

Omniwheels Basismodell

Mit diesem Modell erarbeitest du dir weitere Grundlagen der Programmierung sowie zum Einsatz von vier Omniwheels und deren Funktionsweise.



Im Modell verbaute Aktoren und technisches Zubehör:

Mini Motor	Getriebe	Omniwheelrad
		

Die Erklärung zu den Bauteilen findest du auf der Startseite.

Das Modell „Omniwheels Basismodell“ gliedert sich in 3 Programmieraufgaben:

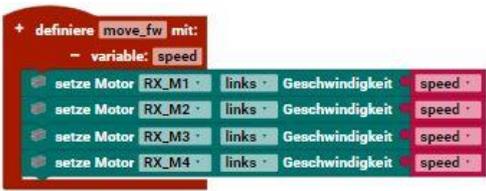
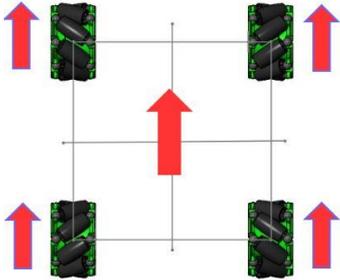
<p>Aufgabe 1</p> <p>Omniwheels_Base_Model_1.ft</p>	<p><u>Programmierlevel 1</u></p> <p>Das Modell fährt 1s gerade aus, 1,4s Sekunden quer nach links, 1s zurück, 1,4s quer nach rechts, dreht sich 3s nach links und dann noch-mals 3s nach rechts. Anschließend stoppt es.</p>
<p>Aufgabe 2</p> <p>Omniwheels_Base_Model_2.ft</p>	<p><u>Programmierlevel 3</u></p> <p>Das Modell soll alle möglichen Fahrtrichtungen abfahren. Dazu sollen für spätere Programmieraufgaben die Fahrtrichtungen in Funktionen geschrieben werden.</p>
<p>Aufgabe 3</p> <p>Omniwheels_Base_Model_3.ft</p>	<p><u>Programmierlevel 3</u></p> <p>Das Modell soll alle möglichen Fahrtrichtungen abfahren. Gesteuert wird das Modell über ein selbst erzeugtes Bedienfeld.</p> 

Aufgabe 1

Da bei den vorherigen Modellen nur 2 Räder verwendet wurden, kannst du leider nicht auf die schon erstellten Unterprogramme zugreifen – du musst einfach neue erstellen.

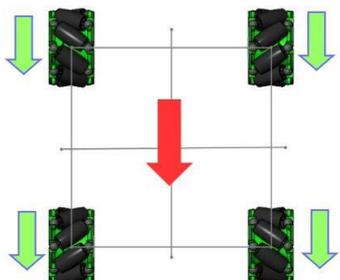
In der Aufgabenbeschreibung bzw. Lösung wird dir erklärt, wie die 4 Omniwheel-Räder drehen müssen, um die gewünschten Bewegungen des Modells durchzuführen. Die **roten Pfeile** stehen für Motordrehung „links“. Die **grünen Pfeile** stehen für Motordrehung „rechts“.

Fahrtrichtung - vorwärts

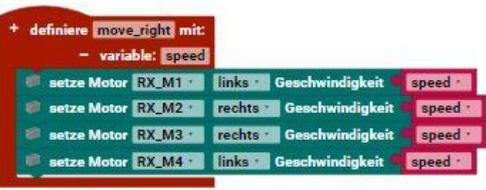
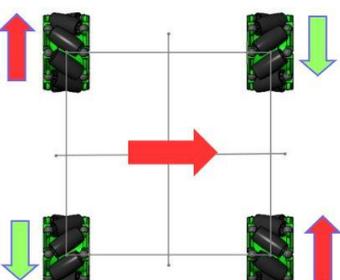
Unterprogramm – move_fw	Räderdrehung
 <pre> + definiere move_fw mit: - variable: speed setze Motor RX_M1 links Geschwindigkeit speed setze Motor RX_M2 links Geschwindigkeit speed setze Motor RX_M3 links Geschwindigkeit speed setze Motor RX_M4 links Geschwindigkeit speed </pre>	

Mit „speed“ greifst du auf eine Variable zurück, die die Motorgeschwindigkeit steuert. Ansonsten sollte die Funktion dir vom Verständnis her keine Probleme bereiten.

