

## **1. Umfeld**

### **1.1. Projektbeschreibung und Zielsetzung**

Der Technikunterricht an der BBS Verden<sup>1</sup> ist in verschiedene Lerngebiete aufgeteilt.

Einer dieser Lerngebiete ist Lerngebiet 12.3, welcher u.A. zur Realisierung Auftragsbezogener Planungen für ein technisches System mitsamt der Präsentation der daraus resultierenden Ergebnisse dient.

Hierfür arbeiten die Schüler an mit dem Lehrer Herr Düren, welcher gleichzeitig als Ansprechpartner zur Verfügung steht vereinbarten Projekte. Die resultierende Projektdokumentation sowie das Aufnehmen, Auswerten und Anzeigen der Messwerte dienen als Leistungsnachweis von Punkt IT 2 des Lerngebietes 12.3. Als Projekt habe ich, Rojat Kadah, mit Herr Düren einen Arduino basierten Personenzähler als Projekt vereinbart.

Im Verlauf der Dokumentation wird der Personenzähler als Projekt, Arduino und Personenzähler bezeichnet.

Bei diesem werden die vorbeigehenden Personen anhand der geplanten Eingabemöglichkeiten als Messwerte aufgenommen. Anschließend werden diese vom Arduino verarbeitet. Zu guter Letzt werden die Ergebnisse Visuell ausgegeben und als Datensatz abgespeichert.

Aufgrund einer nicht genau definierten Vorgehensweise steht es mir frei, auszuwählen, wie ich vorgehe. Die Ergebnisse sind spätestens am 18.06.2021 zu präsentieren. Zu den nicht Unterrichtsbezogenen Zielen gehören vor allem die Vertiefung im Umgang mit Projekten, das verbessern bzw. vertiefen der Hard- sowie Softwarekenntnisse sowie der Umgang mit Kompromissen und einem Begrenzten Budget.

## 2. SOLL-Zustand

### 2.1. Beschreibung

Geplant ist ein Arduino basierter Personenzähler<sup>2</sup>. Jedes Mal, wenn jemand vorbeigeht, wird ein Datensatz<sup>3</sup> erstellt, welcher zusätzlich zur Personenzählung einen Zeitstempel abspeichert, welcher für weitere Zwecke genutzt werden kann. Anhand Visueller Ausgabemethoden wird der Nutzer auf die Zählung hingewiesen.

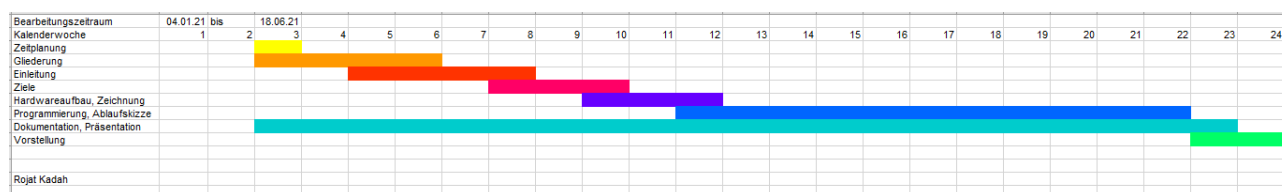
## 3. Planung

### 3.1. Idee

Die Messwerte sollen mithilfe einer Sensorik<sup>4</sup> aufgenommen werden. Knöpfe haben den Nachteil, dass sie berührt werden müssen, was vor allem in Zeiten von Corona<sup>5</sup> eine Zumutung ist. Ein Sensor kann nicht in Frage kommen, da dieser nicht unterscheiden kann, ob jemand rein- oder rausgeht. Zwei Sensoren bieten bereits die Möglichkeit, zwischen rein- und rausgehenden Personen zu unterscheiden, doch können viele Fälle, wie z.B. das umentscheiden der Gehrichtung beim Vorbeigehen oder mehrere vorbeigehende Personen, nicht berücksichtigen. Die größte Beschränkung besteht darin, dass der Arduino nur alle x Sekunden eine vorbeigehende Person zählen kann. Ein Großteil dieser Probleme kann durch Hinzufügung eines dritten Sensors dazwischen, einem Bestätiger, ausgemerzt werden. Der Mittlere Sensor wird im Verlauf der Dokumentation unter den Begriffen Bestätiger sowie Kennzeichnungen, dass er zwischen den anderen Sensoren ist, genannt. Bei dieser Vorgehensweise besteht der Vorteil, dass alles in Echtzeit ablaufen kann und es keine Verzögerungen gibt, da nur gezählt wird, sobald das Vorbeigehen an einer der äußeren Sensoren durch den mittleren Sensor bestätigt wird. Sobald der Bestätiger ausgelöst wird, erhält der Arduino einen Datensatz inklusive des Zeitstempels zur weiteren Verarbeitung. Zur Verarbeitung gehören zum einen die Speicherung der Daten auf einem Speichermedium und zum anderen die Visuelle Ausgabe der Daten für den Vorbeigehenden.

### 3.2. Zeitstrahl

Grafische Darstellung der benötigten Zeit.



Siehe „Dateien/Zeitplan/Zeitplanung.xlsx“ auf Folie 1.

## **4. IST-Zustand**

### **4.1. Beschreibung**

Basierend auf der Planung, Siehe hierfür Kapitel 3, sowie des Sollzustandes, Siehe hierfür Kapitel 2, ist der Personenzähler in einem Zustand, welcher der vereinbarten Funktionalität entspricht.

Für die Speicherung der Datensätze fiel die Entscheidung auf ein Micro SD Modul. Für die Ausgabe der Daten wird ein 20x4 Zeichen I2C LCD-Display genutzt.

Die im 3. Kapitel unter „3.1 Planung“ genannte drei Sensoren Taktik wird beim Personenzähler übernommen. Dank dieser kann der Personenzähler je nach Konfiguration in Echtzeit arbeiten und zwischen der Bewegungsrichtung unterscheiden. Jegliches vorbeigehen wird erst registriert, sobald der Bestätiger aktiviert wird. Die Vorteile sind in Kapitel 3, Planung, dargestellt. Zeitbedingt ist des Weiteren hinzugefügt worden, dass der Personenzähler erkennt, wenn jemand den Durchgang blockiert. Dies erfolgt durch ununterbrochenes Aktivieren des Bestätigers.

