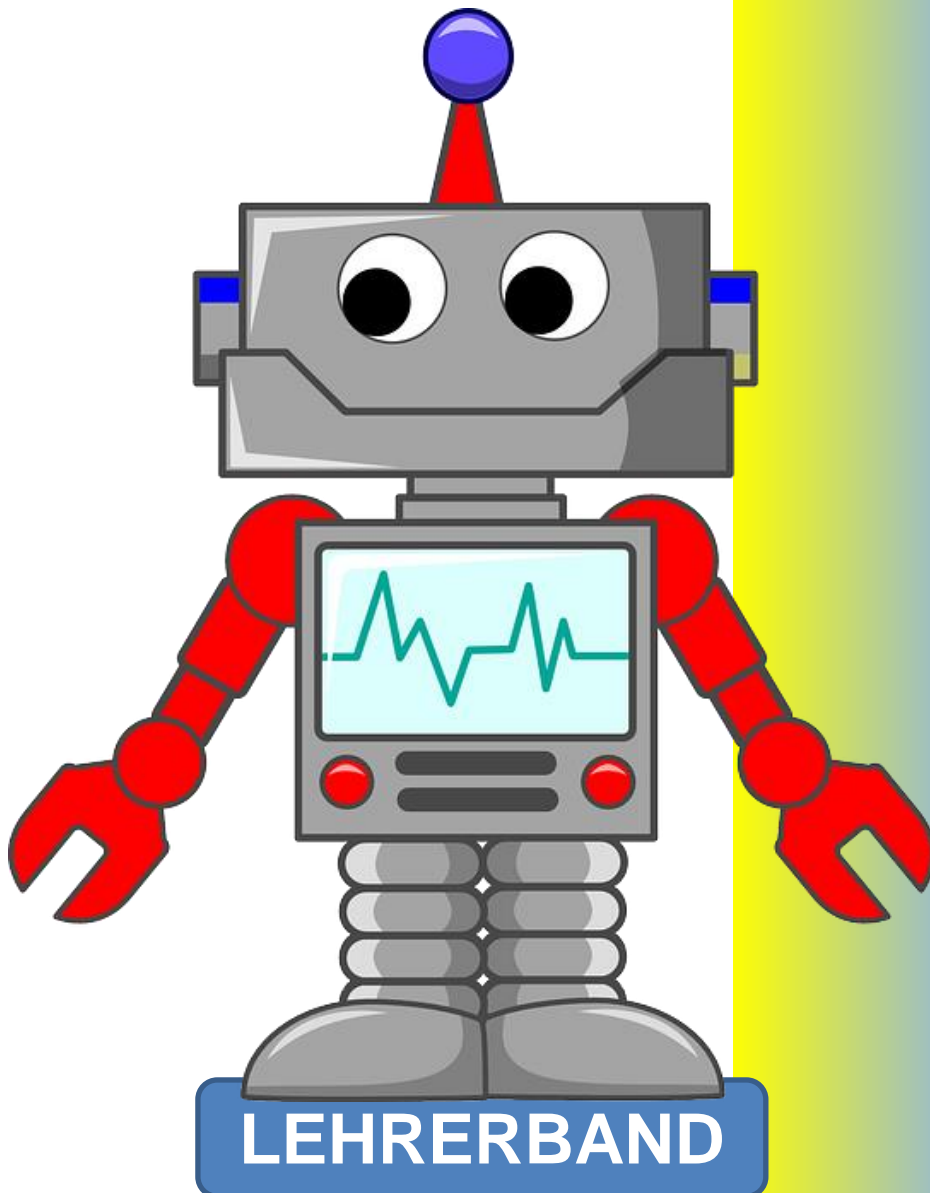


6

ROBOTER

Arbeiten mit den mBots



Hermann Milchram

29.11.2021

Bildnachweis:

sofern nicht anders angegeben befinden sich die Grafiken im Eigentum von NÖ Media oder sind eigene Aufnahmen und Zeichnungen oder Screenshots von Programmen.

Roboter auf der Titelseite robot-312566_640 Quelle Pixabay CC0

LEDs led-306562_640 Quelle Pixabay CC0

zielscheibe-ziel-bogenschießen-2304567 Quelle Pixabay CC0

smiley-1914523_640 Quelle Pixabay CC0

robot-1470108_640 Quelle Pixabay CC0

Herzlichen Dank für das **Korrekturlesen** an **Ilse Doppler** und **Angela Kampichler**!

INHALTSVERZEICHNIS

INHALTSVERZEICHNIS	1
PROGRAMMIERBARE ROBOTER	2
Bestandteile des Bausatzes	2
Hauptbestandteile des mBots	3
Aufbau der Plantine	4
Erste Inbetriebnahme	5
Aktualisierung der Firmware	5
Programmierung	6
mBot – Programmierung mit Scratch	7
Grundfunktionen und Blöcke	8
Kommunikation mit der Umwelt →(Aussehen: RGB LEDs)	10
Kommunikation mit der Umwelt →(Anzeigen: LED Matrix Display)	11
Kommunikation mit der Umwelt →(Audio-Signale)	13
Kommunikation mit der Umwelt →(Helligkeitssensor).....	14
Kommunikation mit der Umwelt →(Ultraschallsensor)	15
Kommunikation mit der Umwelt →(Linienverfolgungssensor)	16
Steuerung mit der Fernbedienung.....	17
Taster	18
LINKSAMMLUNG	FEHLER! TEXTMARKE NICHT DEFINIERT.
INHALT ROBOTIK-KOFFER	FEHLER! TEXTMARKE NICHT DEFINIERT.

