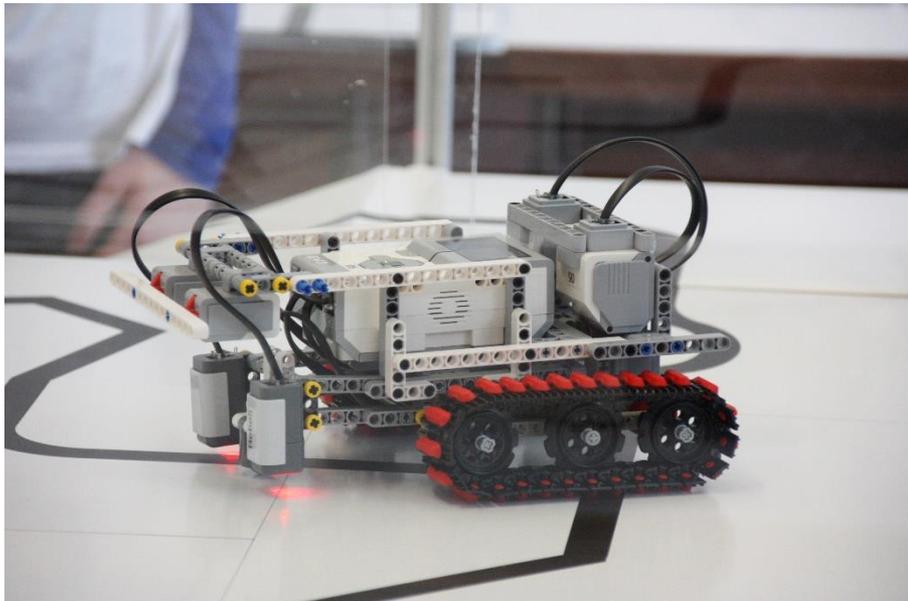


Fakultät für Informatik, Institut für Robotik  
**Laborpraktikum I**  
**Legorobotik – graphische Programmierung**  
Ute Ihme



# DAS LEGO® MINDSTORMS® System

## Das EV3 System



### Prinzip von LEGO® MINDSTORMS®

- Roboter wird gebaut mit
  - programmierbarem LEGO® Stein
  - bis zu 4 Motoren oder Lampen
  - bis zu 4 Sensoren
  - LEGO® TECHNIC Teile
- Erstellung eines Steuerprogramms am Computer
- Übertragen des Programms auf den Roboter
- Testen des Programms



# DAS LEGO® MINDSTORMS® System

## Motoren



Quelle: Lego

Motoren werden an die **Anschlüsse A, B, C und D** angeschlossen.

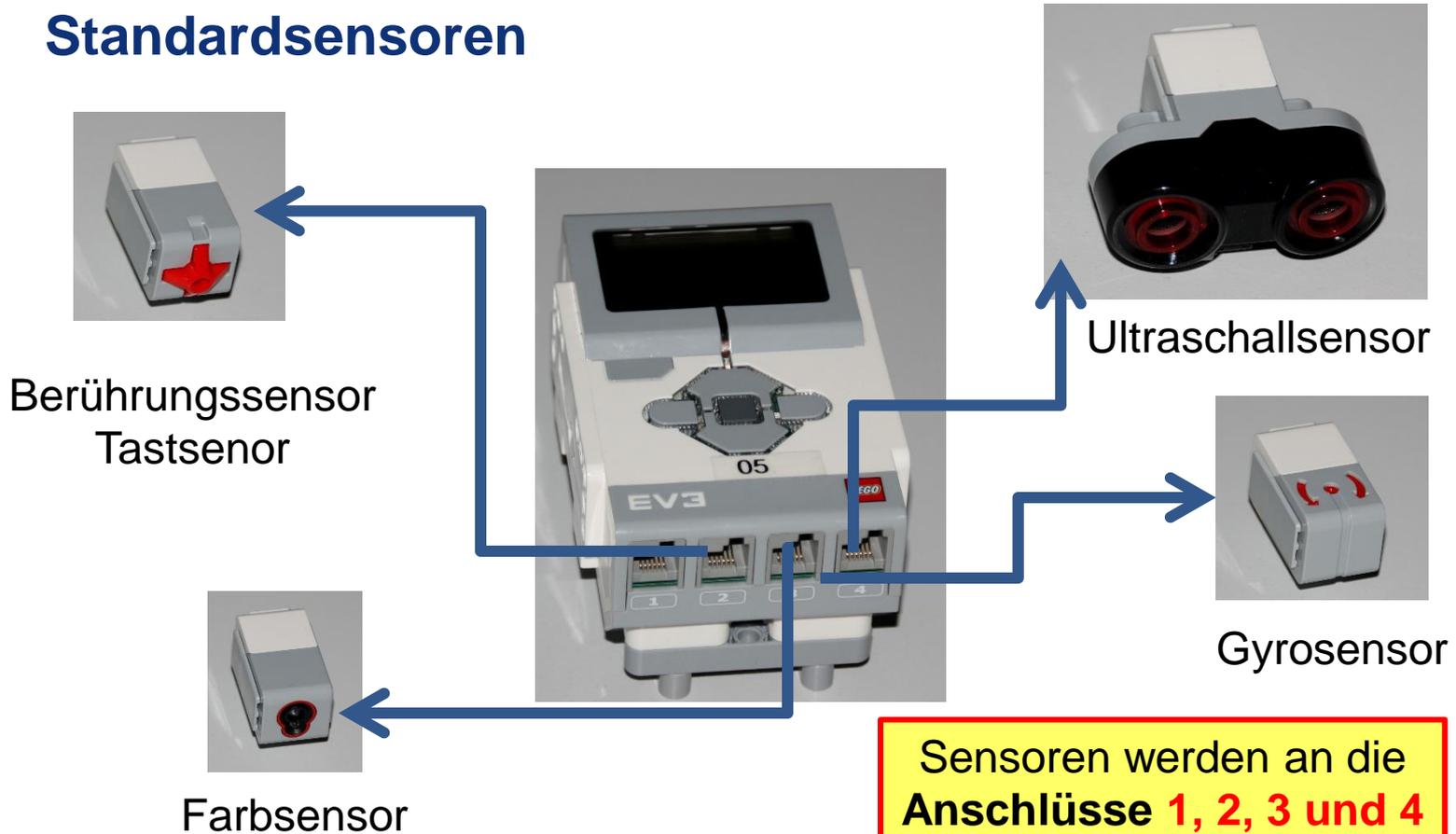
### Servomotor

- Verfügt über integrierten **Rotationssensor**
  - misst Geschwindigkeit und Abstand
  - Leitet Ergebnisse an NXT Stein weiter
- Motor kann auf einen Grad genau gesteuert werden
- Kombinationen mehrerer Motoren möglich
  - arbeiten ggf. mit gleicher Geschwindigkeit



# DAS LEGO® MINDSTORMS® System

## Standardsensoren

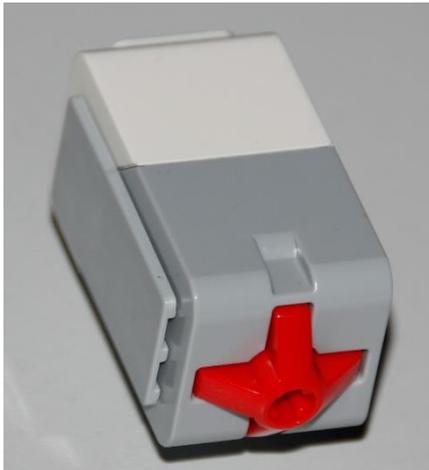


Sensoren werden an die **Anschlüsse 1, 2, 3 und 4** angeschlossen.



