



# Mit MATRIX Robotern Kurven fahren

Beispiel mit MATRIX Essential Set(MR0001)

Erstellt von Water Xu & MATRIX Robotics am 24.09.2025

# Benötigte Materialien



Zwei PET-Flaschen  
(pro Gruppe)



Panzertape (optional)



MATRIX Mini Essential Set  
V2

# Wie kann man mit einem Roboter Kurven fahren?



# Wie kann man mit einem Roboter Kurven fahren?

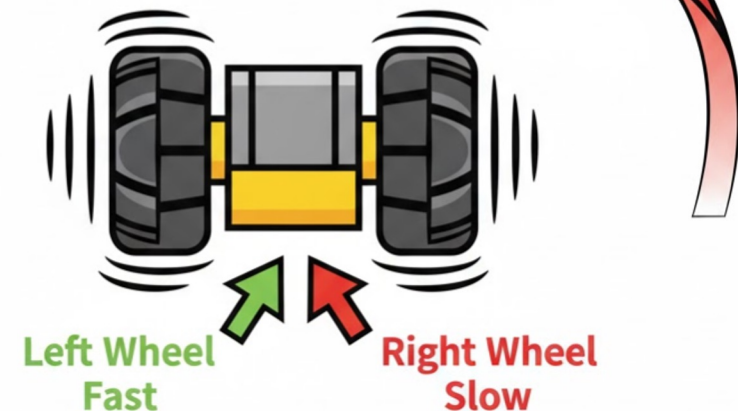
Methode 1: Verändere die Geschwindigkeitsdifferenz der beiden Motoren  
(Differentialsteuerung)

Verändert man die Geschwindigkeiten der Räder,  
fährt man:

- Linkes Rad schnell, rechtes Rad langsam → nach rechts
- rechtes Rad schnell, linkes Rad langsam → nach links

## Differential Steering

Left Wheel Fast + Right Wheel Slow =  
Turn Right



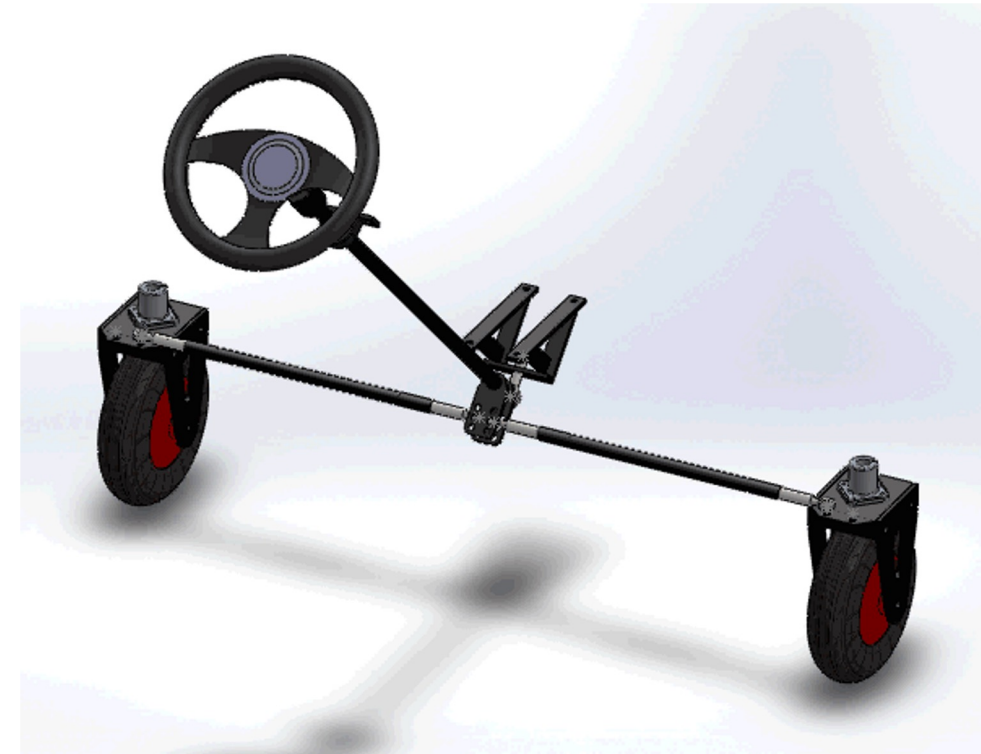
# How can a robot turn left and right?

Methode 2: Verwende eine Lenkachse, um die Räder zu verstellen

Wie bei einem echten Auto, sind die Vorderräder mit einem Servomotor angetrieben.