Programmierbare Roboter

VOM mBot-S Education Robot von Makeblock

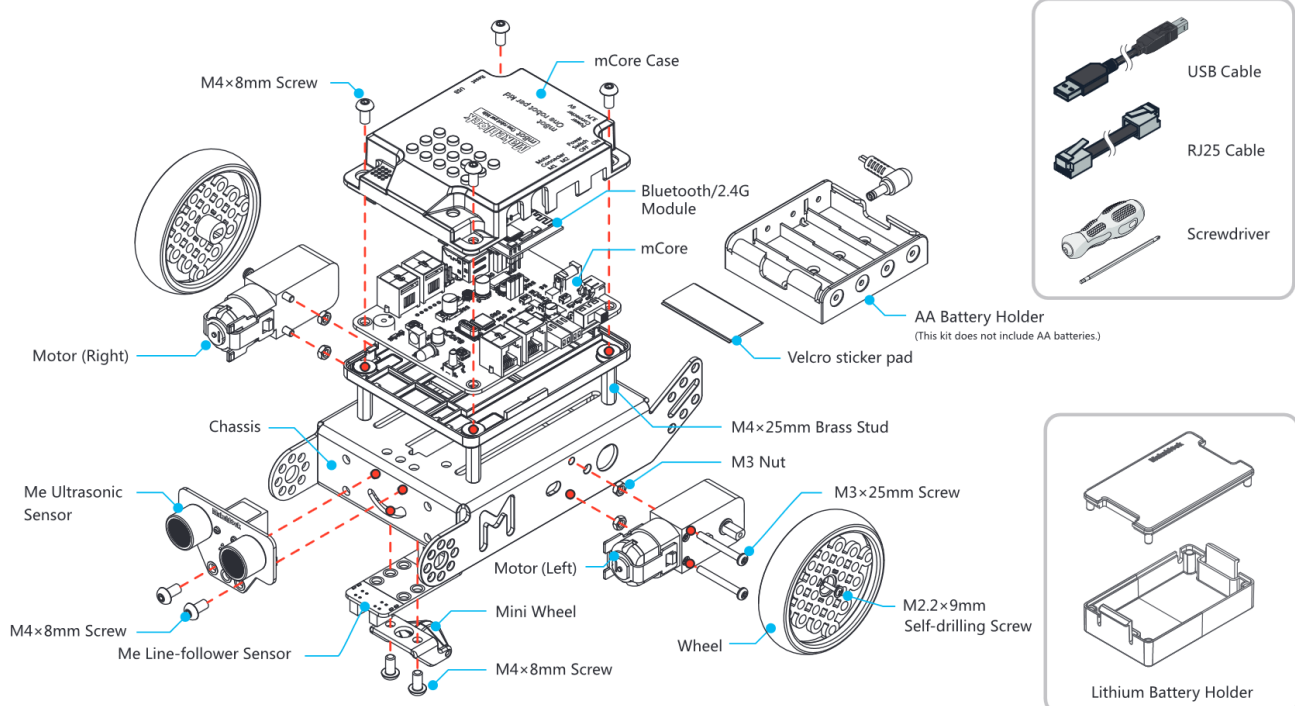
Alter: 8+ (mBot Bluetooth Version bei ca. 95€

<https://austro-tec.at/makeblock> oder <https://www.reichelt.at/>)

Der mBot ist ein **Roboter-Bausatz**, der in zwei verschiedenen Versionen zur Verfügung steht:

- ✓ **Bluetooth-Variante** (Bluetooth 4.0)
das 2,4G Modul kann nachgerüstet werden
- ✓ **2.4G-Variante** mit **USB-Dongle** (unterstützt Windows 32&64Bit Systeme)

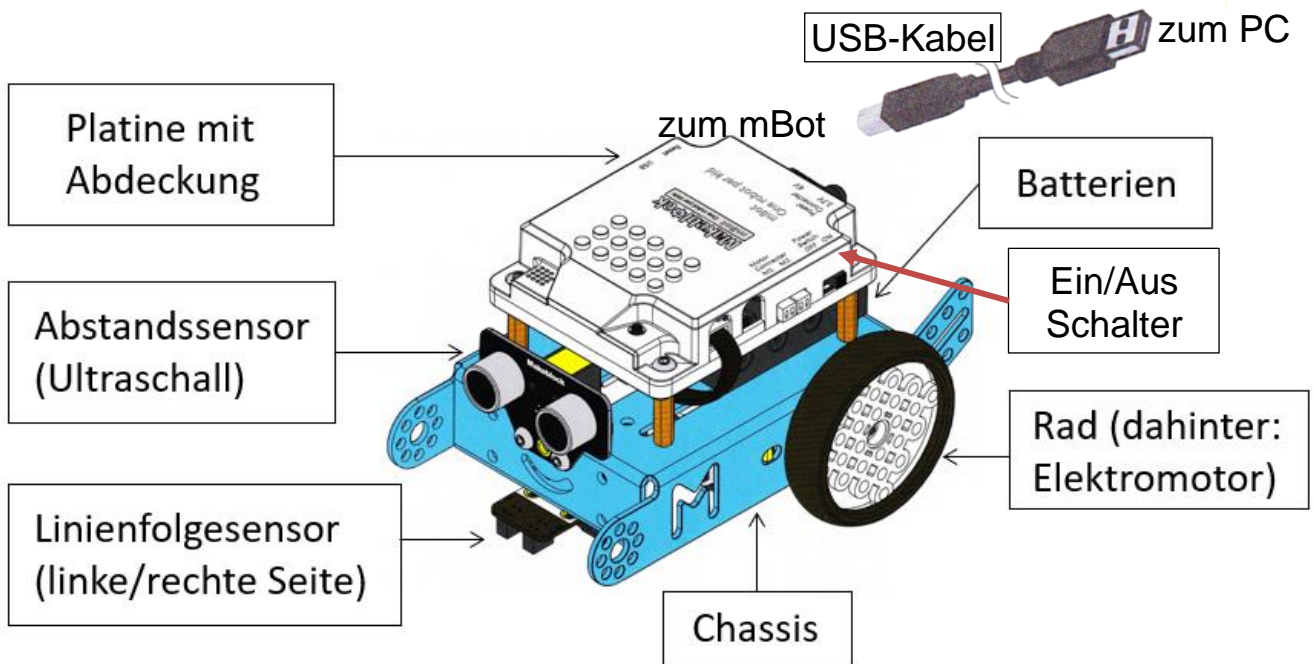
Bestandteile des Bausatzes



Die für die Steuerung verantwortliche **mCore-Platine** basiert auf dem **Arduino Uno**. Als **Programmiersprache** kommt eine an **Scratch** angelehnte, blockorientierte Programmier-Software (**mBlock**) zum Einsatz. Das Basis-Set ist wahlweise in einer Bluetooth- und einer 2.4G-Variante erhältlich. Die folgenden Funktionalitäten können programmiert werden:

- ✓ **Bewegung** erfolgt über zwei Motoren (bis zu 4 Motoren können angeschlossen werden)
- ✓ **Audioausgabe**
- ✓ **Ultraschallsensor** (Erkennen von Hindernissen)
- ✓ **Linienfolgesensor**
- ✓ **Helligkeitssensor**
- ✓ **LED-Lichter** (RGB)
- ✓ **LED Matrix Display**
- ✓ **4 RJ45 Ports** zum Anschließen der Motoren und von Hardware-Erweiterungen
- ✓ **1 Schalter** zur Auswahl dreier verschiedener Modi (Remote Steuerung, Hinderniserkennung, Linienverfolgung)
- ✓ **Lego Konnektoren**

Hauptbestandteile des mBots

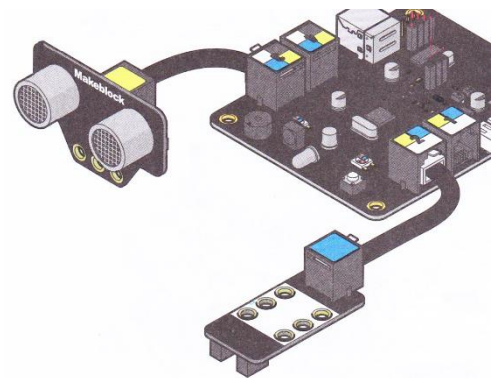


Ports und Module

An den Ports (RJ25-Buchsen) sind standardmäßig drei Module angeschlossen.

