

Anlagen

I. Projekttagbuch

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 recherchieren	-Arduino ist wichtig für	1
01.04.2021	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. Projekttagbuch	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 Programmwürfen
21	NSD	Dateiendung für Dateien vom Structorizer
22	Infrarot	Elektromagnetische Strahlung im Spektralbereich zwischen sichtbarem Licht 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
Programmierungsumgebung	Arduino IDE
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	int ₁₅	Maximal im Raum erlaubte Anzahl an Personen
aktuelleAnzahlPersonen	Int	Aktuelle Personenzahl im Raum
zeitMesser	float ₁₆	Wird als Zeitstempel genutzt
zeitSprung	int	Zeitabstand zwischen Durchläufen. In Millisekunden
raumNummer	string ₁₇	Gibt Raumnummer wieder
Status	string	Gibt aus, ob noch Platz ist oder sich jemand vor den mittleren Sensoren stellt.
anzahlAenderung	char ₁₈	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	file ₁₉	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

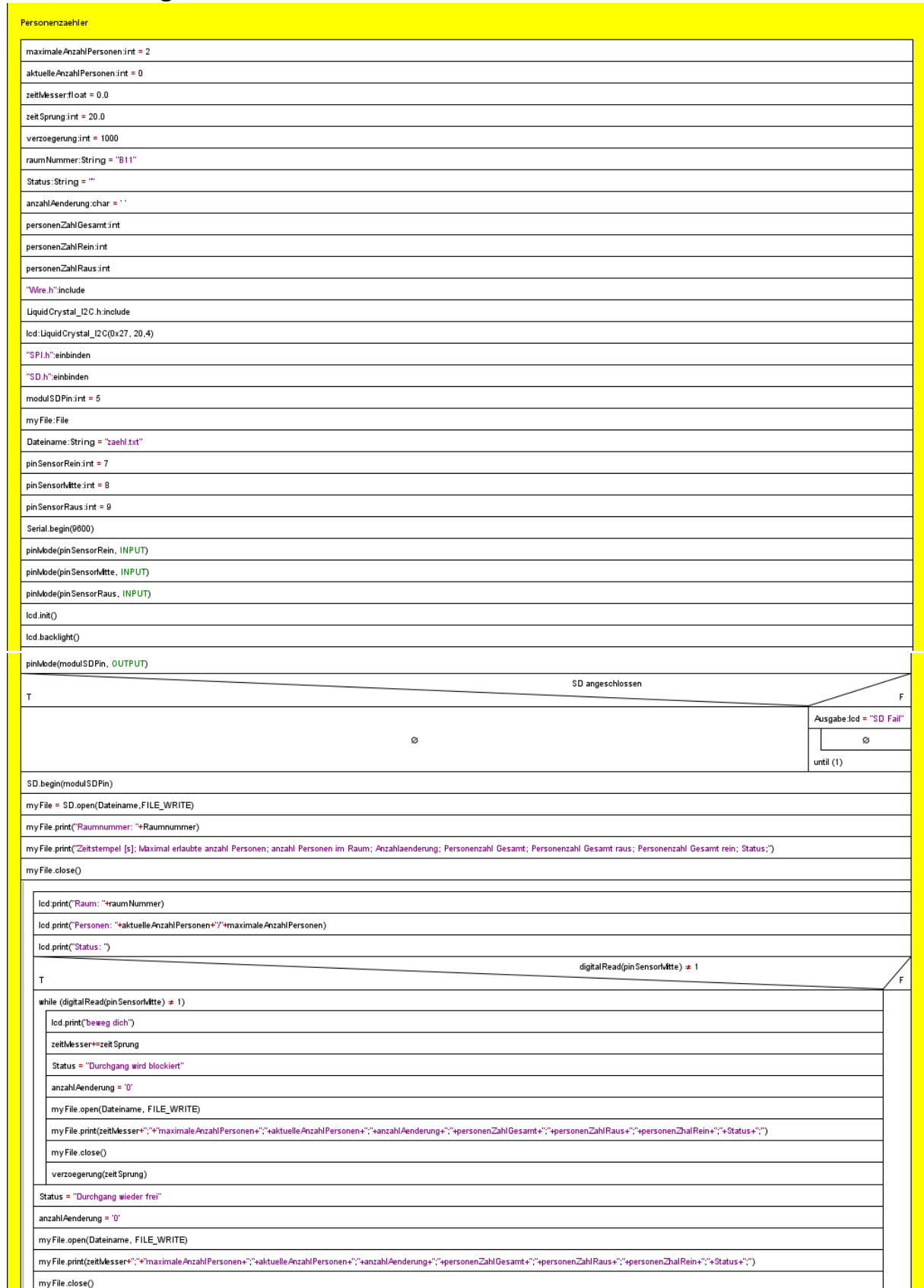
IV.III.III. Zeitstempel erhöhen

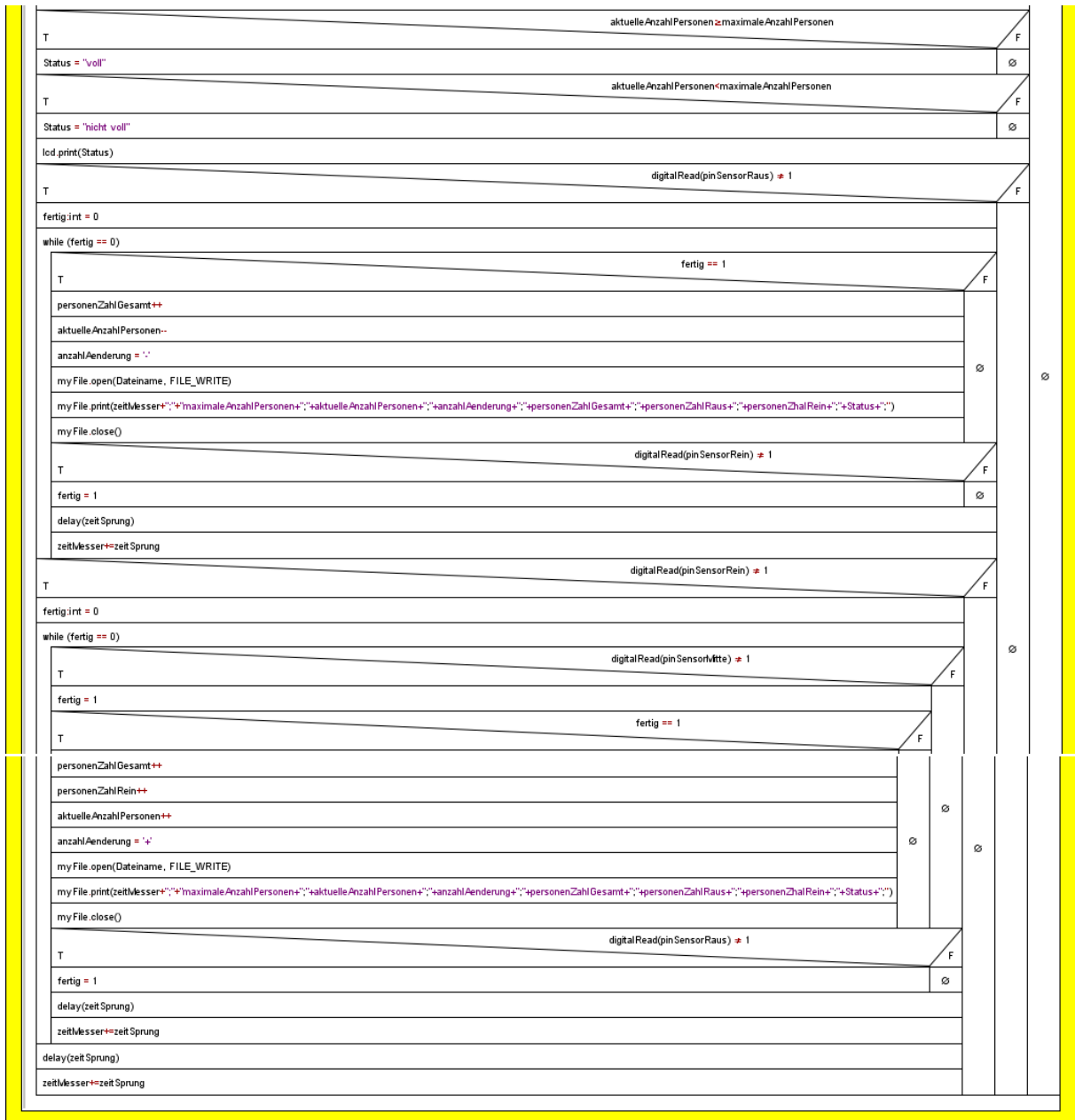
Die Funktion dient zum erhöhen des Zeitstempels um die Dauer des Delays
Für eine Erklärung der Variablenfunktion siehe Kapitel „IV.II. Variablen“.

```
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

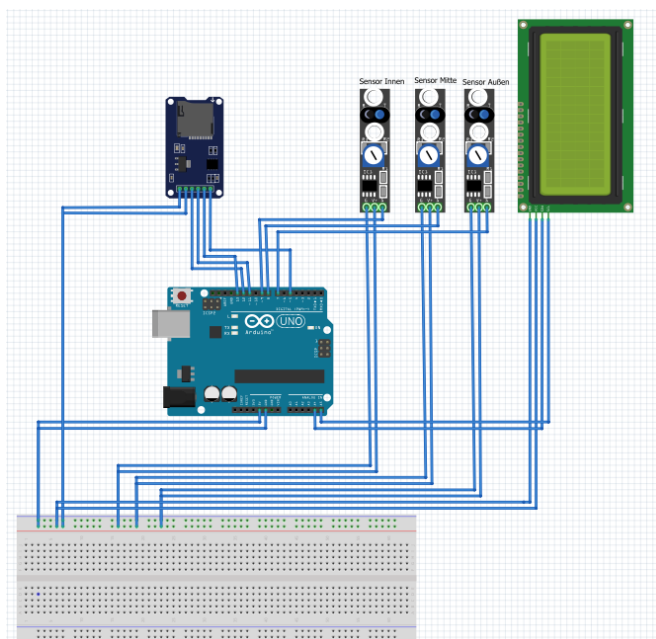
Für das Projekt wird eine Infrarot²³ Sensorik bestehend aus drei Sensoren für die Eingabe genutzt, ein SD Kartenmodul für die Speicherung der Datensätze und ein LCD-Display für die Ausgabe der aktuellen Personenzahl verwendet.

