Name: \_\_\_\_\_\_\_\_\_\_ Klasse: \_\_\_ Datum: \_\_\_\_\_\_\_\_

# Aufgabe 1

# Mecanum-Omniwheels-Fahrzeug mit Hinderniserkennung

## Konstruktionsaufgabe

Baue das Mecanum-Omniwheels-Fahrzeug „Basismodell mit Sensoren“ nach der Bauanleitung. Die Naben der Zahnräder Z20 und der Mecanum-Omniwheels müssen fest angezogen werden, damit beim Fahren kein „Schlupf“ entsteht.

In dieser Aufgabe werden der Spursensor (vorne) und der Ultraschall-Abstandssensor zur Hinder­nis­erkennung verwendet.

Der Ultraschall-Abstandssensor wird an I6 (schwarzes Kabel), die beiden Infrarot-   
(IR-) Sensoren werden an I7 (rechter Sensor, gelb/blaues Kabel) und I8 (linker Sensor, blaues Kabel) angeschlossen. Die IR-Sensoren und der Ultraschallsensor müssen außerdem über den 9V-Spannungsausgang des TXT mit Strom versorgt werden.

Prüfe mit dem Interface-Test, ob die vier Motoren korrekt angeschlossen (vorwärts: Linkslauf, d. h. Drehung gegen den Uhrzeigersinn), die vier Encoder mit den richtigen Zählereingängen verbunden sind (M1: C1, M2: C2, M3: C3 und M4: C4) und die beiden IR-Sensoren die richtigen Werte liefern (weiße Fläche: 1, schwarze Linie: 0).

## Programmieraufgaben

Ein Bild, das Text, Galerie, Szene, Raum enthält.

Automatisch generierte Beschreibung

*Wichtigste Bewegungsarten mit Mecanum-Omniwheels*

1. Synchroner Antrieb in alle Richtungen

Die Abbildung veranschaulicht die Steuerung des Mecanum-Omniwheels-Fahrzeugs. Die oberen drei Bewegungsarten zeigen den für die Geradeausfahrt, die Schrägfahrt und die seitliche Bewegung erforderlichen Radantrieb. Jede der drei Bewegungsarten umfasst die Vorwärts- und die Rückwärtsbewegung bzw. die Bewegung nach links und rechts. Ist der schwarze Pfeil neben dem Rad nach oben gerichtet, soll sich das zugehörige Rad vorwärts drehen.

Damit lassen sich die folgenden acht Bewegungen des Mecanum-Omniwheels-Fahrzeugs unter­scheiden (siehe auch die Animation in [1]):

* Vorwärts
* Rückwärts
* Seitwärts nach links
* Seitwärts nach rechts
* Schräg nach links vorne
* Schräg nach rechts vorne
* Schräg nach links hinten
* Schräg nach rechts hinten

Entwickle zu jeder dieser Bewegungen ein Blockly-Unterprogramm (eine Funktion), an das du die Geschwindigkeit der Motoren als Parameter übergeben kannst.

Die Funktionen werden in den folgenden Aufgaben immer wieder benötigt. Damit werden die Steuerungs­programme übersicht­licher und einfacher zu verstehen. Teste deine Funktionen mit einfachen Beispielprogrammen.

**Beachte**: Bei den Mecanum-Omniwheels ist die Synchronisation der Motoren besonders wichtig (siehe dazu auch Aufgabe 6 des Robotics TXT 4.0 Base Set).

2. Synchrones Drehen

Die unteren beiden Bewegungsarten in der Abbildung zeigen die beiden wichtigsten mit Mecanum-Omniwheels möglichen Drehbewegungen: das Drehen auf der Stelle und das Fahren einer Kurve. Auch darin sind jeweils mehrere Bewegungs­richtun­gen umfasst, wie das Drehen nach links und rechts oder das Abbiegen nach links oder rechts sowie die Vorwärts- und Rückwärtskurvenfahrt.

Insgesamt ergeben sich so sechs weitere Bewegungen:

* Drehen auf der Stelle nach rechts (mit dem Uhrzeigersinn)
* Drehen auf der Stelle nach links (gegen den Uhrzeigersinn)
* Abbiegen nach rechts
* Abbiegen nach links
* Abbiegen rückwärts nach rechts
* Abbiegen rückwärts nach links

Entwickle zu jeder dieser Bewegungen ein Blockly-Unterprogramm (eine Funktion), an das du die Geschwindigkeit der Motoren als Parameter übergibst. Teste deine Funk­tionen mit einfachen Beispielprogrammen.

3. Synchroner Antrieb mit Distanz-Vorgabe

Um das Mecanum-Omniwheels-Fahrzeug präzise zu navigieren benötigst du nun einen Ausschnitt des Befehlssatzes aus den Programmieraufgaben 1 und 2, jeweils zuzüglich einer Distanz-Vorgabe – der Angabe der Zahl der Impulse, um die die Motoren sich drehen sollen.

