



# Lehrplan

**Block #1: Einführung in die Robotik und das Programmieren.  
Themengebiet: Robo's erste Schritte.**

© 2019 durch Robo Technologies GmbH, Vienna, Austria

Alle Rechte vorbehalten. Kein Teil dieser Publikation darf ohne schriftliche Genehmigung des Herausgebers in irgendeiner Form oder auf irgendeine Weise elektronisch, mechanisch, durch Fotokopieren, Aufzeichnen oder auf andere Weise reproduziert, in einem Abrufsystem gespeichert oder übertragen werden.

# Willkommen beim Robo Wunderkind Lehrplan!

Wir freuen uns darüber, dass du dich dafür entschieden hast mit dem ersten Block der Robo Wunderkind Unterrichtseinheiten zum Thema: Robotik, Programmieren und Computerwissenschaften zu arbeiten.

**Der Block #1: Robo's erste Schritte** besteht aus **10 Einheiten**, in welchen SchülerInnen durch selbstständige und kooperative Arbeit die **grundlegenden Kompetenzen der Robotik, des Programmierens und der Computerwissenschaft** erlernen. Unser Lehrplan dient zur stetigen Unterstützung während der Arbeit mit Robo Wunderkind und ist sofort einsetzbar. Dadurch kannst auch du, gemeinsam mit den SchülerInnen, neue Erfahrungen sammeln, mit Robo eigene Ideen in die Realität umsetzen und währenddessen Programmieren und digitale Bildung im Unterricht einbinden.

## **Unser sofort einsetzbarer Lehrplan beinhaltet:**

- Alle **Kerninformationen** und **Details** zur Organisation der Einheiten.
- **Kompetenzen, Lernergebnisse** sowie **Lehrplanbezug**.
- **10 Unterrichtsplanungen** – übersichtliche und einfach nachvollziehbare Gestaltung mit **Zusatzaufgaben**, die differenzierten Unterricht ermöglichen.
- **Zusatzmaterialien:** Präsentationsfolien, Keywords, Druckvorlagen und Challenge Cards.

1. Kerninformation	3
<hr/>	
2. Überblick: Konzepte der Robotik und Computerwissenschaften	4–5
<hr/>	
3. Lernergebnisse	6–7
<hr/>	
4. Robo Wunderkind Unterrichtsaufbau	8–9
<hr/>	
5. Stundenplanungen: Einheit 1-10	10–39

---

**Thema:** STEAM Fächer

**Schulstufe:** 1–4

**Gruppengröße:** 6-12 SchülerInnen

---

Alle Einheiten in Block #1 basieren auf **Geschichten** und werden durch ein Thema verknüpft: **Robo's erste Schritte**. In dieser Geschichte ist Robo ein kleiner Roboter, der das Klassenzimmer besucht. Die SchülerInnen werden Robo dabei helfen, neue Fähigkeiten zu erlernen und die Welt um sich herum zu entdecken. Auf diese Weise lernen sie neue Konzepte in der Robotik und Informatik kennen, bauen eigene Roboter und entdecken die einzigartige **Robo Code App**, um ihre ersten auf einen Zustandsautomaten basierenden Programme zu erstellen.

**Komplexität:** Jede Einheit beinhaltet unterschiedliche Komplexitätsstufen. Die Grundlagen sind für alle SchülerInnen dieselben. Durch die Zusatzaufgaben, kann die Komplexität der Aufgabe an das Niveau der SchülerInnen angepasst werden.

**Empfohlene Vorkenntnisse:** Die SchülerInnen benötigen keine Vorkenntnisse, da sie mit dem Robo Wunderkind Baukasten und der Robo Code App die grundlegenden Begriffe und Prinzipien der Robotik und des Programmierens erlernen.

## **Benötigte Materialien:**

- Robo Wunderkind Baukästen
- Tablet(s)
- Ergänzende Materialien um individuelle Roboter zu gestalten: Lego™ Bausteine, Farbpapier, Karton, etc.
- Zusatzmaterialien: Präsentationsfolien, Keywords, Druckvorlagen und Challenge Cards

# Konzepte der Robotik und Computerwissenschaften in Block #1

Konzepte	Einheit 1	Einheit 2	Einheit 3	Einheit 4	Einheit 5	Einheit 6	Einheit 7	Einheit 8	Einheit 9	Einheit 10
<b>Robotik</b>										
<b>1. Robotik</b> , Ingenieurwesen	+	+	+	+	+	+	+	+	+	+
<b>2. Elektrizität:</b>										
• Elektrischer Strom	+	+	+	+	+	+	+	+	+	+
• Örtliche Verbindungen	+	+	+	+	+	+	+	+	+	+
<b>3. Kabellose Kommunikation</b>	+	+	+	+	+	+	+	+	+	+
<b>4. Fernsteuerung</b>	+									
<b>5. Design Thinking Prozesse:</b>										
• Steuerungs Design	+									
• (mechanisches) Design			+	+	+	+	+	+	+	+
• Code Design		+	+	+	+	+	+	+	+	+
<b>6. Nutzer Input:</b> Verwendung der Steuerelemente (Geräusche, Licht, Motoren, Servomotor)	+									
<b>7. Outputs:</b> Funktionen der Module										
• Outputs: Geräusche		+				+		+	+	+
• Outputs: (RGB) Licht			+			+		+	+	+
• Outputs: DC Motoren				+	+	+	+	+	+	+
• Outputs: Servo Motor							+	+	+	+

# Konzepte der Robotik und Computerwissenschaften in Block #1

Konzepte	Einheit 1	Einheit 2	Einheit 3	Einheit 4	Einheit 5	Einheit 6	Einheit 7	Einheit 8	Einheit 9	Einheit 10
<b>Computerwissenschaften</b>										
<b>1. Programmieren</b>		+	+	+	+	+	+	+	+	+
<b>2. State-Machine basiertes Programmieren:</b>										
• Befehl		+	+	+	+	+	+	+	+	+
• Verbindung		+	+	+	+	+	+	+	+	+
• Zustand						+		+	+	+
<b>3. Sequenzielle Logik</b>		+	+	+	+	+	+	+	+	+
<b>4. Schleifen</b>		+	+	+	+		+			+
<b>5. Parallele Ausführung</b>						+		+	+	+
<b>6. Nutzer Input: Parameter</b>			+	+	+	+	+	+	+	+
<b>7. Digitale Bildung</b>	+	+	+	+	+	+	+	+	+	+
<b>8. Konstruktionsprozess</b>									+	+

# Lernergebnisse der Robotik und Computerwissenschaften in Zusammenhang mit den RW Baukästen

## I. Robotik:

- SchülerInnen verstehen den Umgang mit digitalen Medien – **Digitale Bildung**.
- SchülerInnen verstehen was Roboter sind und wie diese im Alltag eingesetzt werden – **Robotik**.
- SchülerInnen wissen was elektrischer Strom und kabellose Kommunikation (Bluetooth) ist und warum Roboter diese benötigen.
- SchülerInnen verstehen, wie Roboter gesteuert werden: Sie kennen den Unterschied zwischen Steuerung und Programmierung.
- SchülerInnen können die Konzepte des mechanischen Designs und Programmier Designs bei Robo Projekten anwenden.

