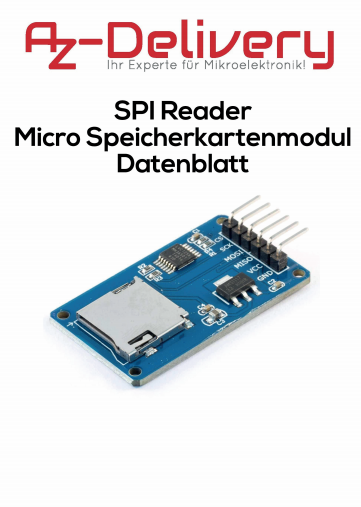


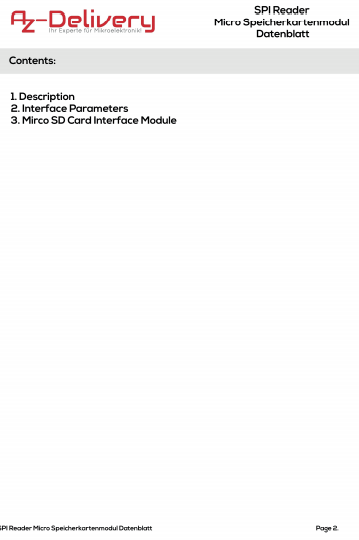
Vorschau. Siehe „Dateien/Datenblätter/Infrarotsensor/TRCT5000 Datenblatt.pdf“ für Datenblatt.

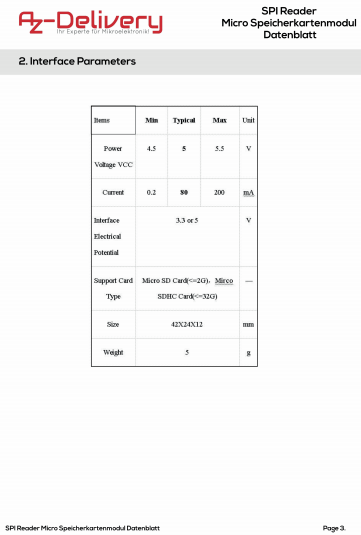
**VI.III.III. Micro SD Shield**  
Pinout

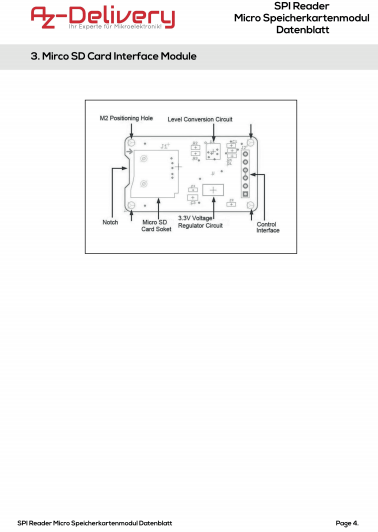
Vorschau. Siehe „Dateien/Datenblätter/SD Modul//SPI Reader Micro Speicherkartenmodul Pinout.pdf“ für Datenblatt.

Arduino Schematics  
  
Vorschau. Siehe „Dateien/Datenblätter/SD Modul/SPI Reader Micro Speicherkartenmodul Arduino Schematics.pdf“ für Datenblatt.