# DAS LEGO® MINDSTORMS® System

## Berührungssensor / Tastsensor



- Abfrage, ob Sensor gedrückt
- Werte des Sensors
  - 0: Sensor nicht gedrückt
  - 1: Sensor gedrückt

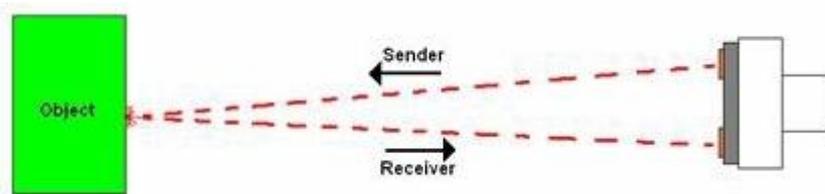


## DAS LEGO® MINDSTORMS® System

### Ultraschallsensor



- Sensor sendet Ultraschall aus
- Schall wird von Hindernis reflektiert
- Reflektierter Schall wird vom Empfänger registriert
- Aus Laufzeit des Schalls kann auf die Entfernung geschlussfolgert werden
- Messbereich: 3 bis 250 cm
- Messgenauigkeit: +/- 1 cm





## DAS LEGO® MINDSTORMS® System

### Colorsensor



- Verfügt über mehrere Moden, z. B.
  - Bestimmung des Farbwertes (ColorID)
  - Bestimmung der reflektierten Helligkeit
- Zur Ausleuchtung kann eine LED eingeschaltet werden

# DAS LEGO® MINDSTORMS® System

## Colorsensor – ColorID Mode



- Bestimmung der Farbe
- Jede Farbe hat einen Wert
- Werte für EV3 Colorsensor

Wert	Farbe
-1	keine
0	Rot
1	Grün
2	Blau
3	Gelb
4	Magenta
5	Orange
6	Weiß
7	Schwarz
8	Pink
9	Grau
10	Hellgrau
11	Dunkelgrau
12	Zyan
13	Braun



## DAS LEGO® MINDSTORMS® System

### Colorsensor – ambient Light Mode



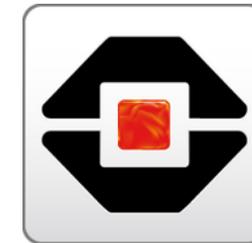
- Messung der Helligkeit mittels Fotodiode
- Helle Fläche reflektiert mehr Licht als dunkle
- Messbereich:
  - 0: dunkel
  - 100: hell
- Zur Ausleuchtung kann eine LED eingeschaltet werden



# Die graphische Programmieroberfläche für EV3

## Erste Schritte

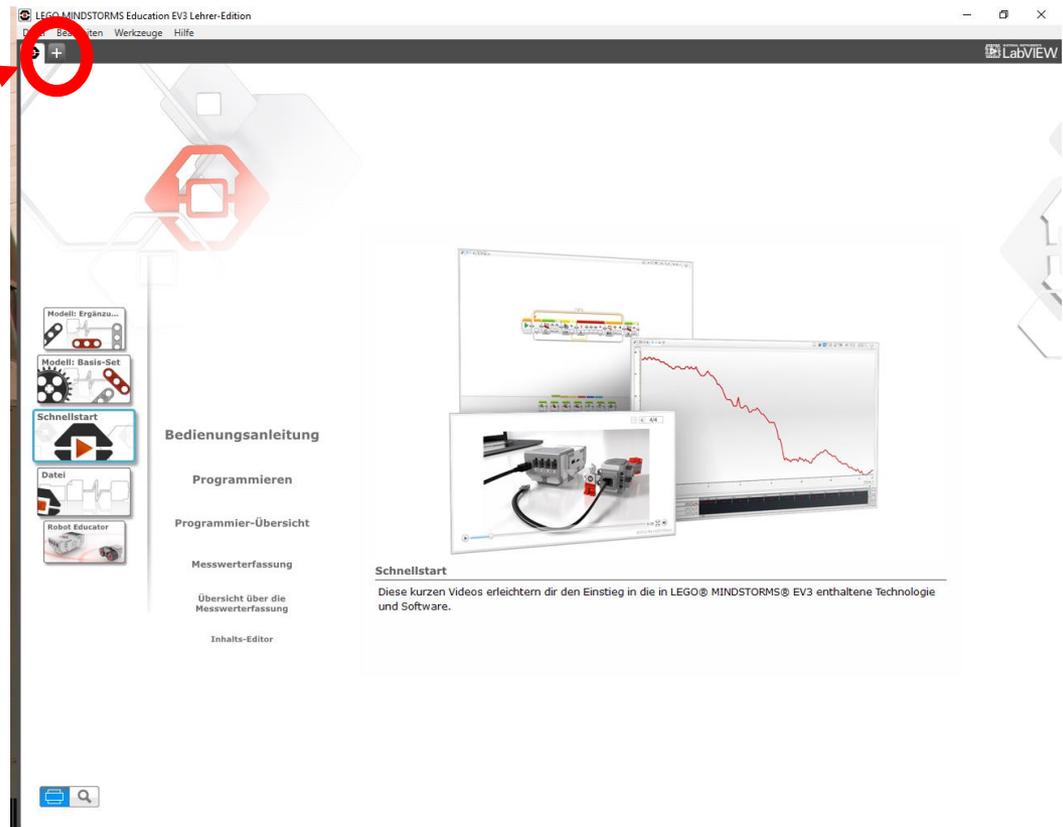
### Starten von Lego Mindstorms Education EV3