## Robo Wunderkind Baukästen:

- SchülerInnen kennen die **RW Module** und **Verbindungen** und verstehen deren Funktionen – **Outputs**.
- SchülerInnen verstehen die Funktionen des Hauptblocks und können erklären warum diese für jedes Projekt notwendig sind.
- SchülerInnen verstehen und berücksichtigen die grundlegende **Baulogik** der RW Baukästen.
- SchülerInnen verwenden die Funktionen der einzelnen Module um **konkrete Projekte umzusetzen**. – Anwendung mechanischen Designs.

## Robo Live App:

- SchülerInnen können die **Robo Live App** verwenden: Sie können den Hauptblock verbinden, neue Projekte erstellen und steuern.

## II. Computerwissenschaften:

- SchülerInnen wissen, was ein **Programm** und **Programmieren** ist.
- SchülerInnen verstehen die Begriffe des **State-Machine Based Programmierens** wie **Befehl, Verbindung, Zustand, Schleife** und können diese in der visuell gestalteten Programmiersprache der **Robo Code App** umsetzen.
- SchülerInnen verstehen den Unterschied zwischen **sequenziellen** und **parallelen Befehlen** und können diese programmieren.
- SchülerInnen kennen die Eigenschaften von Befehlen (**Nutzer Input**) und können diese für die Nutzung eines Robos verwenden.
- SchülerInnen verstehen und berücksichtigen den **konstruktiven Design Prozess** um ein Robo Projekt zu erstellen.

## Robo Code App:

- SchülerInnen können die **Robo Code App** verwenden: Sie können den Hauptblock verbinden, ein neues Projekt erstellen und das Befehlsmenü bedienen.
- SchülerInnen können die **visuell aufgebaute Programmiersprache** der Robo Code App verwenden um in Projekten **Problemstellungen zu lösen**.

## Kognitive und Verhaltensbezogene Inhalte:

- SchülerInnen können ihre Aufmerksamkeit den Informationen zuwenden, die für das erstellen der Aufgaben nötig ist (dauerhafte, selektive, abwechselnde und geteilte Aufmerksamkeit).
- SchülerInnen können die Konzentration während der für die Erledigung von Aufgaben / Projekten vorgesehenen Zeit beibehalten.
- SchülerInnen können Informationen für kurzfristige Aufgaben sowie für längere Zeiträume auswendig lernen.
- SchülerInnen können ihre Fähigkeit zum räumlichen Denken anwenden, um nach ihren Vorstellungen funktionierende Roboter zu bauen.
- SchülerInnen können ihr **logisches und analytisches Denken** anwenden, um verschiedene Roboter zu steuern und zu programmieren.
- SchülerInnen können **Problemstellungen** durch ihr kritisches Denkvermögen **lösen**: Ziele setzen, planen, handeln, reflektieren, verbessern, bewerten und dabei Kritik akzeptieren und daran wachsen
- SchülerInnen können ihre **Fantasie und Kreativität** einsetzen, um ein eigenes Projekt zu erarbeiten.
- SchülerInnen können in **Paaren / kleinen Gruppen** zusammenarbeiten und **soziale Fähigkeiten** nutzen, um ein gemeinsames Projekt zu erarbeiten.
- SchülerInnen können erarbeitete Projekte vor ihrer Klasse **präsentieren, konstruktives Feedback** annehmen und geben.

# Aufbau einer Unterrichtseinheit mit den Robo Wunderkind Baukästen

## Aufbau:

- Fokus auf bestimmte Konzepte in Bezug auf die RW Module, die Robo Live oder Robo Code App.
- Lernziele, Lernergebnisse und Keywords.
- Zusatzmaterialien.

## Unterrichtsabschnitte: 8 Schritte

### Einleitung 7 – 10 min

- 1 **Aktivieren der SchülerInnen:** Bezug zu Vorwissen und persönlichen Erfahrungen der SchülerInnen.
- 2 **Analysieren:** Erzählung der Robo Story um die SchülerInnen emotional zu erreichen, Gemeinsames erarbeiten der **Problemstellung und Lösungen** zur Bewältigung.

### Angeleitete Aufgabe 15 – 20 min

- 3 **Vorbereitung: Erinnerung** an zuvor erlernte Inhalte zu den RW Baukästen und den Apps; Verbinden des Hauptblocks mit den Tablets und Gestaltung eines neuen Projektes.
- 4 **Learning by doing – Gemeinsames Bauen und Programmieren:** SchülerInnen lösen unterschiedliche Aufgaben durch kooperatives Programmieren und Diskutieren.
- 5 **Zusammenfassen** der neuen Informationen, bevor zur selbstständigen Aufgabe gewechselt wird.

### Selbstständige Aufgabe 15 – 20 min

- 6 **Gestaltung eines eigenen Projektes:** Die SchülerInnen arbeiten einzeln, paarweise oder in kleinen Gruppen, um ein eigenes Projekt zu erstellen. Die Lehrperson unterstützt SchülerInnen, welche Schwierigkeiten haben einzeln. Für die Gestaltung der Umgebung und individueller Roboter, können unterschiedliche Materialien verwendet werden. Um Projekte mit zwei oder mehreren Robotern zu erstellen, können SchülerInnen in größeren Gruppen zusammenarbeiten. **Präsentation** (optional): Die SchülerInnen stellen ihre Projekte der Klasse vor und geben sich gegenseitig konstruktives Feedback.
- \* **Variation:** (optional für fortgeschrittene SchülerInnen) um die Einheit komplexer zu gestalten.

### Reflection & Feedback 7 – 10 min

- 7 **Zusammenfassung** der gelernten Inhalte. **Feedback** über die Schwierigkeit der Aufgaben **einholen**.
- 8 **Aufräumen:** SchülerInnen lernen den sorgfältigen Umgang mit Geräten und Baukästen kennen. Nach der Verwendung werden die elektronischen Geräte **ausgeschaltet** und die Module in die Box einsortiert. Die Tablets werden sorgfältig abgesammelt.

Dauer: 45 – 60 min

# Überblick der Projekte: Robo's erste Schritte

Projekte	Konzepte	Schwierigkeitsgrad	Seite
1. Robo kennenlernen!	Robotik, Ingenieurwesen; mechanisches Design	☆	10
2. Programmier deinen ersten Robo!	Programmieren einer Befehlskette; sequentielle Logik; Befehl, Verbindung, Schleife.	☆	13
3. Robo beleuchtet das Klassenzimmer	sequentielle Logik; Befehl, Verbindung, Schleife.	☆	16
4. Das Robo-Auto fährt herum	Befehle: Fahren, Drehen; (Einstellungen: Entfernung, Winkel, Geschwindigkeit)	☆	19
5. Robo's erste Reise	parallele Ausführung, Zustand (+Befehlsdauer), mechanisches Design	☆☆	22
6. Robo reist in die Spielzeugstadt	mechanisches Design und Verwendung der DC Motoren	☆☆	25
7. Robo schaut sich um	Motoren: Servo vs. DC Motor – Entfernung vs. Winkel; mechanisches Design	☆	28
8. Robo lernt Freunde kennen	parallele Ausführung, Zustand (+Befehlsdauer), mechanisches Design	☆☆	31
9. Robo macht eine Überraschung	Design Konstruktion, mechanisches und Code Design	☆☆☆	34
10. Was macht dein Robo?	Design Konstruktion, mechanisches und Code Design	☆☆	37

# Einheit 1: Robo kennenlernen!

**Konzepte:** Robotik, Ingenieurwesen; Mechanisches Design

**Schwierigkeitsgrad:** ★☆☆

Die Einheit #1 dient zur Einführung in die Robo Wunderkind Baukästen. Die Einheit eignet sich für SchülerInnen, die zum ersten Mal mit den Robo Wunderkind Modulen in Kontakt treten oder bereits die Einführungsprojekte absolviert haben. Sie können dabei unterschiedliche Ansätze verwenden: 1) Wenn das Thema in Ihrer Klasse neu ist, können Sie durch die Einheit neues Wissen über die RW Baukästen vermitteln. 2) Sind Ihre SchülerInnen bereits mit den Baukästen vertraut, so eignet sich eine eher unabhängige Herangehensweise um das vorhandene Vorwissen durch Übung und kooperatives Spiel zu verfestigen.