Sobald der Bestätiger aktiviert ist, sei es nach vorbeigehen an einem der äußeren Sensoren oder durch Blockierung, erstellt der Arduino auf der angeschlossenen SD-Karte einen Datensatz. Dieser enthält:

- Raumnummer:     Gibt wieder, wo der Personenzähler aufgebaut ist.
- 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.
- 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.

Die Daten werden im CSV Format abgespeichert, doch da der Arduino dieses nicht kennt, werden die Daten als Textdatei abgespeichert, sodass die Dateiendung nur noch abgeändert werden muss. Der Grund hierfür besteht darin, dass Excel das CSV kennt, sodass über Excel die Möglichkeit zur Speicherung im XLSX Format besteht um Diagramme auszugeben.

Sollte die SD nicht erkannt werden oder fehlen, so gibt der Arduino dies beim Start über den LCD-Display wieder.

Anschließend erfolgt die Ausgabe der Personenzahl über das vierzeilige I2C Display.

In der ersten Zeile wird die Raumnummer ausgegeben.

Bei der zweiten Zeile werden die aktuell im Raum befindliche sowie die maximal erlaubte Anzahl an Personen im Raum wiedergegeben.

Der Übersicht halber wird die dritte Zeile lediglich gefüllt, sobald der Bestätiger blockiert wird. Diese weist den Nutzer darauf hin, dass er den Sensor blockiert. Die letzte Zeile, Zeile vier, gibt wieder, ob der Raum voll oder nicht voll ist. „Nicht voll“ wird auch bei keiner Person im Raum ausgegeben, da der Nutzer lediglich darauf hingewiesen wird, dass der Raum nicht voll ist. Die Ausgabe erfolgt Dynamisch je nach Anzahl an Personen gegenüber der Maximal erlaubten Anzahl an Personen im Raum. Die drei Möglichen Programmabläufe sind unter Kapitel „4.2 Ablaufbeschreibung“ abrufbar

## **4.2. Ablaufbeschreibung**

### **4.2.1. Präposition<sub>6</sub>**

Zur Vermeidung komplexer Erklärungen wird bei den folgenden Programmabläufen davon ausgegangen, dass die Sensoren Horizontal in Reihe an den Türrahmen eines Raumes befestigt worden sind.

### **4.2.2. Reingehen**

Wenn jemand in den Raum geht, wird erst der Außensensor aktiviert. Solange die Person ab jetzt nicht am Bestätiger vorbeigeht, wird die Anzahl an Personen im Raum nicht erhöht.

Wenn die Person nun aber am mittleren Sensor vorbeigeht, so wird die Zählung bestätigt, ein Datensatz auf der SD-Karte erstellt und die Anzahl an im Raum befindlichen Personen auf dem LCD-Display aktualisiert. Des Weiteren wird, solange die Anzahl an Personen im Raum kleiner als die maximal erlaubte Anzahl ist, in der vierten Zeile der Status „Nicht Voll“ ausgegeben. Ansonsten wird der Status auf Voll gesetzt.

Wenn die Person nun weiter in den Raum und damit am Innensensor vorbeiläuft, so wird die Anzahl an Personen im Raum nicht um eins verringert, da hierfür eine Betätigung des Bestätigers notwendig ist.

#### **4.2.3. Rausgehen**

Wenn jemand aus dem Raum geht, wird erst der Innensensor aktiviert. Solange die Person ab jetzt nicht am Bestätiger vorbeigeht, wird die Anzahl an Personen im Raum nicht verringert.

Wenn die Person nun am mittleren Sensor vorbeigeht, so wird die Zählung bestätigt, ein Datensatz auf der SD-Karte erstellt und die Anzahl an im Raum befindlichen Personen auf dem LCD-Display aktualisiert. Aufgrund der Programmierung ist es nicht möglich, dass die Anzahl an Personen im Raum kleiner als Null ist. Entsprechend wird standardmäßig keine Personenzahl kleiner als Null auf dem LCD-Display ausgegeben. Wenn die Person nun weiter aus dem Raum und damit am Außensensor vorbeiläuft, so wird die Anzahl an Personen im Raum nicht um eins erhöht, da hierfür eine Bestätigung des Bestätigers notwendig ist.

#### **4.2.4. Stehenbleiben**

Sollte eine Person den Durchgang blockieren, so wird dies entsprechend solange auf dem LCD-Display ausgegeben wie die Blockierung dauert. Des Weiteren wird dies als entsprechender Datensatz abgespeichert.

Ein weiterer, entsprechender Datensatz wird erstellt, sobald der Durchgang wieder frei ist.

## **5. Kostenkalkulation**

### **5.1. Präposition**

Aufgrund mangelnder Vorkenntnisse mit dem Arduino sowie Fein- und Grobmechanischer, gesundheitlicher Probleme wurde jede Komponente mindestens zweimal erworben, um selbst für den Fall, dass der Personenzähler kaputt geht, noch ein Backup zu haben.

### **5.2. Budget**

Wegen mangelnder Wirtschaftlicher Mittel besteht nicht die Möglichkeit, die besten verfügbaren Komponenten zu erwerben. Vor allem auch deshalb, dass aufgrund Möglicher Hardwareausfälle noch ein ausreichendes Budget für den Erwerb von Ersatzkomponenten erhalten bleiben muss. Entsprechend ist das Budget auf 300€ beschränkt.

