

# Sensoren

- Eingang
- Zähler
- I2C
- USB

# Eingang

## Der "Starte jedes Mal"-Block

Der "Starte jedes Mal"-Block bietet die Möglichkeit ein Programm ablaufen zu lassen, wenn eine Bedingung erfüllt ist. Er funktioniert also ähnlich wie eine Fallunterscheidung, wird aber nicht nur einmal durchlaufen, sondern jedes Mal, wenn die Bedingung erfüllt ist, während des gesamten Ablaufs des Programms. Der "Starte jedes Mal"-Block:



ist eine Abkürzung für folgendes Konstrukt:



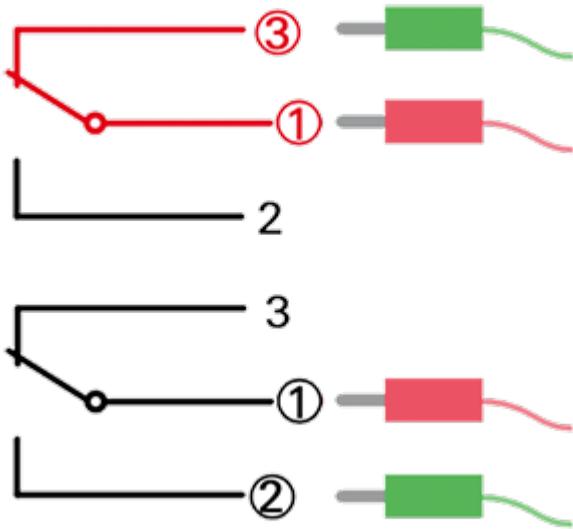
Man kann in den "Starte jedes Mal"-Block der Kategorie Eingaben alle Bedingungen aus ebendieser Kategorie einsetzen.

**Hinweis: Der Programmabschnitt innerhalb des "Starte jedes Mal"-Blocks sollte kurz gehalten werden und keine blockierenden Aufrufe oder Endlosschleifen enthalten, sodass dieser Teil des Programms schnell abgearbeitet werden kann.**

## Taster



Der Taster ist ein digitaler Sensor, der "Es fließt Strom" von "Es fließt kein Strom" unterscheidet. Ob Strom fließt, hängt dabei sowohl von der Verkabelung, als auch davon ab, ob der Taster gedrückt ist. Man kann den Taster also auf zwei verschiedenen Arten verwenden:



Als "Schließer":

Kontakte 1 und 3 werden angeschlossen.

Taster gedrückt: Es fließt Strom.

Taster nicht gedrückt: Es fließt kein Strom.

Als "Öffner":

Kontakte 1 und 2 werden angeschlossen.

Taster gedrückt: Es fließt kein Strom.

Taster nicht gedrückt: Es fließt Strom.

## Abrufen

Mit "hole Mini-Taster [] Status" erhält man Information darüber, ob durch den Taster Strom fließt oder nicht. Fließt Strom, wird 1 zurückgegeben, fließt kein Strom, 0.

## Abfragen

Um abzufragen, ob der Taster in einem bestimmten Zustand ist, wird der Block "ist Taster [] []" genutzt. Über das Dropdown-Menü (kleines Dreieck) kann ausgewählt werden, nach welchem Zustand gefragt wird. Dieser Block kann als Bedingung genutzt werden.

## Reedkontakt



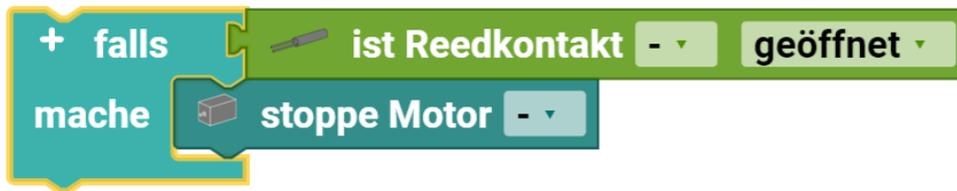
Der Reedkontakt ist ein magnetisch betätigter Schalter, der verwendet wird, um die Anwesenheit eines magnetischen Feldes zu erkennen. Der Kontakt schließt sich, wenn ein Magnet in der Nähe ist, und öffnet sich, wenn der Magnet entfernt wird.

## Abrufen

Mit "hole Reedkontakt [] Status" erhält man Informationen darüber, ob der Reedkontakt geschlossen oder offen ist. Ist der Kontakt geschlossen, wird 1 zurückgegeben, ist er offen, 0.

## Abfragen

Um abzufragen, ob der Reedkontakt in einem bestimmten Zustand ist, wird der Block "ist Reedkontakt [] []" genutzt. Über das Dropdown-Menü (kleines Dreieck) kann ausgewählt werden, nach welchem Zustand gefragt wird. Dieser Block kann als Bedingung genutzt werden. Im Beispiel wird der Motor gestoppt, wenn der Kontakt geöffnet ist.



## Ultraschallsensor



Der Ultraschallsensor wird genutzt, um Entfernungen zu messen.

## Abrufen

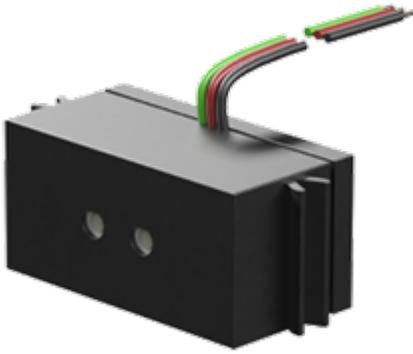
Mit "hole Ultraschallsensor [] Abstand" erhält man die Information, wie weit der Sensor vom nächsten Gegenstand entfernt ist. Der Abstand wird in cm zurückgegeben.