## Story of Robo:

Heute haben wir einen besonderen Gast in unserer Klasse! Das ist Robo, ein intelligenter Roboter, der in unsere Klasse gekommen ist, um unser Freund zu sein und mit uns zu lernen!



## Unterrichtsziel:

Wir werden Ingenieure, um unseren ersten Roboter zusammenzubauen und zu steuern!

# Einheit 1: Robo kennenlernen!

## Module:



Main Block



DC Motors



Servo



RGB LED



Button



Distance Sensor



Wheels



Small Wheel



Connectors



Connector Flat Lego™



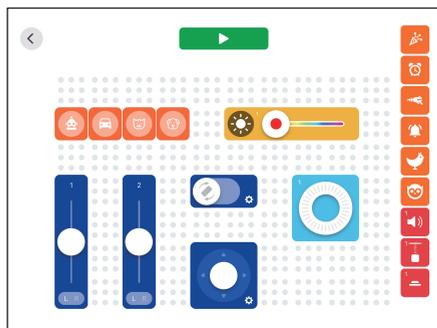
Connector Wire



Connector Block

## Steuerung:

Verwenden Sie die Robo Live App um die Module zu steuern und um deren Funktionen kennenzulernen.



## Fokus:

- **Robotik:** Roboter und deren Aufgaben im Alltag, Fernsteuerung als Möglichkeit zur Bedienung von Robotern.
- **RW Baukasten:** Module, Verbindungsstücke und deren Funktionen.

## Lernziele:

- Die SchülerInnen können einen Robo unter Berücksichtigung der unterschiedlichen Funktionen der einzelnen Module zusammenbauen.

## Lernergebnisse:

- Die SchülerInnen können erklären warum jeder Robo einen Hauptblock benötigt.
- Die SchülerInnen kennen verschiedene Module und deren Funktionen.
- Die SchülerInnen verstehen wie Module miteinander zu kombinieren sind um einen Roboter zu bauen.

## Keywords:

- Robotik, Ingenieurwesen, Roboter, Hauptblock, Verbindungsstücke, Trennungs-Tool.

## Zusatzmaterialien:

- Graphiken der Module zum ausdrucken.
- Challenge Karte #1.
- *Optional: Lego™ Bausteine, Farbpapier, und/oder andere Materialien.*

# Einheit 1: Robo kennenlernen!

## Unterrichtsabschnitte:

**Einleitung** 7 – 10 min

1 **Frage:** Wisst ihr was ein Roboter ist? Warum bauen Menschen Roboter? Wer von euch hat bereits Erfahrung mit Robotern? Wie werden Roboter gesteuert? Wer beschäftigt sich mit Robotern? **Besprechung** folgender Begriffe: **Ingenieurwesen, Robotik, Roboter.**

2 **Analysieren:** Erzählen der Robo Story; Gemeinsames erarbeiten von **Problemstellung** und Lösungen zur Bewältigung.

3 **Öffnen** und erkunden **der Robo Boxen:** kurzes durchschauen der Module.

4 **Entdecken der Robo Live App:** **Verwenden Sie die Challenge Karte #1**

- **Besprechen** Sie die Verwendung des Tablets für die Steuerung des Robos; Teilen Sie die Tablets aus und besprechen Sie die **Benutzungsregeln der Tablets.**
- Einschalten des Hauptblocks und Verbindung mit der Robo Live App; Besprechen Sie die Benutzungsoberfläche mit den SchülerInnen.
- Weitere Module werden nun an den Hauptblock angesteckt. Die SchülerInnen können nun herausfinden, wie sich das in der App bemerkbar macht. Anschließend werden diese wieder auseinandergelöst. Besprechen Sie die Verbindungsstücke und das Trennungs-Tool mit Ihren SchülerInnen.
- Verwenden der Steuerelemente um die einzelnen Module zu steuern. Besprechen Sie, wie diese funktionieren.
- Besprechen des Begriffs "Design" und wie unterschiedliche Bauweisen sich auf die Funktionen eines Robos auswirken.

5 **Zusammenfassen** der neue Inhalte bevor zur eigenständigen Aufgabe übergegangen wird. **Bemerkung:** Die SchülerInnen müssen am Ende der Einheit nicht alle einzelnen Module perfekt verstehen und kennen. Vielmehr sollte ihnen eine Möglichkeit geboten werden die Module durch selbstständiges erkunden kennenzulernen. Dabei werden die Module berührt, verbunden und gesteuert um die Funktionen herauszufinden.

**Achten Sie jedoch darauf, dass jede/r SchülerIn die Funktionen des Hauptblocks kennt.**

**Selbstständige Aufgabe** 15 – 20 min

6 **Gestalte dein eigenes Projekt:** Die SchülerInnen können nun beliebige Variationen von Robo bauen und diese mithilfe der Robo Live App steuern. Stellen Sie folgende Frage: Wieso hast du den Robo auf diese Art gebaut und inwiefern wird dir der Robo im Alltag helfen? Es können Zusatzmaterialien verwendet werden um einen individuellen Roboter zu gestalten.

\* **Zusatzaufgabe:** *Besprechen Sie **technische Details** und das **Design** der Robo Module. Besprechen Sie den eingebetteten Computer und den Akku im Hauptblock und die Pogo-Pins auf den Verbindungsstücken und ihre Funktionen. Besprechen Sie des Weiteren die DC Motoren und den Servo Motor und deren rotierende Teile ohne Pogo-Pins. Lassen Sie die SchülerInnen die Motoren andersrum anzuschließen und herausfinden, ob sie in der Robo Live App erscheinen oder nicht.*