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.

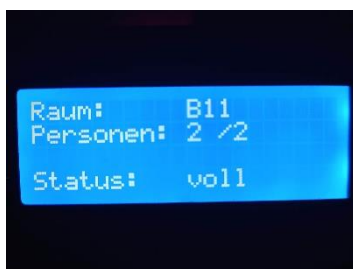
Empfohlen wird ein 9V 2A Netzteil.

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

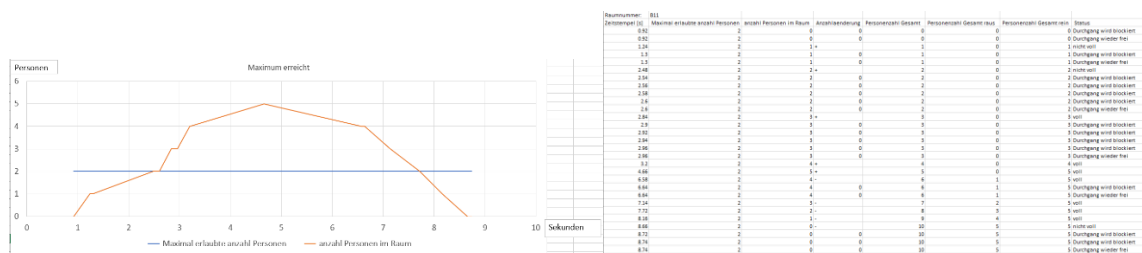
Sobald der mittlere Sensor betätigt wird, egal ob alleine oder nach Betätigung einer der äußeren Sensoren, wird ein Datensatz erstellt. Der Datensatz enthält:

- 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:



Konfigurationsteil.

```
//organisatorisches
int maximaleAnzahlPersonen = 2; //Konfigurierbar
int aktuelleAnzahlPersonen = 0;

float zeitMessser = 0.0;
int zeitSprung = 20; //in ms. Konfigurierbar.
int verzoeigerung = 1000;

String raumNummer = "B11"; //Konfigurierbar
String status = "";
char anzahlAenderung = ' ';

//Globale Messwertaenderungen
int personenZahlGesamt;
int personenZahlRein;
int personenZahlRaus;

//Einrichtung des 4 zeilen lods
#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
//
```

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

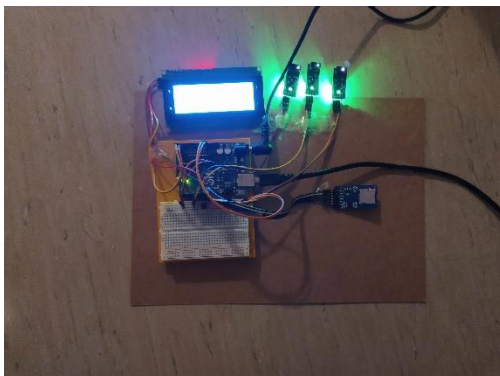
Bei den farblich markierten Feldern handelt es sich um durch den Benutzer konfigurierbare Variablen.

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 ausgegeben.

-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

Zugestellt: 10.03.2021



Arduino Starter Kit für Anfänger K040007 [Projektbuch auf Deutsch]

Verkauf durch: **ARDUINO** | Haben Sie eine Frage zum Produkt? [Frage an den Verkäufer stellen](#)

Zeitraum für Rückgabe endet am 09.04.2021

EUR 79,87

Zustand: Neu



Nochmals kaufen

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

V.II.II. Infrarotsensoren

Zugestellt: 26.04.2021

Die Sendung wurde einem Hausbewohner übergeben.



AZDelivery 3 x TRCT5000 IR Infrarot Linien Folger Hindernis Vermeidung Modul kompatibel mit Arduino

Verkauf durch: **AZ-Delivery Vertriebs GmbH** | Haben Sie eine Frage zum Produkt? [Frage an den Verkäufer stellen](#)

Zeitraum für Rückgabe endet am 26.05.2021

EUR 4,99

Zustand: Neu



Nochmals kaufen

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

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

Zugestellt: 26.04.2021

Die Sendung wurde einem Hausbewohner übergeben.



Leicke ULL-Netzteil 9V 2A 18W, 5,5 x 2,5mm Stecker, Ladegerät, Für Arduino Systeme, UNO R3 REV 3, Mega 2560 R3, Elegoo UNO R3, IEIK UNO R3 etc.

Verkauf durch: **LEICKE - Leipzig**

Zeitraum für Rückgabe endet am 26.05.2021

EUR 11,49

Zustand: Neu



Nochmals kaufen



Intenso Micro SDHC 32GB Class 10 Speicherkarte inkl. SD-Adapter

Verkauf durch: **Amazon EU S.a.r.L.**

Zeitraum für Rückgabe endet am 26.05.2021

EUR 4,79

Zustand: Neu



Nochmals kaufen

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

Zugestellt: 03.04.2021

Die Sendung wurde einem Hausbewohner übergeben.



AZDelivery 3 x Set SPI Reader Micro Speicher SD TF Karte Memory Card Shield

Modul kompatibel mit Arduino inklusive E-Book!

Verkauf durch: [AZ-Delivery Vertriebs GmbH](#) | Haben Sie eine Frage zum Produkt? [Frage an den Verkäufer stellen](#)

Zeitraum für Rückgabe endet am 03.05.2021

EUR 5,29

Zustand: Neu



Nochmals kaufen

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

V.II.V. LCD-Display



WayinTop 20x4 2004 LCD Display Zeichen Bildschirm mit TWI IIC I2C LCD

Schnittstelle Adapter für Arduino für Mega 2560 (Blau/2004)

Verkauf durch: [WayinTop](#)

Zeitraum für Rückgabe endet am 03.05.2021

EUR 9,99

Zustand: Neu



Nochmals kaufen

VorschauBild. Siehe „Dateien/Rechnungen/LCD und Taster/rechnungAmazonLCD.png“ für vergrößerte Ansicht.

V.II.VI. Jumperkabel

Zugestellt: 22.05.2021

Die Sendung wurde im Briefkasten hinterlegt



Elegoo Jumper Wire 40x 20cm Female-Female, Male-Female, Male-Male Kabel

Steckbrücken 28AWG Drahtbrücken für Arduino Raspberry Pi (3er Set)

Verkauf durch: [ELEGOO Official - DE](#)

Rücksendung bis zum 21.06.2021 möglich.

EUR 7,99

Zustand: Neu

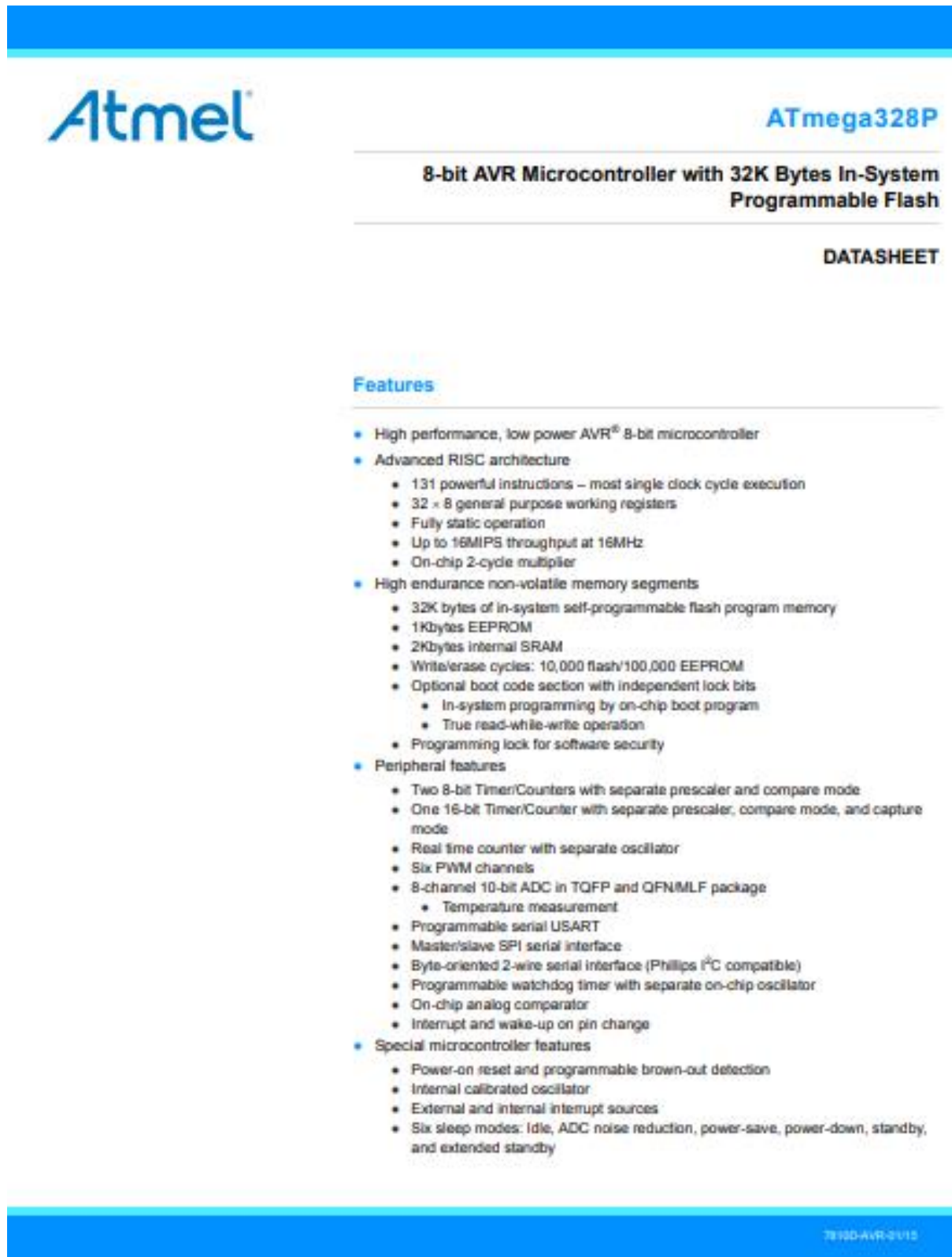


Nochmals kaufen

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

V.III. Datenblätter

V.III.I. Arduino Uno Rev 3



Vorschau bild. Siehe „Dateien/Datenblätter/Arduino/ATmega328P_Datasheet.pdf“ für 294 Seiten langes Datenblatt.