## Abfragen

Um abzufragen, ob der Sensor einen bestimmten Abstand zum nächsten Gegenstand hat, wird der Block "ist Ultraschallsensor [] Abstand [] ..." genutzt. Über das Dropdown-Menü (kleines Dreieck) kann ausgewählt werden, wie der gemessene Abstand mit einem eingegebenen Wert verglichen werden soll (<, =, >). Der Vergleichswert wird in das Zahlenfeld am Ende des Blocks eingegeben. Dieser Block kann als Bedingung genutzt werden. Im Beispiel wird der Motor gestoppt, wenn der gemessene Abstand kleiner als 2 cm ist.



## Farbsensor



Der Farbsensor sendet rotes Licht aus und misst, wie viel davon zurück reflektiert wird. Je nachdem, wie stark die Reflexion ist, gibt der Farbsensor Werte von 0 bis 2000 zurück. Er eignet sich gut, um vorher kalibrierte Farben zu erkennen.

## Abrufen

Mit "hole Farbsensor [] Wert" erhält man die Information, wie stark eine Oberfläche das Licht reflektiert.

## Abfragen

Um abzufragen, ob der Sensor eine bestimmte Farbe vor sich hat, wird der Block "ist Farbsensor [] Wert [] ..." genutzt. Über das Dropdown-Menü (kleines Dreieck) kann ausgewählt werden, wie der gemessene Farbwert mit dem eingegebenen Wert verglichen werden soll (<, =, >). Der Vergleichswert wird in das Zahlenfeld am Ende des Blocks eingegeben. Dieser Block kann als Bedingung genutzt werden. Im Beispiel wird der Motor gestoppt, wenn der gemessene Farbwert kleiner als 1000 ist.



## IR-Spursensor



Der Infrarot-Spursensor ist ein digitaler Sensor zur Erkennung einer schwarzen Spur auf weißem Untergrund, der bei einem Abstand von 5 bis 30 mm von Sensor zu Untergrund arbeitet.

## Abrufen

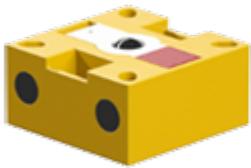
Mit "hole IR-Spursensor [] Status" erhält man 0, wenn der Sensor keine Spur erkennt. Erkennt der Sensor eine Spur, wird 1 zurückgegeben.

## Abfragen

Um abzufragen, ob der IR-Spursensor eine Spur erkennt, vergleicht man den aktuellen Spurstatus mit 0 oder 1. Hierzu eignet sich der Block "ist IR-Spursensor [] Status [] ...". Über das Dropdown-Menü (kleines Dreieck) kann ausgewählt werden, wie der Spurstatus mit dem eingegebenen Wert verglichen werden soll (<, =, >). Der Vergleichswert wird in das Zahlenfeld am Ende des Blocks eingegeben. Dieser Block kann als Bedingung genutzt werden. Im Beispiel wird der Motor gestoppt, wenn der Spurstatus 0 ist.



## Fototransistor



Der Fototransistor ist ein digitaler Sensor, der hell von dunkel unterscheidet.

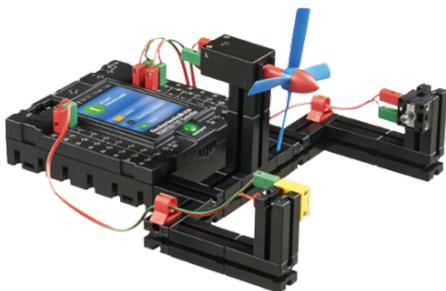
## Abrufen

Mit "hole Fototransistor [] Status" erhält man 0, wenn der Sensor kein Licht erkennt. Erkennt der Sensor ausreichend Licht, wird "1" zurückgegeben.

## Abfragen

Um abzufragen, ob der Fototransistor hell oder dunkel erkennt, vergleicht man den Helligkeitsstatus mit 0 oder 1. Hierzu eignet sich der Block "ist Fototransistor [] Status []". Über das Dropdown-Menü (kleines Dreieck) kann ausgewählt werden, ob hell oder dunkel abgefragt werden soll.

Eine mögliche Verwendung für einen Fototransistor ist in einer Lichtschranke, wie diesem Modell.



## Fotowiderstand



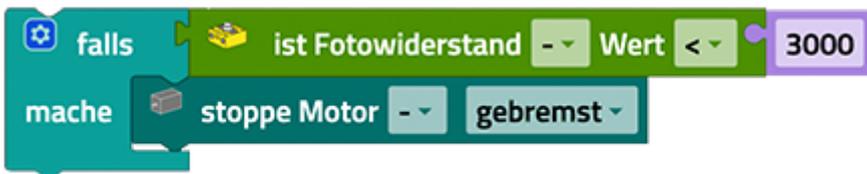
Der Widerstand des Fotowiderstands sinkt, wenn er mehr Helligkeit ausgesetzt ist. Der ausgegebene Wert des Fotowiderstands ist also ein Maß für Helligkeit.

## Abrufen

Mit "hole Fotowiderstand [] Wert" erhält man die Information, wie hell es ist. Je kleiner der ausgegebene Wert, desto heller ist es.