3a. Ergänze die folgenden sechs Bewegungsfunktionen aus den Programmieraufga­ben 1 und 2 jeweils um eine Distanz-Vorgabe: eine feste Anzahl von Impulsen

* zur Geradeausfahrt (vorwärts/rückwärts),
* zur Seitwärtsfahrt (links/rechts) und
* zum Drehen um die eigene Achse (links: im Uhrzeigersinn, rechts: gegen den Uhrzeigersinn).

3b. Teste diese sechs Funktionen an einer zuvor ausgemessenen Test­strecke und bestimme experimentell die Faktoren (Impulse je cm bzw. Impulse je °) zur Bestim­mung der erforderlichen Impulse aus

* einer in cm angegebenen Distanz bei Geradeausfahrt
* einer in cm angegebenen Distanz bei Seitwärtsfahrt und
* einem in ° angegebenen Drehwinkel

Für den Umrechnungsfaktor bei Geradeausfahrt kannst du (analog Aufgabe 6 des Robotics TXT 4.0 Base Set) als erste Näherung den Umfang eines Mecanum-Omniwheels-Rads messen und daraus den Faktor ableiten.

4. Linienerkennung

Mit dem Spur­sensor soll das Mecanum-Omniwheels-Fahrzeug nun außerdem Begrenzungslinien und Kanten identifizieren und sie umfahren: Wird eine (schwarze) Begrenzungslinie oder ein Abgrund erkannt, soll das Fahrzeug 10 cm zurücksetzen, sich um eine Achteldrehung (45°) von der Kante abwenden und dann die Fahrt fortsetzen.

4a. Zeichne ein entsprechendes Zustands­übergangsdiagramm.

4b. Schreibe nun unter Verwendung deiner Navigationsfunktionen aus den Program­mieraufgaben 1 und 2 ein passendes Blockly-Programm (siehe auch Aufgabe 6 des Robotics TXT 4.0 Base Set).

## Experimentieraufgaben

1. Hinderniserkennung mit Ultraschall

Das Mecanum-Omniwheels-Fahrzeug ist mit einem Ultraschallsensor ausgestattet, der dir den Abstand zu einem Objekt in cm liefert (siehe auch Aufgabe 1 des Robotics TXT 4.0 Base Set).

Schreibe ein Blockly-Programm, das verhindert, dass das Fahrzeug einem Hindernis näher als 15 cm kommt. Erkennt es ein Hindernis, soll es diesem durch eine Seitwärtsfahrt ausweichen, bis der Ultraschallsensor kein Hindernis mehr in weniger als 25 cm Entfernung feststellen kann (siehe auch Aufgabe 6 des Robotics TXT 4.0 Base Set), und dann seine Fahrt fortsetzen.

2. Encoder-Navigation

Auch beim Mecanum-Omniwheels-Fahrzeug kannst du aus den Werten des Encoders die zurück­gelegte Strecke bestimmen. Damit sollst du es nun zu einem vorgegebenen Ziel navigieren, zu dem ihm die Luftlinien-Entfernung in cm vorgegeben ist. Markiere für deine Tests einen 3 m (Luftlinie) entfernten Zielpunkt auf dem Boden.

Verstelle nun den direkten Weg zum Ziel durch zunächst eines, später auch mehrere Hindernisse. Überlege dir eine Strategie, nach der das Fahrzeug den Hindernissen ausweicht und anschließend die Fahrt zu dem vorgegebenen Zielpunkt auf dem dann kürzesten Weg fortsetzt.

2a. Beschreibe deine Strategie mit einer Skizze.

2b. Erweitere dein Blockly-Programm aus Experimentieraufgabe 1 entsprechend.

**Tipp**: Die Drehung des Mecanum-Omniwheels-Fahrzeugs erfolgt um die Fahrzeugmitte. Alle Berechnungen zur Fahrstrecke beziehen sich daher immer auf den Mittelpunkt des Fahrzeugs. Soll die Fahrt an einer Linie vor der Stoßstange beginnen und enden, muss das Fahrzeug zum Schluss wieder in Fahrtrichtung gedreht werden, damit es vor der Ziellinie zum Stehen kommt.

Anlagen

# Mecanum-Omniwheels-Fahrzeug mit Hinderniserkennung

## Erforderliches Material

* PC für Programmentwicklung, lokal oder über Web-Schnittstelle.
* USB-Kabel oder BLE- bzw. WLAN-Verbindung für die Übertragung des Programms auf den TXT4.0.
* Parcours-Bogen mit schwarzer, 2 cm breiter, geschlossener Kreislinie (aus Robotics TXT 4.0 Base Set).

## Weiterführende Informationen

[1] FRC Team 2605 (Bellingham, WA): [*How a Mecanum Drive Works*](https://seamonsters-2605.github.io/archive/mecanum/). github.io

[2] Online-Diagrammeditor zur Erstellung von Zustandsübergangsdiagrammen (Format drawio): <https://www.diagrammeditor.de/>