# Die graphische Programmieroberfläche für EV3

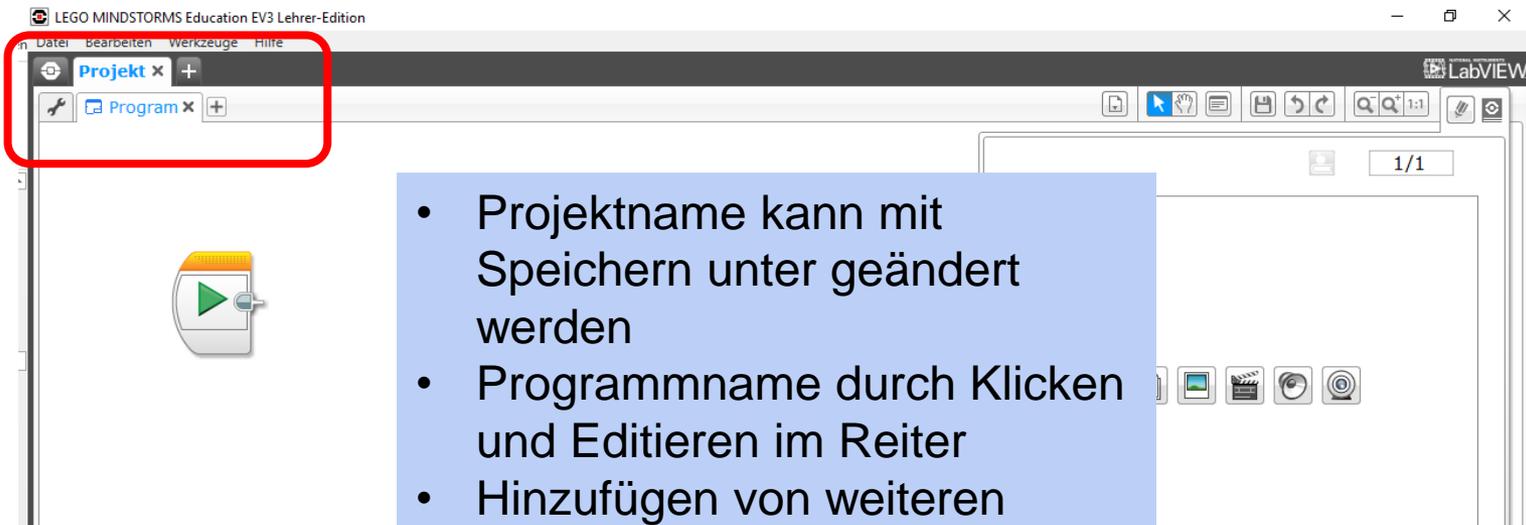
## Erste Schritte

Neues Projekt hinzufügen



# Die graphische Programmieroberfläche für EV3

## Erste Schritte



- Projektname kann mit Speichern unter geändert werden
- Programmname durch Klicken und Editieren im Reiter
- Hinzufügen von weiteren Programmen durch Klicken auf das Plus

# Die graphische Programmieroberfläche für EV3

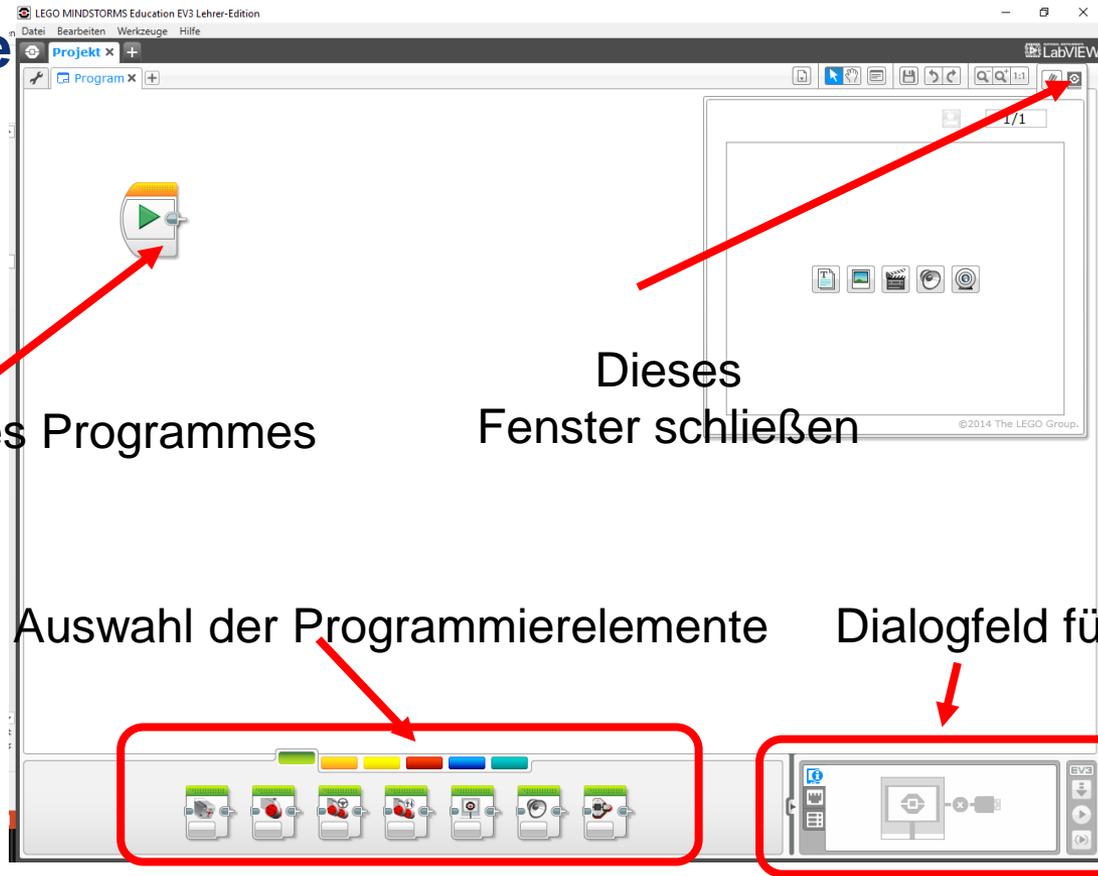
## Erste Schritte

Startelement eines Programmes

Dieses Fenster schließen

Auswahl der Programmierelemente

Dialogfeld für EV3





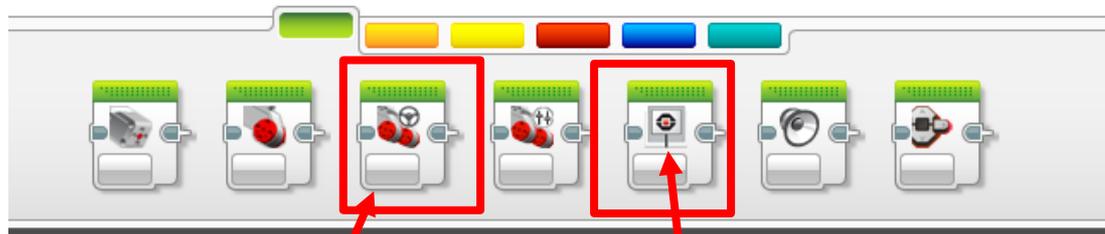
## Hinweise zur Bearbeitung der Praktikumsaufgaben