## Abfragen

Um abzufragen, ob der Fotoresistor einen bestimmten Helligkeitswert misst, wird der Block "ist Fotowiderstand [] Wert [] ..." genutzt. Über das Dropdown-Menü (kleines Dreieck) kann ausgewählt, wie der Helligkeitswert mit dem eingegebenen Wert verglichen werden soll (<, =, >). Der Vergleichswert wird in das Zahlenfeld am Ende des Blocks eingegeben. Dieser Block kann als Bedingung genutzt werden. Im Beispiel wird der Motor gestoppt, wenn der Helligkeitswert kleiner als 3000 ist.



## NTC-Widerstand



Der NTC-Widerstand ist ein nichtbinärer Temperatursensor. Sein elektrischer Widerstand sinkt, wenn die Temperatur steigt, und ist damit ein Maß für die Temperatur.

## Abrufen

Mit "hole NTC-Widerstand []" erhält man entweder einen Widerstandswert oder die daraus errechnete Temperatur. Was zurückgegeben werden soll, kann über das Dropdown-Menü (kleines Dreieck) gewählt werden.

## Abfragen

Um abzufragen, ob der NTC-Widerstand einen bestimmten Wert misst, wird der Block "ist NTC-Widerstand [] [] ..." genutzt. Über die Dropdown-Menüs (kleines Dreieck) kann gewählt werden, was und mit welchem Vergleichsoperator verglichen werden soll. Der Vergleichswert wird in das Zahlenfeld am Ende des Blocks

eingetragen. Dieser Block kann als Bedingung genutzt werden. Im Beispiel wird der Motor gestoppt, wenn die Temperatur kleiner als 20 ist.



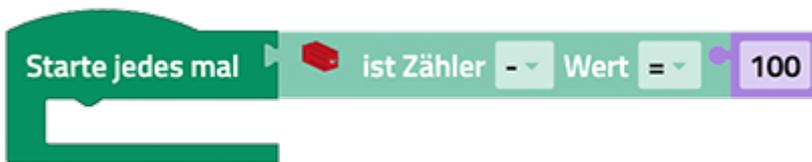
# Zähler



Der Encodermotor kann als Zähler eingesetzt werden. Gezählt wird die Anzahl seiner Umdrehungen, und zwar nicht nur, wenn er sich selbst als Motor dreht, sondern auch, wenn er von außen mechanisch angetrieben wird.

## Der "Starte jedes Mal"-Block

Der "Starte jedes Mal"-Block bietet die Möglichkeit ein Programm ablaufen zulassen, wenn eine Bedingung erfüllt ist. Er funktioniert also ähnlich wie eine Fallunterscheidung, wird aber nicht nur einmal durchlaufen, sondern jedes Mal, wenn die Bedingung erfüllt ist, während des gesamten Ablauf des Programms. Der "Starte jedes Mal"-Block:



Ist eine Abkürzung für folgendes Konstrukt:



Man kann in den "Starte jedes Mal"-Block der Kategorie Zähler alle Bedingungen aus ebendieser Kategorie einsetzen.

**Hinweis:** Der Programmabschnitt innerhalb des "Starte jedes Mal"-Block sollte kurz gehalten werden und keine blockierenden Aufrufe oder Endlosschleifen enthalten, sodass dieser Teil des Programms schnell abgearbeitet werden kann.

## Abrufen

Mit "hole Zähler [] Wert" erhält man den vom Zähler gezählten Wert.

## Abfragen

Um abzufragen, ob der Zähler einen bestimmten Wert gezählt hat, wird dieser Block



genutzt. Über das Dropdown-Menü (kleines Dreieck) kann ausgewählt, wie der gezählte Wert mit dem eingegebenen Wert verglichen werden soll (<, ?, =, ?, ?, >).

## Zurücksetzen

Der Zähler beginnt wieder bei 0, wenn der Block "setze Zähler zurück []" ausgeführt wird.

# I2C

Die in diesem Kapitel beschriebenen I2C-Sensoren werden über ein geeignetes Flachbandkabel mit dem TXT-4.0-Controller verbunden.

## Der "Starte jedes Mal"-Block

Der "Starte jedes Mal"-Block bietet die Möglichkeit ein Programm ablaufen zulassen, wenn eine Bedingung erfüllt ist. Er funktioniert also ähnlich wie eine Fallunterscheidung, wird aber nicht nur einmal durchlaufen, sondern jedes Mal, wenn die Bedingung erfüllt ist, während des gesamten Ablauf des Programms. Der "Starte jedes Mal"-Block:



Ist eine Abkürzung für folgendes Konstrukt:



Man kann in den "Starte jedes Mal"-Block der Kategorie I2C alle Bedingungen aus ebendieser Kategorie einsetzen.

**Hinweis: Der Programmabschnitt innerhalb des "Starte jedes Mal"-Block sollte kurz gehalten werden und keine blockierenden Aufrufe oder Endlosschleifen enthalten, sodass dieser Teil des Programms schnell abgearbeitet werden kann.**

## Gestensensor



Der Gestensensor kombiniert Funktionen zur Erkennung von Handgesten, Farbmessung, Umgebungshelligkeit und Distanzmessung in einem Bauteil.

Aktivieren / Deaktivieren

Der Block "Gestensensor [ ] [ ]" ermöglicht das Ein- und Ausschalten spezifischer Funktionen des Gestensensors. Über das Dropdown-Menü (kleines Dreieck) kann ausgewählt, ob das Licht, die Entfernungsmessung oder die Gestenerkennung aktiviert oder deaktiviert werden soll.