**Reflection & Feedback** 7 – 10 min

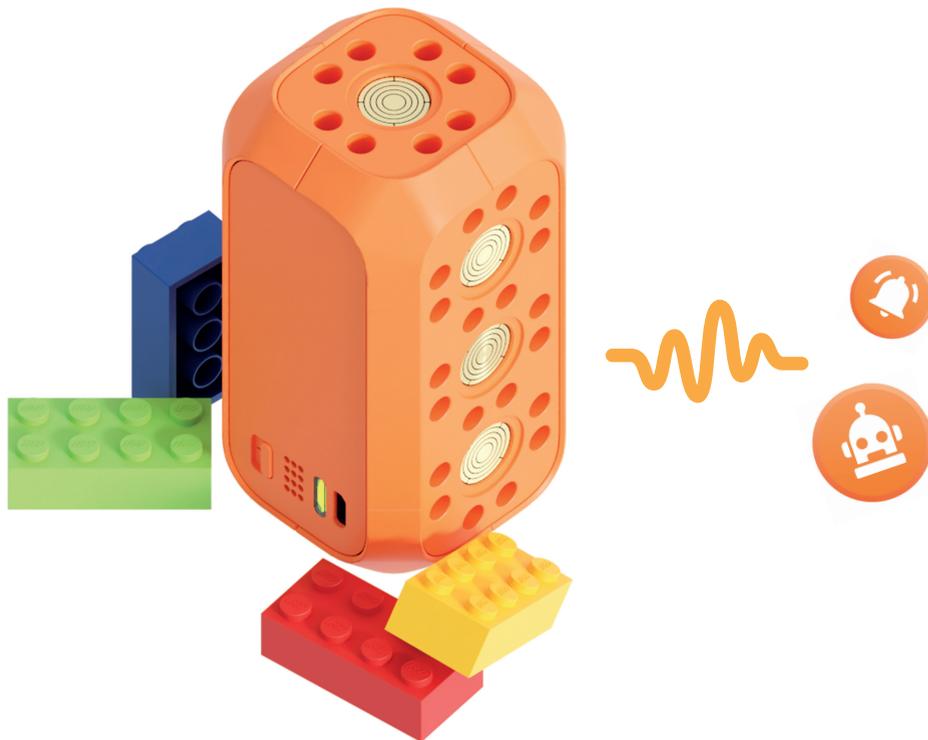
7 **Zusammenfassung:** Besprechung einiger Module (und deren Funktionen), welche in den Projekten verwendet wurden. **Feedback erhalten:** Waren die Aufgaben schwierig oder einfach? Interessant oder langweilig? Welcher Teil der Einheit war am spannendsten? Wieso?

8 **Aufräumen:** Die SchülerInnen sollen lernen auf die Geräte die sie verwenden aufzupassen und diese ordnungsgemäß aufzuräumen – RW Baukästen und Tablets.

# Einheit 2: Programmierere deinen ersten Robo!

**Konzepte:** Programmieren einer Befehlskette; Sequentielle Logik; Befehl, Verbindung, Schleife

**Schwierigkeitsgrad:** ★☆☆



## Story of Robo:

Unser neuer Freund freut sich sehr uns wieder zu sehen und möchte uns "Hallo" sagen. Denkt ihr, Roboter können so etwas von selbst tun? Mit der Robo Code App können wir Programme erstellen, die unser Roboter ausführen kann.



## Unterrichtsziel:

Wir wollen Robo dabei helfen Geräusche von sich zu geben. Werden wir ProgrammiererInnen und erstellen unseren ersten Programmcode in der Robo Code App.

## Einheit 2: Programmiere deinen ersten Robo!

### Module:



Main Block

### Programmiere:



Sounds



Connect

### Fokus:

- **Computerwissenschaften:** Programmieren um einen Roboter steuern zu können; Die Relevanz des Programmierens im Alltag.
  - **Robo Code App:** Befehl, Verbindung; Schleife.
- 

### Lernziele:

- Die SchülerInnen können ein sequentielles Programm kodieren, welches Geräusche und Verbindungen beinhaltet. Anschließend können Sie diese anpassen um eine Schleife zu erstellen.

### Lernergebnisse:

- Die SchülerInnen verstehen was ein Programm ist.
  - Die SchülerInnen kennen die Code Tasten und Befehle in der Robo Code App und können Geräusche programmieren.
  - Die SchülerInnen können zwei oder mehrere Geräusche miteinander verbinden um ein sequentielles Programm zu erstellen.
  - Die SchülerInnen können ein Programm anpassen um eine Schleife zu erstellen.
- 

### Keywords:

- Programmieren, Programm.
  - Projektbildschirm, Programmierbildschirm, Befehl, Startpunkt, Start Knopf, Verbindungsknopf, Verbindung, Papierkorb (Löschen Modus).
- 

### Zusatzmaterialien:

- Challenge Karte #2.
- Grafiken zum ausdrucken: Hauptblock, Befehle, Knöpfe.
- *Optional: Lego™ Bausteine, Farbpapier, und/oder andere Materialien.*

## Einheit 2: Programmiere deinen ersten Robo!

### Unterrichtsabschnitte:

- Einleitung**  
7 – 10 min
- 1 Frage:** Wer ist euer neuer Freund? Was haben wir gestaltet? Wie können Menschen Roboter steuern? Was ist der Unterschied zwischen fernsteuern und programmieren? Habt ihr bereits zuvor programmiert? Was ist Programmieren? **Besprechen** der Begriffe: **Programmieren, Code.**
  - 2 Analysieren:** Erzählen der Robo Story; Gemeinsames erarbeiten von **Problemstellung** und Lösungen zur Bewältigung.
- 
- Angeleitete Aufgabe**  
15 – 20 min
- 3 Öffnen der Baukästen; Abrufen** der Funktionen des Hauptblocks und deren Wichtigkeit.
  - 4** Kennenlernen der **Robo Code App** und programmieren von **Geräusch-Befehlen:** Bringe deinem Robo sprechen bei!  
**Verwenden Sie die Challenge Karte #2**
    - **Verbinden** des Hauptblocks mit der Robo Code App; **Besprechen Sie:** Mein Robo Bildschirm, Robo Projektbildschirm, Neues-Projekt-Knopf und die Befehlsleiste.
    - Finden der Geräusche auf der **Befehlsleiste:** Besprechen Sie die Farbe der Befehle.
    - Programmieren des **ersten Geräusches:** Besprechen Sie den Startpunkt und den Start-Knopf.
    - Programmieren des **zweiten Geräusches:** Besprechen Sie wie und warum der Startpunkt verschoben werden kann.
    - **Verbinden von zwei Geräuschen:** Vermittlung der Begriffe des **Verbindungsknopfes**, Verbindungs-Modus und **Verbindungen** zwischen Befehlen.
    - **Löschen** eines Geräuschs oder einer Verbindung: Vermittlung der Begriffe des **Papierkorbs** und des Lösch-Modus.
    - **Programmieren einer Schleife:** Abänderung eines Programms zu einer Schleife – Besprechen Sie den Begriff.
  - 5 Zusammenfassung** Hauptblock und seine Funktionen, Robo Code App Interface; die Rolle von Robotern im Alltag.
- 
- Selbstständige Aufgabe**  
15 – 20 min
- 6 Gestaltung eines eigenen Projektes:** Überlegt, wie Robo Geräusche macht und wie er euch im Alltag helfen könnte. Erstellt nun eine beliebige Variation von Robo Geräuschen für bestimmte Situationen. Ihr könnt auch Materialien verwenden um euren Robo selbst zu gestalten.
  - \* Zusatzaufgabe:** *Inhalte über **Programmiersprachen**; Stellen Sie Fragen: Was ist eine Programmiersprache? Kennt ihr bereits Programmiersprachen? Wie sieht die Programmiersprache der Robo Code App aus? Besprechen Sie folgende Begriffe: Befehl, Verbindung, Übergang, Schleife.*
- 
- Reflection & Feedback**  
7 – 10 min
- 7 Zusammenfassung:** Was ist Programmieren, was ist Code (Programm); Hauptblock und seine Funktionen; Geräusche und Verbindungen; sequenzielles Programm, Schleife; und die Projekte welche die SchülerInnen programmiert haben. **Feedback erhalten:** über die Schwierigkeit der Aufgaben.
  - 8 Aufräumen:** Die SchülerInnen sollen lernen auf die Geräte die sie verwenden aufzupassen und diese ordnungsgemäß aufzuräumen – RW Baukästen und Tablets.