- Jede Aufgabe des Spielfeldes ist eine eigenständige Aufgabe. D. h. jede Aufgabe kann einzeln gelöst werden und muss nicht mit anderen Aufgaben kombiniert werden.
- Beim Programmieren wird am besten den EV3 immer über USB Kabel mit Rechner verbinden  
→ Ports für Motoren und Sensoren werden automatisch gesetzt



## Graphische Programmierung EV3

### Wichtige Elemente – Grüne Palette



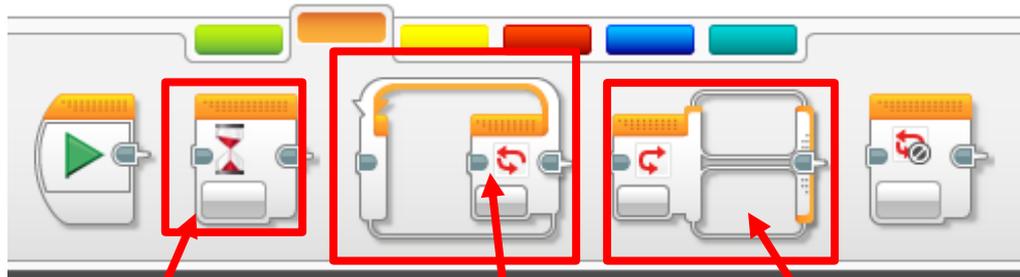
Steuerung zweier Motoren

Bildschirmanzeige



## Graphische Programmierung EV3

### Wichtige Elemente – Orange Palette



Warte-Block

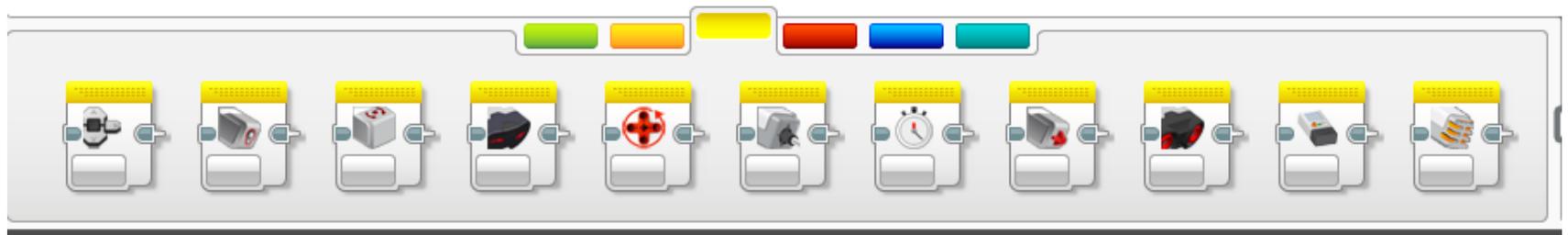
Schleife

Schalter  
(if-else-Block)



## Graphische Programmierung EV3

### Wichtige Elemente – Gelbe Palette

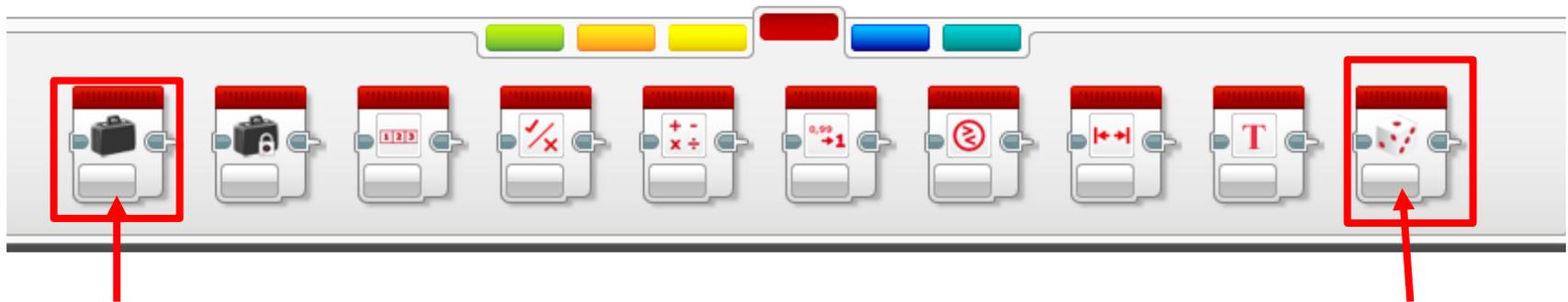


Blöcke werden benötigt zur Abfrage von Sensorwerten



# Graphische Programmierung EV3

## Wichtige Elemente – Rote Palette



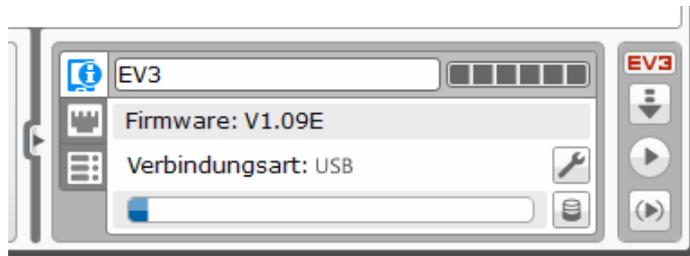
Variablen

Zufallszahl



# Graphische Programmierung EV3

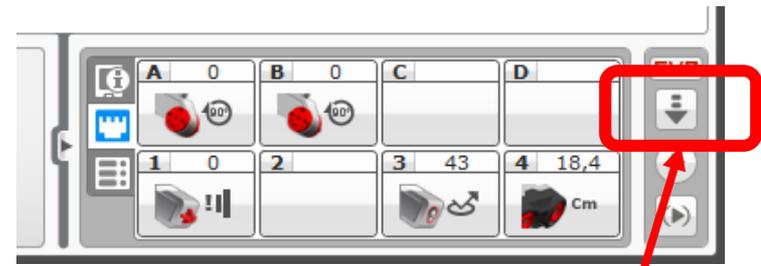
## EV3 Dialogfeld



Systeminformationen



Motoren und Sensorenbelegung



Herunterladen  
des Programms  
auf den EV3-  
Stein



# Graphische Programmierung EV3

## Erste Schritte: Bildschirmanzeigen

The image shows two instances of the 'Hello Word' block in the EV3 graphical programming environment. The top instance is a complete block with a 'Warte-Block' (wait block) attached to its right side. A red arrow points to the 'Warte-Block' with the label 'Warte-Block'. The bottom instance shows the 'Hello Word' block with a context menu open over it. The menu lists options: 'Text', 'Formen', 'Bild', and 'Bildschirm zurücksetzen'. The 'Text' option is selected, and a sub-menu is visible with 'Pixel' and 'Raster' options. A red arrow points from the 'Warte-Block' in the top instance to the 'Hello Word' block in the bottom instance.