## Abrufen

Der Block "hole Gestensensor [ ] [ ]" ermöglicht es, viele verschiedene Daten vom Gestensensor abzurufen. Je nach Auswahl im Dropdown-Menü können folgende Informationen erhalten werden:

- HEX: Dies gibt den Farbwert im hexadezimalen Format an.
- RGB: Liefert die Intensitätswerte für die roten, grünen und blauen Komponenten einer Farbe. Dies ermöglicht eine detaillierte Analyse der Farbzusammensetzung. Optional kann sich auch auf eine einzelne Farbkomponente konzentriert werden.
- HSV: Bietet eine Darstellung der Farben durch Hue (Farbton), Saturation (Sättigung) und Value (Helligkeitswert). Diese Methode wird oft für eine genauere und intuitivere Farbanalyse verwendet als RGB. Auch hier kann optional eine spezifische Komponente zur detaillierten Analyse gewählt werden.
- Umgebungslicht: Misst die Lichtintensität in der Umgebung des Sensors und gibt sie als numerischen Wert aus.
- Entfernung: Zeigt die gemessene Distanz zu einem Objekt oder einer Oberfläche vor dem Sensor.
- Geste: Dient dazu, die aktuell erkannte Geste von dem Gestensensor abzurufen

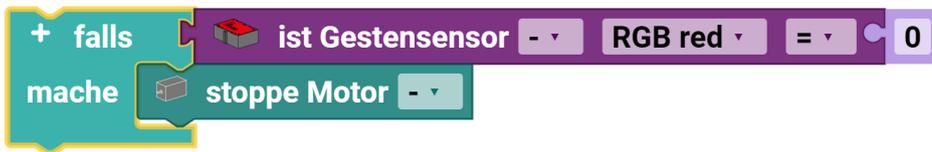


## Abfragen

Der Block "ist Gestensensor [ ] [ ] ...", der als Bedingung dient, übernimmt folgende Funktionen:

- RGB-Werte (Rot, Grün, Blau): Mit der Auswahl einer RGB-Farbe ermöglicht der Block "ist Gestensensor" das Reagieren auf spezifische Farbwerte. Im Dropdown-Menü kann ausgewählt werden, ob Rot-, Grün- oder Blauwerte abgefragt werden. Vergleichsoperatoren (<, =, >) und ein Wert im Zahlenfeld legen fest, ob die Bedingung erfüllt ist, z.B. um bei einem bestimmten Farbwert einen Motor zu stoppen oder eine andere Aktion auszulösen.
- HSV-Werte (Hue, Saturation, Value): Mit der Auswahl einer HSV-Eigenschaft bietet der Block eine alternative Methode zur Farberkennung, die oft für präzisere Farberkennung verwendet wird. Über das Dropdown-Menü kann der Hue-, Saturation- oder Value-Wert eingestellt und verglichen werden, was eine feinere Steuerung ermöglicht.
- Umgebungslicht: Mit der Auswahl von "Umgebungslicht" im Block "ist Gestensensor" kann das allgemeine Lichtniveau gemessen werden. Diese Funktion kann programmatisch genutzt werden, um bei geringer Beleuchtung automatisch Aktionen wie das Ausschalten eines Motors zu veranlassen.
- Entfernung: Durch die Auswahl von "Entfernung" kann bestimmt werden, wie weit ein Objekt entfernt ist. Über das Dropdown-Menü und das Zahlenfeld lassen sich spezifische Abstände einstellen, bei denen das Programm reagieren soll, etwa durch das Anhalten oder Starten von Bewegungen.
- Geste: Prüft, ob die vom Sensor erkannte Geste der angegebenen Geste entspricht.

Das folgende Beispiel zeigt, wie der Block "ist Gestensensor" verwendet wird, um auf einen spezifischen Zustand zu reagieren. Im dargestellten Fall wird geprüft, ob der Rotwert (RGB red) des Gestensensors gleich 0 ist. Ist diese Bedingung erfüllt, führt das Programm eine Aktion aus - hier das Stoppen eines Motors.



# Umweltsensor



Der Umweltsensor vereint die vier Funktionen Luftqualitätssensor, Luftfeuchtigkeitssensor, Barometer und Thermometer in einem Bauteil.

## Luftfeuchtigkeitssensor

### Abrufen

Mit dem Block "hole Umweltsensor [] Luftfeuchtigkeit" kann man die Luftfeuchtigkeit messen.

### Abfragen

Um abzufragen, ob man eine bestimmte Luftfeuchtigkeit misst, wird der Block "ist Umweltsensor [] Luftfeuchtigkeit [] ..." genutzt. Über das Dropdown-Menü (kleines Dreieck) kann ausgewählt, wie die Luftfeuchtigkeit mit dem eingegebenen Wert verglichen werden soll (<, ?, =, ?, ?, >). Der Vergleichswert wird in das Zahlenfeld am Ende des Blocks eingegeben. Dieser Block kann als Bedingung genutzt werden. Im Beispiel wird der Motor gestoppt, wenn die Luftfeuchtigkeit größer als 10 ist.



## Luftqualitätssensor

### Abrufen

Mit dem Block "hole Umweltsensor [] Luftqualität als []" kann man die Luftqualität messen. Über das Dropdown-Menü (kleines Dreieck) kann gewählt werden, ob die Luftqualität als Zahlenwert (von 0 bis 500) oder als Text zurückgegeben werden soll.