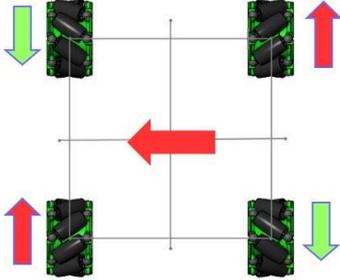
Fahrtrichtung - rückwärts

Unterprogramm – move_bw	Räderdrehung
 <pre> + definiere move_bw mit: - variable: speed setze Motor RX_M1 rechts Geschwindigkeit speed setze Motor RX_M2 rechts Geschwindigkeit speed setze Motor RX_M3 rechts Geschwindigkeit speed setze Motor RX_M4 rechts Geschwindigkeit speed </pre>	

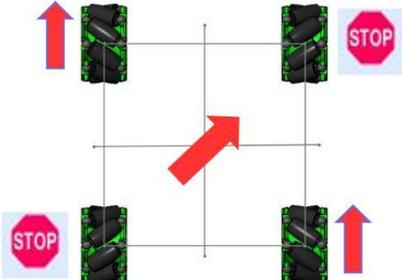
Fahrtrichtung – seitwärts rechts

Unterprogramm – move_right	Räderdrehung
 <pre> + definiere move_right mit: - variable: speed setze Motor RX_M1 links Geschwindigkeit speed setze Motor RX_M2 rechts Geschwindigkeit speed setze Motor RX_M3 rechts Geschwindigkeit speed setze Motor RX_M4 links Geschwindigkeit speed </pre>	

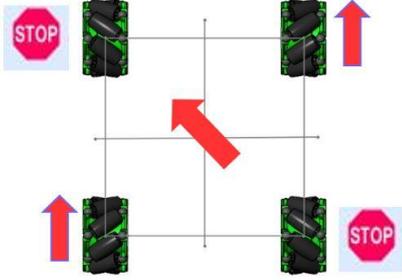
Fahrtrichtung – seitwärts links

Unterprogramm – move_left	Räderdrehung
<pre> + definiere move_left mit: - variable: speed setze Motor RX_M1 rechts Geschwindigkeit speed setze Motor RX_M2 links Geschwindigkeit speed setze Motor RX_M3 links Geschwindigkeit speed setze Motor RX_M4 rechts Geschwindigkeit speed </pre>	

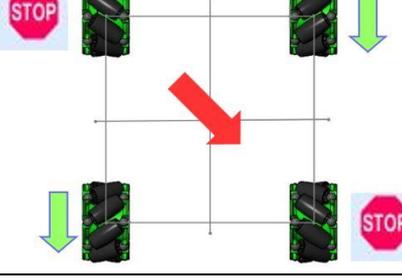
Fahrtrichtung – diagonal vorwärts rechts

Unterprogramm – diagFR	Räderdrehung
<pre> + definiere diagFR mit: - variable: speed setze Motor RX_M2 links Geschwindigkeit speed setze Motor RX_M3 links Geschwindigkeit speed stoppe Motor RX_M4 stoppe Motor RX_M1 </pre>	

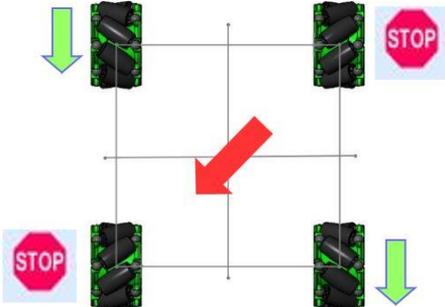
Fahrtrichtung – diagonal vorwärts links

Unterprogramm – diagFL	Räderdrehung
<pre> + definiere diagFL mit: - variable: speed setze Motor RX_M1 links Geschwindigkeit speed setze Motor RX_M4 links Geschwindigkeit speed stoppe Motor RX_M3 stoppe Motor RX_M2 </pre>	