## DAS SPIELFELD: Legostadt

### **Aufgabe 1: Fahrt zum Flughafen**

Start: P1

Ende: Flughafenhalle

Der Roboter soll aus P1 zum  
Parkfläche am Flughafen fahren.

Ziel:

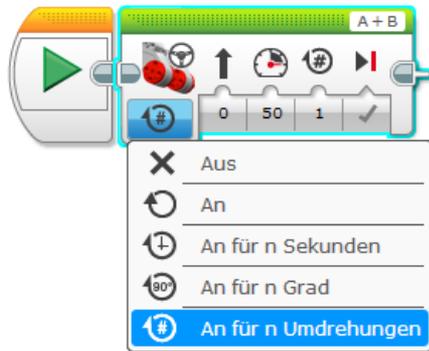
Lernen der Steuerung des Roboters.

- Geradeausfahren
- Kurvenfahren



# DAS SPIELFELD: Legostadt

## Motorsteuerung





## DAS SPIELFELD: Legostadt

### Aufgabe 1: Fahrt zum Flughafen

Start: P1

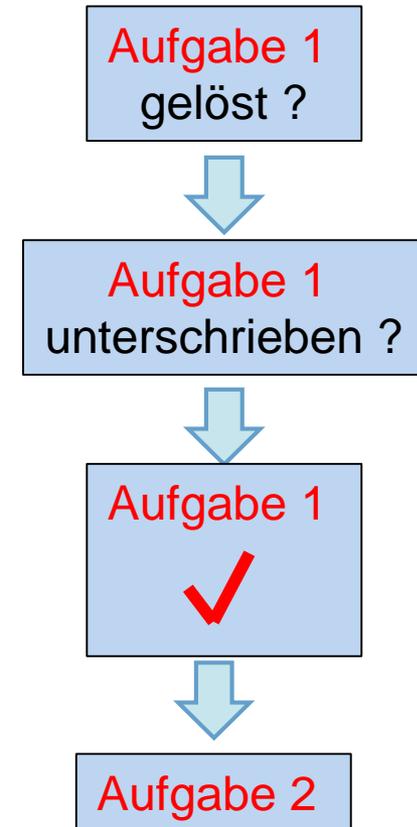
Ende: Flughafenhalle

Der Roboter soll aus P1 zum  
Parkfläche am Flughafen fahren.

Ziel:

Lernen der Steuerung des Roboters.

- Geradeausfahren
- Kurvenfahren





## DAS SPIELFELD: Legostadt

### **Aufgabe 2: Fahrt zum Krankenhaus auf verschiedenen Wegen**

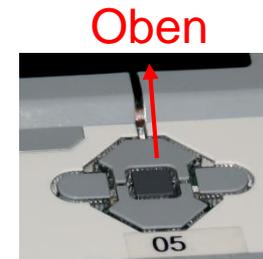
Start: P2

Ende: Parkfläche Krankenhaus

Der Roboter soll von P2 aus über 2 verschiedene Weg zum Krankenhaus fahren. Die Auswahl des Weges ist abhängig vom gedrückten Knopf des EV3 Steines.