### **5.3. Tabellarische Rechnung**

Tabellarische Aufstellung der Kosten pro Komponente sowie des Gesamtpreises

<b>Komponente</b>	<b>Anzahl</b>	<b>Einzelpreis</b>	<b>Gesamtpreis</b>
Arduino Uno Revision 3	2	22,47€	22,47€+80€
Jumperkabel 20cm	2 Packungen a 40 Kabel	7,99€	15,98€
KY-004 Taster	1 Packung a 3 Taster	5,99€	5,99€
20x4 Zeichen 2004 LCD-Display	2	9,99€	19,98€
Lesezeichen Sticker	1 Packung a 840 Sticker	6,99€	6,99€
Lötkolben 60W	1	18,99€	18,99€
Netzteil 9V 2A	1	11,49€	11,49€
32GB Micro SD	1	4,79€	4,79€
Micro SD Shield	1	5,29€	5,29€
TRCT 5000	2 Packungen a 3 Sensoren	4,99€	9,98€
<b>Gesamt</b>		<b>98,98€</b>	<b>277,96 €</b>

Siehe Digitalen Anhang unter „Dateien/Rechnungen/Rechnung.xlsx“

## **6. Implementierung**

### **6.1. Programm**

Das Programm besteht aus mehreren Teilen, welche allesamt zum Großteil im Setup sowie Loop Teil des Programms implementiert worden sind.

Global werden vor Beginn der Funktionen die Organisatorischen Variablen wie die Raumnummer und der Zeitstempel, Bibliotheken zur Verwendung der verbauten Komponenten und zu guter Letzt die Sensoren deklariert.

Im Setupteil werden die Sensoren sowie der LCD konfiguriert. Abschließend wird der Fall, dass keine SD-Karte eingefügt worden ist, ins Setup integriert.

Sollte keine SD-Karte eingelegt sein, so gibt der Arduino dies beim Start wieder und stoppt die Arbeit durch eine leere Dauerschleife.

Im Setupteil wird erst die LCD-Ausgabe durchgeführt und anschließend die Sensorabfragen. Siehe hierfür „6.2. Eingabe“.

Anschließend wird die Funktion zur Erhöhung des Zeitstempels aufgerufen. Siehe hierfür „6.3. Zeitstempel“.

Diese erhöht den Zeitstempel um die jeweils beim delay<sup>9</sup> vergangene Anzahl an Millisekunden. Siehe hierfür „6.3. Zeitstempel“. Im Verlauf dieses Kapitels wird davon ausgegangen, dass der Personenzähler Horizontal im Türrahmen aufgebaut worden ist.

Siehe Kapitel „V.I. Bibliotheken“ für die jeweils genutzten Bibliotheken.

### **6.2. Eingabe**

Die Eingabe erfolgt innerhalb des Loop Teils. Dabei wird erstmal abgefragt, ob der Bestätiger aktiviert worden ist. Wenn, dann wird über das Display ausgegeben, dass der Durchgang frei gemacht werden soll.

Folgend findet dann die Betätigung der äußeren Sensoren statt. Wenn jemand in den Raum geht wird erstmal der Außensensor betätigt. Wenn nun der Bestätiger betätigt wird, so wird die Personenzahl um eins erhöht. Wenn stattdessen der Innensensor aktiviert wird, dann wird stattdessen der Rausgehprozess in Gang gesetzt, welcher genauso funktioniert, außer dass das anschließend folgende betätigen des Bestätigers zur Reduzierung der Personenzahl um eins führt. Des Weiteren führt die Betätigung des Bestätigers zum Aufruf der Speicherfunktion. Siehe Hierfür „6.5. Speichern“

### **6.3. Zeitstempel**

Der Zeitstempel wird genutzt, da es Budgetbedingt nicht möglich ist, eine RTC<sup>10</sup>, Real Time Clock, zu verwenden. Die Erhöhung von diesem findet in einer Funktion vom Typ Void statt und wird vor jedem Delay aufgerufen. In der Funktion haben wir zwei Werte. Einmal den Zeitstempel und den Wert, welche die Anzahl an Millisekunden pro Delay angibt. Folgend als Zeitsprung bezeichnet. Zeitstempel wird bei jedem Aufruf um Zeitsprung erhöht. Da Zeitsprung aber in Millisekunden ist, wird er dabei jedes Mal durch 1000 geteilt.

#### 6.4. Verarbeitung

Bei jeder Eingabe wird immer der Bestätiger betätigt. Mit Aufruf von diesem wird immer ein Datensatz auf dem Arduino erstellt, bestehend aus dem Zeitstempel in Sekunden, der maximalen sowie aktuellen Anzahl an Personen, der Anzahländerung, der Gesamten Personenzahl sowie der Gesamten rein- und rausgehenden Anzahl an Personen sowie dem Status. Der Status gibt an, ob der Durchgang blockiert ist, wieder frei ist oder ob der Raum voll bzw. nicht voll ist. Die Anzahländerung gibt je nachdem, ob jemand rein- oder rausgeht oder den Durchgang blockiert, +, – oder 0 wieder.

#### 6.5. Speichern

Bei der Speicherfunktion werden der Zeitstempel, die maximale sowie aktuelle Anzahl an Personen im Raum, die Anzahländerung sowie die Gesamte Anzahl vorbeigehender Personen sowie rein und rausgehend gespeichert. Zuvor wird mithilfe von SD.open der Zugriff auf die Datei gewährt, deren Dateiname am Anfang des Programms vom Nutzer festgelegt werden kann. Am Ende wird dieser Zugriff wieder beendet, sodass die SD-Karte während der Personenzählung sicher entfernt und wieder eingefügt werden kann. Gespeichert wird die Datei mit der Endung einer Textdatei. Das verwendete Format ist das CSV<sup>11</sup> Format, was an der Semikolontrennung zwischen jeder Variablen und dem Zeilenumbruch am Ende jeder Zeile erkennbar ist. Die Wahl fiel auf diese Lösung, da der Arduino standardmäßig keine Möglichkeit zur Speicherung im XLSX<sup>12</sup> Format hat. Dennoch besteht bei Excel die Möglichkeit, CSV Dateien im XLSX Format abzuspeichern.

Diese Vorgehensweise wurde aufgrund ihrer Einfachheit gegenüber anderen Möglichkeiten gewählt und weil das Erneute aufrufen der CSV Datei zur erneuten Erstellung der Diagramme führt.

#### 6.6. Ausgabe

Zur Ausgabe dient ein I2C<sup>13</sup> Display. Die Wahl ist auf dieses gefallen, da es gegenüber herkömmlichen Displays nur vier Anschlusspins besitzt und so mehr Pins für andere Zwecke zur Verfügung stehen.