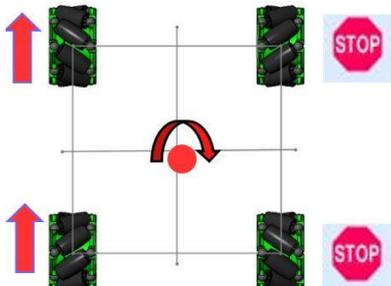
Fahrtrichtung – diagonal rückwärts rechts

Unterprogramm – diagBR	Räderdrehung
<pre> + definiere diagBR setze Motor RX_M2 rechts Geschwindigkeit 300 setze Motor RX_M3 rechts Geschwindigkeit 300 stoppe Motor RX_M1 stoppe Motor RX_M4 </pre>	

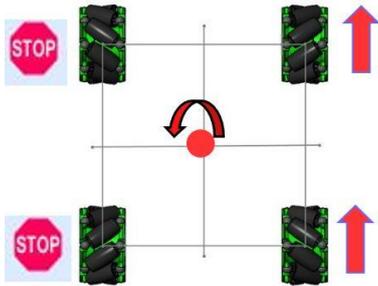
Fahrtrichtung – diagonal rückwärts links

Unterprogramm – diagBL	Räderdrehung
<pre> + definiere diagBL setze Motor RX_M1 links Geschwindigkeit 300 setze Motor RX_M4 rechts Geschwindigkeit 300 stoppe Motor RX_M2 stoppe Motor RX_M3 </pre>	

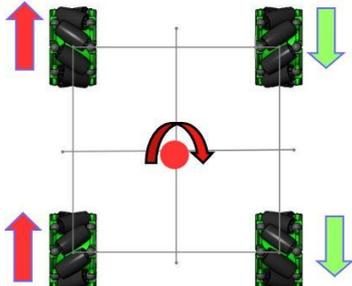
Fahrtrichtung – Kurvenfahrt rechts

Unterprogramm – curve_right_fw	Räderdrehung
<pre> + definiere curve_right_fw setze Motor RX_M1 links Geschwindigkeit 300 setze Motor RX_M3 links Geschwindigkeit 300 stoppe Motor RX_M2 stoppe Motor RX_M4 </pre>	

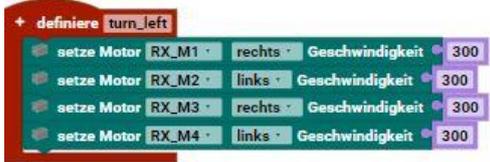
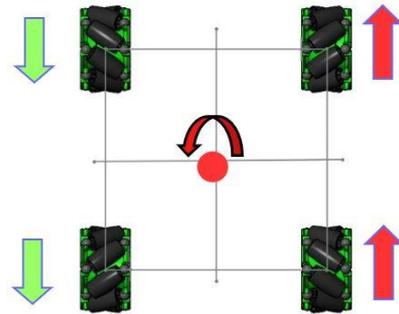
Fahrtrichtung – Kurvenfahrt links

Unterprogramm – curve_left_fw	Räderdrehung
<pre> + definiere curve_left_fw setze Motor RX_M2 links Geschwindigkeit 300 setze Motor RX_M4 links Geschwindigkeit 300 stoppe Motor RX_M1 stoppe Motor RX_M3 </pre>	