- ✓ Abstandssensor (Port 3)
- ✓ Linienverfolgungssensor (Port 2)
- ✓ LED-Matrix Modul (Port 1 oder Port 4)

Nach dem Einschalten erklingen 3 Töne und die LEDs leuchten nacheinander in unterschiedlichen Farben (rot, grün, blau und weiß) auf. Die rote LED in der Mitte zeigt die Stromversorgung an. Die LED am **Bluetooth** bzw. **WLAN 2.4G-Modul** blinkt. Sobald eine Verbindung mit dem PC, Tablet hergestellt ist, leuchtet die LED konstant.



Verbindung mit dem PC oder Tablet

1. Verbindung mit dem **USB-Kabel** (nur PC)
2. Verbindung **Bluetooth**
Notebooks und Tablet-PCs haben meist standardmäßig bereits Bluetooth integriert, für PCs ist meistens ein zusätzlicher Bluetooth-Dongle erforderlich!
 - ✓ Bluetooth am PC, Tablet ... einschalten
 - ✓ Am Windows PC muss der mBot vor der ersten Verwendung über die **Einstellungs-APP** hinzugefügt werden!
3. Verbindung mit dem **WLAN 2.4G-Modul**
<https://bit.ly/2Z8OYxY>

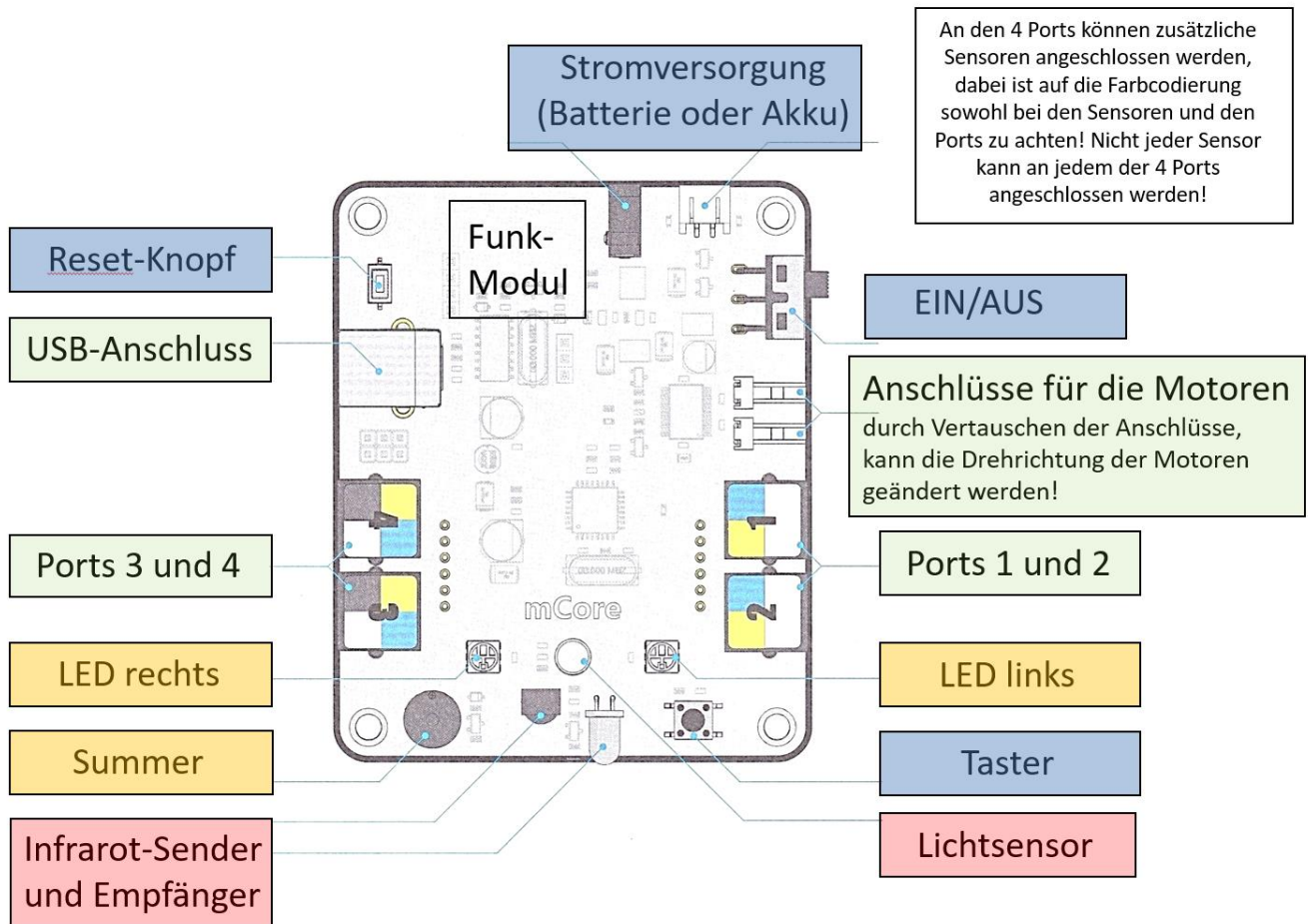


Bluetooth- und andere Geräte

+ Bluetooth- oder anderes Gerät hinzufügen



Aufbau der Plantine



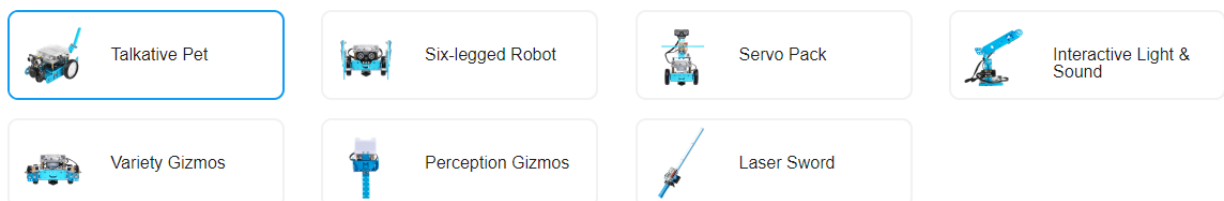
Ein paar Beispiele für Sensoren die über die 4 Ports angeschlossen werden können:

- ✓ Display
- ✓ Geräuschsensor
- ✓ Temperatur- und Luftfeuchtigkeitssensor
- ✓ Bewegungssensor
- ✓ Kompasssensor
- ✓ Schieberegler/Potentiometer



Die Farbe am Sensor gibt Auskunft darüber, an welchen Port er angeschlossen werden darf!