### Abfragen

Um abzufragen, ob man eine bestimmte Luftqualität misst, wird der Block "ist Umweltsensor [] Luftqualität [] ..." genutzt. Über das Dropdown-Menü (kleines Dreieck) kann ausgewählt, wie die Luftqualität mit dem

eingeegebenen Wert verglichen werden soll (<, =, >). Der Vergleichswert wird in das Zahlenfeld am Ende des Blocks eingegeben. Dieser Block kann als Bedingung genutzt werden. Im Beispiel wird der Motor gestoppt, wenn die Luftqualität größer als 10 ist.



## Barometer

### Abrufen

Mit dem Block "hole Umweltsensor [] Luftdruck" kann man den Luftdruck messen.

### Abfragen

Um abzufragen, ob man einen bestimmten Luftdruck misst, wird der Block "ist Umweltsensor [] Luftdruck [] ..." genutzt. Über das Dropdown-Menü (kleines Dreieck) kann ausgewählt, wie der Luftdruck mit dem eingegebenen Wert verglichen werden soll (<, =, >). Der Vergleichswert wird in das Zahlenfeld am Ende des Blocks eingegeben. Dieser Block kann als Bedingung genutzt werden. Im Beispiel wird der Motor gestoppt, wenn der Luftdruck größer als 10 ist.



## Thermometer

### Abrufen

Mit dem Block "hole Umweltsensor [] Temperatur" kann man die Temperatur messen.

### Abfragen

Um abzufragen, ob man eine bestimmte Temperatur misst, wird der Block "ist Umweltsensor [] Temperatur [] ..." genutzt. Über das Dropdown-Menü (kleines Dreieck) kann ausgewählt, wie die Temperatur mit dem eingegebenen Wert verglichen werden soll (<, =, >). Der Vergleichswert wird in das Zahlenfeld am Ende des Blocks eingegeben. Dieser Block kann als Bedingung genutzt werden. Im Beispiel wird der Motor gestoppt, wenn die Temperatur größer als 10 ist.



## Kalibrierungsmanagement

### Genauigkeit abrufen

Der Block "hole Umweltsensor [] Genauigkeit" ermöglicht das Abrufen der Genauigkeitsstufe der Luftqualitätsmessung des Umweltsensors. Die Genauigkeit wird auf einer Skala von 0 bis 3 bewertet, wobei jede Zahl eine spezifische Bedeutung hat:

- 0: Dieser Zustand dient als Startzustand. Der Sensor ist bisher nicht stabilisiert.
- 1: Kalibrierung erforderlich. Eine Kalibrierung sollte initiiert werden.
- 2: Kalibrierung läuft. Der Sensor ist aktiv in der Kalibrierungsphase.
- 3: Kalibrierung abgeschlossen. Der Sensor erreicht nun seine beste Leistung bei der Luftqualitätsmessung.

In dem Beispiel wird, wenn die Genauigkeit der vom Umweltsensor gemessenen Daten gleich 3 ist, d.h. der Sensor stabilisiert ist, eine Aktion ausgelöst.



## Prüfen, ob Kalibrierung erforderlich ist

Der Block "ist Kalibrierung von Umweltsensor [] erforderlich" dient spezifisch dazu, zu prüfen, ob die Genauigkeit des Umweltsensors den Wert 1 erreicht hat, was anzeigt, dass eine Kalibrierung erforderlich ist.



## Kalibrierungsprozess starten

Der Block "kalibriere Umweltsensor []" startet den Kalibrierungsprozess des Sensors. Dieser Vorgang soll die Genauigkeit des Sensors verbessern und sicherstellen, dass die gemessenen Umweltdaten verlässlich sind.

# Kombisensor



Der Kombisensor vereint die drei Funktionen Beschleunigungssensor, Gyroskop und Kompassensensor in einem Bauteil.

## Beschleunigungssensor

### Initialisierung

Der Block "Init Accelerometer [] Bereich: [] (g) Bandbreite: [] (Hz) Kompensation: []" dient zur Initialisierung des Beschleunigungssensors mit bestimmten Konfigurationen. Die Einstellungen dieses Blocks legen fest, wie der Sensor Daten erfasst und wie sensibel er auf Bewegungen reagiert. Hier sind die spezifischen Einstellungen erklärt:

- Bereich (2g): Diese Einstellung bestimmt den maximalen Beschleunigungsbereich, den der Sensor erfassen kann. Ein Bereich von 2g bedeutet, dass der Sensor Beschleunigungen bis zum Zweifachen der Erdbeschleunigung messen kann, was etwa 19,62 m/s<sup>2</sup> entspricht.
- Bandbreite (7.81 Hz): Die Bandbreite bestimmt die Frequenz, mit der der Sensor Beschleunigungsdaten erfasst und aktualisiert. Eine Bandbreite von 7.81 Hz bedeutet, dass der Sensor etwa alle 128 Millisekunden (ms) Daten aktualisiert (entsprechend der Umkehrung der Frequenz: 1/7.81 Hz ≈ 128 ms). Diese Einstellung beeinflusst, wie schnell der Sensor auf Änderungen in der Beschleunigung reagiert und ist wichtig für Anwendungen, bei denen es auf die Reaktionsgeschwindigkeit ankommt.
- Kompensation: Durch Aktivieren der Kompensation werden Messungenauigkeiten, die durch Umgebungseinflüsse oder Sensorimperfectionen verursacht werden, automatisch korrigiert, was zu präziseren und zuverlässigeren Daten führt.



Init Accelerometer - Bereich: 2 (g) Bandbreite: 7.81 (Hz) Kompensation: [ ]

## Abrufen

Mit "hole Kombisensor [] Beschleunigung in []" erhält man die Beschleunigung in einer Raumrichtung. Die gewünschte Raumrichtung kann über das Dropdown-Menü (kleines Dreieck) gewählt werden. Die Beschleunigung wird in g angegeben.