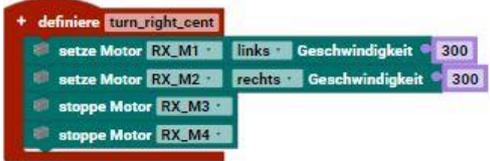
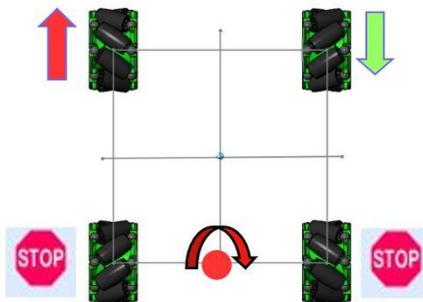
Fahrtrichtung – drehen rechts

Unterprogramm – turn_right	Räderdrehung
<pre> + definiere turn_right setze Motor RX_M1 links Geschwindigkeit 300 setze Motor RX_M2 rechts Geschwindigkeit 300 setze Motor RX_M3 links Geschwindigkeit 300 setze Motor RX_M4 rechts Geschwindigkeit 300 </pre>	

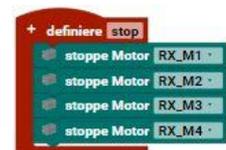
Fahrtrichtung – drehen links

Unterprogramm – turn_left	Räderdrehung
 <pre> + definiere turn_left setze Motor RX_M1 · rechts · Geschwindigkeit · 300 setze Motor RX_M2 · links · Geschwindigkeit · 300 setze Motor RX_M3 · rechts · Geschwindigkeit · 300 setze Motor RX_M4 · links · Geschwindigkeit · 300 </pre>	

Drehung um den Mittelpunkt einer Achse

Unterprogramm – turn_right_cent	Räderdrehung
 <pre> + definiere turn_right_cent setze Motor RX_M1 · links · Geschwindigkeit · 300 setze Motor RX_M2 · rechts · Geschwindigkeit · 300 stoppe Motor RX_M3 · stoppe Motor RX_M4 · </pre>	

Wenn du alle Motoren stoppen möchtest, musst du nur in die Funktion „stop“ viermal den Befehl „stoppe Motor“ einfügen und die Anschlussparameter ändern.



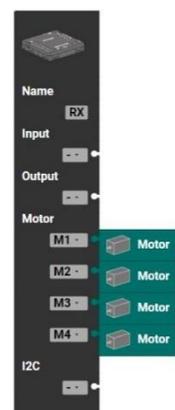
```

+ definiere stop
  stoppe Motor RX_M1 ·
  stoppe Motor RX_M2 ·
  stoppe Motor RX_M3 ·
  stoppe Motor RX_M4 ·
          
```

Jetzt zu den Ausgabenlösungen.

Beginne mit der Controllerkonfiguration. Für die erste Aufgabe benötigst du nur die 4 Antriebsmotoren, die an M1 bis M4 angeschlossen sind.

Wechsle anschließend in das „Hauptprogramm“.




Da das Programm nicht in einer Endlosschleife arbeitet, kannst du diese löschen.

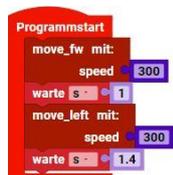
 Erzeuge als Nächstes die Variable „speed“ über „Verarbeitung“ – „Variablen“ – Variable erstellen“.

Beginne mit den benötigten Funktionen für das Hauptprogramm. Diese sind:

„move_fw“, „move_left“, „move_bw“, „move_right“, „turn_left“, „turn_right“ und „stop“.

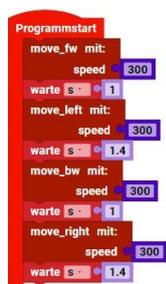
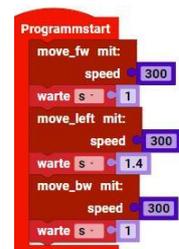
Aus den Vorlagen kannst du die einzelnen Funktionsstrukturen übernehmen.

Welche Funktionen musst du anhand der Aufgabenstellung ins Hauptprogramm einbauen? Da wäre zuerst der Befehl bzw. die Funktion „move_fw“. Als Geschwindigkeit benötigst du den Wert „300“ sowie den Befehl „warte s“ mit dem Wert „1“ für 1 Sekunde.



