

INFO: VARIABLEN

Bei der Programmierung des Verkehrszählers musst du immer wieder auf gewisse Werte zurückgreifen. Damit dies unkompliziert machbar ist, bietet die Informatik mit dem Konzept der Variablen eine hilfreiche Lösung. Sie sind eine Art Kiste, die mit einem Namen versehen ist, in dieser Kiste kannst du nun verschiedene Dinge hinterlegen, zum Beispiel Zahlen oder auch Texte und diese später erneut abrufen.



MExLab

 (\mathbf{i})

Je nachdem, was du in einer Variable speichern möchtest, musst du den richten Datentyp auswählen:

Zeichen (char): Für einzelne Textzeichen Text (string): Für ganze Wörter oder Sätze Zahl (int): Für Zahlen von -32768 bis +32768 Große Zahl (long): Für Zahlen von –214748364 bis +2147483648 Dezimalzahl (float): Für Kommazahlen (z.B. 25,3) Zustand (boolean): wahr oder falsch

Variablen können ihren Wert im Laufe des Programmes verändern, sodass du zum Beispiel der Variable "Anzahl" nach jedem vorbeifahrenden Auto um eins wachsen lässt.



Der Code funktioniert nicht? Tipps zur Fehlerbehebung

Stecken deine Kabel exakt wie in der Abbildung?
Sind deine Befehlsblöcke wirklich wie kleine "Puzzleteile" verbunden?
Hast du alle Blöcke gelöscht, die nicht mit deinem Hauptblock verbunden sind?

• Hast du den aktuellsten Programmcode kompiliert

und nach Änderungen in Blockly erneut übertragen?

Noch Schwierigkeiten? Dann wende dich an eine:n Mentor:in!





Ministerium für Wirtschaft, Industrie, Klimaschutz und Energie des Landes Nordrhein-Westfalen



Schritt 1A

1. Zum Programmieren: <u>blockly.sensebox.de</u>

2. Im Setup werden einige Komponenten integriert, die zu Beginn des Programmcodes einmalig eingeschaltet werden müssen.



3. Das Display muss nun im Setup initialisiert werden:



Display initialisieren

Schritt 1B

1. Um die Messwerte auf dem Display anzeigen zu lassen, benötigst du die Blöcke 'Zeige auf dem Display' und 'Schreibe Text/ Zahl' in der Endlosschleife.



2. Anstatt nei 'Wert' nun den Block des jeweiligen Sensors einfügen, definierst du eine Variable und fügst diese anschließend bei 'Wert' ein. Wähle dafür in der Kategorie 'Variablen' den Block 'Typisierte Variable erstellen' aus.

3. Definiere die Variable als Zahl (int), benenne sie als 'Distanz' und füge sie in die Endlosschleife ein.





Verkehrszähler von Autos

Verbinde das OLED-Display und die Sensoren mit dem Mikrocontroller.

1) Erstelle ein Programm, sodass dir Messwerte des Distanzsensors auf dem Display angezeigt werden

2) Erweitere deinen Programmcode, sodass ebenfalls die Anzahl an erkannten Autos auf dem Display angezeigt wird.



Schritt 1C

Die Variable muss nun einem Messwert zugeordnet werden. Da die Distanz mit dem Ultraschall-Distanzsensor gemessen wird, wird dieser Sensorblock mit der Variable 'Distanz' verbunden.

> Schreibe int Distanz - Ultraschall-Abstandssensor an Port A - Trigger A1 - Echo A2 - Maximale Distanz 250 cm

> > Teste deinen Code!

Schritt 2

1. Das Messgerät erkennt nun Distanzen, aber noch keine Autos. Um diese zu zählen, muss eine neue Variable (int) namens 'Auto' erstellt werden. Da die Zählung der Autos bei null starten soll, legst du dies im 'Setup' fest (Kategorie Mathematik).



2. Mit einer 'Wenn – Mache' Bedingung legst du nun fest, in welchem Fall ein neues Auto dazugezählt wird: Wenn die Distanz kleiner als 40 cm ist, dann erhöht sich die Variable der Autos um eins. Damit kein Auto doppelt gezählt wird, füge den Block 'Warte 1000 Millisekunden' ein.



3. Damit die Anzahl der Autos auch auf dem Display angezeigt wird, füge erneut den Block 'Schreibe Text/ Zahl' hinzu und füge bei 'Wert' die Variable 'Autos' ein. Mit dem regelmäßigen Löschen des Displays stellst du sicher, dass dir immer die aktuellste und richtige Distanz zum vorbeifahrenden Auto angezeigt wird. Passe zudem die y-Koordinate an, sodass die Distanz und die Anzahl untereinander angezeigt werden.



Damit ist die Programmierung abgeschlossen. Simuliere nun mit gegenständen vorbeifahrende Autos und teste, ob alles funktioniert.