## Abfragen

Um abzufragen, ob man eine bestimmte Beschleunigung misst, wird der Block "ist Kombisensor Beschleunigung in [] [] ..." genutzt. Über die Dropdown-Menüs (kleines Dreieck) kann ausgewählt, wie die Beschleunigung mit dem eingegebenen Wert verglichen werden soll (<, =, >) und welche Raumrichtung abgefragt werden soll. Der Vergleichswert wird in das Zahlenfeld am Ende des Blocks eingegeben. Dieser Block kann als Bedingung genutzt werden. Im Beispiel wird der Motor gestoppt, wenn die Beschleunigung in x-Richtung größer als 10 ist.



falls ist Kombisensor Beschleunigung in x-Richtung > 10 mache stoppe Motor gebremst

## Kompassensensor

### Initialisierung

Der Block "Init Magnetometer" initialisiert das Magnetometer mit einer spezifischen Datenaktualisierungsrate. Diese Rate bestimmt, wie oft das Gerät magnetische Felder misst und Daten aktualisiert, in diesem Beispiel 2 Hz, was einer Messung alle 0,5 Sekunden entspricht.



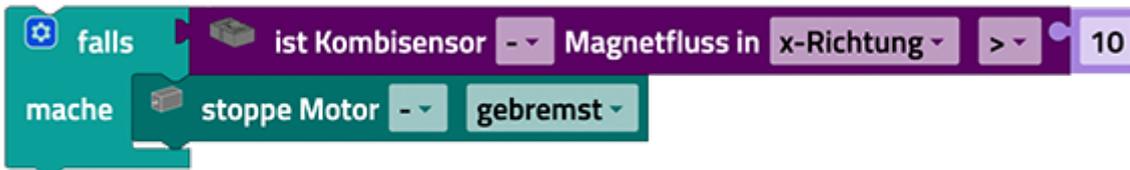
Init Magnetometer - Rate: 2 (Hz)

## Abrufen

Mit "hole Kombisensor Magnetfluss in []" erhält man den magnetischen Fluss in einer Raumrichtung. Die gewünschte Raumrichtung kann über das Dropdown-Menü (kleines Dreieck) gewählt werden. Der magnetische Fluss wird in ?T angegeben.

## Abfragen

Um abzufragen, ob man einen bestimmten magnetischen Fluss misst, wird der Block "ist Kombisensor Magnetfluss in [] [] ..." genutzt. Über die Dropdown-Menüs (kleines Dreieck) kann ausgewählt, wie der magnetische Fluss mit dem eingegebenen Wert verglichen werden soll (<, =, >) und welche Raumrichtung abgefragt werden soll. Der Vergleichswert wird in das Zahlenfeld am Ende des Blocks eingegeben. Dieser Block kann als Bedingung genutzt werden. Im Beispiel wird der Motor gestoppt, wenn der magnetische Fluss in x-Richtung größer als 10 ist.



## Gyroskop

### Initialisierung

Der Block "Init Gyroskop [] Bereich: [] (°/s) Bandbreite: [] (Hz) Kompensation: []" dient zur Initialisierung des Gyroskops mit spezifischen Einstellungen, die festlegen, wie es Drehbewegungen erfasst und darauf reagiert. Hier sind die spezifischen Einstellungen erklärt:

1. Bereich:
  - Diese Einstellung definiert den maximalen Messbereich des Gyroskops. Ein Bereich von 125 Grad pro Sekunde bedeutet, dass das Gyroskop Drehbewegungen bis zu dieser Geschwindigkeit präzise erfassen kann.
2. Bandbreite:
  - Die Bandbreite bestimmt die Frequenz, mit der das Gyroskop seine Messungen aktualisiert. Eine Bandbreite von 12 Hz bedeutet, dass das Gyroskop Daten etwa alle 83 Millisekunden aktualisiert (entsprechend der Umkehrung der Frequenz:  $1/12 \text{ Hz} \approx 83 \text{ ms}$ ). Diese Einstellung beeinflusst die Reaktionsfähigkeit des Gyroskops auf Änderungen in der Drehbewegung.
3. Kompensation:
  - Das Kontrollkästchen für die Kompensation ermöglicht es, Fehler und Abweichungen in den Messungen des Gyroskops automatisch zu korrigieren. Durch das Aktivieren der Kompensation werden Messungenauigkeiten, die durch Umgebungseinflüsse oder Geräteimperfectionen verursacht werden, ausgeglichen, was zu präziseren und zuverlässigeren Daten führt.

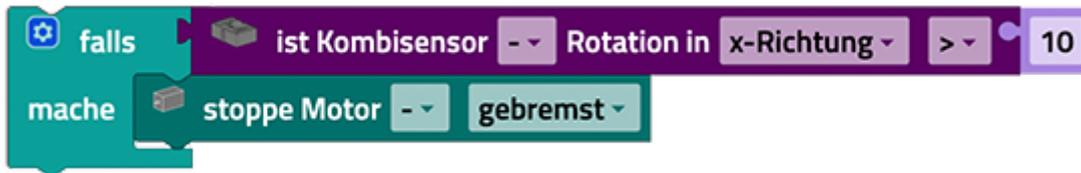


## Abrufen

Mit "hole Kombisensor [] Rotation in []" erhält man die Rotation in einer Raumrichtung. Die gewünschte Raumrichtung kann über das Dropdown-Menü (kleines Dreieck) gewählt werden. Die Rotation wird in %/s angegeben.