Anschließend fährt das Modell „1.4“ Sekunden mit der Geschwindigkeit von „300“ quer nach links.

Nach der Linksfahrt soll das Modell „1“ Sekunde zurückfahren - ebenfalls mit der Geschwindigkeit von „300“.

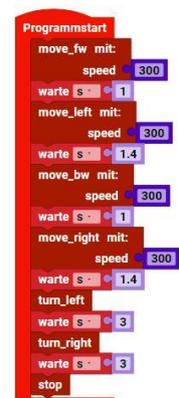


Mit dem nächsten Block fährt das Modell eine Fahrstrecke von „1.4“ Sekunden quer nach rechts.

Die letzten beiden Blöcke sollen das Modell jeweils „3“ Sekunden nach „links“ und dann nach „rechts“ drehen.

Anschließend soll das Modell stoppen.

Danach wird das Programm beendet.



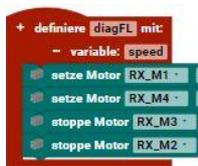
Teste das Programm und speichere es unter dem Namen **Omniwheels_Base_Model_1** auf deinem Rechner ab.

Aufgabe 2

Nun kommst du zur zweiten Aufgabe. Hier erzeugst du ein Programm, ähnlich der ersten Fahrtrichtungsfunktionen. Diese sind:

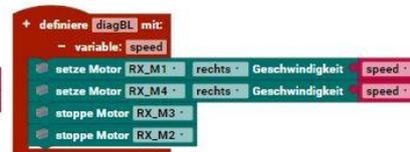
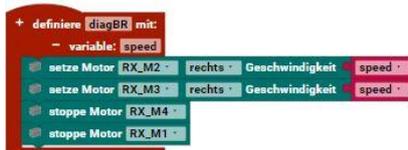
„diagFL“, „diagFR“, „diagBL“, „diagBR“.

Erzeuge zuerst die Funktion „diagFL“. Hier soll das Modell diagonal nach vorne und nach links fahren. Die Geschwindigkeit wird auch in dieser Aufgabe durch die Variable „speed“ festgelegt.



Die Programmstruktur kannst du aus den Grundlagen zum Beginn des Dokuments entnehmen.

Erstelle anschließend die drei weiteren Funktionsblöcke zur Querfahrt. Dazu duplizierst du die Funktion 3 mal und änderst deren Namen um. Im zweiten Schritt musst du die Angaben zu den Motoren ändern. Die Änderungen kannst du aus den nachfolgenden Darstellungen entnehmen.

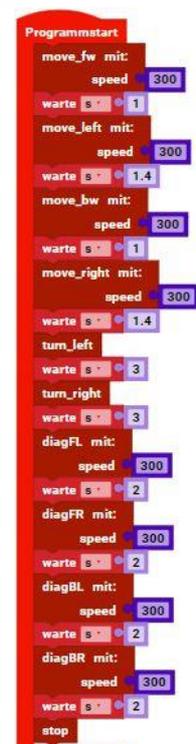


Somit sind alle Funktionen erstellt und du kannst mit dem Hauptprogramm beginnen.

Die nebenstehende Programmstruktur ist nur ein Beispiel. Du kannst natürlich dein Modell auch anders steuern. Ändere z.B. die Geschwindigkeit mit denen das Modell fährt.

Probiere einfach verschiedene Möglichkeiten aus.

Speichere es unter dem Namen **Omniwheels_Base_Model_2** auf deinem Rechner ab.



Aufgabe 3

In dieser Aufgabe erstellst du ein Bedienfeld, mit dem du das Modell vom Rechner aus steuern kannst.

Da du auf die Fahrbefehle in Aufgabe 2 zugreifen musst bzw. benötigst, speichere die Aufgabe 2 unter **Omniwheels_Base_Model_3** auf deinem Rechner ab.