Zusätzlich gibt es eine Fülle von Erweiterungsbausätzen mit denen unterschiedliche Roboter gebaut werden können. <https://www.makeblock.com/mbot-add-on-packs-talkative-pet>



Erste Inbetriebnahme

Beim mBot handelt es sich um einen Roboter-Bausatz, der in einzeln verpackten Teilen geliefert wird. Geleitet durch eine Kurzanleitung montiert man mit Hilfe des mitgelieferte Schraubenziehers, die beiden Motoren, Sensoren und Räder an das ALU-Chassis und schließt sie laut Anleitung an das mCore-Board an.

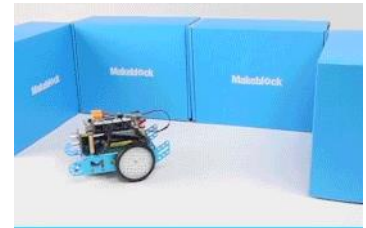
Der mBot kann wahlweise mit vier Stück **AA-Batterien** oder einem **Akku** betrieben werden. Beides ist jedoch nicht im Lieferumfang enthalten und muss extra gekauft werden.

Nach dem Zusammenbau kann der mBot auf Grund der **drei einprogrammierten Funktionsmodi** sofort einige grundlegende Funktionen ausführen. Die 3 Modi können direkt am mBot über einen kleinen schwarzen Taster an der Oberseite ausgewählt oder über die Fernbedienung (Knöpfe A B C) gestartet werden. (Firmware der Fabrik muss eingespielt sein siehe → **Aktualisierung der Firmware!**)

- ✓ **Remote Control** (Fernbedienung → **A**, LED-Farbe „weiß“) Steuerung über die mitgelieferte Fernbedienung oder mit Hilfe der Makeblock APP für Android und iOS-Geräte



- ✓ **Hinderniserkennung** (Fernbedienung → **B**, LED-Farbe grün) gesteuert durch die eingebauten Abstandssensoren kann der mBot selbständig die Umgebung erkunden und dabei Hindernissen ausweichen

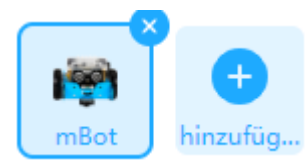


- ✓ **Linienverfolgung** (Fernbedienung → **C**, LED-Farbe blau)



Achtung: Vor der ersten Verwendung ist es sinnvoll, eventuell vorhandene Firmware-Updates zu installieren.

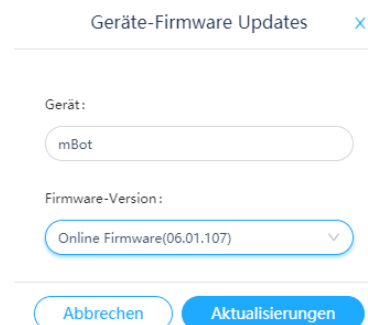
- ✓ mBot mit dem mitgelieferten **USB-Kabel** an den PC anschließen
- ✓ Programm **mBlock** am PC öffnen
- ✓ **Register Geräte** auswählen und aus der Bibliothek **mBot** auswählen und als „meistgenutztes Gerät“ hinzufügen (Cody kann entfernt werden)



Aktualisierung der Firmware

Für die **Aktualisierung der Firmware** stehen zwei Varianten zur Verfügung:

- ✓ **Online Firmware**
- ✓ **Firmware der Fabrik**
Nur hier stehen drei einprogrammierte Funktionsmodi (**Remote Control**, **Hinderniserkennung** und **Linienverfolgung**) zur Verfügung!



Programmierung

Es stehen grundsätzliche 3 Arten von Programmiersoftware für den mBot zur Verfügung:

<https://mblock.makeblock.com/en-us/download/>



mBlock web version

Chrome browser recommended >>

Support Windows/Mac/Linux/Chromebook

Code with blocks

Code with Python



mBlock PC version

Version: V5.2.0

Released: 2020.01.22

Released log >> Previous version >>

Download for Windows

Download for Mac

Win7 or Win10 (64-bit recommended)

macOS 10.12+



mBlock mobile app

Learn coding in phones and tablets



Android
Android 6.0 +
(ARM-based devices only, X86
Android not supported)



iOS
iOS 10.0 +

Die **Makeblock APP** und die **mBlock Blockly APP** bieten eine einfache Benutzeroberfläche, die es den Kindern ermöglicht, ihren mBot per Knopfdruck zu steuern oder die voreingestellten Steuerungseinstellungen beliebig zu kombinieren. Durch einfaches Ziehen, Ablegen und Kombinieren von Befehlsblöcken, genau wie bei Bausteinen, können Kinder jede Bewegung des mBots steuern und ihren eigenen persönlichen Roboter erstellen.

Über die **Makeblock-APP** stehen mehrere Lernpfade zur Erkundung des mBots zur Verfügung:

- ✓ **Spielen** (Fahren, Zeichnen und Laufen, Musiker, Sprachsteuerung)
- ✓ **Erstellen** (eigene interaktive Bedienoberfläche gestalten)
- ✓ **Coding** (vorgegebene Lernpfade und freies Programmieren mit der mBlock Blockly APP)

Coding am PC mit Scratch

Lokal am PC

- ✓ Aktuelle Software herunterladen und installieren
<https://mblock.makeblock.com/en-us/download/>



mBlock PC version

Version: V5.2.0

Released: 2020.01.22

Released log >> Previous version >>

Online im Browser (mLink)

- ✓ **mLink installieren**
<https://mblock.makeblock.com/en-us/download/>

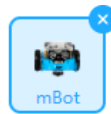


→ **Other mBlock software** (es stehen Versionen für verschiedene Betriebssysteme zur Verfügung (Windows, Mac, Linux, Chromebook))

mBot – Programmierung mit Scratch



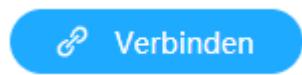
1. **mBlock** kann in der **Offline-Version** (PC-Version) oder über **mLink** als **Online-Version** (Internetverbindung erforderlich) gestartet werden.