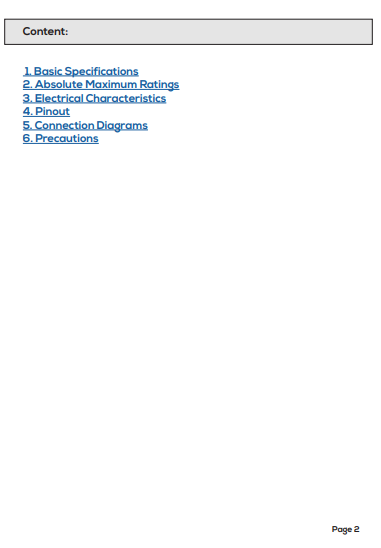
Datenblatt  


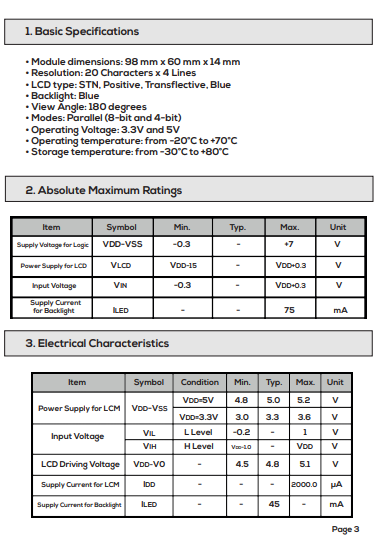


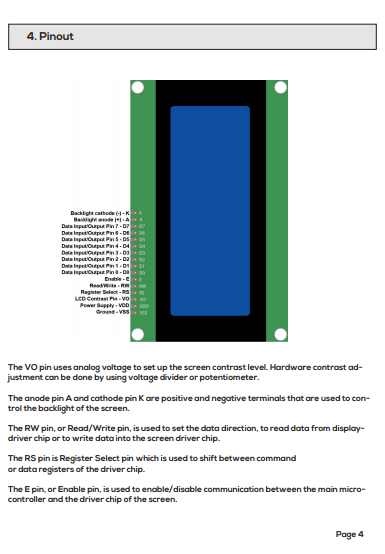


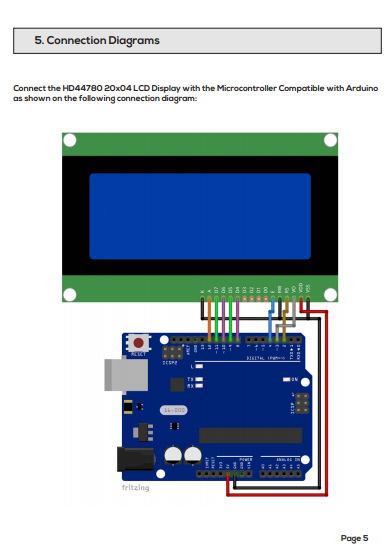
Vorschau. Siehe „Dateien/Datenblätter/SD Modul/SPI Reader Micro Speicherkartenmodul Datenblatt AZ-Delivery Vertriebs GmbH.pdf“ für Datenblatt