Folgende Funktionen benötigst du für das neue Programm:

„move_fw“, „move_bw“, „move_left“, „move_right“, „turn_right“, „turn_left“, „stop“, „diagFL“, „diagFR“, „diagBL“, „diagBR“.

Lösche aus dem Programmstart alle Befehle.



Bevor du mit dem Hauptprogramm und seinen Funktionen beginnst, hier die Erklärung zur Erstellung eines Bedienfeldes.

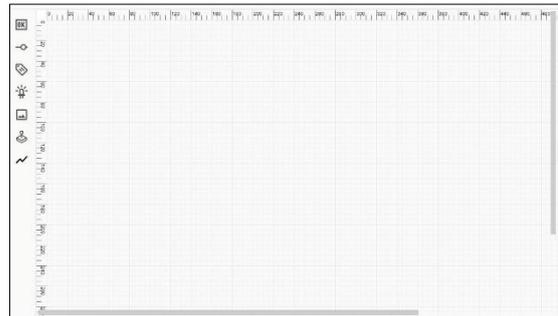


Zuerst musst du im Projektfenster auf „“ klicken. Es öffnet sich ein Kontextfenster aus dem du „Bedienfeld“ auswählst und mit „ERSTELLEN“ bestätigst.



Es öffnet sich ein Bedienfeld-konfigurationsfenster in dem du das Bedienfeld mit verschiedenen Bedienelementen aufbauen kannst.

Neben den Befehlen bzw. Elementen besitzt das Fenster noch eine X/Y-Einteilung. Der Gesamtbildschirm hat eine Einteilung von 640x360 px.



Nun zu den einzelnen Elementen:



Schaltfläche oder „remote_button“, bei Klick wird ein Ereignis ausgelöst

Schieberegler oder „remote_slider“, hier können Werte eingestellt werden

Textausgabe oder „remote_label“, hier kann z.B. ein Infotext dargestellt werden

Virtuelle Lampe oder „remote_status_indikator, opt. Darstellung Schalter ein/aus

Bilddarstellung oder „remote_image“, hier kann ein Bild dargestellt werden

Steuerung mit einem virtuellen Joystick oder „remote_joystick“

Diagramm oder „remote_chart“, anzeigen von Werten in einem Diagramm

Die einzelnen Befehlsicons werden wie auch die Programmbefehle aus dem Menü in den Bildschirm gezogen.

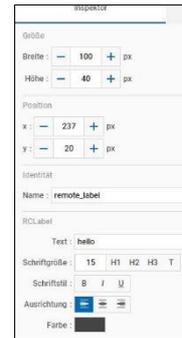
Ziehe zuerst das „remote_button“ für eine Überschrift in den Bildschirm.

Wichtig: Zur Platzierung verwendest du die Einträge im Inspektor. Hier legst du die Größe, die Position, eine ID sowie das Aussehen des Button-Inhaltes fest.

Lege zuerst den auszugebenden Text fest „Steuerprogramm“. Überschreibe Schriftgröße von „40“ und den Schriftstil „B“ für fett.



Ändere die ID von „remote_label“ auf „Überschrift“ um. Lege anschließend noch die Position fest – „X“ auf „210“ und „y“ auf „20“ fest. Zum Schluss änderst du noch die „Größe“ der Darstellung ab auf „Breite 215“ und „Höhe 40“.



Auf dem Bedienfeldkonfigurationsbildschirm wird die Einstellung dargestellt.

Würde das Programm zum Controller übertragen, wird folgende Darstellung angezeigt.



Weitere Elemente, die du benötigst, werde ich im Laufe der Programmerstellung erklären.

Wichtig: Platziere alle Bewegungsrichtungen, die du mit dem Bedienfeld steuern möchtest, in die Bedienfeldkonfiguration.