2. In der Register-Karte „Geräte“ den mBot auswählen



3. Modus für die Programmausführung auswählen (bei Bluetooth Verbindung steht nur der Live-Modus zur Verfügung)
4. Verbindung mit dem mBot herstellen

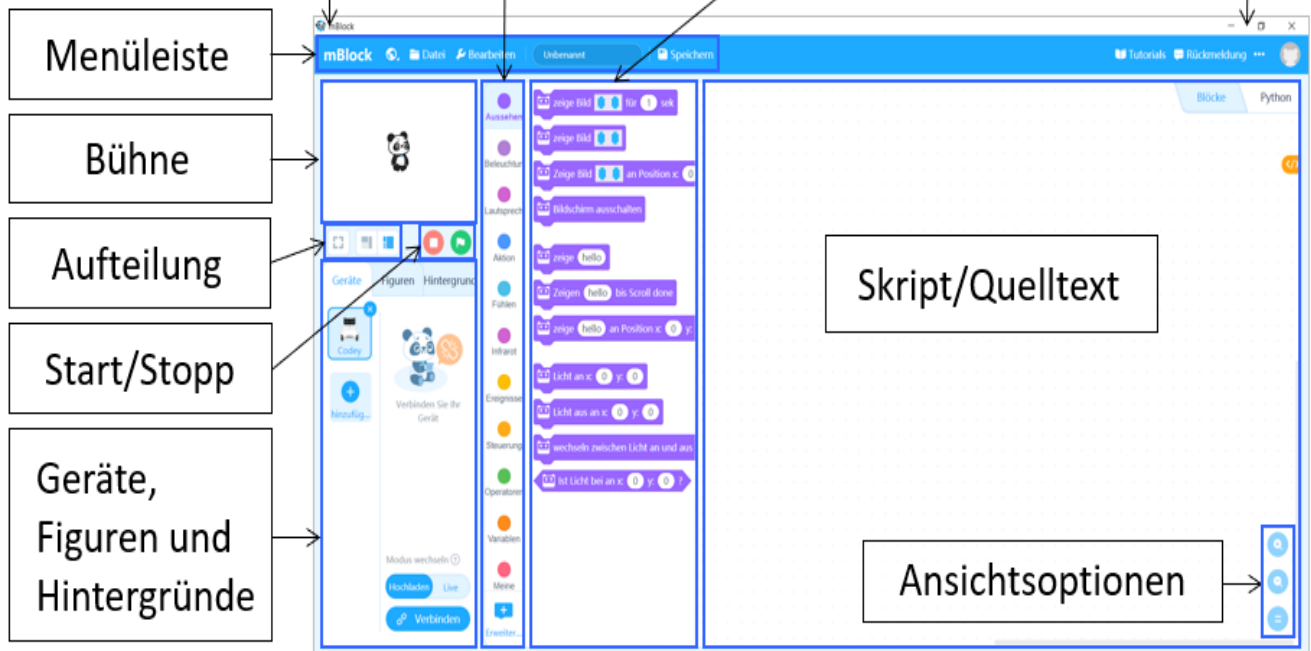
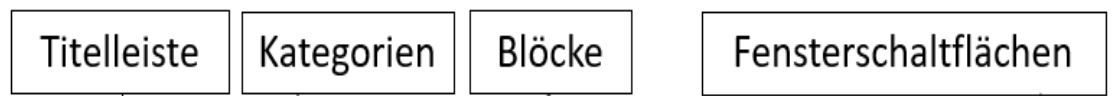


(USB, **Bluetooth**, 2,4G)

Jeder mBot ist eindeutig über seine **MAC-Adresse** (14 stellige Buchstaben-Zahlenkombination erkennbar!) Sobald mehrere mBots verwendet werden ist es sinnvoll, diese jeweils mit der **MAC-Adresse** zu beschriften um eine schnelle Zuordnung zu ermöglichen!

Im Klassenverband ist die Verwendung von **2,4G Dongles** zu überlegen, da jeder mBot eindeutig einem WLAN-Dongle zugeordnet ist, mit dem ers sich automatisch verbindet!

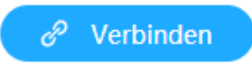



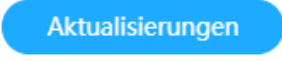
5. Gewünschte Programmblöcke per Drag&Drop ins Skriptfenster ziehen



Die Programmierung mit mBlock basiert auf der visuellen, **objektorientierten** Programmiersprache **Scratch**.

Grundfunktionen und Blöcke

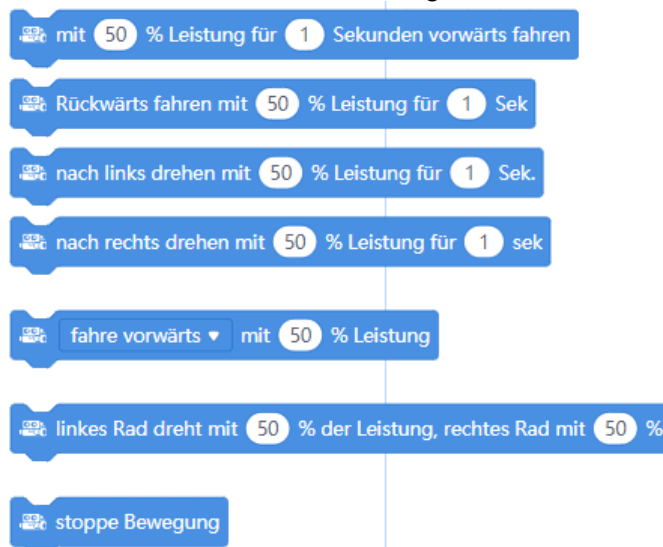
Vor der Programmierung wird empfohlen die vorinstallierten Programme zu löschen:

- ✓ Verbindung zum mBot mit **USB-Kabel** herstellen
- ✓ Auf  klicken und USB auswählen
- ✓  →  → 
(mBot muss eingeschaltet sein!)
- ✓ Online Firmware auswählen und auf  klicken!

Steuerung der Motoren (Register Aktionen)

Aktion

Im Register „Aktionen“ stehen sieben Blöcke zur Steuerung der Elektromotoren bereit.