## Abfragen

Um abzufragen, ob man eine bestimmte Winkelgeschwindigkeit misst, wird der Block "ist Kombisensor [] Rotation in [] [] ..." genutzt. Über die Dropdown-Menüs (kleines Dreieck) kann ausgewählt, wie die Rotation mit dem eingegebenen Wert verglichen werden soll (<, =, >) und welche Raumrichtung abgefragt werden soll. Der Vergleichswert wird in das Zahlenfeld am Ende des Blocks eingegeben. Dieser Block kann als Bedingung genutzt werden. Im Beispiel wird der Motor gestoppt, wenn die Rotation in x-Richtung größer als 10 ist.



# USB

Über den USB-Anschluss kann die Kamera mit integriertem Mikrofon angeschlossen werden. Kamera und Mikrofon werden hier getrennt betrachtet. Um die Funktionen der Kamera zu nutzen, muss man sie erst im Kamera Konfigurator konfigurieren. Wie man zum Kamerakonfigurator gelangt, wird [hier](#) erklärt.

## Kamera



Die Kamera kann insbesondere als Bewegungsdetektor, als Farbdetektor, als Balldetektor und als Liniendetektor genutzt werden.

## Bewegungsdetektor

Um die Kamera als Bewegungsdetektor zu nutzen, muss man im Kamerakonfigurator das Männchen-Symbol in die gerasterte Fläche ziehen, dann öffnet sich rechts ein Fenster in dem man unter Inspektor

- den Pixelbereich, in dem auf Bewegung überprüft wird,
- die Position dieses Bereichs (auf dem angegebenen Punkt liegt die obere linke Ecke des Bereichs),
- den Namen des Bewegungsdetektors und
- die Toleranz

festlegen kann.

## Bewegungsdetektor Programm

Das Bewegungsdetektor-Programm läuft ab, wenn eine Bewegung erkannt wurde. Es wird separat vom Hauptprogramm geschrieben. Variablen funktionieren global über beide Programme hinweg. Das Eingabe-Programm läuft im Block "wenn Bewegung erkannt" ab.

## Farbdetektor

Um die Kamera als Farbdetektor zu nutzen, muss man im Kamerakonfigurator das Pipetten-Symbol in die gerasterte Fläche ziehen, dann öffnet sich rechts ein Fenster in dem man unter Inspektor

- den Pixelbereich, in dem auf Farbe überprüft wird,
- die Position dieses Bereichs (auf dem angegebenen Punkt liegt die obere linke Ecke des Bereichs),
- den Namen des Farbdetektors und
- den Kontrast

festlegen kann.

## Abrufen

Mit "hole Farbe als []" erhält man die erkannte Farbe im Hexadezimal oder im RGB Format. Das Format kann über das Dropdown-Menü (kleines Dreieck) eingestellt werden.

## Abfragen

Um abzufragen, ob der Detektor eine Farbe wahrgenommen hat, wird der Block "ist Farbe detektiert" genutzt. Dieser Block kann als Bedingung genutzt werden.

Um abzufragen, ob der Detektor eine bestimmte Farbe wahrnimmt, wird dieser Block



genutzt. Mit dem Block kann man die aufgenommene Farbe mit einer eingegeben vergleichen. Über das Dropdown-Menü (kleines Dreieck) kann gewählt werden, ob die eingestellte Farbe gleich oder ungleich der gefilmten Farbe sein soll. Dieser Block kann als Bedingung genutzt werden. Im Beispiel wird der Motor gestoppt, wenn die aufgenommene Farbe Grün ist.



## Farbdetektor Programm

Das Farbdetektor-Programm läuft ab, wenn eine Farbe erkannt wurde. Es wird separat vom Hauptprogramm geschrieben. Variablen funktionieren global über beide Programme hinweg. Das Eingabe-Programm läuft im Block "wenn Farbe erkannt" ab.

## Balldetektor

Um die Kamera als Balldetektor zu nutzen, muss man im Kamerakonfigurator das Bälle-Symbol in die gerasterte Fläche ziehen, dann öffnet sich rechts ein Fenster in dem man unter Inspektor

- den Pixelbereich, in dem auf Bälle überprüft wird,
- die Position dieses Bereichs (auf dem angegebenen Punkt liegt die obere linke Ecke des Bereichs),
- den Namen des Balldetektors,
- den Bereich, in dem der Durchmesser des Balles liegt,
- den Bereich der x-Achse,
- die Farbe des Balles und
- die Farbtoleranz

festlegen kann.

## Abrufen

Mit "hole [] des Balls" erhält man x-Position, y-Position, Radius oder Durchmesser, des Balles.

## Abfragen

Um abzufragen, ob der Detektor einen Ball wahrgenommen hat, wird der Block "ist Ball detektiert" genutzt. Dieser Block kann als Bedingung genutzt werden.

Um abzufragen, ob der Detektor einen Ball mit einer bestimmten x-Position, y-Position, Radius oder Durchmesser wahrnimmt, wird dieser Block



genutzt. Mit dem Block kann man Spezifikationen, des aufgenommenen Balles, mit einem eingegebenen Wert vergleichen. Über die Dropdown-Menüs (kleines Dreieck) kann gewählt werden, was und mit welchem Vergleichsoperator verglichen werden soll. Dieser Block kann als Bedingung genutzt werden. Im Beispiel wird der Motor gestoppt, wenn der Durchmesser, des aufgenommenen Balles, 5 ist.