VI.III.II. Infrarotsensor

Az-Delivery
Ihr Experte für Mikroelektronik

Tracker Sensor TCRT5000 Linienfolger Modul Datenblatt





**Tracker Sensor
TCRT5000 Linienfolger Modul
Datenblatt**

Contents:

- 1. Features**
- 2. Description**
- 3. Applications**
- 4. Product Summary**
- 5. Absolute Maximum Ratings**
- 6. Basic Characteristics**
- 7. Dimensions**
- 8. Tube Specification Figures**

1. Features

- Package type: leaded
- Detector type: phototransistor
- Dimensions (L x W x H in mm): 10.2 x 5.8 x 7
- Peak operating distance: 2.5 mm
- Operating range within $\pm 20\%$ relative collector current: 0.2 mm to 15 mm
- Typical output current under test: $I_C = 1\text{ mA}$
- Daylight blocking filter
- Emitter wavelength: 950 nm
- Lead (Pb)-free soldering released
- Compliant to RoHS directive 2002/95/EC and in accordance to WEEE 2002/96/EC



2. Description

The TCRT5000 and TCRT5000L are reflective sensors which include an infrared emitter and phototransistor in a leaded package which blocks visible light.

3. Applications

- Position sensor for shaft encoder
- Detection of reflective material such as paper, IBM cards, magnetic tapes etc.
- Limit switch for mechanical motions in VCR
- General purpose - wherever the space is limited

4. Product Summary

PART NUMBER	DISTANCE FOR MAXIMUM CTR _{rel} ⁽¹⁾ (mm)	DISTANCE RANGE FOR RELATIVE I _{out} > 20 % (mm)	TYPICAL OUTPUT CURRENT UNDER TEST ⁽²⁾ (mA)	DAYLIGHT BLOCKING FILTER INTEGRATED
TCRT5000	2.5	0.2 to 15	1	Yes
TCRT5000L	2.5	0.2 to 15	1	Yes

Notes

(1) CTR: current transfere ratio, I_{out}/I_{in}

(2) Conditions like in table basic charactristics/sensors

5. Absolute Maximum Ratings

Absoulte Maximum Ratings ⁽¹⁾

PARAMETER	TEST CONDITION	SYMBOL	VALUE	UNIT
INPUT (EMITTER)				
Reverse voltage		V _{in}	5	V
Forward current		I _f	80	mA
Forward surge current	I _f ≤ 10 μs	I _{fsm}	3	A
Power dissipation	T _{amb} ≤ 25 °C	P _v	100	mW
Junction temperature		T _j	100	°C
Collector emitter voltage		V _{ceo}	70	V
Emitter collector voltage		V _{eco}	5	V
Collector current		I _c	100	mA
Power dissipation	T _{amb} ≤ 55 °C	P _v	100	mW
Junction temperature		T _j	100	°C
SENSOR				
Total power dissipation	T _{amb} ≤ 25 °C	P _{tot}	200	mW
Ambient temperature range		T _{amb}	- 25 to + 85	°C
Storage temperature range		T _{stg}	- 25 to + 100	°C
Soldering temperature	2 mm from case, t ≤ 10 s	T _{sd}	260	°C

Note

(1) T_{amb} = 25 °C, unless otherwise specified

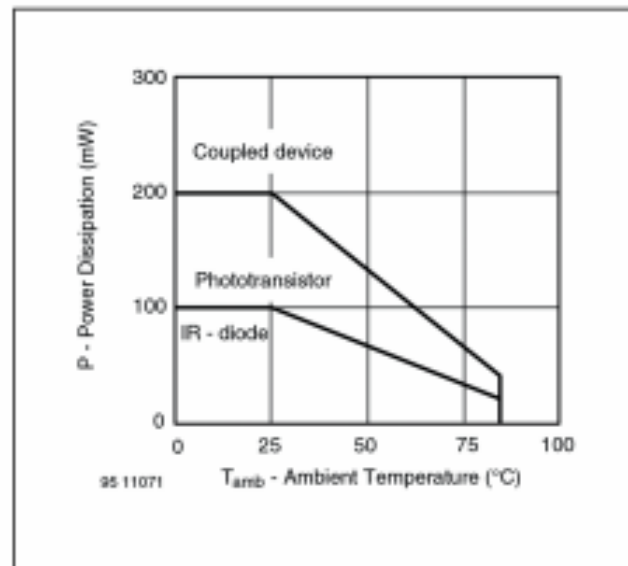


Figure 1 - Power Dissipation Limit vs. Ambient Temperature

6. Basic Characteristics

Basic Characteristics ⁽¹⁾

PARAMETER	TEST CONDITION	SYMBOL	MIN.	TYP.	MAX.	UNIT
INPUT (EMITTER)						
Forward voltage	$I_F = 60 \text{ mA}$	V_F		1.25	1.5	V
Junction capacitance	$V_D = 0 \text{ V}, f = 1 \text{ MHz}$	C_j		17		pF
Radiant intensity	$I_F = 60 \text{ mA}, t_p = 20 \text{ ms}$	I_a			21	mW/ster
Peak wavelength	$I_F = 100 \text{ mA}$	λ_p	940			nm
Virtual source diameter	Method: 63 % encircled energy	d		2.1		mm
OUTPUT (DETECTOR)						
Collector emitter voltage	$I_C = 1 \text{ mA}$	V_{CE0}	70			V
Emitter collector voltage	$I_E = 100 \text{ }\mu\text{A}$	V_{EC0}	7			V
Collector dark current	$V_{CE} = 20 \text{ V}, I_F = 0 \text{ A}, E = 0 \text{ lx}$	I_{CE0}		10	200	nA
SENSOR						
Collector current	$V_{CE} = 5 \text{ V}, I_F = 10 \text{ mA}, D = 12 \text{ mm}$	$I_C (2) (3)$	0.5	1	2.1	mA
Collector emitter saturation voltage	$I_F = 10 \text{ mA}, I_C = 0.1 \text{ mA}, D = 12 \text{ mm}$	$V_{CEsat} (2) (3)$			0.4	V

Note

- (1) $T_{amb} = 25^\circ\text{C}$, unless otherwise specified
- (2) See figure 3
- (3) Test surface: mirror (Mfr. Spindler a. Hoyer, Part No. 340005)

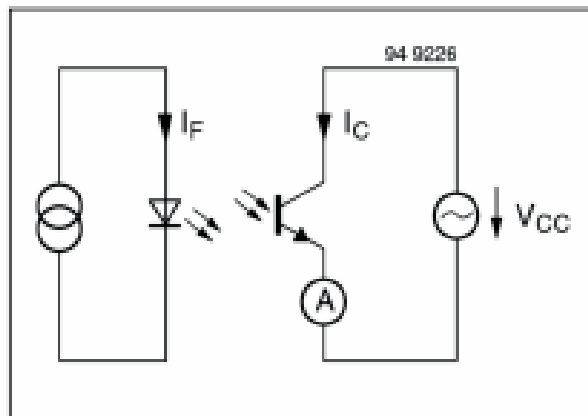


Figure 2 - Test Circuit

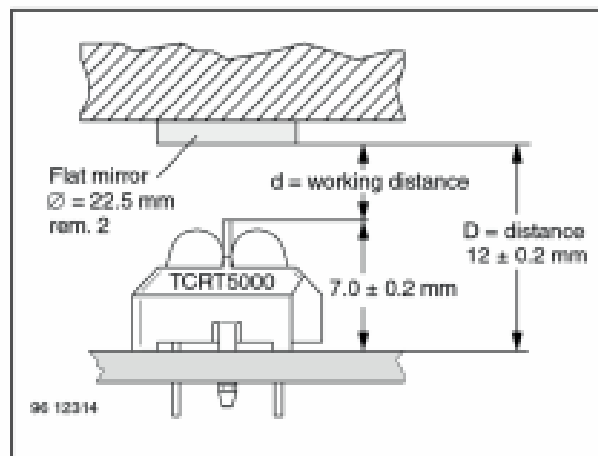


Figure 3 - Test Circuit



Tracker Sensor TCRT5000 Linienfolger Modul Datenblatt

Basic Characteristics

$T_{amb} = 25^{\circ}\text{C}$, unless otherwise specified

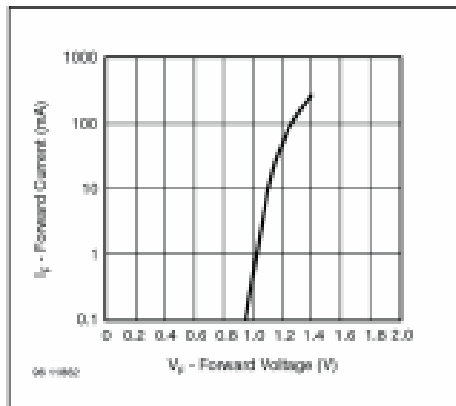


Figure 4 - Forward Current vs. Forward Voltage

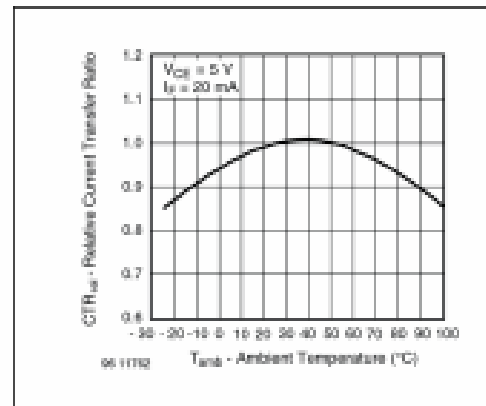


Figure 5 - Relative Current Transfer Ratio vs. Ambient Temperature

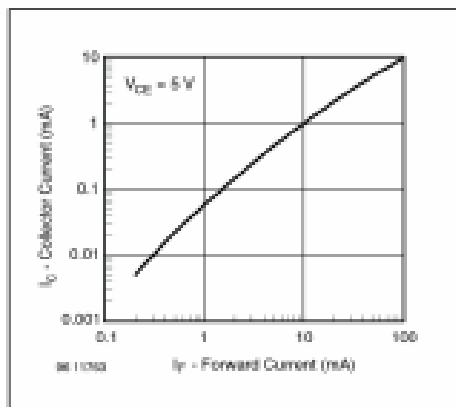


Figure 6 - Collector Current vs. Forward Current

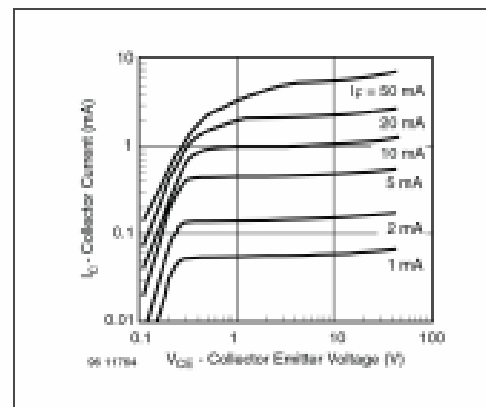


Figure 7 - Collector Emitter Saturation Voltage vs. Collector Current



Tracker Sensor TCRT5000 Linienfolger Modul Datenblatt

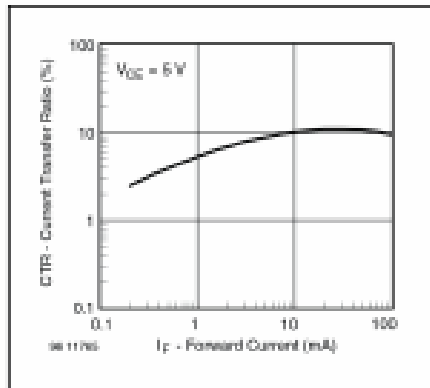


Figure 8 - Current Transfer Ratio vs.