- ✓ Der Prozentsatz (0% - 100%) bestimmt die Leistung (**Drehgeschwindigkeit**) der Motoren
- ✓ Die ersten 4 Blöcke ermöglichen eine zeitliche Begrenzung der Bewegung
- ✓ Der letzte Block dient zum Unterbrechen der Bewegung beim Verwenden von Blöcken ohne zeitlicher Begrenzung oder beim Erkennen eines Hindernisses

Dein mBot soll beim Anklicken des Fähnchens  folgende Aufgaben erledigen:



- (1) mit 50% der Leistung für 3 Sekunden vorwärts fahren.
- (2) mit 30% der Leistung für 5 Sekunde rückwärts fahren.
- (3) mit 90% der Leistung für 1 Sekunden nach links drehen.
- (4) mit 75% der Leistung für 1 Sekunde vorwärts fahren dann 2 Sekunden warten und mit 100% der Leistung für 1 Sekunde rückwärts fahren.

Tipp: Um ein Programm mit dem Fähnchen starten zu können, muss das Ereignis  am Beginn eines Skripts stehen! Probiere auch andere Ereignisse zum Starten aus



Der Befehl fürs Warten befindet sich im Register  → 

Versuche die Skripts sowohl im **Live-Modus** als auch durch **Hochladen** auszuführen!



1

```
wenn  geklickt wird
  mit 50 % Leistung für 3 Sekunden vorwärts fahren
```

2

```
wenn  geklickt wird
   Rückwärts fahren mit 30 % Leistung für 5 Sek
```

3

```
wenn  geklickt wird
   nach links drehen mit 90 % Leistung für 1 Sek.
```

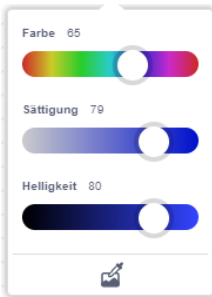
4

```
wenn  geklickt wird
   mit 75 % Leistung für 1 Sekunden vorwärts fahren
  warte 2 Sekunde(n)
   Rückwärts fahren mit 100 % Leistung für 1 Sek
```


Kommunikation mit der Umwelt → Aussehen: RGB LEDs



Register Zeigen



OnBoard-LEDs:

Im vorderen Bereich der Platine sitzen zwei **RGB-LEDs**. Die beiden LEDs können einzeln angesteuert werden. Die Farbeinstellung erfolgt dabei entweder über **Schieberegler** oder durch die Eingabe der sogenannten **RGB-Werte** (Zahlen von 0-255. → siehe InO-Bot: additive Farbmischung →. Bei den beiden Blöcken ohne zeitliche Begrenzung, müssen die LEDs durch einen Block mit den Werten 0-0-0 ausgeschaltet werden.

LED alle in Farbe ● für 1 Sekunden einschalten

LED alle in Farbe ● einschalten

Einschalten alle Licht mit Farbe Rot ● 255 Grün ● 0 Blau ● 0



- (1) beide LEDs leuchten 2 Sekunden rot
- (2) die beiden LEDs leuchten in unterschiedlichen Farben
- (3) beide LEDs sollen zuerst 1 Sekunde gelb leuchten, dann für 2 Sekunden ausgeschaltet werden und dann für drei Sekunden blau leuchten.
- (4) die beiden LEDs sollen abwechselnd in 2 unterschiedlichen Farben 5x blinken
Tipp: Verwende dazu den Block „wiederhole Anzahl“ → Register Ereignisse
- (5) Programmiere eine Verkehrsampel

<https://www.wien.gv.at/verkehr/ampeln/ampelkunde.html>

mBot AB2 Lösung

Wenn du hier mit der linken Maustaste anklickst kommst du zu den Schiebereglern zum Einstellen der Farbwerte!

1

wenn ■ geklickt wird

LED alle in Farbe ● für 2 Sekunden einschalten

2

wenn ■ geklickt wird

LED links in Farbe ● einschalten

LED rechts in Farbe ● einschalten

3

wenn ■ geklickt wird

LED alle in Farbe ● für 1 Sekunden einschalten

LED alle in Farbe ● einschalten

warte 2 Sekunde(n)

LED alle in Farbe ● für 3 Sekunden einschalten

4

wenn ■ geklickt wird

wiederhole 5

LED links in Farbe ● einschalten

LED links in Farbe ● einschalten

LED rechts in Farbe ● einschalten

LED rechts in Farbe ● einschalten

5

wenn ■ geklickt wird

wiederhole fortlaufend

LED alle in Farbe ● für 5 Sekunden einschalten

LED alle in Farbe ● für 3 Sekunden einschalten

LED alle in Farbe ● für 4 Sekunden einschalten

wiederhole 4

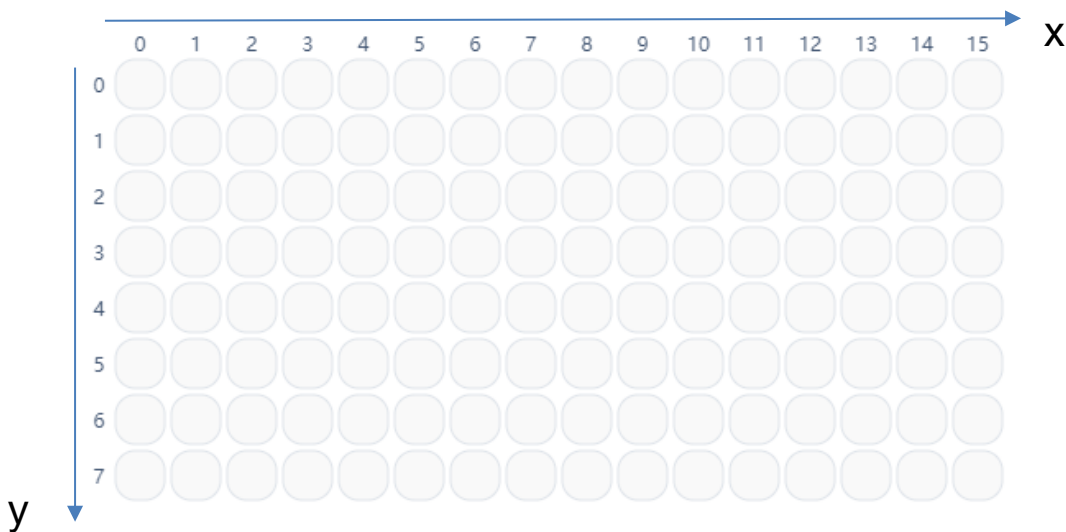
LED alle in Farbe ● für 0.5 Sekunden einschalten

LED alle in Farbe ● für 3 Sekunden einschalten

Kommunikation mit der Umwelt → Anzeigen: LED Matrix Display)

✓ LED-Matrix-Display (Register Aussehen)

Das LED-Matrix-Display kann an den freien Ports 1 oder 4 angeschlossen werden. Sie besteht aus einer Matrix von 16x8 Pixel (x=0-15, y=0-7, 128 LEDs) die einzeln angesteuert werden können. Es können damit sowohl kurze Texte, Zahlen oder Pixelbilder angezeigt werden.