Über das Display findet die Gesamte Ausgabe statt. Hierzu zählen die Ausgabe der SD Fehlermeldung am Anfang und die Datenausgabe im Verlauf des Loop Teils.

Die SD Fehlermeldung gibt aus, dass die SD entweder falsch formatiert ist oder dass keine angeschlossen ist. Eine Unterscheidung zwischen diesen beiden Fällen kennt der Personenzähler in seiner aktuellen Version nicht.

Teil der Ausgabe im Loop Teil sind die Ausgabe der Raumnummer in der ersten Zeile, die Angabe der aktuellen sowie maximalen Anzahl an Personen im Raum in der zweiten Zeile, bei Blockierung des Durchgangs die zugehörige Meldung in der dritten Zeile und die Ausgabe des Status in der vierten Zeile.



## **7. SOLL/IST Vergleich**

Im Sollzustand geht es vor allem darum, wie das Programm im groben aussehen soll. Als erster Punkt ist das Erstellen eines Datensatzes genannt worden, sobald jemand vorbeigeht. Dies ist im Programm entsprechend umgesetzt worden und wurde insofern erweitert, dass der Arduino zwischen der Bewegungsrichtung unterscheiden kann. Der zweite Punkt des Soll-Zustandes ist der Zeitstempel. Die Umsetzung ist ohne eine RTC erfolgt und wird mithilfe eines auf drei Kommastellen genau eingetragenen Floats gezählt.

Punkt drei ist, dass der Zeitstempel für weitere Zwecke genutzt werden soll. Beim Projekt wird er aktuell lediglich mitgespeichert um zu erfahren, wann jemand vorbeigegangen ist bzw. wann der Durchgang blockiert worden ist. Dies kann sich mit zukünftigen Versionen ändern.

Der Letzte Punkt im Soll-Zustand ist die Visuelle Ausgabe, mit welcher der Nutzer auf die Zählung hingewiesen wird.

Hierfür wird ein I2C Display verwendet, welches dank seiner vier Zeilen mit jeweils zwanzig Zeichen ausreichend Platz hat, um die aktuelle sowie maximal erlaubte Anzahl an Personen wiederzugeben.

Des Weiteren wird das Display genutzt, um die Raumnummer sowie den Status wiederzugeben.

Der Soll-Zustand wurde erfüllt und um weitere Funktionalitäten erweitert.

Dank der modularen Programmierung besteht weiterhin die Möglichkeit, das Programm zu erweitern.

## 8. Nutzungsanalyse

### 8.1. Beschreibung

Um sicherzustellen, dass der Personenzähler korrekt funktioniert, wurde dieser mithilfe einer Testumgebung getestet. Hierbei sind die Sensoren statt in einem Türrahmen an eine Platte befestigt worden. Die Maximal erlaubte Anzahl an Personen wird auf zwei und der Zeitsprung auf 20ms gesetzt. Aufgrund der geringen Reichweite der Sensoren wird statt einer vorbeilaufenden Person eine Hand genutzt, um mehrere Szenarien zu testen. Um Komplexe Erklärungen zu vermeiden, wird dennoch davon ausgegangen, dass der Personenzähler am Türrahmen des Raumes B11 befestigt worden ist und statt Händen Personen sich daran vorbeibewegen.

Diese sind:

- Es gehen solange Personen in den Raum, bis der Raum voll ist. Anschließend gehen diese Personen wieder raus.  
Siehe hierfür „I.II. Maximum erreicht“.
- Es gehen mehr Personen raus als rein.  
Siehe hierfür „I.III. Minimum erreicht“.
- Eine Person stellt sich vor den Bestätiger und blockiert so den Durchgang.  
Siehe hierfür „I.IV. Durchgang blockiert“.
- Die SD Fehlt bzw. ist fehlerhaft.  
Siehe hierfür „I.V. SD Fehlerhaft“

### 8.2. Maximum erreicht

#### 8.2.1. Beschreibung

Bei diesem Testdurchlauf soll geprüft werden, wie der Personenzähler bei Erreichen einer Personenzahl reagiert, welche höher als die maximal erlaubte ist. In diesem Fall fünf. Anschließend soll überprüft werden, wie er beim Rückgang zurück auf null Personen reagiert.

Dieser Test dient zum einen der Überprüfung der Auf- bzw. Abwärtszählung mitsamt der dazugehörigen Statusänderung und zum anderen der Überprüfung der Richtigkeit der erstellten Datensätze.

#### 8.2.2. Erwartung

Voraussichtlich wird der Arduino bis zum Erreichen von zwei Personen im Raum ausgeben, dass der Raum nicht voll ist. Anschließend, von zwei bis hin zu fünf Personen im Raum, wird der Personenzähler voraussichtlich wiedergeben, dass der Raum voll ist. Dies wird sich ggf. bis zur Rückkehr auf zwei nicht ändern. Bei Erreichen von weniger als zwei Personen im Raum wird mit großer Wahrscheinlichkeit ausgegeben, dass der Raum nicht voll ist.

Nun zum Datensatz.

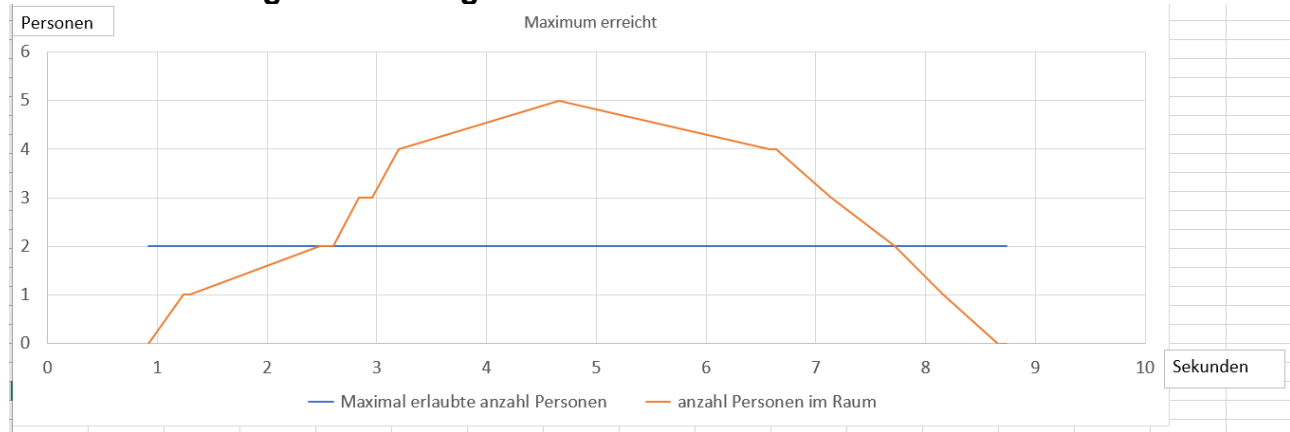
Während der Aufwärtszählung werden die Spalten C, die Personenanzahl, E sowie F, die Globalen Personenzählungen, wahrscheinlich um eins erhöht. Dabei wird ggf. Spalte D, welche die Anzahländerung pro Datensatz angibt, auf + gesetzt, um die Erhöhung darzustellen.