Koordinaten, Aussehen und Inhalte:

RC-Button

Text	Name	x	y	Breite	Höhe	Schriftgr.	Ausricht.
turn left	bt_turn_left	60	20	100	40	22	
turn right	bt_turn_right	220	20	100	40	22	
forward left	bt_fw_left	20	80	100	40	22	
forward	bt_fw	150	80	100	40	22	
forward right	bt_fw_right	280	80	100	40	22	
left	bt_left	20	140	100	40	22	
right	bt_right	280	140	100	40	22	
backward left	bt_bw_left	20	200	100	40	22	
backward	bt_bw	150	200	100	40	22	
backward right	bt_bw_right	280	200	100	40	22	

RC Label

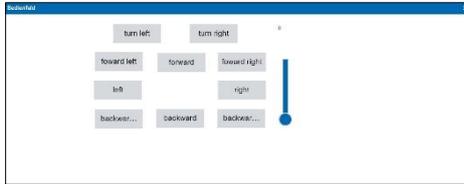
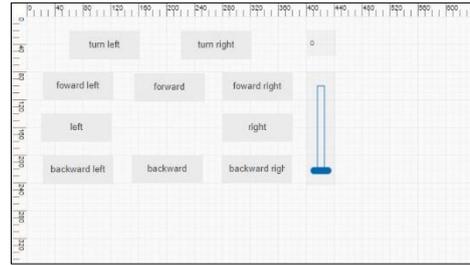
0	remote_label	400	20	40	35	22	
---	--------------	-----	----	----	----	----	--

RC Slider

Test	Name	x	y	Breite	Höhe	Ausricht.	Von-Bis
	Slider_speed	280	200	40	160	vertikal	0,0,512

Das erzeugte Bedienfeld sollte folgendes Aussehen haben.

Lösche hier die Überschrift „Steuerprogramm“.



Startest du das Programm wird dir das Bedienfeld wie folgt dargestellt.



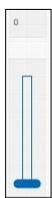
So, das wäre erledigt und du kannst das Hauptprogramm bzw. den Programmstart definieren.



Setze zuerst die Variable „speed“ auf „0“. Anschließend fügst du den Befehl „setze Beschriftungsfeld ... Text“ aus dem Block „Kommunikation“ – „Fernbedienung“ ein.

Füge für „Text“ die Variable „speed“ ein.

Somit ist die Definition des Programmstarts abgeschlossen.



Baue als nächstes eine Abfrage des Schiebereglers ins Programm ein. Der eingestellte Wert soll im „remote_label“ angezeigt werden und dient als Variable für die Geschwindigkeit „speed“.



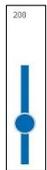
Dazu benötigst du den Befehl „wenn Schieberegler ... bewegt: Ereignis“ aus dem Block „Kommunikation“ – „Fernbedienung“ ein.



Ziehe den Befehl „setze ... auf“ in den Freiraum und ergänze den Befehl mit „Ereignis ...“ aus dem Block „Ereignis Schaltfläche“ – „Kommunikation“ – „Fernbedienung“.



Anschließend wird das Beschriftungsfeld „remote_label ... Text“ auf den Wert von „speed“ festgelegt. Dazu baust du den Befehl „setze Beschriftungsfeld ... Text“ ein und erweiterst ihn mit der Variable „speed“.



Wichtig: Der eingestellte Variablenwert „speed“ wird in allen Programmteilen mit dem gleichen Wert verarbeitet.

Erstelle jeweils eine Abfrage zur Schaltfläche „bt_fw“ und „bt_bw“.

Der Übersicht halber, habe ich die Darstellung so gewählt, dass beide Definitionen immer nebeneinander stehen. Die ersten Beiden erkläre ich kurz.

Aus dem Block „Kommunikation“ – „Fernbedienung“ ziehst du zweimal den Befehl „wenn Schaltfläche ... angeklickt“. Ändere ihn auf „bt_fw“ und „bt_bw“ um.



In den freien Bereich fügst du den Befehl „falls mache sonst“ ein. Ergänze die Abfrage mit dem Befehl „Ereignis ...“ aus dem Block „Kommunikation“ – „Fernbedienung“ ein.