In der Kategorie „Aussehen“ gibt es hierfür acht Blöcke.



- (1) Zeige für 3 Sekunden dein Lieblingssymbol aus der vorhandenen Auswahl an
- (2) Zeichne dein eigens ICON und zeige es für 5 Sekunden am Display
- (3) Lasse einen Punkt in der Mitte des Display von oben nach unten wandern
- (4) Lasse für jeweils 0.2 Sekunden einen Punkt zufällig am Display aufleuchten

Verwende dazu „Zufallszahl“ aus dem Register Operatoren

wähle eine zufällige Zahl zwischen 1 und 10

mBot AB3 Lösung

1

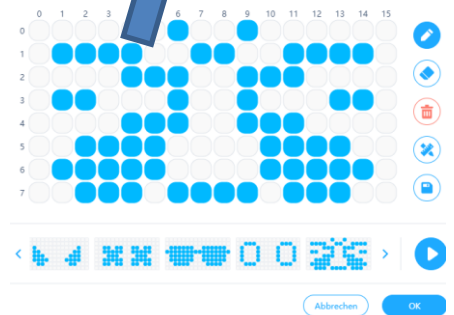
```
wenn  geklickt wird
  LED Panel port4 zeige Bild  für 3 sek
```

2

```
wenn  geklickt wird
  LED Panel port4 zeige Bild  für 5 sek
```

3

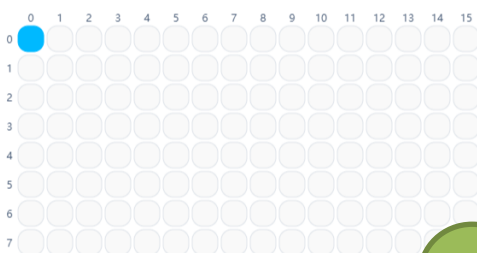
```
wenn  geklickt wird
  wiederhole fortlaufend
    LED Panel port1 zeige Bild  bei x: 7 y: 0
    warte 0.2 Sekunde(n)
    LED Panel port1 zeige Bild  bei x: 7 y: 1
    warte 0.2 Sekunde(n)
    LED Panel port1 zeige Bild  bei x: 7 y: 2
    warte 0.2 Sekunde(n)
    LED Panel port1 zeige Bild  bei x: 7 y: 3
    warte 0.2 Sekunde(n)
    LED Panel port1 zeige Bild  bei x: 7 y: 4
    warte 0.2 Sekunde(n)
    LED Panel port1 zeige Bild  bei x: 7 y: 5
    warte 0.2 Sekunde(n)
    LED Panel port1 zeige Bild  bei x: 7 y: 6
    warte 0.2 Sekunde(n)
    LED Panel port1 zeige Bild  bei x: 7 y: 7
    warte 0.2 Sekunde(n)
```



3a

So geht es auch!!

```
wenn  geklickt wird
  wiederhole fortlaufend
    setze y-wert auf 0
    wiederhole 7
      LED Panel port1 zeige Bild  bei x: 7 y: y-wert
      warte 0.2 Sekunde(n)
      ändere y-wert um 1
```



Am virtuellen Display wird 1 Punkt im Koordinatenursprung (0/0) gezeichnet!

4

```
wenn  geklickt wird
  wiederhole fortlaufend
    setze x-wert auf wähle eine zufällige Zahl zwischen 0 und 15
    setze y-wert auf wähle eine zufällige Zahl zwischen 0 und 7
    LED Panel port1 zeige Bild  bei x: x-wert y: y-wert
    warte 0.2 Sekunde(n)
```

Kommunikation mit der Umwelt → Audio-Signale



Die SOUNDS befinden sich im Register

Der **Summer/Buzzer** befindet sich auf der Platine deines mBot und kann Töne erzeugen und über den eingebauten Lautsprecher wiedergeben!

```

    spiele Note C4 für 0.25 Schläge
    Spiele Ton in Frequenz 700 Hz für 1 Sek
  
```

Die Tonhöhe wird dabei über die Frequenz (Anzahl der Schwingungen pro Sekunde) oder über die Notennamen und die Oktavbezeichnung angegeben. Die Tonlänge kann in Sekunden bzw. Schlägen (ganze Note = 1, halbe Note = 0.5, Viertelnote = 0.25, Achtelnote = 0.125) angegeben werden. **Achtung:** Die Bezeichnungen in mBlock stimmen nicht mit den in der Musiklehre verwendeten Oktavbezeichnungen überein! Kammerton **a¹** → mBlock **A4**



- (1) Dein mBot soll 8 Viertelnoten C2 spielen.
- (2) Erzeuge den **Kammerton**, er soll für 3 Sekunden erklingen.
- (3) Versuche die Melodie von „Alle meine Entchen zu spielen“ (Beginne mit C4). Zum Schreiben von Noten kannst du den Online-Noteneditor „Scorio“ verwenden! http://www.scorio.com/de_DE/web/scorio/new-score

mBot AB4 Lösung

```

    3 wenn Flagge geklickt wird
      spiele Note C4 für 0.25 Schläge
      spiele Note D4 für 0.25 Schläge
      spiele Note E4 für 0.25 Schläge
      spiele Note F4 für 0.25 Schläge
      wiederhole 2
        spiele Note G4 für 0.5 Schläge
      wiederhole 2
        wiederhole 4
          spiele Note A4 für 0.25 Schläge
          spiele Note G4 für 1 Schläge
        wiederhole 4
          spiele Note F4 für 0.25 Schläge
        wiederhole 2
          spiele Note E4 für 0.5 Schläge
        wiederhole 4
          spiele Note D4 für 0.25 Schläge
      spiele Note C4 für 1 Schläge
  
```

```

    1 wenn Flagge geklickt wird
      wiederhole 8
        spiele Note A2 für 0.25 Schläge
  
```

```

    2 wenn Flagge geklickt wird
      Spiele Ton in Frequenz 440 Hz für 3 Sek
  
```

Kommunikation mit der Umwelt → Helligkeitssensor

Der Helligkeitssensor befindet sich vorne auf der Platine.

Der zugehörige Block ist im Register  zu finden.