Beim anschließenden herunterzählen wird Spalte C voraussichtlich bis zu einem Wert

von null heruntergezählt, während die Globalen Personenzählungen in Spalte E und F um eins erhöht werden.

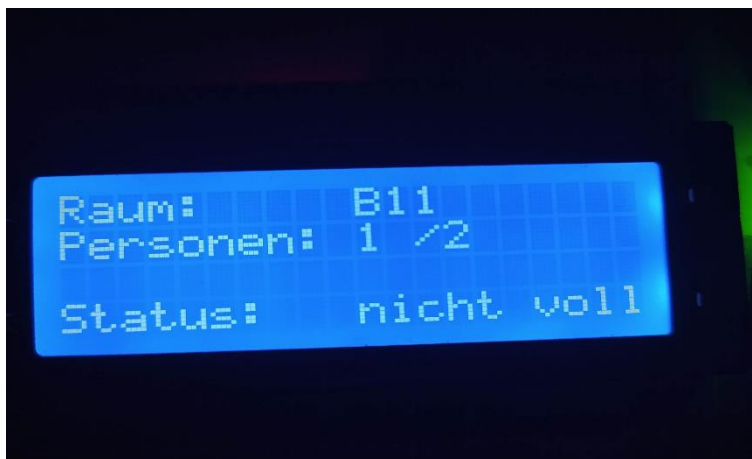
Der Status wird mit großer Wahrscheinlichkeit ab zwei Personen im Raum auf „voll“ gesetzt und sonst auf „nicht voll“

### 8.2.3. Auswertung der Messergebnisse



Vorschaubild. Siehe „Dateien/Testdurchlauf/maximum erreicht/Diagramm Maximum Durchlauf.png“ aus Seite 2 im Digitalen Anhang für eine vergrößerte Ansicht.

Am Diagramm ist an der blauen Linie die maximal erlaubte Anzahl an Personen ersichtlich. Die Orange Linie gibt die aktuelle Anzahl an Personen im Raum wieder. Hier muss nun, solange die Orange Linie unter der blauen ist, beim LCD-Display „nicht voll“ als Status wiedergegeben werden. In diesem Fall mit null sowie einer Person im Raum.



Vorschaubild. Siehe „Dateien/Fotos/LCD/einePerson.png“ im Digitalen Anhang für eine vergrößerte Ansicht.

Wie an dem Foto erkennbar, wird dies erfüllt. Anschließend, ab Erreichen von zwei oder mehr Personen im Raum, wird als Status „voll“ ausgegeben.



Vorschaubild. Siehe „Dateien/Fotos/LCD/zweiPersonen.png“ im Digitalen Anhang für eine vergrößerte Ansicht des linken Bildes.

Siehe „Dateien/Fotos/LCD/fuenfPersonen.png“ im Digitalen Anhang für eine vergrößerte Ansicht des rechten Bildes.

Wie erkennbar ist, wird dies ebenso erfüllt. So ist erkennbar, dass der Personenzähler die Zählung samt der Statussynchronisierung draufhat. Am Diagramm ist des Weiteren erkennbar, dass die Personenzahl im Raum gegen Ende wieder unter zwei, auf eins, geht. Entsprechend hat der Status sich auf „nicht voll“ zu stellen.



Vorschaubild. Siehe „Dateien/Fotos/LCD/einePerson.png“ im Digitalen Anhang für eine vergrößerte Ansicht.

Erkennbar ist, dass die Personenzahl unter zwei ist und der Status entsprechend auf „nicht voll“ steht. Entsprechend funktioniert die LCD-Ausgabe.

## Datensatz:

Raumnummer:	B11						
Zeitstempel [s]	Maximal erlaubte anzahl Personen	anzahl Personen im Raum	Anzahländerung	Personenzahl Gesamt	Personenzahl Gesamt raus	Personenzahl Gesamt rein	Status
0.92	2	0	0	0	0	0	0 Durchgang wird blockiert
0.92	2	0	0	0	0	0	0 Durchgang wieder frei
1.24	2	1 +	1	1	0	0	1 nicht voll
1.3	2	1	0	1	0	0	1 Durchgang wird blockiert
1.3	2	1	0	1	0	0	1 Durchgang wieder frei
2.48	2	2 +	2	2	0	0	2 nicht voll
2.54	2	2	0	2	0	0	2 Durchgang wird blockiert
2.56	2	2	0	2	0	0	2 Durchgang wird blockiert
2.58	2	2	0	2	0	0	2 Durchgang wird blockiert
2.6	2	2	0	2	0	0	2 Durchgang wird blockiert
2.6	2	2	0	2	0	0	2 Durchgang wieder frei
2.84	2	3 +	3	3	0	0	3 voll
2.9	2	3	0	3	0	0	3 Durchgang wird blockiert
2.92	2	3	0	3	0	0	3 Durchgang wird blockiert
2.94	2	3	0	3	0	0	3 Durchgang wird blockiert
2.96	2	3	0	3	0	0	3 Durchgang wird blockiert
2.96	2	3	0	3	0	0	3 Durchgang wieder frei
3.2	2	4 +	4	4	0	0	4 voll
4.66	2	5 +	5	5	0	0	5 voll
6.58	2	4 -	4	6	1	1	5 voll
6.64	2	4	0	6	1	1	5 Durchgang wird blockiert
6.64	2	4	0	6	1	1	5 Durchgang wieder frei
7.14	2	3 -	7	7	2	2	5 voll
7.72	2	2 -	8	8	3	3	5 voll
8.16	2	1 -	9	9	4	4	5 voll
8.66	2	0 -	10	10	5	5	5 nicht voll
8.72	2	0	0	10	5	5	5 Durchgang wird blockiert
8.74	2	0	0	10	5	5	5 Durchgang wird blockiert
8.74	2	0	0	10	5	5	5 Durchgang wieder frei

Vorschau. Siehe „Dateien/Testdurchlauf/maximum erreicht/Tabelle.xlsx“ im Digitalen Anhang für eine vergrößerte Ansicht.