Dreht sich der Servomotor, verändert sich der Winkel der Räder:

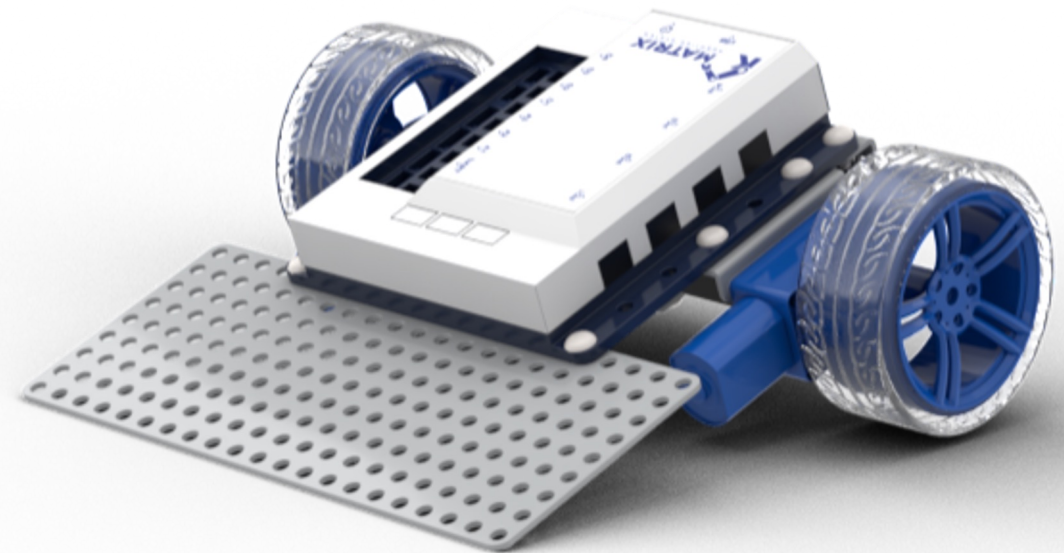
- Vorderräder drehen links → Roboter fährt nach links
- Vorderräder drehen rechts → Roboter fährt nach rechts





# Lernziele:

1. Verstehen, wie unterschiedliche Motorgeschwindigkeiten die Fahrzeugrichtung beeinflussen.
2. Den Roboter vorwärts, rückwärts und drehen lassen können durch einstellen der Motorgeschwindigkeit.
3. Richtige Verwendung von Warteblocks, um die Reihenfolge und das Timing der Bewegungen präzise zu kontrollieren



# Praktisches Beispiel (20 Minuten)

1. Gleiche Motorgeschwindigkeit einstellen, um geradeaus zu fahren.
2. Linke und rechte Motorgeschwindigkeit einstellen um den Roboter zu drehen.
3. Nutzung von Warteblocks, um die Länge der Fahrt einzustellen.
4. Kombination aller Bewegungen für eine kurze Fahrsequenz.

# Verwende das Programm der letzten Lektion

The screenshot displays the MATRIX Robotics System block programming environment. The top bar includes the MATRIX logo, navigation icons, and a file name field showing `*Example6.mbn2`.

The left sidebar lists the following categories and their corresponding block types:

- Mini Core**: Mini Begin
- Serial**: if, then
- Sensing**: if, then
- Extension**: if, then
- Boxs**: else
- 3rd Party**: else
- Sensors**: else
- Control**: repeat, 10
- Operators**: forever
- Variables**: forever

The main workspace shows a program sequence:

- Mini Begin**: 18650x2, UART: On, Baud: 9600
- Setup**:
  - Mini DC Motor M1, Power: 55
  - Mini DC Motor M2, Power: 55
  - wait 3000 millisecond
  - Mini DC Motor M1, Power: 0
  - Mini DC Motor M2, Power: 0
  - wait 100 millisecond
- Loop**: (empty loop body)



**Control**

- Mini Core
- Serial
- Sensing
- Extension
- Boxs
- 3rd Party
- Sensors
- Control
- Operators
- Variables

wait 1000 millisecond

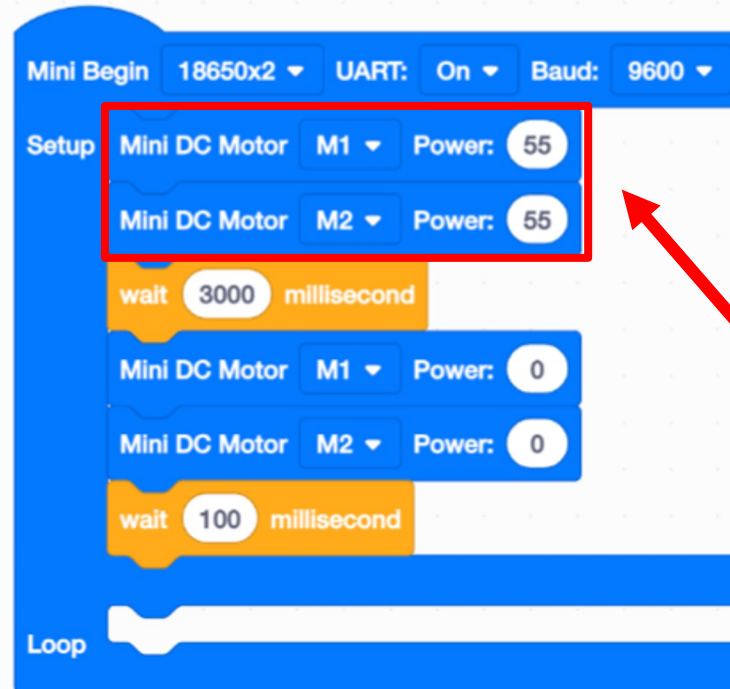
if then

if then


else

repeat 10

forever



Sind beide Geschwindigkeiten gleich, wird der Roboter gerade fahren.