# Einheit 3: Robo bringt das Klassenzimmer zum leuchten!

**Konzepte:** Sequenzielle Logik; RW visuell basiertes Programmieren: Befehl und Befehl-Einstellungen

**Schwierigkeitsgrad:** ★☆☆



## Story of Robo:

Robo freut sich sehr unser Freund zu sein und möchte eine kleine Party mit bunten Lichtern feiern! Können wir unseren Robo so programmieren, dass er das Klassenzimmer zum leuchten bringt?



## Unterrichtsziel:

Wir wollen Robo dabei helfen eine Party zu schmeißen. Bauen wir einen strahlenden Robo und programmieren ihn so, dass er in verschiedenen Farben leuchtet.

# Einheit 3: Robo bringt das Klassenzimmer zum leuchten!

## Module:



Main Block



RGB LED

## Programmiere:



Visuals



Constant Light



Blink

## Fokus:

- **Robotik:** Output – RGB Licht und dessen Funktion.
  - **Computerwissenschaften:** Nutzer Input – Eigenschaften der Befehle.
- 

## Lernziele:

- Die SchülerInnen können ein einfaches sequenzielles Programm kodieren, welches verschiedene visuelle Signale und unterschiedliche Einstellungen und Verbindungen zwischen diesen beinhaltet.

## Lernergebnisse:

- Die SchülerInnen können durch den Zusammenbau der Module einen Robo bauen, der leuchten kann.
  - Die SchülerInnen können die Robo Code App anwenden, um ein neues Projekt zu gestalten.
  - Die SchülerInnen können ein einfaches sequenzielles Programm erstellen, welches folgende visuelle Signale beinhaltet: Konstantes Licht, Blinklicht.
  - Die SchülerInnen können die Einstellungen der verschiedenen Lichter verändern.
- 

## Keywords:

- Programmierbildschirm, Befehlsleiste, Befehl, Verbindung, visuelle Signale: konstantes Licht, Blinklicht.
- 

## Zusatzmaterialien:

- Challenge Karte #3.
- Graphiken zum ausdrucken: Hauptblock, RGB Licht, visuelle Signale.
- *Optional: Lego™ Bausteine, Farbpapier, andere Materialien.*

# Einheit 3: Robo bringt das Klassenzimmer zum leuchten!

## Unterrichtsabschnitte:

- Einleitung**  
7 – 10 min
- 1 Fragen:** Was haben wir beim letzten Mal gemacht? Was ist ein Code (Programm)? Was haben wir zuletzt programmiert und warum? Welches Modul haben wir verwendet und warum?
  - 2 Analysieren:** Erzählen der Robo Story; Gemeinsames erarbeiten von **Problemstellung** und Lösungen zur Bewältigung.
- 
- Angeleitete Aufgabe**  
15 – 20 min
- 3 Verbinden** des Hauptblocks mit dem Tablet; **Abrufen erlernter Begriffe:** Mein Robo Bildschirm, Robo Projektbildschirm, Programmierbildschirm Neues Projekt Knopf und die Befehlsleiste.
  - 4 Bauen und Programmieren:** Bringen wir unserem Robo bei, verschieden zu leuchten. **Verwenden Sie die Challenge Karte #3**
    - Finden und anstecken des **RGB Lichtes** an den Hauptblock,
    - Finden der **visuellen Signale** auf der Befehlsleiste; Besprechen Sie die Farbe der Befehle in Bezug zum RGB Licht Modul,
    - Programmieren eines konstanten Licht-Befehls; Verwenden verschiedener Einstellungen: Farbe, Dauer, Helligkeit,
    - Programmieren eines **Blinklicht-Befehls**; Verwenden verschiedener Einstellungen: Farbe, Blink-Anzahl, Geschwindigkeit.
    - Vergleich: **Blinklicht** und **konstantes Licht**; Was sind die Unterschiede?
    - **Spielerisches Ausprobieren:** Programmieren der **visuellen Signale** mit unterschiedlichen **Einstellungen** und **Verbindungen** zwischen den Befehlen. **Besprechen Sie:** Wie beeinflussen die Einstellungen der Befehle was der Robo ausführt? Schalten Sie das Licht im Klassenzimmer aus und feiern sie zusammen eine kleine Robo Party mit vielen bunten Lichtern!
  - 5 Zusammenfassung** Konstantes Licht, Blinklicht und deren Einstellungen; Wie kann uns der Robo mit dem RGB Licht im Alltag unterstützen?
- 
- Selbstständige Aufgabe**  
15 – 20 min
- 6 Gestaltung eines eigenen Projektes:** Wie kann uns der Robo mit dem RGB Licht im Alltag unterstützen? Baue und programmiere deinen eigenen Robo und verwende dabei visuelle Signale, welche für eine bestimmte Situation hilfreich sind. Ihr könnt auch Materialien verwenden um euren Robo selbst zu gestalten.
- \* Zusatzaufgabe:** Inhalte über die **Befehlsdauer** und das **Unendlichkeitssymbol**; Stellen Sie Fragen: Was ist die Dauer eines Befehls? Was ist der Unterschied zwischen einer endlichen und einer unendlichen Befehlsdauer? Wie verändert sich dadurch, was der Robo ausführt?
- 
- Reflection & Feedback**  
7 – 10 min
- 7 Zusammenfassung:** visuelle Signale – konstantes Licht, Blinklicht und deren Einstellungen; sequenzielles Programm, Schleife und die Projekte welche die SchülerInnen programmiert haben. **Feedback erhalten:** über die Schwierigkeit der Aufgaben.
  - 8 Aufräumen:** Die SchülerInnen sollen lernen auf die Geräte die sie verwenden aufzupassen und diese ordnungsgemäß aufzuräumen – RW Baukästen und Tablets.

# Einheit 4: Das Robo-Auto fährt herum

**Konzepte:** Mechanisches Design; Sequentielle Logik; RW visuell basiertes Programmieren: Befehl, Verbindung, Schleife

**Schwierigkeitsgrad:** ★☆☆



## Story of Robo:

Robo ist sehr neugierig und möchte gerne die Welt um sich herum entdecken! Um zu reisen muss Robo aber zuerst das Fahren lernen. Wie wir wissen, können Roboter viele Dinge tun, aber nur, wenn wir sie programmieren!



## Unterrichtsziel:

Wir möchten Robo dabei helfen, Fahren zu lernen. Wir werden ein Rob-Auto bauen und programmieren.

## Einheit 4: Das Robo-Auto fährt herum

### Module:



Main Block



DC Motors



Wheels



Small Wheel



Connectors

### Programmieren:



Movement



Drive



Turn

### Fokus:

- **Robotik:** Output – DC Motoren und deren Funktionen.
- **Computerwissenschaften:** Nutzer Input – Eigenschaften der Befehle.

### Lernziele:

- Die SchülerInnen können ein einfaches sequenzielles Programm kodieren, welches folgende Befehle der Bewegung beinhaltet: Fahren, Drehen und Verbindungen zwischen diesen Befehlen; Sie können das Programm zu einer Schleife umprogrammieren.