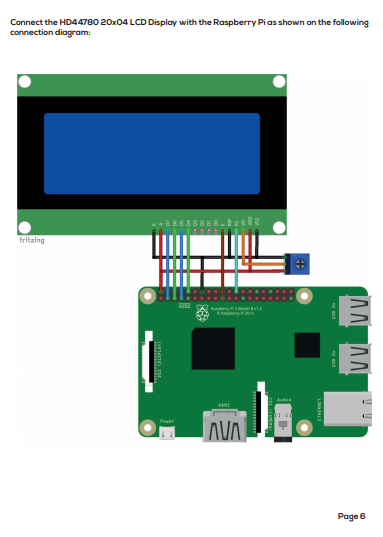
**VI.III.IV. LCD-Display**  
Datenblatt  

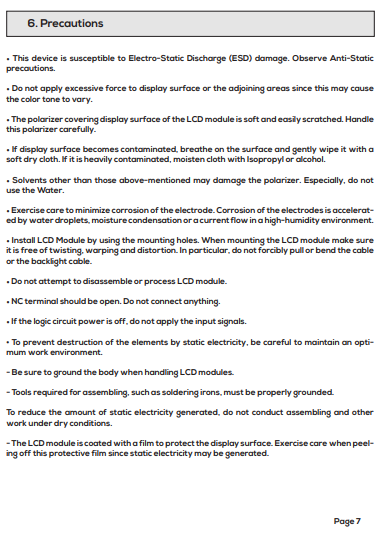



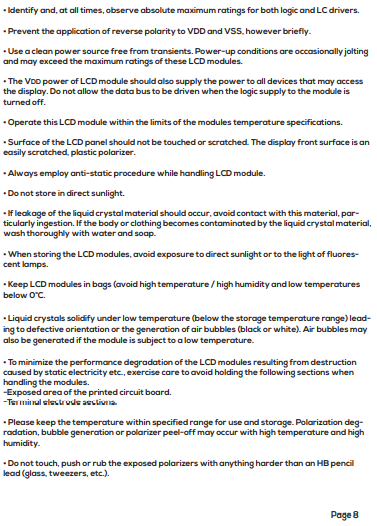












Vorschau. Siehe „Dateien/Datenblätter/lcl/lcd datenblatt.pdf“ für Datenblatt

**VI. Quelltextdokumentation**

**VI.I. Programmcode**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**//organisatorisches**

**int maximaleAnzahlPersonen = 2; //Konfigurierbar**

**int aktuelleAnzahlPersonen = 0;**

**float zeitMesser = 0.0;**

**int zeitSprung = 20; //in ms. Konfigurierbar.**

**int verzoegerung = 1000;**

**String raumNummer = "B11"; //Konfigurierbar**

**String Status = "";**

**char anzahlAenderung = ' ';**

**//Globale Messwertaenderungen**

**int personenZahlGesamt;**

**int personenZahlRein;**

**int personenZahlRaus;**

**//Einrichtung des 4 zeilen lcds**

**#include <Wire.h> // Library for I2C communication**

**#include <LiquidCrystal\_I2C.h> // bibliothek für i2c lcd**

**LiquidCrystal\_I2C lcd = LiquidCrystal\_I2C(0x27, 20, 4); //lcd objekt erstellen**

**//\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**//SD Bezogen**

**#include "SPI.h"**

**#include "SD.h"**

**int modulSDPin = 5;**

**File myFile;**

**String Dateiname = "zaehl3.txt"; //konfigurierbar**

**//\_\_\_\_\_\_\_\_\_\_**

**//festlegen der sensorpins**

**int pinSensorRein = 7;**

**int pinSensorMitte = 8;**

**int pinSensorRaus = 9;**

**//\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**void setup() {**

**//baudrate**

**Serial.begin(9600); //baudrate festlegen**

**//\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**//sensoren als eingabe festlegen**

**pinMode(pinSensorRein, INPUT);**

**pinMode(pinSensorMitte, INPUT);**

**pinMode(pinSensorRaus, INPUT);**

**//\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**//LCD starten**

**lcd.init();**

**lcd.backlight();**

**//\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**//SD einrichtung und erste zeilen mit beschreibung fuellen**

**pinMode(modulSDPin, OUTPUT);**

**//wenn nicht an fehlermeldung ausgeben und dauerschleife**

**if(!SD.begin(modulSDPin))**

**{**

**Serial.print("SD Fail");**

**lcd.setCursor(0,3);**

**lcd.print("SD Fail");**

**while(1);**

**}**

**//Dateiobjekt starten und füllen**

**SD.begin(modulSDPin);**

**myFile = SD.open(Dateiname, FILE\_WRITE);**

**myFile.print("Raumnummer: ;");**

**myFile.print(raumNummer);**

**myFile.println(";");**

**myFile.println("Zeitstempel [s]; Maximal erlaubte anzahl Personen; anzahl Personen im Raum; Anzahlaenderung; Personenzahl Gesamt; Personenzahl Gesamt raus; Personenzahl Gesamt rein; Status;");**