Example6.mbn2

Mini Core

wait 1000 millisecond

Serial

if then

Sensing

if then

Extension Boxes

if then

3rd Party Sensors

else

Control

repeat 10

Operators

Variables

forever

My Blocks

Mini Begin18650x2UART: OnBaud: 9600

Setup

Mini DC MotorM1Power: 55

Mini DC MotorM2Power: 0

wait 3000 millisecond

Mini DC MotorM1Power: 0

Mini DC MotorM2Power: 0

wait 100 millisecond

Loop

```
1 #include "Matri
2
3 void setup()
4 {
5     Mini.begin(LI
6     Serial.begin(
7     Mini.M1.set(5
8     Mini.M2.set(0
9     delay(3000);
10    Mini.M1.set(0
11    Mini.M2.set(0
12    delay(10);
13 }
14
15 void loop(
16 {
17
18 }
```

Linkes Rad schnell, rechtes Rad langsam  
→ Drehung nach rechts

Rechtes Rad schnell, linkes Rad langsam  
→ Drehung nach links

Mini Core

wait 1000 millisecond

Serial

if then

Sensing

if then

Extension

if then

Boxes

else

3rd Party

repeat 10

Sensors

forever

Control

wait until

Operators

Variables

My Blocks

Mini Begin

18650x2

UART: On

Baud: 9600

Setup

Mini DC Motor M1 Power: 0

Mini DC Motor M2 Power: 55

wait 3000 millisecond

Mini DC Motor M1 Power: 0

Mini DC Motor M2 Power: 0

wait 100 millisecond

Loop

Linkes Rad schnell, rechtes Rad langsam

→ Drehung nach rechts

Rechtes Rad schnell, linkes Rad langsam

→ Drehung nach links

```

1  #include "MatrixMini.h"
2
3  void setup()
4  {
5      Mini.begin(LI_2, 0, 9600);
6      Serial.begin(9600);
7      Mini.M1.set(0);
8      Mini.M2.set(55);
9      delay(3000);
10     Mini.M1.set(0);
11     Mini.M2.set(0);
12     delay(100);
13 }
14
15 void loop()
16 {
17 }
18

```

Mini Core

Serial

Sensing

Extension

Boxes

3rd Party

Sensors

Control

Operators

Variables

My Blocks

Mini Core

Mini Begin 18650x2 UART:

Setup

Loop

Mini RGB LED LED1 R: 255

Mini DC Motor M1 Power: 50

Mini Servo Motor RC1 Angle: 90

Mini Button Button1 is Pressed

Mini Ultrasonic D1 Distance: 10

Mini GPIO

Mini D1 Digital Signal

Mini D1 Set to HIGH

Mini Begin 18650x2 UART: On Baud: 9600

Setup

Mini DC Motor M1 Power: 50

Mini DC Motor M2 Power: 50

wait 1000 millisecond

Mini DC Motor M1 Power: 0

Mini DC Motor M2 Power: 0

wait 100 millisecond

Mini DC Motor M1 Power: 50

Mini DC Motor M2 Power: 0

wait 1100 millisecond

Mini DC Motor M1 Power: 0

Mini DC Motor M2 Power: 0

wait 100 millisecond

Mini DC Motor M1 Power: 50

Mini DC Motor M2 Power: 50

wait 1000 millisecond

Mini DC Motor M1 Power: 0

Mini DC Motor M2 Power: 0

wait 100 millisecond

Mini DC Motor M1 Power: 50

Mini DC Motor M2 Power: 50

wait 1000 millisecond

Mini DC Motor M1 Power: 0

Mini DC Motor M2 Power: 0

wait 100 millisecond

Mini DC Motor M1 Power: 50

Mini DC Motor M2 Power: 50

**Versuche, die Motorgeschwindigkeit so anzupassen, dass er die Drehungen und Fahrten macht, wie in der Aufgabe auf der nächsten Folie.**

# Aufgabe für den Unterricht

(20 Minuten )

Versuche ein Programm zu schreiben, mit dem dein Roboter diese Aufgabe erfüllen kann:

Der Roboter startet aus dem Quadrat, berührt beide Flaschen nacheinander und kehrt in das Quadrat zurück.