### Lernergebnisse:

- Die SchülerInnen können die Module so zusammenbauen, dass sie ein Robo-Auto bauen.
- Die SchülerInnen können ein einfaches sequenzielles Programm kodieren, welches folgende Bewegungen beinhaltet: Drehen, Fahren.
- Die SchülerInnen können die Einstellungen der Dreh- und Fahr-Befehle ändern.
- Die SchülerInnen können eine Schleife erstellen.

### Keywords:

- Befehlsleiste, Befehl, Verbindung, Bewegung: Fahren, Drehen, Schleife, Design.

### Zusatzmaterialien:

- Challenge Karte #4.
- Grafiken zum ausdrucken: Hauptblock, DC Motoren, Bewegung: Fahren, Drehen.
- *Optional: Lego™ Bausteine, Farbpapier, andere Materialien.*

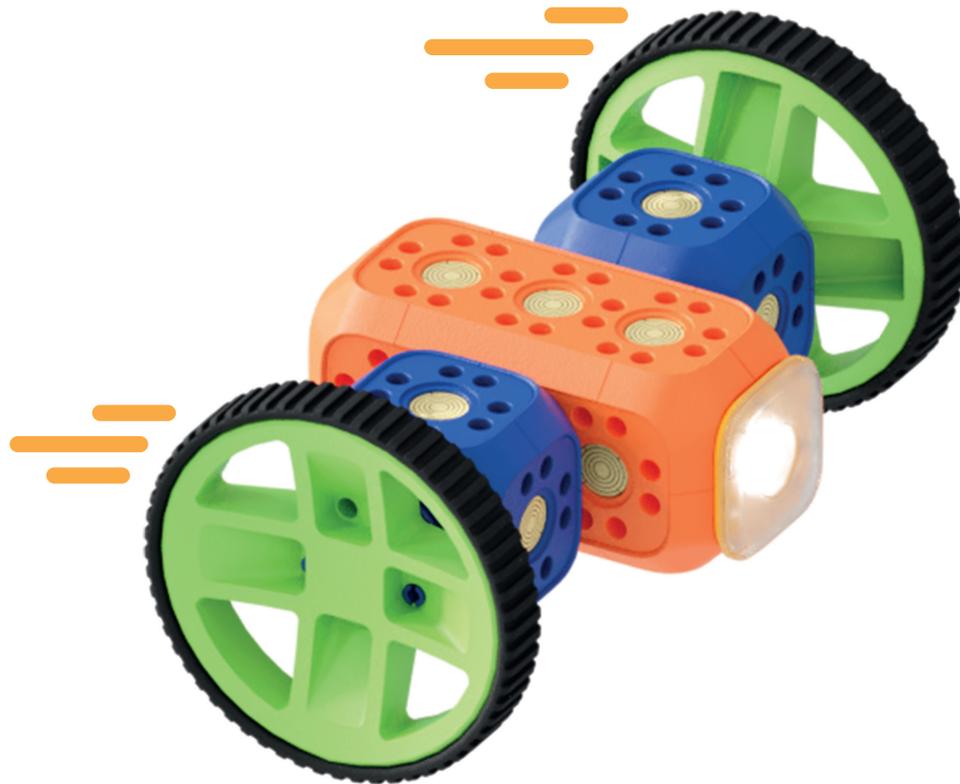
## Unterrichtsabschnitte:

- Einleitung**  
7 – 10 min
- 1 Fragen:** Welche Module haben wir bereits verwendet? Was kann euer Robo alles? Wie können wir das im Alltag einsetzen? Abrufen der Begriffe: Robotik, Programmieren, Programm (Code), kabellose Verbindung, Design.
  - 2 Analysieren:** Erzählen der Robo Story; Gemeinsames erarbeiten von **Problemstellung** und Lösungen zur Bewältigung.
- 
- Angeleitete Aufgabe**  
15 – 20 min
- 3 Verbinden** des Hauptblocks mit dem Tablet; **Abrufen** der Begriffe des Robo Code App Interface: Programmierbildschirm, Knöpfe, Befehlsleiste, Befehle, Verbindungen.
  - 4 Bauen und Programmieren:** Bring deinem Robo fahren bei! **Verwenden Sie die Challenge Karte #4**
    - Stecken Sie die **DC Motoren** an den Hauptblock. **Besprechen** Sie die mechanischen Details: das **Rotationsstück** ohne Pogo-Pins im Vergleich zu den **Verbindungsstücken**. Lassen Sie die SchülerInnen die DC Motoren unterschiedlich mit dem Hauptblock verbinden um zu sehen, ob die Befehle der Bewegung in der App erscheinen oder nicht. **Stellen Sie Fragen:** Wie sollten wir die DC Motoren verbinden und warum? **Zusammenfassen:** Die DC Motoren sollen mithilfe der Verbindungsstücke an den Hauptblock gesteckt werden, damit ihre Funktionen abgerufen werden können. Das Rotationsstück sollte mit den Rädern verbunden werden.
    - Finden der **Bewegungs-Befehle** auf der **Befehlsleiste**: Besprechen Sie die Farbe der Befehle in Bezug auf die Module der DC Motoren.
    - Programmieren des Fahr-Befehls; Verwendung verschiedener Einstellungen: Entfernung und Geschwindigkeit; Reverse Knopf.
    - Programmieren des Dreh-Befehls; Verwendung verschiedener Einstellungen: Winkel und Geschwindigkeit.
    - **Spielerisches Ausprobieren:** Programmieren der **Fahr- und Dreh-Befehle** mit unterschiedlichen **Einstellungen** und **Verbindungen** zwischen den Befehlen. **Besprechen Sie:** Wie beeinflussen die Einstellungen der Befehle was der Robo ausführt?
  - 5 Zusammenfassung** wie werden DC Motoren an den Hauptblock gesteckt und warum? Arten der Bewegung; deren Einstellungen; Welche anderen Roboter im echten Leben haben Motoren und was sind deren Aufgaben?
  - \* Zusatzaufgabe:** Vermitteln der Inhalte zum technischen Design. Lassen Sie Ihre SchülerInnen Variationen des Robo-Autos bauen und anschließend dasselbe Programm abspielen. Besprechen Sie: Wie beeinflusst die Bauweise des Robos dessen Ausführungen? Was heißt es, Robos Design effizienter zu gestalten?
- 
- Selbstständige Aufgabe**  
15 – 20 min
- 6 Gestaltung eines eigenen Projektes:** Wie kann uns der Robo mit den DC Motoren im Alltag unterstützen? Baue und programmiere deinen eigenen Robo und verwende dabei Bewegungs-Befehle, welche für eine bestimmte Situation hilfreich sind. Ihr könnt auch Materialien verwenden um euren Robo selbst zu gestalten.
- 
- Reflection & Feedback**  
7 – 10 min
- 7 Zusammenfassung:** Technische Details zu den DC Motoren – Fahr-, Dreh-Bewegungen und deren Einstellungen; sequenzielles Programm, Schleife und die Projekte welche die SchülerInnen programmiert haben. **Feedback erhalten:** über die Schwierigkeit der Aufgaben.
  - 8 Aufräumen:** Die SchülerInnen sollen lernen auf die Geräte die sie verwenden aufzupassen und diese ordnungsgemäß aufzuräumen – RW Baukästen und Tablets.