Forward Current

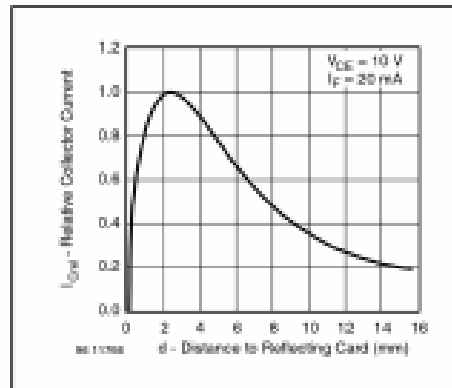
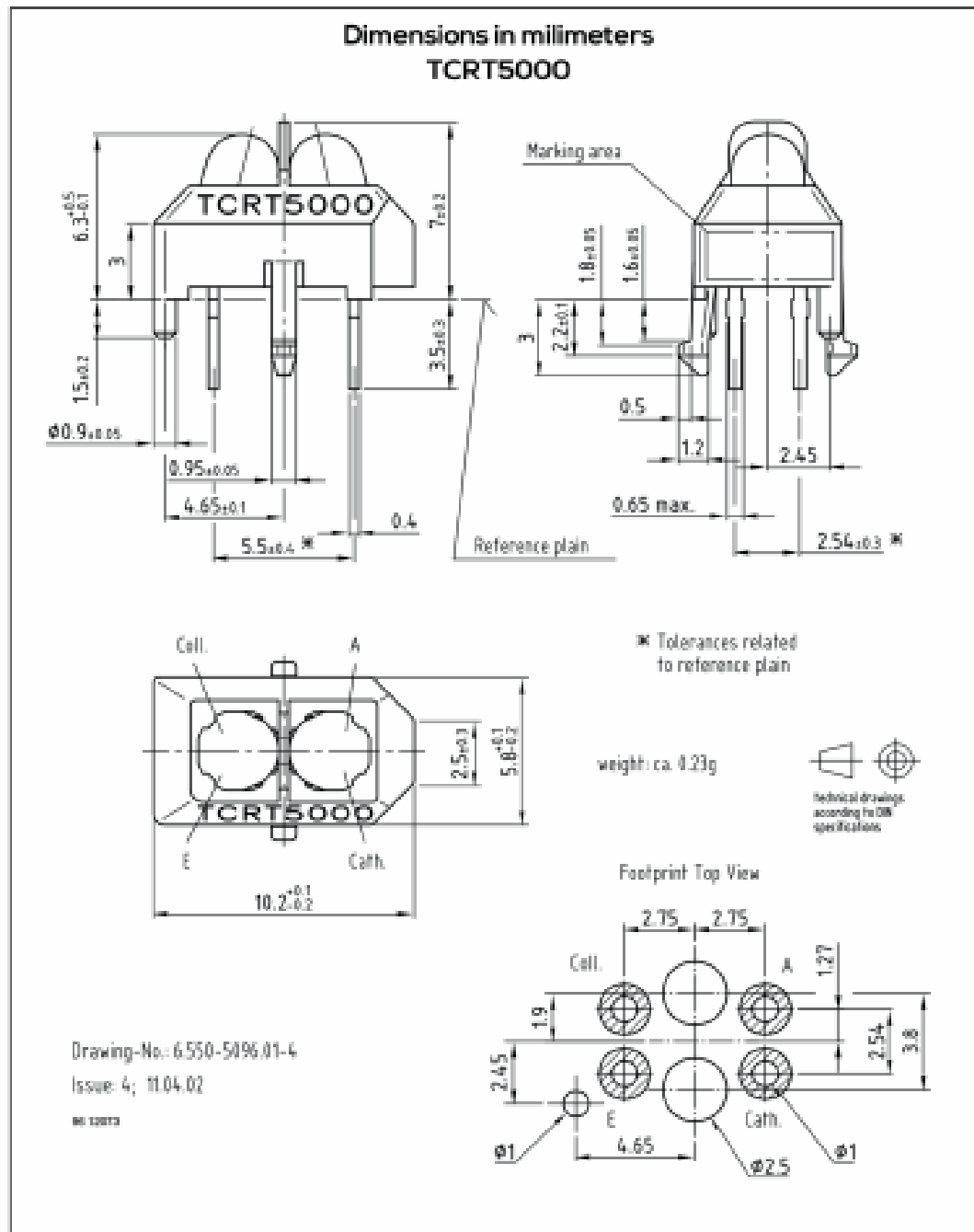
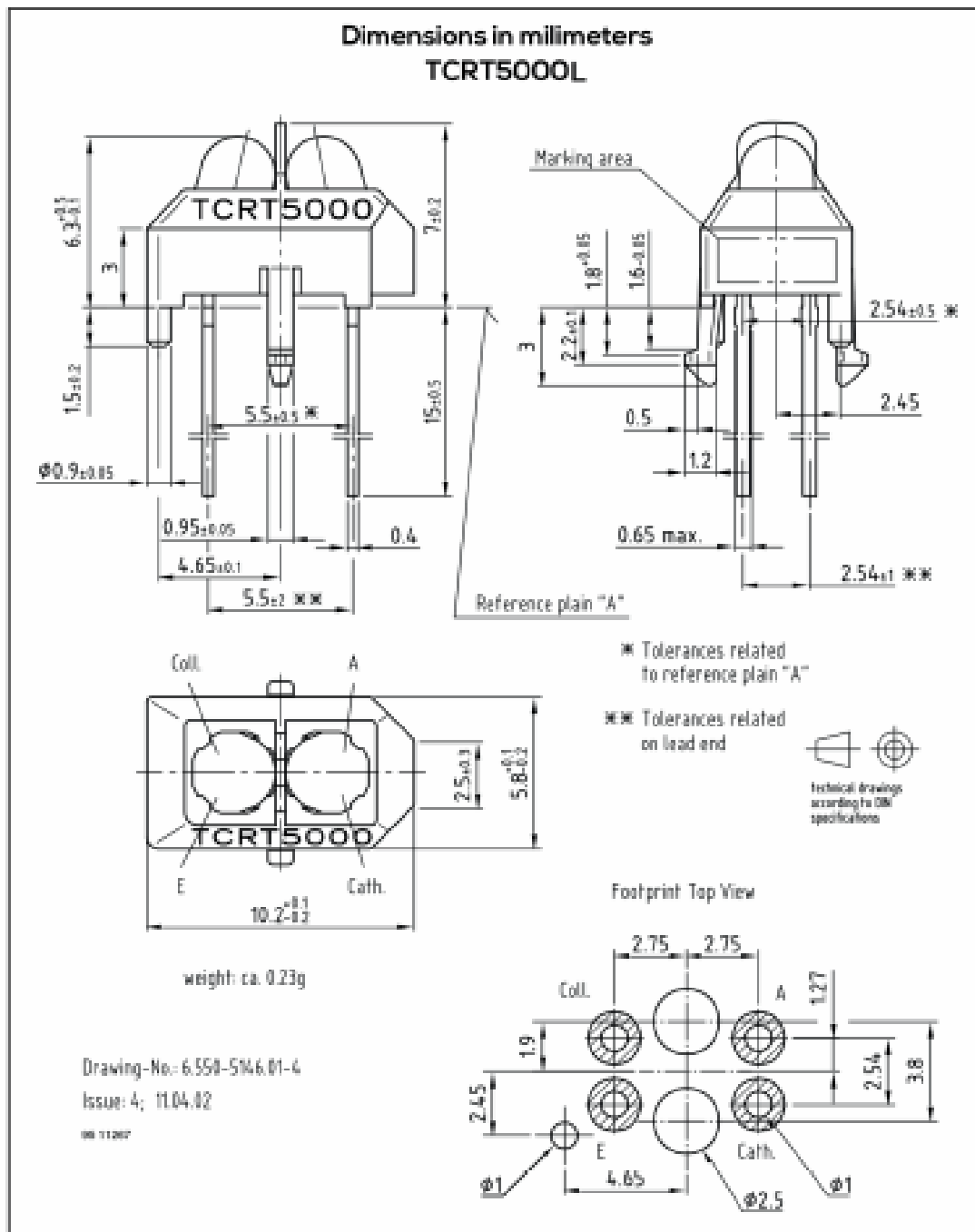


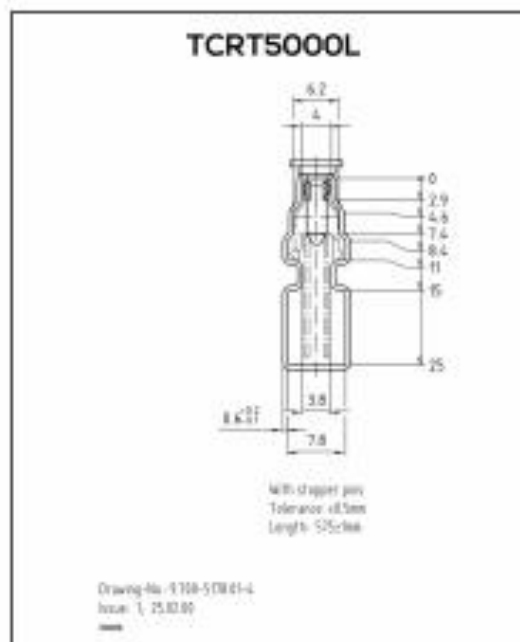
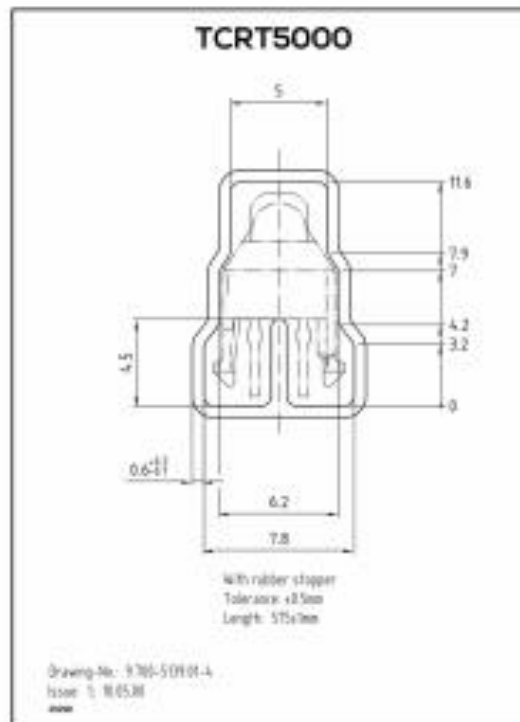
Figure 9 - Relative Collector Current vs.