Wertebereich: 0 (ganz dunkel) bis ca. 1030 (sehr hell)

Mit folgendem Skript wird der aktuelle Wert der Helligkeit abgefragt und auf dem LED-Matrix-Modul angezeigt:



- (1) Setze den Sensor unterschiedlichen Helligkeiten (abdecken; aufdecken) aus und beobachte die angezeigten Werte. Abhängig von der Helligkeit, sollen die LEDs unterschiedlich leuchten.

mBot AB5 Lösung

1



Kommunikation mit der Umwelt → Ultraschallsensor

Der Ultraschallsensor (Abstandssensor) befindet sich vorne am Roboter und sollte mit dem Port 3 verbunden sein.

Der zugehörige Block ist im Register **Fühlen** zu finden.



Ultraschall-Sensor port3 Entfernung

Die Entfernungen werden in cm angegeben

- ✓ Mit dem folgendem Skript, wird die Entfernung am LED-Matrix-Display angezeigt.

```

wenn geklickt wird
  wiederhole fortlaufend
    LED-Panel port1 zeigt Zahl ultrasonic sensor port3 distance(cm)
    warte 0.1 Sekunde(n)
  
```



- (1) Schreibe ein Programm das die jeweilige Entfernung auf dem LED-Matrix-Display anzeigt und abhängig von der Entfernung einen hohen Ton (Abstand < 10) bzw. einen tiefen Ton (Abstand > 10) ausgibt. Solange kein Hindernis in der Nähe ist, soll dein mBot vorwärts fahren. Taucht ein Hindernis auf, soll er bei einem Abstand <10 stoppen und die LEDs sollen rot leuchten, während der Vorwärtsfahrt sollen die LEDs grün leuchten.

1

```

wenn geklickt wird
  wiederhole fortlaufend
    LED-Panel port1 zeigt Zahl ultrasonic sensor port3 distance(cm)
    wenn ultrasonic sensor port3 distance(cm) < 10 , dann
      spiele Note A6 für 0.05 Schläge
      LED alle in Farbe rot einschalten
      stoppe Bewegung
    sonst
      spiele Note C4 für 0.25 Schläge
      LED alle in Farbe grün einschalten
      fahre vorwärts mit 50 % Leistung
  
```

mBot AB6 Lösung

Kommunikation mit der Umwelt → Linienverfolgungssensor

Der Linienverfolgungssensor befindet sich vorne an der Unterseite deines mBots und sollte an Port 2 angeschlossen sein. Der Linienverfolgungssensor besteht aus zwei Sensoren (links/rechts), die unabhängig voneinander überprüfen, ob der Untergrund hell oder dunkel ist.

Es gibt zwei Blöcke in der Kategorie **Fühlen**



Rückgabewerte:

- ✓ 0 li: schwarz re: schwarz → Roboter ist in der Spur
- ✓ 1 li: schwarz re: weiß → Roboter zu weit rechts
- ✓ 2 li: weiß re: schwarz → Roboter zu weit links
- ✓ 3 li: weiß re: weiß → Roboter ist neben der Spur



Die **Entfernungen** werden in cm angegeben

Mit dem folgenden Skript werden die **Sensorwerte am LED-Matrix-Display** angezeigt.



(1) Schreibe ein Programm das deinen mBot auf einer schwarzen Linie dahinfahren lässt.

mBot AB7 Lösung



Steuerung mit der Fernbedienung

Der Infrarot-Sender befindet sich in der Fernbedienung, der Empfänger auf der Platine.

Der zugehörige Block ist unter **Fühlen** zu finden.



Hier kann jede Taste der Fernbedienung eingestellt werden.

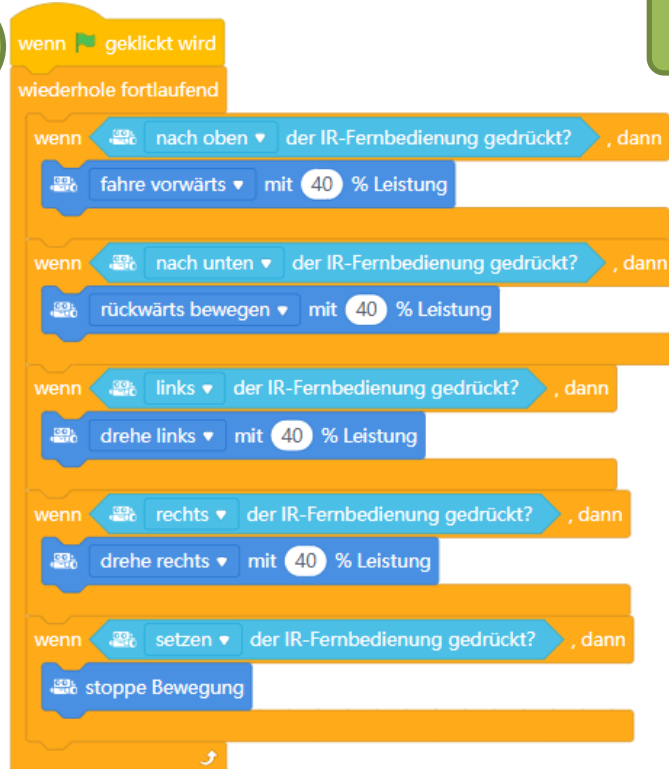


- ✓ Teste die Fernbedienung mit folgendem Skript. Das LED-Matrix-Display soll „A“ zeigen, wenn die Taste A gedrückt wird, sonst soll das Display leer bleiben.



- (1) Schreibe ein Programm, mit dem du deinen Roboter über die Fernbedienung steuern kannst.

1



mBot AB8 Lösung

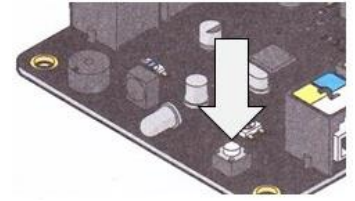


Achtung: Um den mBot über die Fernbedienung steuern zu können, muss die Firmware der Fabrik(06.01.009) eingespielt sein!

Taster

Auf dem mBot befindet sich ein Taster, mit dem im Auslieferungszustand die drei verschiedenen Modi (Remote-Control, Linienverfolgung und Hinderniserkennung) eingestellt werden können.

Der zugehörige Block ist unter **Fühlen** zu finden.



- (1) Teste den Taster mit folgendem Skript. Beim Drücken des Tasters (OnBoard-Taste) soll auf dem LED-Matrix-Modul „ups“ zu lesen sein. Wird der Taster losgelassen, soll das Display leer bleiben.

mBot AB9 Lösung



Achtung: Um den mBot über den Taster steuern zu können, muss die Firmware der Fabrik(06.01.009) eingespielt sein!



Weitere Informationen und Materialien:

<http://hemi.bplaced.net/Robotik/Roboter.htm>