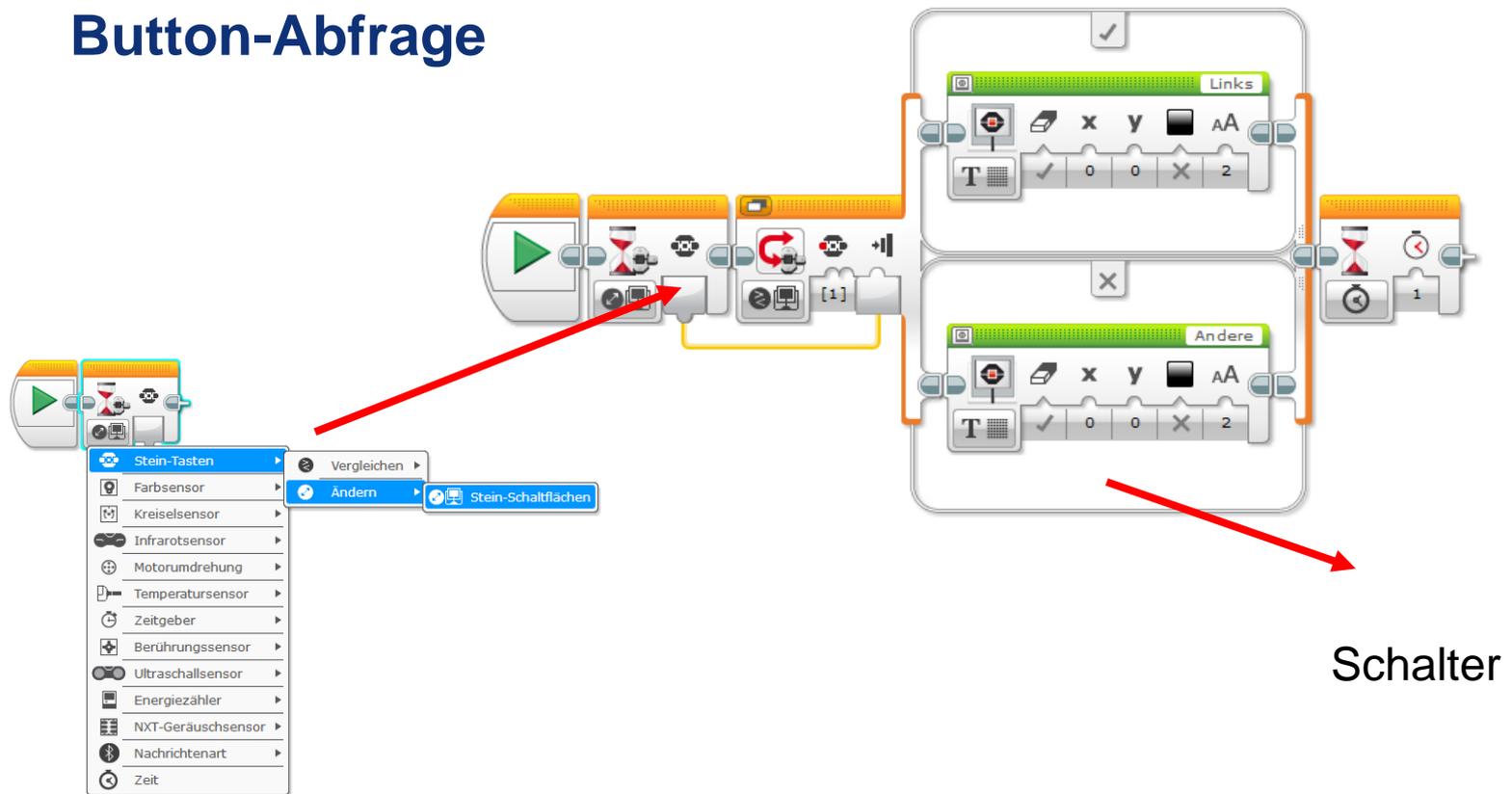
Knopf Oben: über Cafe

alle anderen: über Hotel



# DAS SPIELFELD: Legostadt

## Button-Abfrage





## DAS SPIELFELD: Legostadt

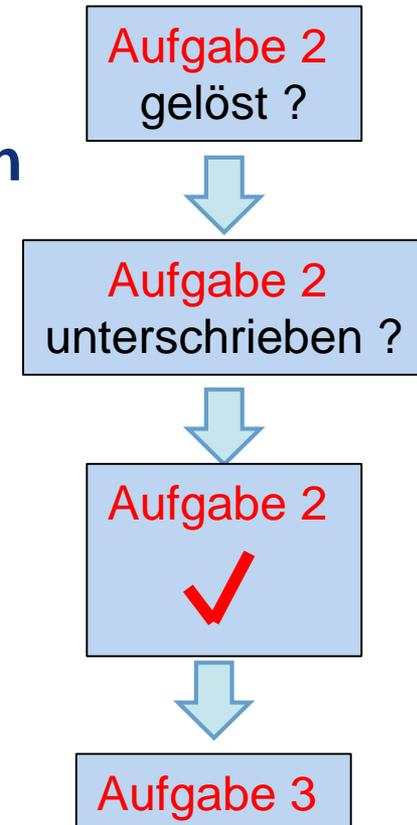
### **Aufgabe 2: Fahrt zum Krankenhaus auf verschiedenen Wegen**

Start: P2

Ende: Parkfläche Krankenhaus

Der Roboter soll von P2 aus über 2  
verschiedene Weg zum Krankenhaus fahren.  
Die Auswahl des Weges ist abhängig vom  
gedrückten Knopf des  
EV3 Steines.

Knopf Oben: über Cafe  
alle anderen: über Hotel





DAS SPIELFELD: Legostadt

## **Aufgabe 3: Beförderung von Fahrgästen zwischen Flughafen und Hotel**

Start und Ende: Parkfläche Flughafen

Der Roboter soll als Shuttlebus Gäste zwischen Flughafen und Hotel hin und zurück befördern. An jedem Ort warten 3 Gäste. Es soll jeweils ein Gast transportiert werden.

Der Roboter startet per Knopfdruck, wenn der Gast eingestiegen ist. Der Roboter fährt die Strecke vom Flughafen zum Hotel vorwärts. Lässt den Gast ein- und aussteigen und fährt nach Knopfdruck die gleiche Strecke rückwärts zurück.

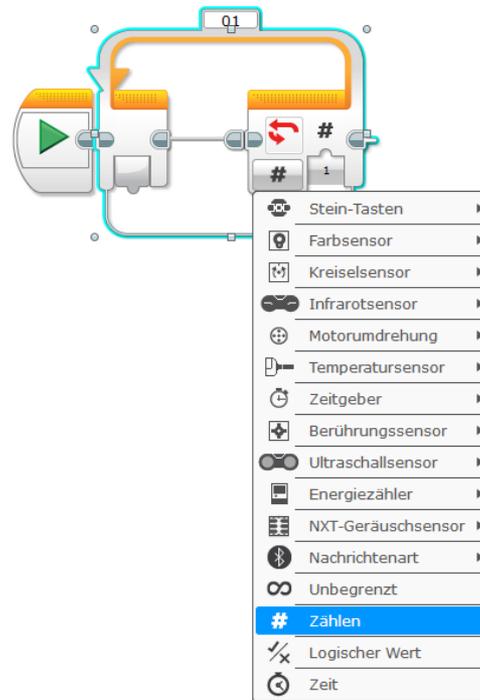
(Realisierung mit einer Schleife)

**Auf den Parkflächen darf der Roboter neu ausgerichtet werden!**



# DAS SPIELFELD: Legostadt

## Die for-Schleife





## DAS SPIELFELD: Legostadt

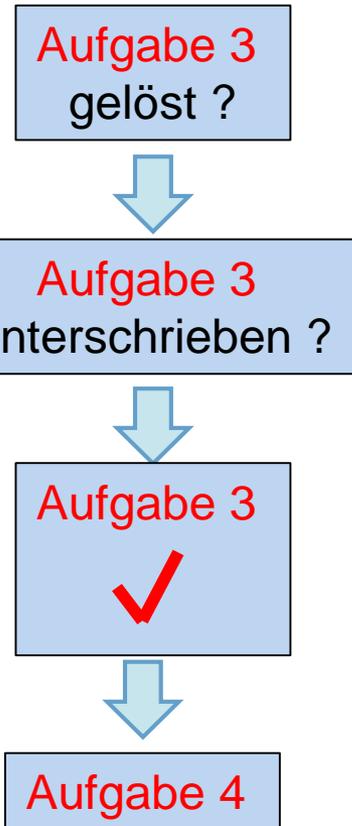
### **Aufgabe 3: Beförderung von Fahrgästen zwischen Flughafen und Hotel**

Start und Ende: Parkfläche Flughafen

Der Roboter soll als Shuttlebus Gäste zwischen Flughafen und Hotel hin und zurück befördern. An jedem Ort warten 3 Gäste. Es soll jeweils ein Gast transportiert werden.

Der Roboter startet per Knopfdruck, wenn der Gast eingestiegen ist. Der Roboter fährt die Strecke vom Flughafen zum Hotel vorwärts. Lässt den Gast ein- und aussteigen und fährt nach Knopfdruck die gleiche Strecke rückwärts zurück.

**Auf den Parkflächen darf der Roboter neu ausgerichtet werden!**





DAS SPIELFELD: Legostadt

## **Aufgabe 4: Einparken mittels Tastsensor**

Start: Parkfläche vor Hotel

Ende: P3

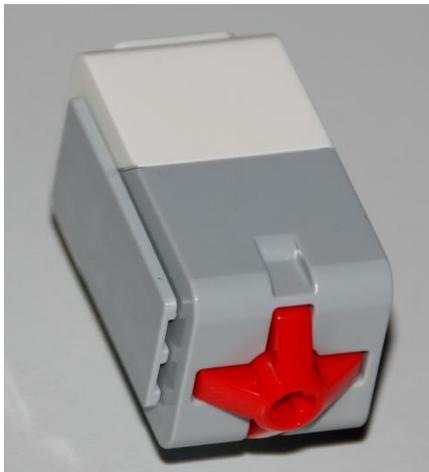
Der Roboter soll rückwärts einparken. Er soll anhalten, wenn der Tastsensor die Bande berührt.