**myFile.close();**

**}**

**void loop() {**

**//LCD ausgabeeinrichtung**

**//Raumnummer**

**lcd.setCursor(0, 0);**

**lcd.print("Raum: ");**

**lcd.setCursor(10, 0);**

**lcd.print(raumNummer);**

**//\_\_\_\_\_\_\_\_\_\_\_**

**//Personenanzahl**

**lcd.setCursor(0,1);**

**lcd.print("Personen: ");**

**lcd.setCursor(10,1);**

**lcd.print(" ");**

**lcd.setCursor(10,1);**

**lcd.print(aktuelleAnzahlPersonen);**

**lcd.setCursor(12,1);**

**lcd.print("/");**

**lcd.setCursor(13,1);**

**lcd.print(maximaleAnzahlPersonen);**

**//\_\_\_\_\_\_\_\_\_\_\_\_**

**//Ausgabe Status**

**lcd.setCursor(0,3);**

**lcd.print("Status: ");**

**//\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**//raumfuelle abfrage Sensor**

**//mitte, falls jemand sich einfach hinstellt**

**if(digitalRead(pinSensorMitte) !=1) //Wenn der mittleresensor ein hindernis hat**

**{**

**while(digitalRead(pinSensorMitte) !=1) //solange der mittlere Sensor ein hindernis hat**

**{**

**lcd.setCursor(0,2);**

**lcd.print("beweg dich");**

**zeitMesserErhoehung();**

**Status = "Durchgang wird blockiert";**

**anzahlAenderung = '0';**

**Speichern();**

**delay(zeitSprung);**

**}**

**lcd.setCursor(0,2);**

**lcd.print(" ");**

**Status = "Durchgang wieder frei";**

**anzahlAenderung = '0';**

**Speichern();**

**}**

**//raumfuelle abfrage**

**//wenn mehr im raum sind als erlaubt**

**if(aktuelleAnzahlPersonen>=maximaleAnzahlPersonen)**

**{**

**Status = "voll";**

**}**

**//wenn weniger im raum sind als erlaubt**

**if(aktuelleAnzahlPersonen<maximaleAnzahlPersonen)**

**{**

**Status = "nicht voll";**

**}**

**//\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**//Status ausgeben**

**lcd.setCursor(10,3);**

**lcd.print(" ");**

**lcd.setCursor(10,3);**

**lcd.print(Status);**

**//\_\_\_\_\_\_\_\_**

**//aussen**

**if(digitalRead(pinSensorRaus) !=1) //wenn der aussensensor ausgeloest wurde beginne prozess zur personenzahlreduzierung**

**{**

**int fertig = 0; //schleifenbeender**

**while(fertig == 0)**

**{**

**if(digitalRead(pinSensorMitte) != 1) //wenn bestaetigung mittlerer Sensor**

**{**

**fertig = 1;**

**if(fertig == 1)**

**{**

**personenZahlGesamt++;**

**if(aktuelleAnzahlPersonen>0)**

**{**

**personenZahlRaus++;**

**aktuelleAnzahlPersonen--;**

**anzahlAenderung = '-';**

**Speichern();**

**}**

**}**

**}**

**if(digitalRead(pinSensorRein) != 1)//wenn statt dem mittleren sensor der reingehsensor betaetigt wird**

**{**

**fertig = 1;**

**}**

**delay(zeitSprung);**

**zeitMesserErhoehung();**

**}**

**}**

**//innen**

**if(digitalRead(pinSensorRein) !=1) //wenn der innensensor ausgeloest wurde beginne prozess zur personenzahlerhoehung**

**{**

**int fertig = 0; //schleifenbeender**

**while(fertig == 0)**

**{**

**if(digitalRead(pinSensorMitte) != 1) //wenn bestaetigung mittlerer Sensor**

**{**

**fertig = 1;**

**if(fertig == 1)**

**{**

**personenZahlGesamt++;**

**personenZahlRein++;**

**aktuelleAnzahlPersonen++;**

**anzahlAenderung = '+';**

**Speichern();**

**}**

**}**

**if(digitalRead(pinSensorRaus) != 1)//wenn statt dem mittleren sensor der reingehsensor betaetigt wird**

**{**

**fertig = 1;**

**}**

**delay(zeitSprung);**

**zeitMesserErhoehung();**

**}**

**}**

**//\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**delay(zeitSprung);**

**zeitMesserErhoehung();**

**}**

**void zeitMesserErhoehung() //Funktion um zeitMesser um zeitSprung Sekunden erhoehen**

**{**

**zeitMesser = zeitMesser+((float)zeitSprung/1000);//zeitMesser um zeitSprung Sekunden erhöhen**

**}**

**void Speichern() //Speicherfunktion**

**{**

**myFile = SD.open(Dateiname, FILE\_WRITE); //SD objekt zuweisen**

**myFile.print(zeitMesser, 3);**

**myFile.print(";");**

**myFile.print(maximaleAnzahlPersonen);**

**myFile.print(";");**

**myFile.print(aktuelleAnzahlPersonen);**

**myFile.print(";");**

**myFile.print(anzahlAenderung);**

**myFile.print(";");**

**myFile.print(personenZahlGesamt);**

**myFile.print(";");**

**myFile.print(personenZahlRaus);**

**myFile.print(";");**

**myFile.print(personenZahlRein);**

**myFile.print(";");**

**myFile.print(Status);**

**myFile.println(";");**

**myFile.close();**

**}**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

Vorschau. Siehe „Dateien/Programm/Personenzaehler/Personenzaehler.ino“ im digitalen Anhang für vollen Programmcode.