Distance

7. Dimensions

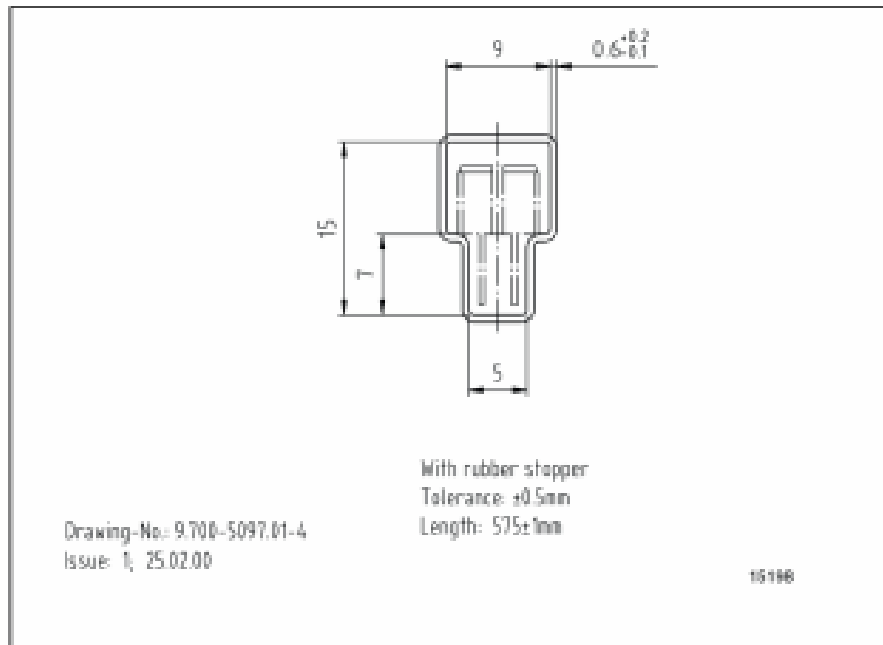




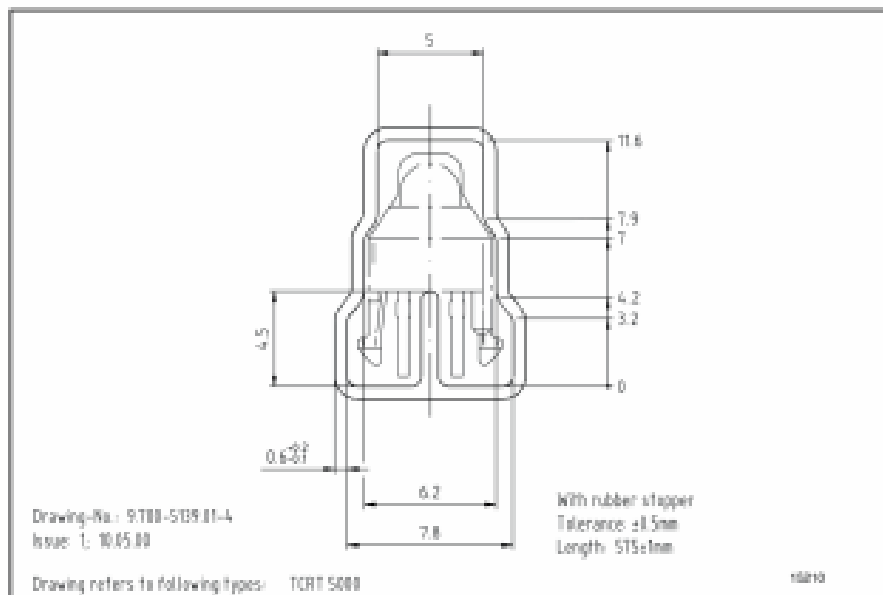
Tube dimensions in millimeters



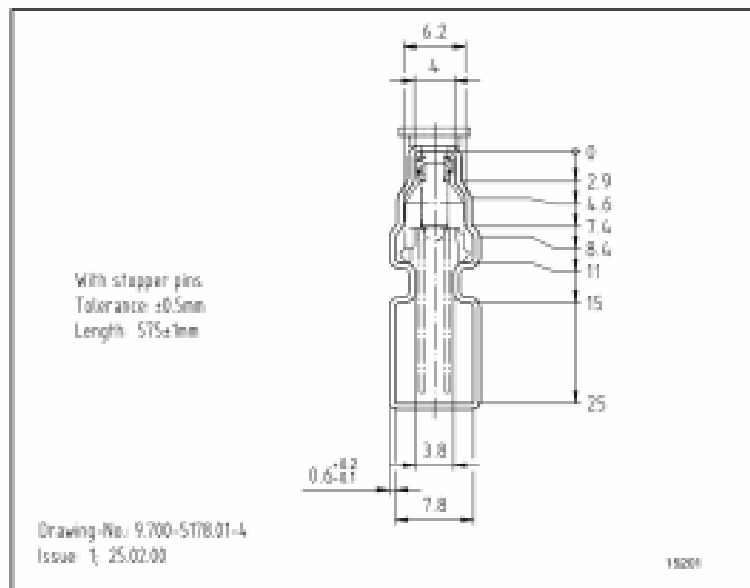
8. Tube Specification Figures



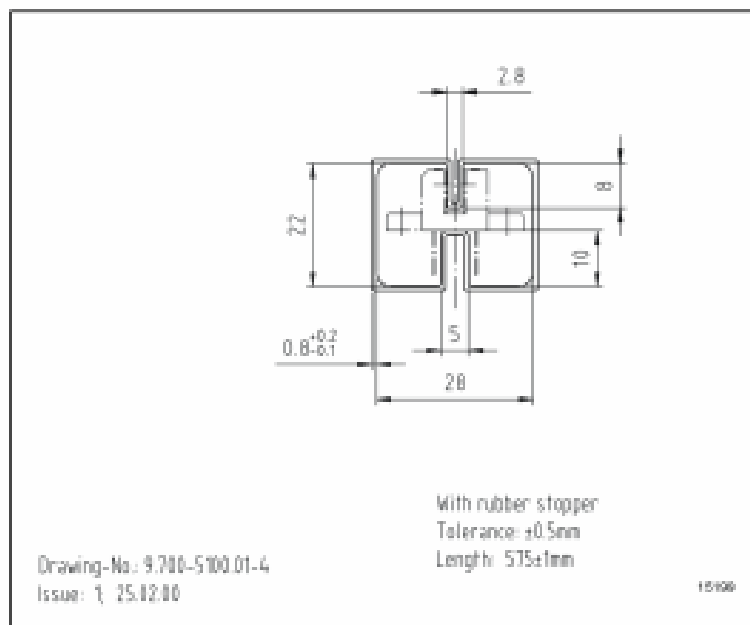
Tube Specification Figure 1



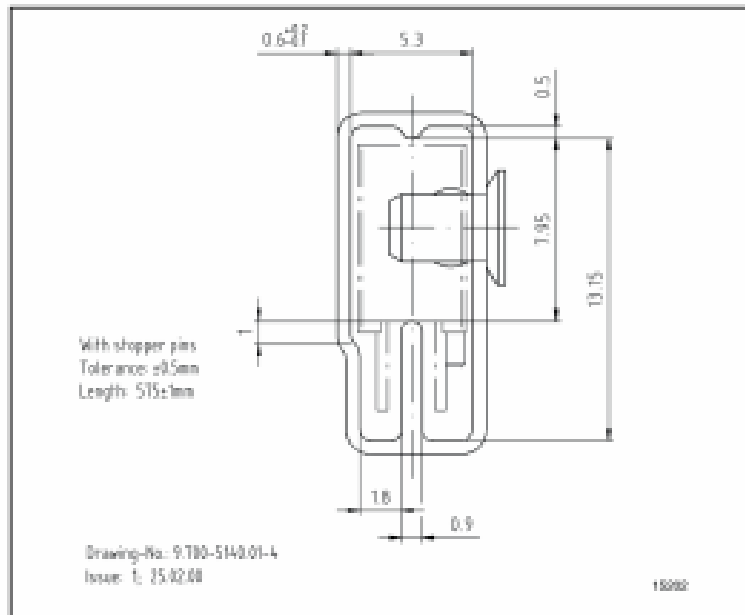
Tube Specification Figure 2



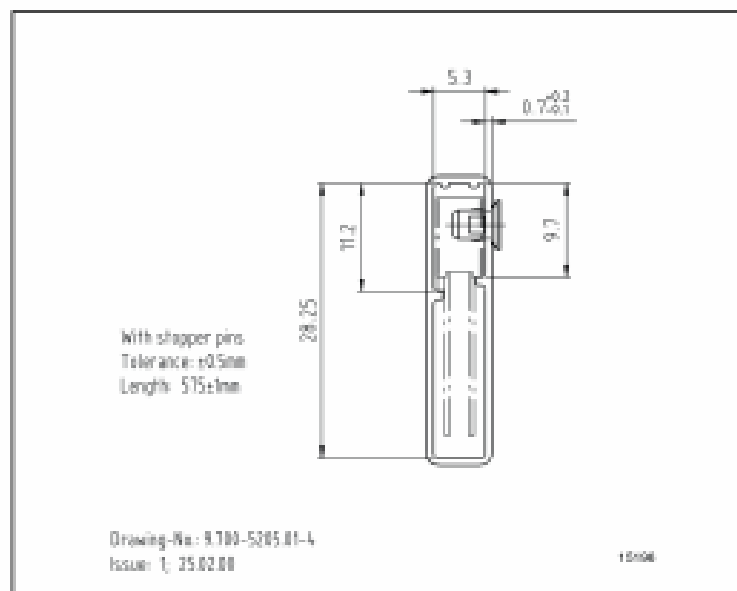
Tube Specification Figure 3



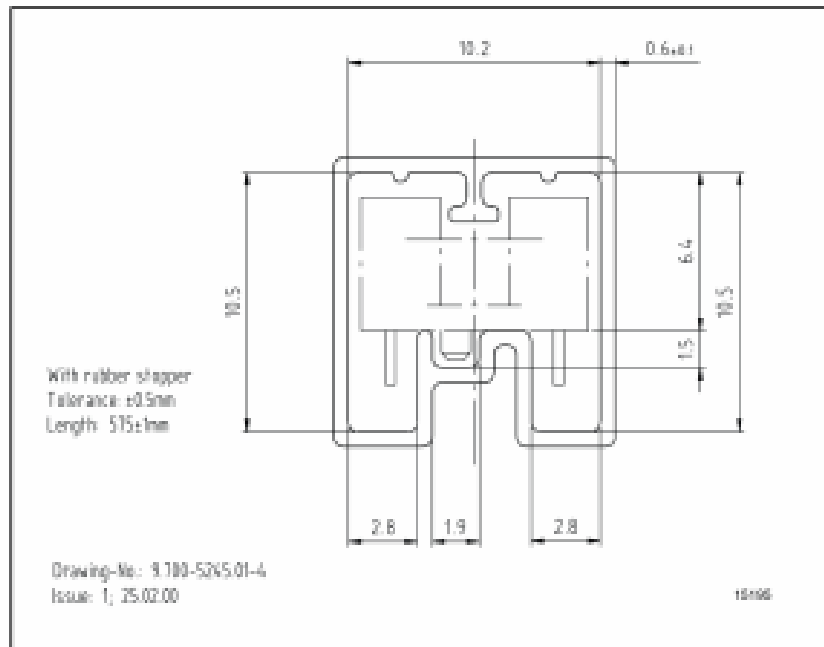
Tube Specification Figure 4



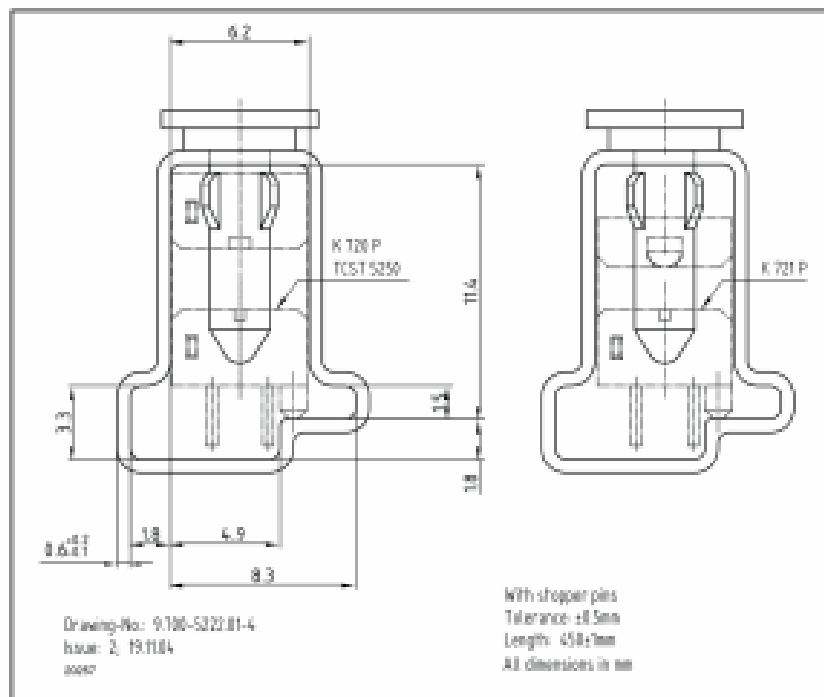
Tube Specification Figure 5