# Einheit 5: Robo's erste Reise

**Konzepte:** Parallele Ausführung; RW visuell basiertes Programmieren: Zustand; Mechanisches und Code Design

**Schwierigkeitsgrad:** ★★☆☆



## Story of Robo:

Jetzt ist es Zeit für Robos erste Reise! Wohin fährt euer Robo und warum? Wen wird er treffen? Welche Befehle braucht Robo für das Abenteuer?



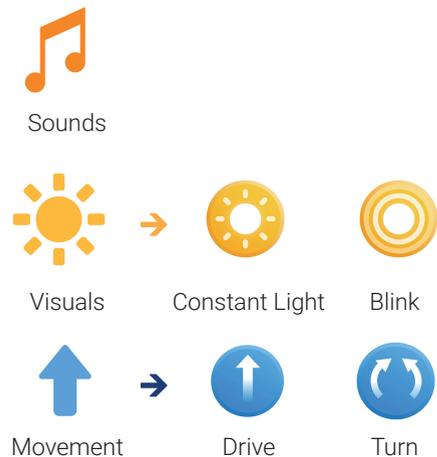
## Unterrichtsziel:

Um Robo auf seine erste Reise vorzubereiten, bauen und programmieren wir einen Reise-Robo. Dabei kombinieren wir alle Befehle, die wir bereits gelernt haben.

## Module:



## Programmiere:



## Fokus:

- **Robotik:** Output – Geräusche, RGB Licht, DC Motoren.
- **Computerwissenschaften:** Nutzer input – Eigenschaften der Befehle, Parallele Ausführung.

## Lernziele:

- Die SchülerInnen können ein Programm erstellen, in welchem parallele Ausführungen durch Geräusche, visuelle Signale und Bewegungen (Fahren, Drehen) beinhaltet sind. Sie können diese Befehle miteinander verbinden.

## Lernergebnisse:

- Die SchülerInnen können die Module so zusammenbauen, dass sie einen Robo bauen, der fahren kann.
- Die SchülerInnen können Befehle so programmieren, dass diese parallel ausgeführt werden und folgende Befehle beinhalten: Geräusche, visuelle Signale, Bewegungen (Fahren und Drehen).
- Die SchülerInnen können die Befehle miteinander verbinden.
- Die SchülerInnen können die Einstellungen für die unterschiedlichen Befehle verändern.

## Keywords:

- Befehlsleiste, Befehl, Geräusche, visuelle Signale, Bewegung.
- RW visuell basiertes Programmieren: Zustand, Befehlsdauer.

## Zusatzmaterialien:

- Challenge Karte #5.
- Grafiken zum ausdrucken: Hauptblock, RGB Licht, DC Motoren, Geräusche, visuelle Signale, Bewegung.
- *Optional: Lego™ Bausteine, Farbpapier, andere Materialien.*

# Einheit 5: Robo's erste Reise

## Unterrichtsabschnitte:

Einleitung  
7 – 10 min

- 1 **Fragen:** Welche Module haben wir bereits verwendet? Was kann euer Robo alles? Wie können wir das im Alltag einsetzen?
- 2 **Analysieren:** Erzählen der Robo Story; Gemeinsames erarbeiten von **Problemstellung** und Lösungen zur Bewältigung.

Angeleitete Aufgabe  
15 – 20 min

- 3 **Verbinden** des Hauptblocks mit dem Tablet; Abrufen: Hauptblock, RGB Licht, DC Motoren und RW visuell basiertes Programmieren.
- 4 **Bauen und Programmieren:** Bereitet euren Robo auf die erste Reise vor! **Verwenden Sie die Challenge Karte #5.**
  - **Bauen eines Reise-Robos;** Dazu wird der Hauptblock, RGB Licht und die DC Motoren verwendet.
  - Programmieren eines **sequenziellen Codes (Programmes):** Der Reise-Robo fährt, dreht sich und macht ein Geräusch.
  - Programmieren einer parallelen **Ausführung:** Ihre SchülerInnen sollen einen bestimmten Code programmieren – Der Robo macht ein Geräusch, jedes mal wenn er vorwärts fährt – gleichzeitig. Lassen Sie Ihre SchülerInnen selbstständig überlegen, wie das möglich ist: Durch das Zusammenfügen eines **Bewegungs-** mit einem **Geräusch-Befehl** in einem "Bubble" (**Zustand**) Besprechen Sie folgende Begriffe: **parallele Ausführung, Zustand.**
  - **Hinzufügen von visuellen Signalen** in einen Zustand; Lassen Sie die SchülerInnen versuchen weitere geräusche und Bewegungen in einen Zustand zu zeichnen. Besprechen Sie, wieso bestimmte **Arten von Befehlen** nicht hinzugefügt werden können.
  - **Programmieren von zwei Zuständen** und einer Verbindung zwischen diesen: Besprechen Sie wie der Übergang von einem zum anderen Zustand funktioniert.
  - **Spielerisches Ausprobieren:** Programmieren verschiedener Zustände mit **Geräuschen, visuellen Signalen, Bewegungen** und **Verbindungen** zwischen diesen.
- 5 **Zusammenfassung** Analysieren Sie die Unterschiede zwischen einer sequentiellen Ausführung und einer parallelen Ausführung; Wie können all diese Befehle, Robo auf seiner Reise helfen?
- \* **Zusatzaufgabe:** *Vermittlung der Inhalte über Code Design. Erinnern, was Design ist und wieso Menschen Dinge unterschiedlich designen. Stellen Sie Fragen: Können wir einen Code designen? Was bedeutet es einen Code zu designen?*

Selbstständige Aufgabe  
15 – 20 min

- 6 **Gestaltung eines eigenen Projektes:** Entscheide selbst, wo dein Roboter verreist. Gestalte das Umfeld des Ortes und programmiere deinen Robo dementsprechend und verwende dabei alle bereits erlernten Befehle um parallele Ausführungen zu programmieren.

Reflection & Feedback  
7 – 10 min

- 7 **Zusammenfassung:** Parallele Ausführung, Zustand. **Feedback erhalten:** über die Schwierigkeit der Aufgaben.
- 8 **Aufräumen:** Die SchülerInnen sollen lernen auf die Geräte die sie verwenden aufzupassen und diese ordnungsgemäß aufzuräumen – RW Baukästen und Tablets.

# Einheit 6: Robo reist in die Spielzeugstadt

**Konzepte:** Parallele Ausführung; RW visuell basiertes Programmieren: Zustand; Mechanisches und Code Design

**Schwierigkeitsgrad:** ★★ ★



## Story of Robo:

Robo wurde in die Spielzeugstadt eingeladen! Um dorthin zu kommen, muss Robo einen weiten Weg zurücklegen, Hindernissen ausweichen und knifflige Kurven fahren. Sind wir dazu bereit ein kompliziertes Programm zu erstellen um Robo zu helfen?



## Unterrichtsziel:

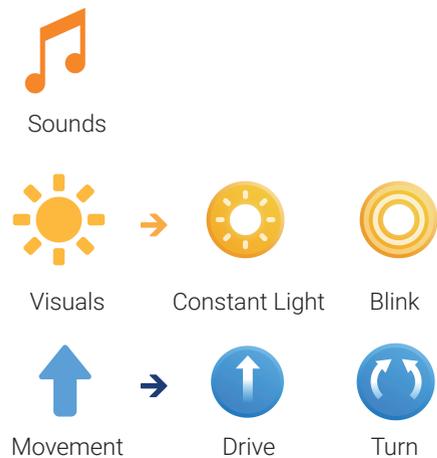
Wir wollen Robo dabei helfen in die Spielzeugstadt zu reisen. Dazu programmieren wir ein Robo-Fahrzeug, das Hindernissen ausweichen und Herausforderungen bewältigen kann.