## Balldetektor Programm

Das Balldetektor-Programm läuft ab, wenn eine Farbe erkannt wurde. Es wird separat vom Hauptprogramm geschrieben. Variablen funktionieren global über beide Programme hinweg. Das Eingabe-Programm läuft im Block "wenn Ball erkannt" ab.

## Linien-detektor

Um die Kamera als Linien-detektor zu nutzen, muss man im Kamerakonfigurator das Symbol mit den Punkten auf einer Linie in die gerasterte Fläche ziehen, dann öffnet sich rechts ein Fenster in dem man unter Inspektor

- den Pixelbereich, in dem auf Linien überprüft wird,
- die Position dieses Bereichs (auf dem angegebenen Punkt liegt die obere linke Ecke des Bereichs),
- den Namen des Linien-detektors,
- die Anzahl an Linien, die erkannt werden sollen, und
- den Bereich, in dem die Breite der Linie(n) liegt

festlegen kann.

## Abrufen

Mit "hole [] der Linie []" erhält man Position oder Breite einer der maximal fünf Linien ab.

Mit "hole Farbe der Linie [] als []" erhält die Farbe einer Linie im Hexadezimal oder im RGB Format ausgegeben lassen. Das Format kann über das Dropdown-Menü (kleines Dreieck) eingestellt werden.

## Abfragen

Um abzufragen, ob der Detektor eine Linie wahrgenommen hat, wird der Block "Ist Linie detektiert" genutzt. Dieser Block kann als Bedingung genutzt werden.

Um abzufragen, ob der Detektor eine Linie mit einer bestimmten Position oder Breite wahrnimmt, wird dieser Block



genutzt. Mit dem Block kann man Spezifikationen, der aufgenommenen Linie(n), mit einem eingegebenen Wert vergleichen. Über die Dropdown-Menüs (kleines Dreieck) kann gewählt werden, was und mit welchem Vergleichsoperator verglichen werden soll. Dieser Block kann als Bedingung genutzt werden. Im Beispiel wird der Motor gestoppt, wenn die Breite, der aufgenommenen Linie, kleiner als 2 ist.



Um abzufragen, ob der Detektor eine Linie mit einer bestimmten Farbe wahrnimmt, wird dieser Block



genutzt. Mit dem Block kann man die aufgenommene Linienfarbe mit einer eingegeben vergleichen. Über das Dropdown-Menü (kleines Dreieck) kann gewählt werden, ob die eingestellte Farbe gleich oder ungleich der gefilmten Farbe seien soll. Dieser Block kann als Bedingung genutzt werden.

## Liniendetektor Programm

Das Liniendetektor-Programm läuft ab, wenn eine oder mehrere Linien erkannt wurden. Es wird separat vom Hauptprogramm geschrieben. Variablen funktionieren global über beide Programme hinweg. Das Eingabe-Programm läuft im Block "wenn Linien erkannt" ab.

## Bild

### Bild holen

Mit dem "Bild von [] holen"-Block kann das neueste Bild in Binärformat von der angeschlossenen Kamera abgerufen werden.



### Bild nach Base64 konvertieren

Der Block "Bild [] nach base64 konvertieren" wandelt ein Bild vom Binärformat in das Base64-Format um. Base64 ist eine Kodierung, die Binärdaten in eine ASCII-String-Repräsentation umwandelt,



### Bild geändert Ereignis

Der Block "wenn Bild ... geändert: Ereignis" reagiert auf ein Ereignis, dass ausgelöst wird, wenn sich das Bild der Kamera ändert.

Im Beispiel wird bei jedem neuen Bild von der Kamera das Bild neu geholt, in Base64 konvertiert und schließlich auf der Anzeige aktualisiert.



## Mikrofon

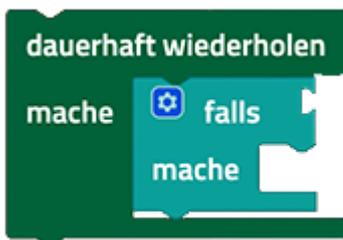
Das in der Kamera integrierte Mikrofon kann als Lautstärkedetektor genutzt werden.

### Der "Starte jedes Mal"-Block

Der "Starte jedes Mal"-Block bietet die Möglichkeit ein Programm ablaufen zulassen, wenn eine Bedingung erfüllt ist. Er funktioniert also ähnlich wie eine Fallunterscheidung, wird aber nicht nur einmal durchlaufen, sondern jedes Mal, wenn die Bedingung erfüllt ist, während des gesamten Ablauf des Programms. Der "Starte jedes Mal"-Block:



Ist eine Abkürzung für folgendes Konstrukt:



Man kann in den "Starte jedes Mal"-Block der Kategorie Mikrofon alle Bedingungen aus ebendieser Kategorie einsetzen.

**Hinweis:** Der Programmabschnitt innerhalb des "Starte jedes Mal"-Blocks sollte kurz gehalten werden und keine blockierenden Aufrufe oder Endlosschleifen enthalten, sodass dieser Teil des Programms schnell abgearbeitet werden kann.

## Lautstärkedetektor

### Abrufen

Mit "Mikrofon [] Lautstärke [] ..." erhält man die Lautstärke in Dezibel.

## Abfragen

Um abzufragen, ob der Lautstärkedetektor eine bestimmte Lautstärke wahrnimmt, wird dieser Block



genutzt. Mit dem Block kann man die aufgenommene Lautstärke mit einer eingegebenen vergleichen. Über das Dropdown-Menü (kleines Dreieck) kann gewählt werden, mit welchem Vergleichsoperator verglichen werden soll. Dieser Block kann als Bedingung genutzt werden.