Tube Specification Figure 6



Tube Specification Figure 7



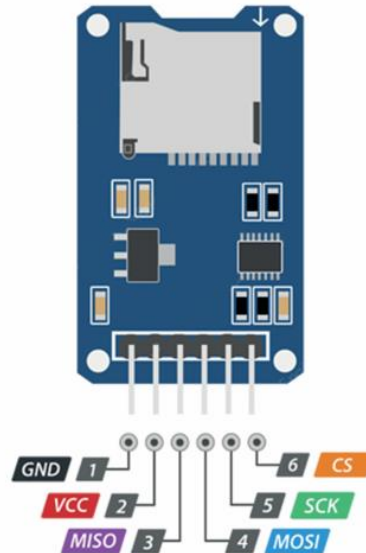
Tube Specification Figure 8

VI.III.III. Micro SD Shield

Pinout



SPI Reader Micro Speicherkartenmodul
Pinout

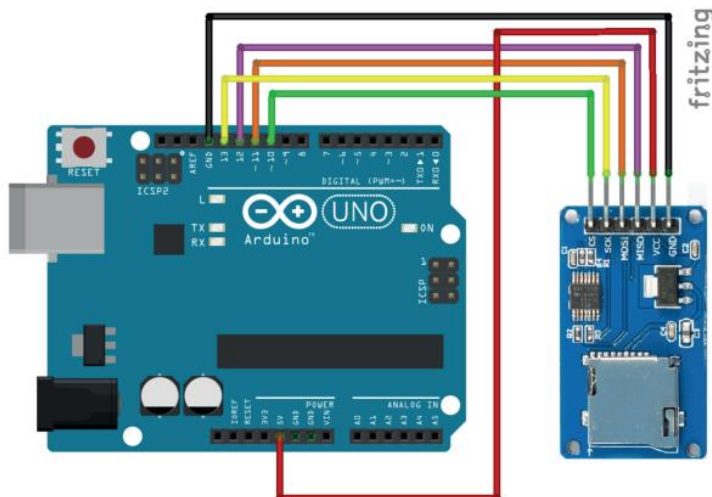


Vorschau. Siehe „Dateien/Datenblätter/SD Modul//SPI Reader Micro Speicherkartenmodul Pinout.pdf“ für Datenblatt.

Arduino Schematics



SPI Reader Micro Speicherkartenmodul
Arduino Schematics



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

Datenblatt

Az-Delivery
Ihr Experte für Mikroelektronik!