(Realisierung ohne Warte-Block; nur mit Schalter und Schleife)



## DAS SPIELFELD: Legostadt

### **Berührungssensor / Tastsensor**

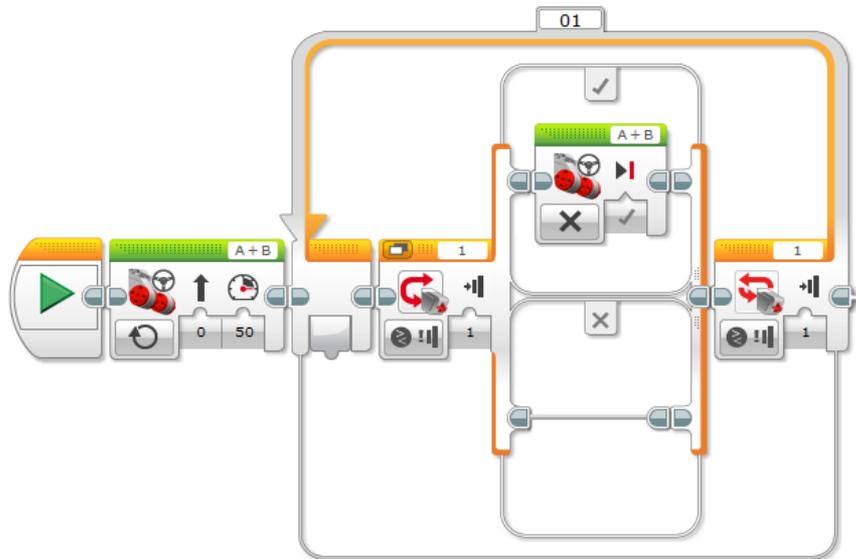


- Abfrage, ob Sensor gedrückt
- Werte des Sensors
  - 0: Sensor nicht gedrückt
  - 1: Sensor gedrückt



## DAS SPIELFELD: Legostadt

### Abfrage Berührungssensor



Analog können alle anderen Sensoren abgefragt werden.



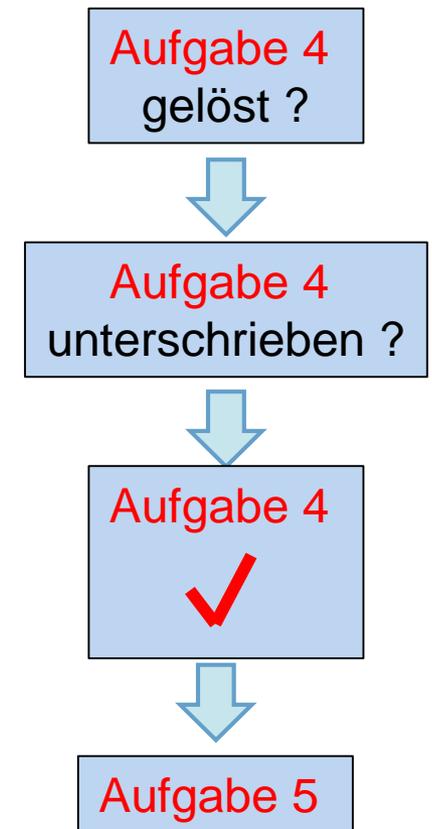
## DAS SPIELFELD: Legostadt

### Aufgabe 4: Einparken mittels Tastsensor

Start: Parkfläche vor Hotel

Ende: P3

Der Roboter soll rückwärts einparken. Er soll anhalten, wenn der Tastsensor die Bande berührt.





DAS SPIELFELD: Legostadt

## **Aufgabe 5: Einparken mittels Ultraschallsensor**

Start: Parkfläche Schule

Ende: P1 – Garage

Der Roboter holt einen Schüler ab. Dabei parkt er selbstständig in die Garage ein. Er soll stehenbleiben, wenn der Abstand zur Wand kleiner als 5 cm ist. Dabei soll der Roboter die aktuelle Entfernung anzeigen.

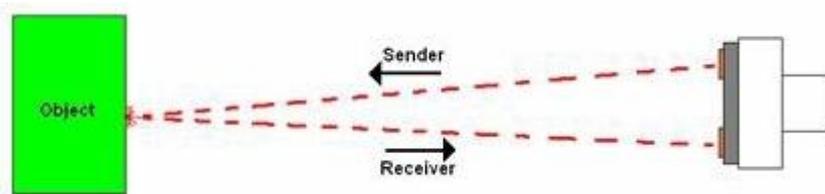


## DAS SPIELFELD: Legostadt

### Ultraschallsensor



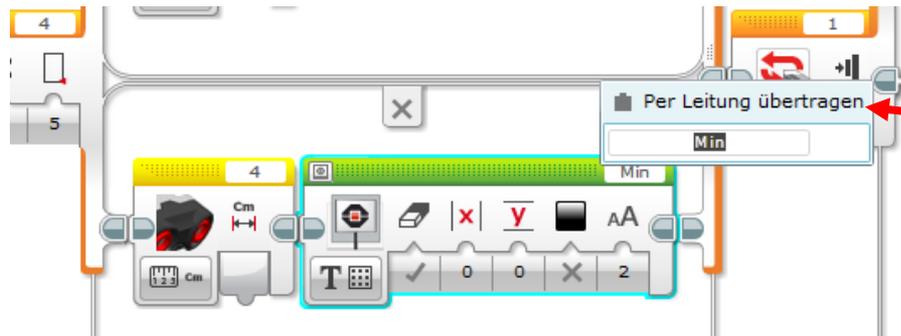
- Sensor sendet Ultraschall aus
- Schall wird von Hindernis reflektiert
- Reflektierter Schall wird vom Empfänger registriert
- Aus Laufzeit des Schalls kann auf die Entfernung geschlussfolgert werden
- Messbereich: 3 bis 250 cm
- Messgenauigkeit: +/- 1 cm