Wie an der Tabelle an u.A. Zeile drei mit einem Zeitstempel von 1,24 Sekunden erkenntlich ist, werden mit der Erhöhung der Personenzahl im Raum auch die Globale Personenzahl sowie die Globale Personenzahl welche reingeht um eins erhöht. Weiterhin wird bei jeder Erhöhung Anzahländerung auf plus gesetzt.

Ebenso ist an u.A. Zeile zwanzig mit dem Zeitstempel 6,58 Sekunden ersichtlich, dass es bei der Verringerung der Personenzahl im Raum andersherum mit demselben Prinzip funktioniert. Die Globale Personenzahl sowie die Globale rausgehende Personenzahl wird um eins erhöht. Die Personenzahl im Raum wird um eins verringert. Dabei wird die Anzahländerung auf minus gesetzt.

Sobald der Durchgang blockiert wird, ist Anzahländerung auf 0 gesetzt, da weder jemand rein- noch rausgeht.

Rechts beim Status wird ausgegeben, ob der Raum voll, nicht voll oder ob der Durchgang blockiert ist. Bei Zeile zwanzig mit dem Zeitstempel 6,58 z.B. ist der Status voll, was widerspiegelt, dass mehr Personen im Raum sind als erlaubt.

### 8.2.4. Interpretation der Ergebnisse

Unter Heranziehung der Erwartungen wird ersichtlich, dass die Programmierung samt der Ausgabe und der Speicherung des Personenzählers stimmen.

Der Personenzähler wurde nicht in gleichen Zeitabständen betätigt, um diesen auf veränderte Zeitabstände zu überprüfen.

Dennoch wird ersichtlich, dass die unter Punkt „9.3. Geplante Features“ genannte Veränderung der Sensorreihenfolge in zukünftigen Updates übernommen wird, da es zeitlich nicht mehr möglich ist, die Anpassung samt Testung durchzuführen.

### 8.3. Minimum erreicht

#### 8.3.1. Beschreibung

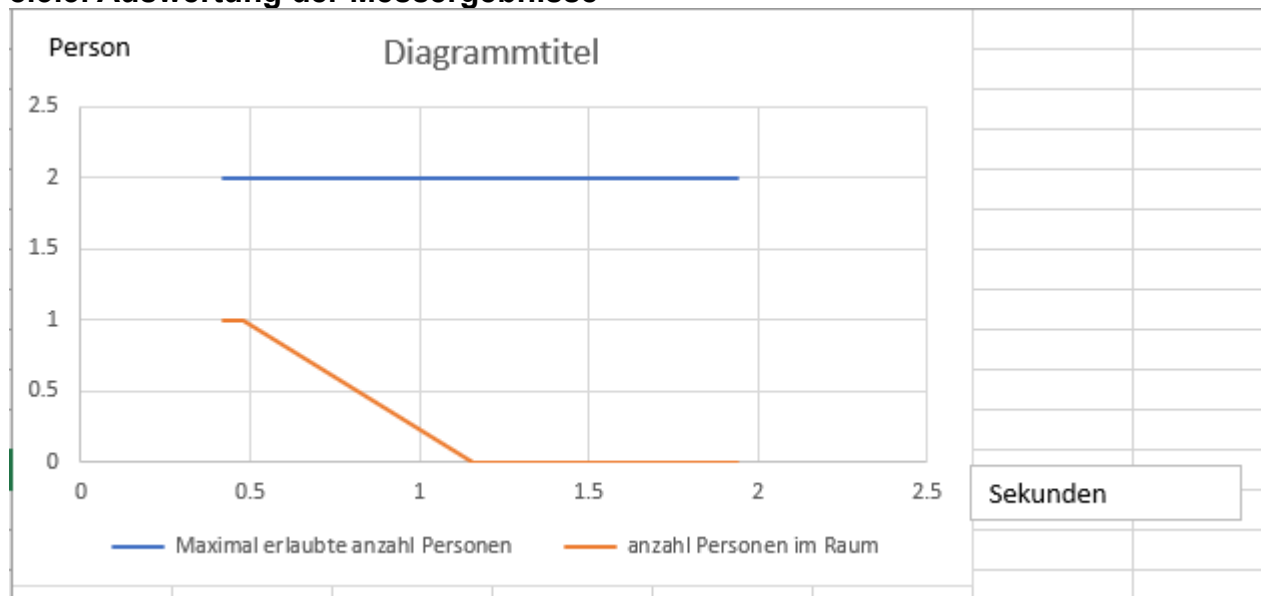
Ziel des Testdurchlaufes soll es sein, zu überprüfen wie der Personenzähler damit umgeht, wenn mehr Personen rausgehen als reingehen. Hierzu soll die Personenzahl erstmal auf eins erhöht werden. Anschließend sollen zwei Personen rausgehen, sodass die Reaktion überprüft werden kann.

Dieser Test dient zum einen der Überprüfung der Programmierung und zum anderen der Überprüfung der Richtigkeit der erstellten Datensätze.

#### 8.3.2. Erwartung

Es wird davon ausgegangen, dass die Personenzahl um eins erhöht und anschließend um eins verringert wird, da die Programmierung eine Verringerung unter null nicht zulässt. Beim zweiten, der rausgeht, wird die Globale Personenzahlbewegung voraussichtlich nicht um diesen erhöht.