SPI Reader Micro Speicherkartenmodul Datenblatt





SPI Reader
Micro Speicherkartenmodul
Datenblatt

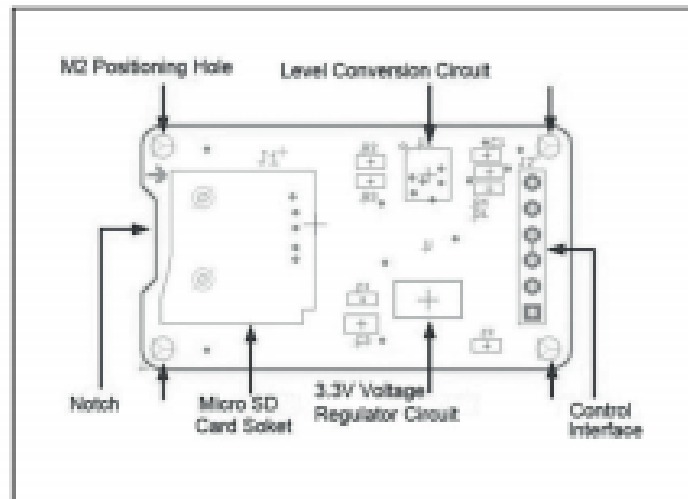
Contents:

- 1. Description**
- 2. Interface Parameters**
- 3. Micro SD Card Interface Module**

2. Interface Parameters

Items	Min	Typical	Max	Unit
Power Voltage VCC	4.5	5	5.5	V
Current	0.2	50	200	mA
Interface Electrical Potential	3.3 or 5			V
Support Card Type	Micro SD Card(<=2G), <u>Micro</u> SDHC Card(<=32G)			—
Size	42X24X12			mm
Weight	5			g

3. Micro SD Card Interface Module

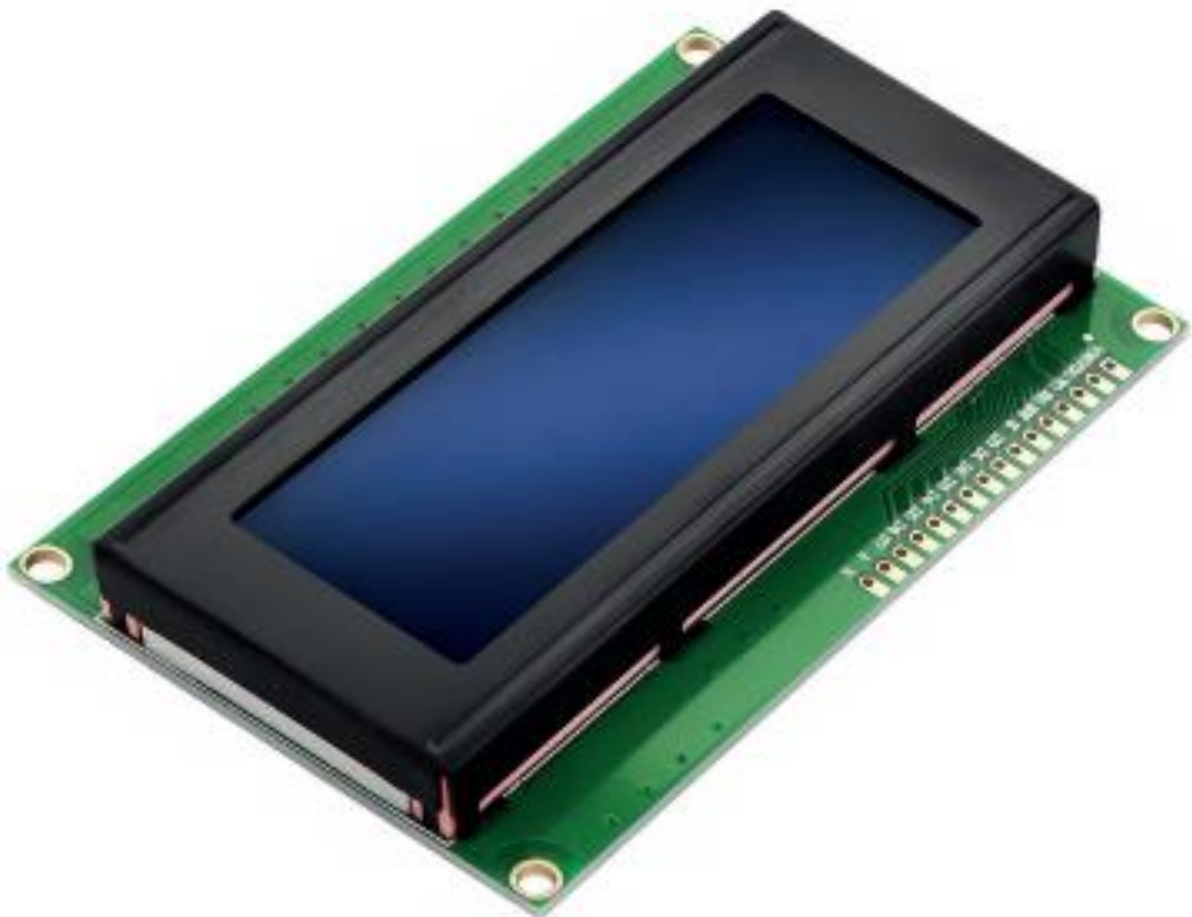


VI.III.IV. LCD-Display

Datenblatt



HD44780 20x04 Blaues Display Datenblatt



Content:

- [1. Basic Specifications](#)
- [2. Absolute Maximum Ratings](#)
- [3. Electrical Characteristics](#)
- [4. Pinout](#)
- [5. Connection Diagrams](#)
- [6. Precautions](#)

1. Basic Specifications

- Module dimensions: 98 mm x 60 mm x 14 mm
- Resolution: 20 Characters x 4 Lines
- LCD type: STN, Positive, Transflective, Blue
- Backlight: Blue
- View Angle: 180 degrees
- Modes: Parallel (8-bit and 4-bit)
- Operating Voltage: 3.3V and 5V
- Operating temperature: from -20°C to +70°C
- Storage temperature: from -30°C to +80°C

2. Absolute Maximum Ratings

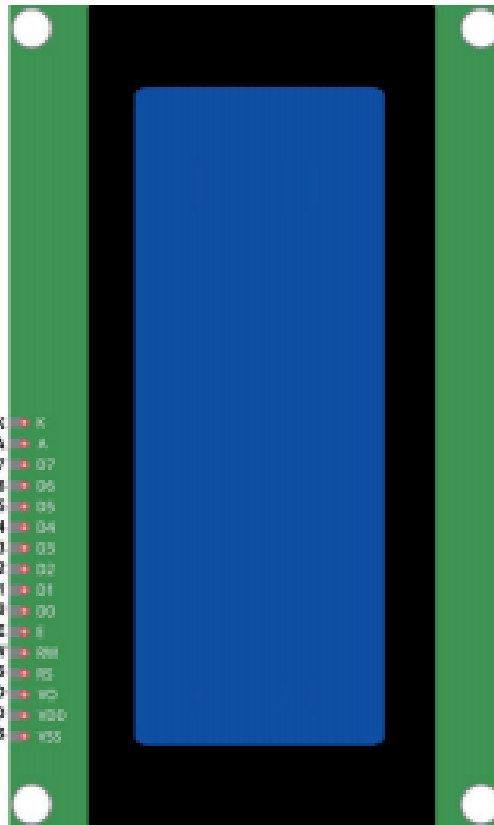
Item	Symbol	Min.	Typ.	Max.	Unit
Supply Voltage for Logic	VDD-VSS	-0.3	-	+7	V
Power Supply for LCD	VLCD	VDD-15	-	VDD+0.3	V
Input Voltage	VIN	-0.3	-	VDD+0.3	V
Supply Current for Backlight	I _{LED}	-	-	75	mA

3. Electrical Characteristics

Item	Symbol	Condition	Min.	Typ.	Max.	Unit
Power Supply for LCM	VDD-VSS	VDD=5V	4.8	5.0	5.2	V
		VDD=3.3V	3.0	3.3	3.6	V
Input Voltage	V _{IL}	L Level	-0.2	-	1	V
	V _{IH}	H Level	VDD-1.0	-	VDD	V
LCD Driving Voltage	VDD-V0	-	4.5	4.8	5.1	V
Supply Current for LCM	I _{DD}	-	-	-	2000.0	μA
Supply Current for Backlight	I _{LED}	-	-	45	-	mA

4. Pinout

Backlight cathode (-) - K	→ K
Backlight anode (+) - A	→ A
Data Input/Output Pin 7 - D7	→ D7
Data Input/Output Pin 6 - D6	→ D6
Data Input/Output Pin 5 - D5	→ D5
Data Input/Output Pin 4 - D4	→ D4
Data Input/Output Pin 3 - D3	→ D3
Data Input/Output Pin 2 - D2	→ D2
Data Input/Output Pin 1 - D1	→ D1
Data Input/Output Pin 0 - D0	→ D0
Enable - E	→ E
Read/Write - RW	→ RW
Register Select - RS	→ RS
LCD Contrast Pin - VO	→ VO
Power Supply - VDD	→ VDD
Ground - VSS	→ VSS



The VO pin uses analog voltage to set up the screen contrast level. Hardware contrast adjustment can be done by using voltage divider or potentiometer.

The anode pin A and cathode pin K are positive and negative terminals that are used to control the backlight of the screen.

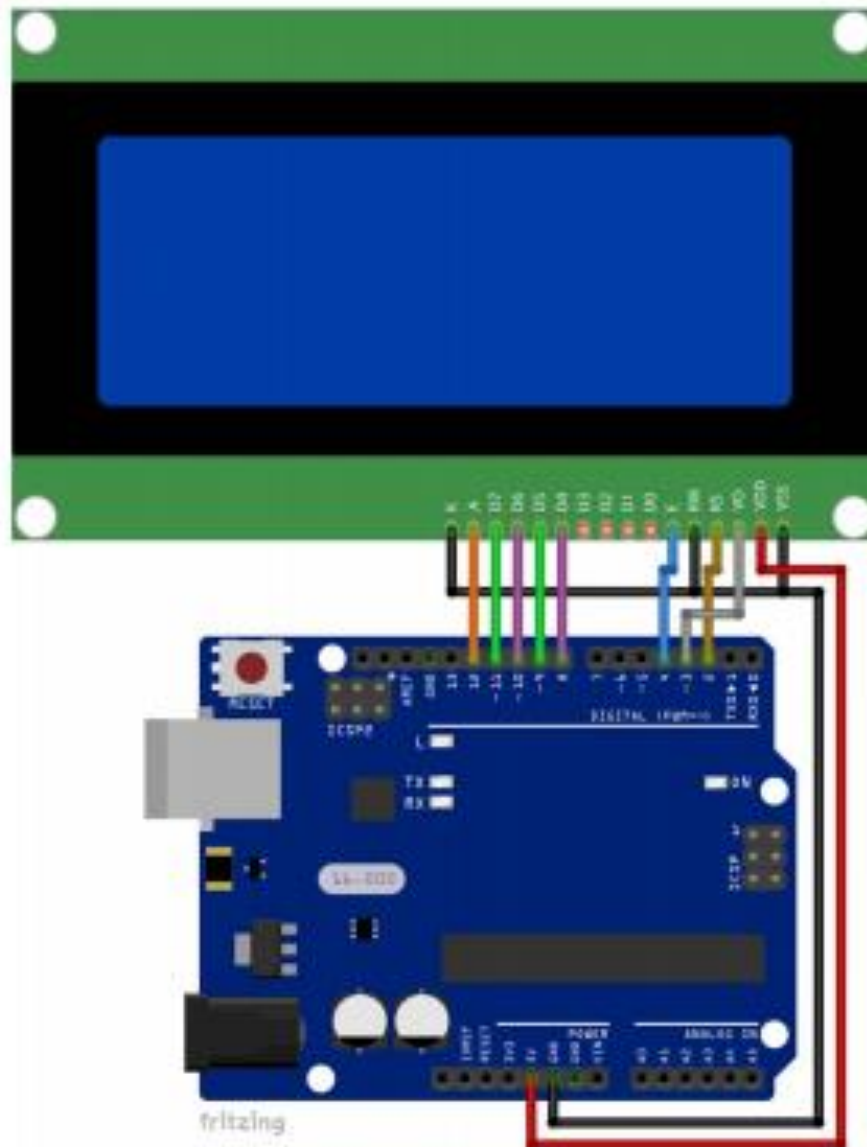
The RW pin, or Read/Write pin, is used to set the data direction, to read data from display-driver chip or to write data into the screen driver chip.