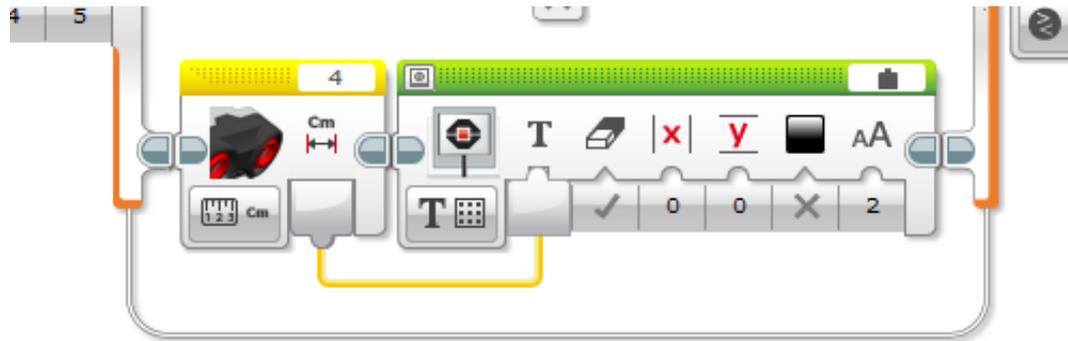


# DAS SPIELFELD: Legostadt

## Abfrage Sensorwerte



Per Leitung übertragen auswählen



Datenleitung ziehen



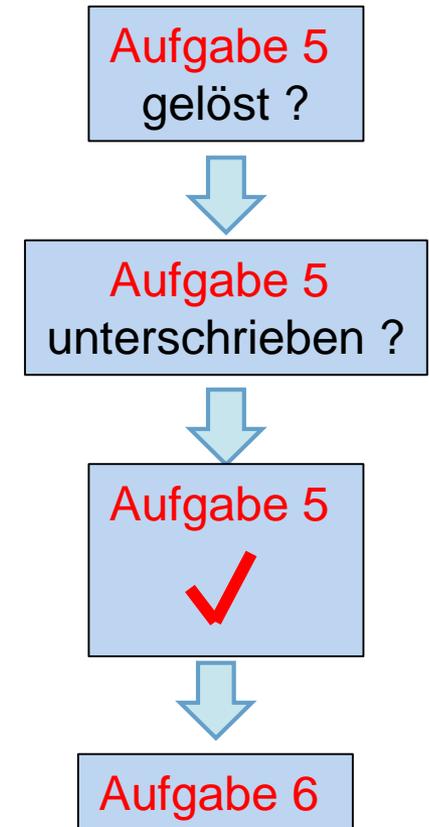
## DAS SPIELFELD: Legostadt

### **Aufgabe 5: Einparken mittels Ultraschallsensor**

Start: Parkfläche Schule

Ende: P1 – Garage

Der Roboter holt einen Schüler ab. Dabei parkt er selbstständig in die Garage ein. Er soll stehenbleiben, wenn der Abstand zur Wand kleiner als 5 cm ist. Dabei soll der Roboter die aktuelle Entfernung anzeigen.





## DAS SPIELFELD: Legostadt

### **Aufgabe 6: Ausflugsziel**

Start: P4

Ende: entsprechendes Farbfeld

Der Roboter soll in Abhängigkeit von ermittelten Farbe am entsprechenden Ausflugsziel anhalten. Das Farbfeld wird über eine Zufallszahl ermittelt (siehe Folie73). Die Zufallszahl soll angezeigt werden.

0 – Gelb (Farb-ID: 3)

1 – Blau (Farb-ID: 2)

2 – Schwarz (Farb-ID: 7)

3 – Rot (Farb-ID: 0)

## DAS SPIELFELD: Legostadt

### Colorsensor – ColorID Mode



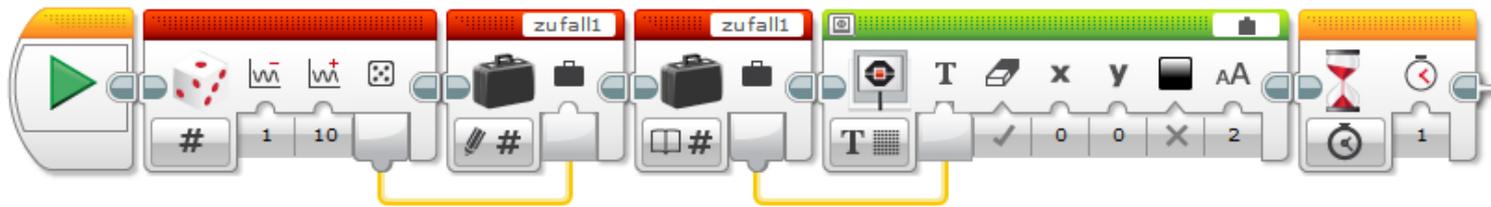
- Bestimmung der Farbe
- Jede Farbe hat einen Wert
- Werte für EV3 Colorsensor

Wert	Farbe
-1	keine
0	Rot
1	Grün
2	Blau
3	Gelb
4	Magenta
5	Orange
6	Weiß
7	Schwarz
8	Pink
9	Grau
10	Hellgrau
11	Dunkelgrau
12	Zyan
13	Braun



# DAS SPIELFELD: Legostadt

## Zufallszahl erzeugen und Arbeit mit Variablen





## DAS SPIELFELD: Legostadt

### Aufgabe 6: Ausflugsziel

Start: P4

Ende: entsprechendes Farbfeld

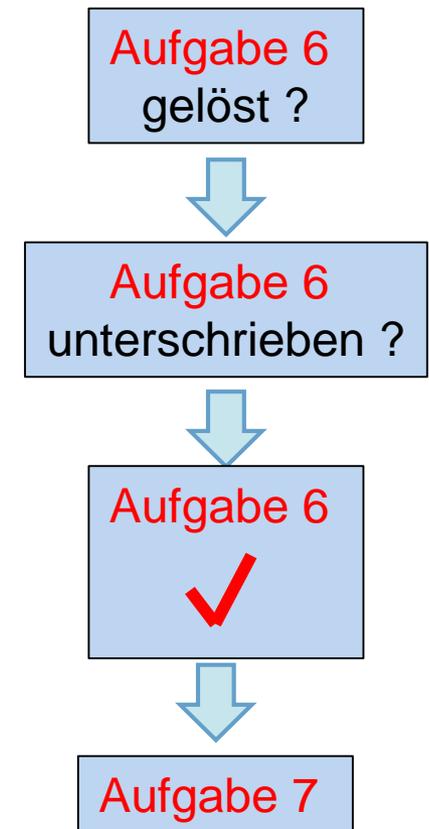
Der Roboter soll in Abhängigkeit von ermittelten Farbe am entsprechenden Ausflugsziel anhalten. Das Farbfeld wird über eine Zufallszahl ermittelt (siehe Folie 73). Die Zufallszahl soll angezeigt werden.

0 – Gelb (Farb-ID: 3)

1 – Blau (Farb-ID: 2)

2 – Schwarz (Farb-ID: 7)

3 – Rot (Farb-ID: 0)





## DAS SPIELFELD: Legostadt

### Aufgabe 7: Folge dem Weg zum Leuchtturm

Start: P3

Ende: Gelbes Feld beim Leuchtturm

Der Roboter soll der schwarzen Linie zum Leuchtturm folgen. Der Roboter soll anhalten, sobald das Endfeld (gelb) erreicht ist.