Ersetze „mache“ mit der Funktion „move_fw“ und „move_bw“. Ersetze „sonst“ mit der Funktion „stop“.

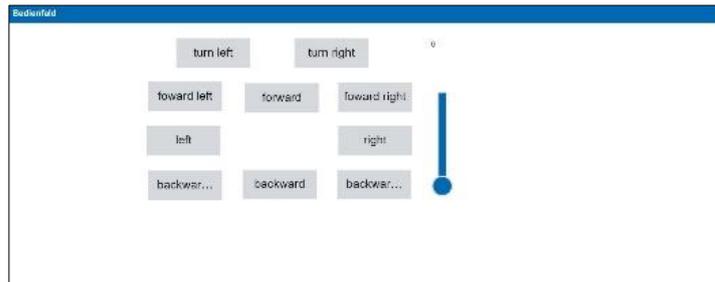


Füge jetzt noch die fehlenden Funktionen bzw. Auswertungen der Schaltflächen hinzu. Dazu duplizierst du einfach „bt_fw“ und „bt_bw“ und änderst die Einträge entsprechend der Bilddarstellungen ab.

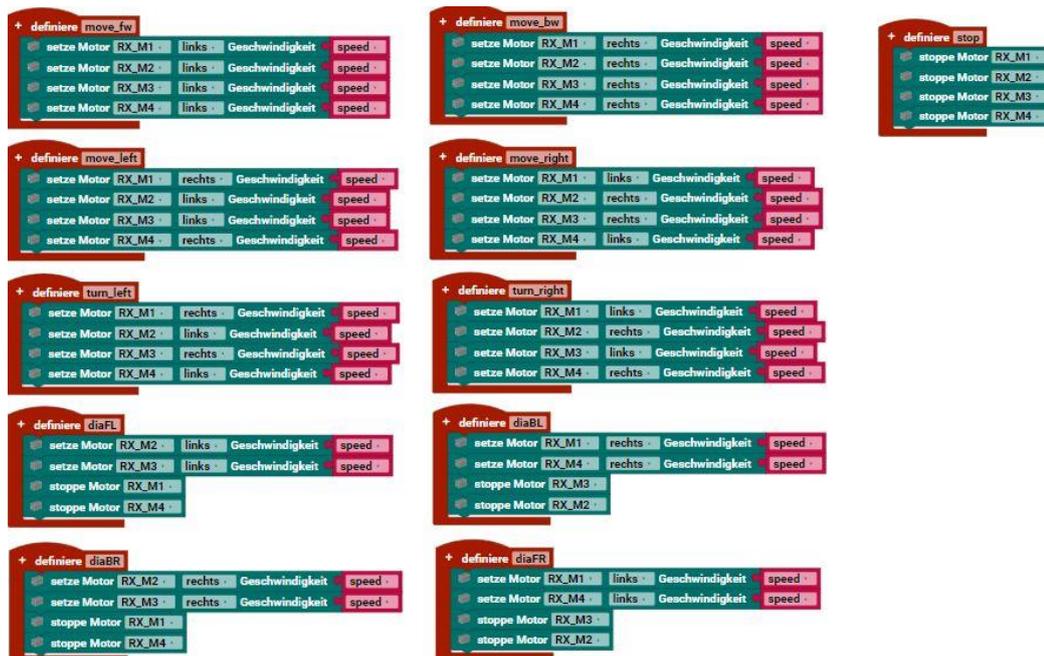




Wenn du alle Funktionen und Abfragen erstellt hast, kannst du das Programm testen. Im erscheinenden Bedienfeld kannst du mit einem Mausklick auf die einzelnen Schaltflächen das Modell steuern.



Hier zum Schluss noch einmal alle Funktionen die du für die 3. Aufgabe benötigst.



Hat alles funktioniert, speichere das Programm nochmals unter dem Namen **Omniwheels_Base_Model_3** auf deinem Rechner ab. Anschließend kannst du mit dem letzten Modell beginnen.