The RS pin is Register Select pin which is used to shift between command or data registers of the driver chip.

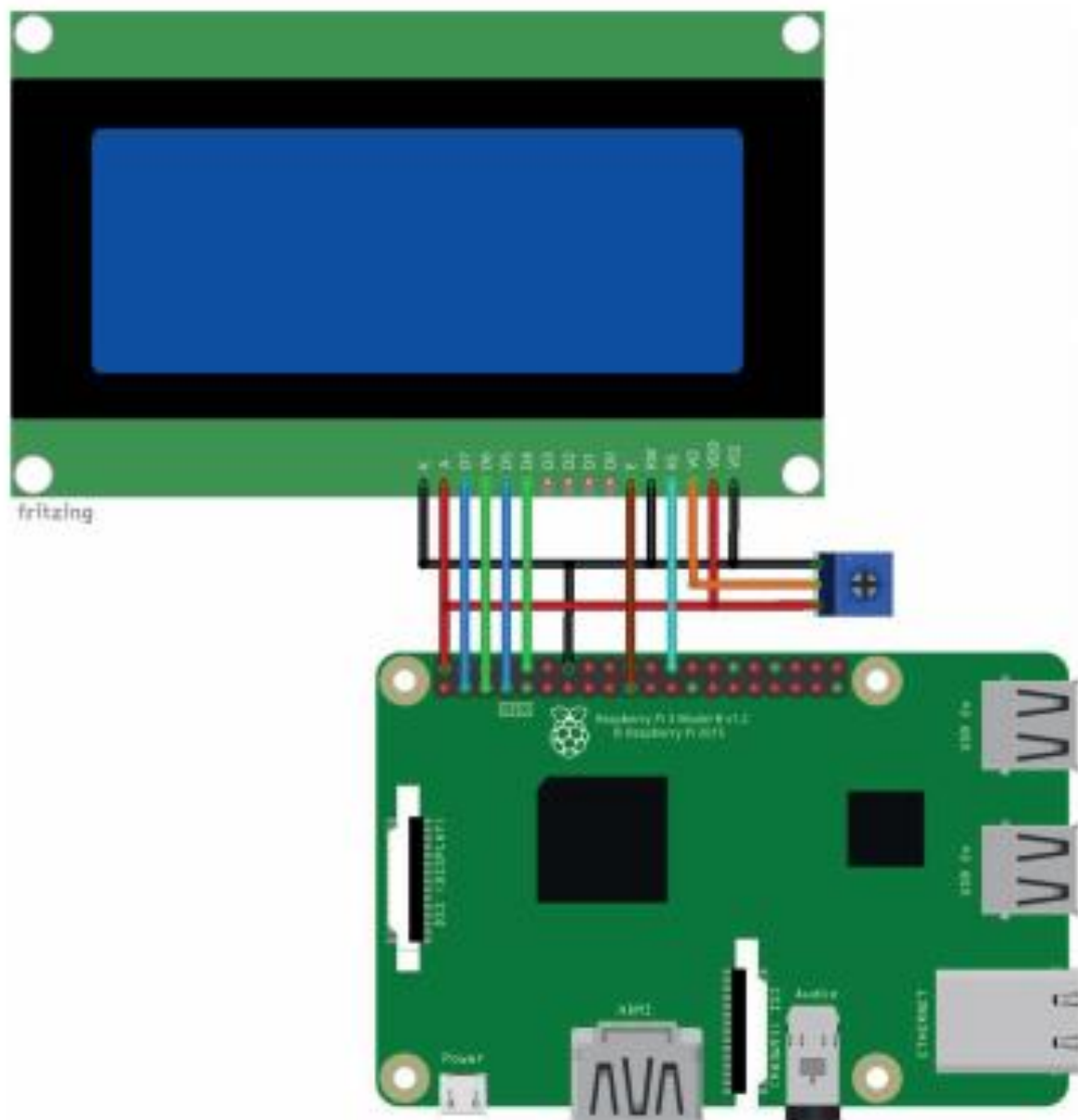
The E pin, or Enable pin, is used to enable/disable communication between the main micro-controller and the driver chip of the screen.

5. Connection Diagrams

Connect the HD44780 20x04 LCD Display with the Microcontroller Compatible with Arduino as shown on the following connection diagram:



Connect the HD44780 20x04 LCD Display with the Raspberry Pi as shown on the following connection diagram:



6. Precautions

- This device is susceptible to Electro-Static Discharge (ESD) damage. Observe Anti-Static precautions.
 - Do not apply excessive force to display surface or the adjoining areas since this may cause the color tone to vary.
 - The polarizer covering display surface of the LCD module is soft and easily scratched. Handle this polarizer carefully.
 - If display surface becomes contaminated, breathe on the surface and gently wipe it with a soft dry cloth. If it is heavily contaminated, moisten cloth with Isopropyl or alcohol.
 - Solvents other than those above-mentioned may damage the polarizer. Especially, do not use the Water.
 - Exercise care to minimize corrosion of the electrode. Corrosion of the electrodes is accelerated by water droplets, moisture condensation or a current flow in a high-humidity environment.
 - Install LCD Module by using the mounting holes. When mounting the LCD module make sure it is free of twisting, warping and distortion. In particular, do not forcibly pull or bend the cable or the backlight cable.
 - Do not attempt to disassemble or process LCD module.
 - NC terminal should be open. Do not connect anything.
 - If the logic circuit power is off, do not apply the input signals.
 - To prevent destruction of the elements by static electricity, be careful to maintain an optimum work environment.
 - Be sure to ground the body when handling LCD modules.
 - Tools required for assembling, such as soldering irons, must be properly grounded.
- To reduce the amount of static electricity generated, do not conduct assembling and other work under dry conditions.
- The LCD module is coated with a film to protect the display surface. Exercise care when peeling off this protective film since static electricity may be generated.

- Identify and, at all times, observe absolute maximum ratings for both logic and LC drivers.
- Prevent the application of reverse polarity to VDD and VSS, however briefly.
- Use a clean power source free from transients. Power-up conditions are occasionally jolting and may exceed the maximum ratings of these LCD modules.
- The VDD power of LCD module should also supply the power to all devices that may access the display. Do not allow the data bus to be driven when the logic supply to the module is turned off.
- Operate this LCD module within the limits of the modules temperature specifications.
- Surface of the LCD panel should not be touched or scratched. The display front surface is an easily scratched, plastic polarizer.
- Always employ anti-static procedure while handling LCD module.
- Do not store in direct sunlight.
- If leakage of the liquid crystal material should occur, avoid contact with this material, particularly ingestion. If the body or clothing becomes contaminated by the liquid crystal material, wash thoroughly with water and soap.
- When storing the LCD modules, avoid exposure to direct sunlight or to the light of fluorescent lamps.
- Keep LCD modules in bags (avoid high temperature / high humidity and low temperatures below 0°C.
- Liquid crystals solidify under low temperature (below the storage temperature range) leading to defective orientation or the generation of air bubbles (black or white). Air bubbles may also be generated if the module is subject to a low temperature.
- To minimize the performance degradation of the LCD modules resulting from destruction caused by static electricity etc., exercise care to avoid holding the following sections when handling the modules.
 - Exposed area of the printed circuit board.
 - Terminal electrode sections.
- Please keep the temperature within specified range for use and storage. Polarization degradation, bubble generation or polarizer peel-off may occur with high temperature and high humidity.
- Do not touch, push or rub the exposed polarizers with anything harder than an HB pencil lead (glass, tweezers, etc.).

Az-Delivery

Ihr Experte für Mikroelektronik!

If you are looking for the high quality products for Arduino and Raspberry Pi, AZ-Delivery Vertriebs GmbH is the right company to get them from. You will be provided with numerous application examples, full installation guides, eBooks, libraries and assistance from our technical experts.

Page 9

Vorschau. Siehe „Dateien/Datenblätter/lcd/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 verzoeigerung = 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 ausgeloeset 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 ausgeloeset 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 erhoeihen
}

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_\(Technik\)](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_\(Technik\)](https://de.wikipedia.org/wiki/Integration_(Technik))
 9. delay
[https://de.wikipedia.org/wiki/Verz%C3%B6gerung_\(Telekommunikation\)](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_\(Dateiformat\)](https://de.wikipedia.org/wiki/CSV_(Dateiformat))
 12. XLSX
https://en.wikipedia.org/wiki/Office_Open_XML
 13. I2C
 13. LCD
<https://de.wikipedia.org/wiki/FI%C3%BCssigkristallanzeige>
 14. int
[https://de.wikipedia.org/wiki/Integer_\(Datentyp\)](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_\(Datentyp\)](https://de.wikipedia.org/wiki/Char_(Datentyp))
 18. file
<https://www.arduino.cc/en/reference/SD>
 19. void
[https://de.wikipedia.org/wiki/Void_\(Schl%C3%BCsselwort\)](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.fzip>
- 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.