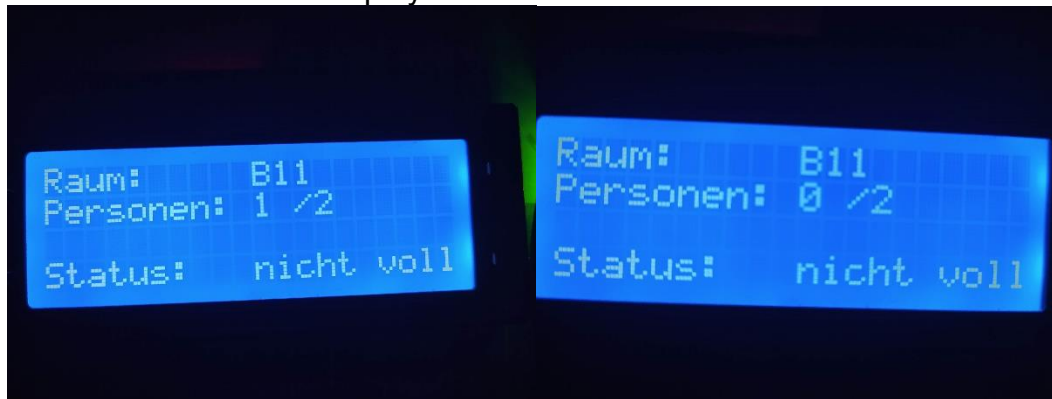
#### 8.3.3. Auswertung der Messergebnisse



Vorschau bild. Siehe „Dateien/Testdurchlauf/minimum erreicht/Tabelle.xlsx“ Auf Folie 2 im Digitalen Anhang für eine vergrößerte Ansicht.

Am Diagramm ist erkennbar, dass die maximale Anzahl an Personen, die blaue Linie, nicht von der aktuellen Personenzahl, der Orangen Linie, erreicht wird. Stattdessen bleibt die aktuelle Anzahl nach den zwei verlassenden Personen bei null.

Dies ist auch am LCD-Display ersichtlich



Vorschaubild. Siehe „Dateien/Fotos/LCD/einePerson.png“ im Digitalen Anhang für eine vergrößerte Ansicht vom linken Bild.

Vorschaubild. Siehe „Dateien/Fotos/LCD/nullPersonen.png“ im Digitalen Anhang für eine vergrößerte Ansicht vom rechten Bild.

Wie am Anfang erkenntlich, ist der Status bei „nicht Voll“. Weiterhin ist die Personenzahl im Raum unter der Maximal erlaubten Anzahl.

Raumnummer	B11							
Zeitstempel	Maximal erla	anzahl Perso	Anzahlaende	Personenzah	Personenzah	Personenzah	Status	
0.42	2	1 +		1	0	1	nicht voll	
0.48	2	1	0	1	0	1	Durchgang wird blockiert	
0.48	2	1	0	1	0	1	Durchgang wieder frei	
1.16	2	0 -		2	1	1	nicht voll	
1.94	2	0	0	3	1	1	Durchgang wird blockiert	
1.94	2	0	0	3	1	1	Durchgang wieder frei	

Vorschaubild. Siehe „Dateien/Testdurchlauf/minimum erreicht/Tabelle.xlsx“ auf Folie 1 im Digitalen Anhang für eine vergrößerte Ansicht.

Wie ersichtlich ist, wird die Globale Personenzahl sowie die sich Global reinbewegenden beim Reingehen um eins erhöht. Beim Verlassen des Raumes wird die Globale Personenzahl um insgesamt zwei erhöht, während die Global rausgehenden Zahl um eins erhöht wird. Dies wird in einem Update ausgebessert, da es zeitlich nicht mehr mit der Testung machbar ist.

#### 8.3.4. Interpretation der Ergebnisse

Unter Heranbeziehung der Erwartungen wird ersichtlich, dass die Programmierung mitsamt der Ausgabe und Speicherung zum Großteil funktionieren, aber es bei der Eintragung der Globalen Personenzahlwerte stockt.

Wie bereits bei Kapitel „8.3.3. Auswertung der Ergebnisse“ ersichtlich, wird dies in einer zukünftigen Version behoben.

## 8.4. Durchgang blockiert

### 8.4.1. Beschreibung

Bei diesem Testdurchlauf wird der Bestätiger blockiert. Es wird überprüft, ob der Status in der dritten LCD<sub>14</sub>-Zeile sowie der Status im Datensatz übereinstimmen.

### 8.4.2. Erwartung

Es wird erwartet, dass in der dritten Zeile beim blockieren des Bestätigers „beweg dich“ ausgegeben wird.

Weiterhin, dass Anzahländerung im Datensatz 0 und Status bis zum Ende der Blockierung „Durchgang wird blockiert“ wiedergibt. Anschließend „Durchgang wieder frei“.

### 8.4.3. Auswertung der Messergebnisse

Raumnummer: B11							
Zeitstempel [s]	Maximal erlaubte anzahl Personen	anzahl Personen im Raum	Anzahlaenderung	Personenzahl Gesamt	Personenzahl Gesamt raus	Personenzahl Gesamt rein	Status
1.64	2	0	0	0	0	0	0 Durchgang wird blockiert
1.66	2	0	0	0	0	0	0 Durchgang wird blockiert
1.68	2	0	0	0	0	0	0 Durchgang wird blockiert
1.7	2	0	0	0	0	0	0 Durchgang wird blockiert
1.72	2	0	0	0	0	0	0 Durchgang wird blockiert
1.72	2	0	0	0	0	0	0 Durchgang wieder frei

Vorschaubild. Siehe „Dateien/Testdurchlauf/Durchgang blockiert/Tabelle.xlsx“ auf Folie 1 im Digitalen Anhang für eine vergrößerte Ansicht.

Wie an der Tabelle ersichtlich wird solange der Durchgang blockiert wird unter Anzahländerung „0“ und unter Status „Durchgang wird blockiert“ ausgegeben. Beim freiwerden „Durchgang wieder frei“.

### 8.4.4. Interpretation der Ergebnisse

Unter Heranziehung der Erwartung wird ersichtlich, dass es funktioniert.