**VI.II. Quellen**1. BBS Verden  
https://www.bbsverden.de/  
2. Personenzähler  
https://de.wikipedia.org/wiki/Handz%C3%A4hler  
3. Datensatz  
https://de.wikipedia.org/wiki/Datensatz  
4. Sensorik  
https://de.wikipedia.org/wiki/Sensorik\_(Technik)  
5. Corona  
https://de.wikipedia.org/wiki/COVID-19  
6. Präposition  
https://de.wikipedia.org/wiki/Pr%C3%A4position  
7. Budget  
https://de.wikipedia.org/wiki/Budget  
8. Integriert  
https://de.wikipedia.org/wiki/Integration\_(Technik)  
9. delay  
https://de.wikipedia.org/wiki/Verz%C3%B6gerung\_(Telekommunikation)  
10. RTC  
https://de.wikipedia.org/wiki/Echtzeituhr  
11. CSV  
https://de.wikipedia.org/wiki/CSV\_(Dateiformat)  
12. XLSX  
https://en.wikipedia.org/wiki/Office\_Open\_XML13. I2C  
13. LCD  
https://de.wikipedia.org/wiki/Fl%C3%BCssigkristallanzeige  
14. int  
https://de.wikipedia.org/wiki/Integer\_(Datentyp)  
15. float  
https://de.wikipedia.org/wiki/Gleitkommazahl  
16. string  
https://de.wikipedia.org/wiki/Zeichenkette  
17. char  
https://de.wikipedia.org/wiki/Char\_(Datentyp)  
18. file  
https://www.arduino.cc/en/reference/SD  
19. void  
https://de.wikipedia.org/wiki/Void\_(Schl%C3%BCsselwort)  
20. Struktogramm  
https://de.wikipedia.org/wiki/Nassi-Shneiderman-Diagramm  
21. nsd  
https://help.structorizer.fisch.lu/index.php?menu=69   
22. Infrarot  
https://de.wikipedia.org/wiki/Infrarotstrahlung  
Fritzing LCD  
https://forum.fritzing.org/uploads/default/original/2X/3/396ac45a9caa834358c9439a959ef2a6203c7a2d.fzpz  
Fritzing Infrarotsensor  
https://forum.fritzing.org/uploads/default/original/2X/3/3198993fad58be49eec4acad545dfe3d66d4beac.fzpz  
Arduino Uno  
https://www.amazon.de/gp/product/B01ELNJAUC/  
LCD  
https://www.amazon.com/WayinTop-Display-Interface-Adapter-Arduino/dp/B07TXGD3WS  
SD Modul  
https://www.amazon.de/AZDelivery-Reader-Speicher-Memory-Arduino/dp/B077MCQS9P/  
Kabel  
https://www.amazon.de/gp/product/B01EV70C78/  
Sensor  
https://www.amazon.de/gp/product/B07DRCKV3X/  
Fritzing  
https://fritzing.org/download/  
Structorizer  
https://www.chip.de/downloads/Structorizer\_64884440.html  
Word Suite  
www.microsoft.com  
Arduino IDE   
https://www.arduino.cc/en/software  
Wire.h  
https://www.arduino.cc/en/Reference/Wire  
LiquidCrystal\_I2C.h  
https://www.arduino.cc/reference/en/libraries/liquidcrystal-i2c/  
SD.h  
https://www.arduino.cc/en/reference/SD  
SPI.h  
https://www.arduino.cc/en/reference/SPI

Vorschau. Siehe “Dateien/Quellen/Quellenverzeichnis.docx” für vergrößerte Ansicht.