## 8.5. SD Fehlerhaft

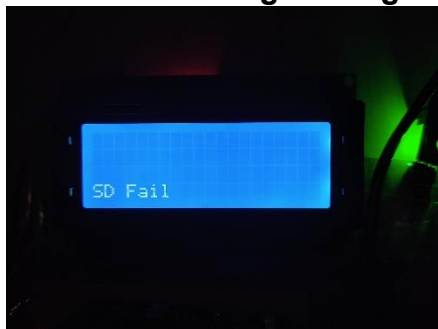
### 8.5.1. Beschreibung

Wenn beim Start die SD Karte fehlt soll eine Fehlermeldung ausgegeben werden.

### 8.5.2. Erwartung

Der Personenzähler gibt bei fehlen eine Fehlermeldung in der letzten Zeile wieder.

### 8.5.3. Auswertung der Ergebnisse sowie Interpretation



Die SD wurde nicht angeschlossen. Dies hat der Personenzähler beim Start erkannt und die Fehlermeldung „SD Fail“ ausgegeben.

Unter Heranziehung der Erwartungen wird ersichtlich, dass die Ausgabe der Fehlermeldung funktioniert.

Vorschaubild. Siehe „Dateien/Fotos/LCD/sd fail.png“ im Digitalen Anhang für eine vergrößerte Ansicht.



## **9. Fazit**

### **9.1. Reflektion**

Anhand von Kapitel 7, dem Soll/Ist Vergleich, ist ersichtlich, dass das Projekt in fast vollem Umfang erfüllt worden ist.

Es wurde geplant, dass der Arduino Datensätze erstellt, einen Zeitstempel erstellt und das Ergebnis Visuell wiedergibt.

Lediglich beim Zeitstempel gibt es Probleme. Diese bestehen nicht in Soft- bzw. Hardwareproblemen, sondern darin, dass der Nutzer die Startuhrzeit Manuell Nachtragen muss, da keine RTC eingebaut ist und entsprechend der Delay für den Zeitstempel genutzt wird

Zum Erreichen des Projektes trugen vor allem meine Vorkenntnisse in der Programmierung sowie Hilfe Seitens der Lehrkräfte und einiger Mitschüler bei.

Zu guter Letzt habe ich das Ziel, meine Kenntnisse mit dem Hard- und Softwareumgang sowie mit dem Arduino umzugehen, insofern vertieft, dass ich sicher im belegen der Pins sowie besser im Umgang mit der Programmierung bin. Beim Softwareumgang vor allem das vertiefen der genutzten Programmierstruktur sowie das verbessern meiner Programmierkenntnisse.

### **9.2. Probleme**

Während des Projektes ist es zu vielen Problemen gekommen. Hauptsächlich im Umgang mit der Hardware sowie des Budgets.

- Angefangen bei der Eingabemöglichkeiten. Es hatte relativ lange gedauert, bis eine Festlegung der Messwerteingabe festgelegt worden ist. Anfangs unter Nutzung von Buttons, doch dann mit der Überlegung zwischen PIRs sowie Infrarotsensoren.
  - Ein PIR hat eine größere Reichweite, aber dafür eine gewisse Wartezeit, bis er wieder für die Zählung genutzt werden kann. Dies würde zwischen jeder Aktivierung zu unnötigen Wartezeiten führen. Weiterhin hat der PIR bei Testdurchläufen nicht funktioniert, weshalb er nicht genutzt worden ist.
  - Ein Infrarotsensor hat in derselben Preisklasse eine geringere Reichweite, arbeitet dafür aber in Echtzeit und ohne Wartezeit. Weiterhin ist die Sonne eine Infrarotquelle, welche zu falschen Aktivierungen der Sensoren führt. Dies kann jedoch durch Abschirmung sowie Umpositionierung der Sensoren vermeiden werden.

Die Entscheidung ist am Ende auf den Infrarotsensor gefallen, da er Preislicher ist, ohne Wartezeiten auskommt und bei den Testdurchläufen eine höhere Zuverlässigkeit vorgewiesen hat.

- Budgetbedingt hat es keine Möglichkeit gegeben, eine RTC sowie höherwertige Komponenten zu verbauen, sodass ein Zeitstempel zur zeitlichen Orientierung dient sowie eine Praktische Umsetzung des Projektes nicht möglich ist.

- Wenn sich jemand vor dem mittleren Sensor aufhält wird eine Meldung ausgegeben, dass diese Person weitergehen soll. Hierbei besteht das Problem, dass durchgehend Datensätze erstellt werden solange sich der Sensor blockiert wird. Aufgrund der übrigen Projektzeit besteht nicht die Möglichkeit, dies zu korrigieren, doch wird in zukünftigen Versionen ausgebessert.

### 9.3. Geplante Features

Für das Projekt sind noch weitere Updates sowie Upgrades geplant, für deren Umsetzung die Zeit bzw. das Budget nicht gereicht hatte.

Geplant sind:

- Verbauen von Infrarotsensoren mit höherer Reichweite.
- Ein Display mit Touchscreen, um die Konfiguration unabhängig von der IDE zu machen.
- Eine Menüführung, sodass die IDE lediglich für Update sowie Wartungszwecke verwendet werden muss.
- Abschirmung für die Infrarotsensoren.
- Gehäuse für den Personenzähler.
- Button, um bei Fehlzählungen die Personenzahl im Raum zurückzusetzen.
- Die Programmierung wird insofern verändert, dass der mittlere Sensor nicht mehr der Bestätiger ist, sondern der Ausgangspunkt, von welchem aus der innere oder äußere Sensor aktiviert werden können. Zeitlich konnte dies aufgrund der dazugehörigen Testphase nicht mehr umgesetzt werden.

Geplant ist:

- Wenn jemand reingeht, wird erstmal der mittlere und anschließend der innere Sensor betätigt. Der äußere Sensor macht solange nichts
- Wenn jemand rausgeht, wird erstmal der mittlere und anschließend der äußere Sensor betätigt. Der innere Sensor macht solange nichts.
- Die Aufgabe der Blockierungserkennung werden entsprechend der innere sowie der äußere Sensor machen, vorausgesetzt diese werden vor dem mittleren Aktiviert.