# Einheit 6: Robo reist in die Spielzeugstadt

## Module:



## Programmiere:



## Fokus:

- **Robotik:** Output – Geräusche, RGB Licht, DC Motoren.
- **Computerwissenschaften:** Nutzer Input; parallele Ausführung.

## Lernziele:

- Die SchülerInnen können ein Programm erstellen, in welchem parallele Ausführungen durch Geräusche, visuelle Signale und Bewegungen (Fahren, Drehen) beinhaltet sind. Sie können diese Befehle verbinden, damit Robo Herausforderungen bewältigen kann.

## Lernergebnisse:

- Die SchülerInnen können die Module so zusammenbauen, dass sie ein Robo Fahrzeug bauen, welches fahren kann.
- Die SchülerInnen können eine Problemstellung analysieren und ein Programm kodieren, welches diese löst.
- Die SchülerInnen können Befehle so programmieren, dass diese parallel ausgeführt werden und folgende Befehle beinhalten: Geräusche, visuelle Signale, Bewegungen (Fahren und Drehen).
- Die SchülerInnen können die Befehle miteinander verbinden.
- Die SchülerInnen können die Einstellungen der unterschiedlichen Befehle verändern und dabei deren Position in einem Zustand berücksichtigen.

## Keywords:

- Befehlsleiste, Befehl, Geräusche, visuelle Signale, Bewegung.
- RW visuell basiertes Programmieren: Zustand, Befehlsdauer.

## Zusatzmaterialien:

- Challenge Karte #6.
- *Optional: Lego™ Bausteine, Farbpapier, andere Materialien.*

# Einheit 6: Robo reist in die Spielzeugstadt

## Unterrichtsabschnitte:

Einleitung  
7 – 10 min

- 1 Fragen:** Welche Module haben wir bereits verwendet? Welche Arten von Codes (Programmen) haben wir bereits programmiert? Inwiefern unterscheiden sich diese voneinander und wie beeinflusst das die Ausführungen von Robo?
- 2 Analysieren:** Erzählen der Robo Story; Gemeinsames erarbeiten von **Problemstellung** und Lösungen zur Bewältigung.

Angeleitete Aufgabe  
15 – 20 min

- 3 Gestaltung der Umgebung:** Verwenden Sie unterschiedliche Materialien um den Weg zur Spielzeugstadt mit verschiedenen Hindernissen zu gestalten – Bäume, Steine etc. – Verbinden des Hauptblocks mit der Robo Code App auf dem Tablet.
- 4 Bauen und Programmieren:** eines Robo-Fahrzeuges: Helfen wir Robo in die Spielzeugstadt zu gelangen! **Verwenden Sie die Challenge Karte #6.**
  - Programmieren eines **sequenziellen Codes:** Bewegung – Fahr- und Dreh-Befehle um Hindernissen auszuweichen und in die Spielzeugstadt zu gelangen.

**Bemerkung:** Es ist am besten, mit einfachen, konkreten Aufgaben zu beginnen und die Komplexität zu erhöhen, wenn die SchülerInnen besser werden. In diesem Fall erstellen die SchülerInnen einen einfachen Code und verbessern ihn schrittweise. Für einige Schüler könnte es einfacher sein, mit einem Code zu beginnen, der echte Robo-Bewegungen nachahmt, oder sogar alle Bewegungen selbst auszuprobieren, bevor sie programmiert werden.

  - **Anpassen des sequenziellen Codes:** fügen Sie visuelle Signale und Geräusche hinzu. Zum Beispiel: Robo beleuchtet die Straße jedes Mal vor dem Abbiegen. Robo macht einen Ton vor jedem Fahr-Befehl.
  - Programmieren einer **parallelen Ausführung:** Fügen Sie den Bewegungen visuelle Signale und Geräusche hinzu. Zum Beispiel: Robo beleuchtet die Straße jedes Mal, wenn er einen Dreh-Befehl ausführt – gleichzeitig; Robo gibt jedes Mal einen Geräusch von sich, wenn er einen Fahr-Befehl ausführt – gleichzeitig.
- 5 Zusammenfassung** Analysieren Sie den Unterschied zwischen einem sequentiellen Code und einer parallelen Ausführung; Wie können all diese Befehle, Robo helfen in die Spielzeugstadt zu reisen?
- \* Zusatzaufgabe:** *Kennenlernen der **Motor 1 und Motor 2 Befehle.** Programmieren eines **sequenziellen Codes** mit einem Motor 1 und einem Motor 2 Befehl und besprechen der Unterschiede zu Fahr- und Dreh-Befehlen. Verwenden Sie unterschiedliche Einstellungen der Befehle um herauszufinden, wie sich diese auf die Fahrbewegungen des Robos auswirken. Anpassen des Codes um durch die Motor 1 und Motor 2 Befehle eine **parallele Ausführung** herbeizuführen. Besprechen Sie, wieso es möglich ist zwei Bewegungs-Befehle in einen Zustand zu ziehen. Die SchülerInnen versuchen nun den ganzen Weg in die Spielzeugstadt zu fahren und dabei nur Motor 1 und Motor 2 Befehle zu verwenden.*

Selbstständige Aufgabe  
15 – 20 min

- 6 Gruppenaufgabe:** Gestalten einer Straße mit verschiedenen Hindernissen. Programmieren des Robos, sodass dieser die Straße bewältigen kann. Anschließende Präsentation des Projektes vor der Klasse.

Reflection & Feedback  
7 – 10 min

- 7 Feedback erhalten:** über die Schwierigkeit der Aufgaben.
- 8 Aufräumen:** Die SchülerInnen sollen lernen auf die Geräte die sie verwenden aufzupassen und diese ordnungsgemäß aufzuräumen – RW Baukästen und Tablets.

# Einheit 7: Robo schaut sich um

**Konzepte:** Sequentielle Logik, RW visuell basiertes Programmieren: Befehle und deren Einstellungen, Schleife; Mechanisches Design

**Schwierigkeitsgrad:** ★★☆☆



## Story of Robo:

Letztes Mal haben wir Robo geholfen, in die Spielzeugstadt zu reisen. Nun ist es an der Zeit, andere Spielzeuge zu treffen! Robo möchte lernen, sich umzusehen, um mit den zu Anderen zu kommunizieren.



## Unterrichtsziele:

Wir wollen Robo dabei helfen sich umzusehen. Dazu werden wir einen Robo mit Kopf bauen und ihn so programmieren, dass er diesen drehen kann.

## Module:



Main Block



DC Motors



Servo



Distance  
Sensor



Connectors

## Programmiere:



Movement



Motor



Servo

## Fokus:

- **Robotik:** Servo und DC Motoren.
- **Computerwissenschaften:** Nutzer Input; Eigenschaften der Befehle.

## Lernziele:

- Die SchülerInnen können ein einfaches sequentielles Programm erstellen, das folgende Bewegungen beinhaltet: Servo oder Motor 1 Bewegung. Sie verstehen den Unterschied zwischen dem Servo und dem DC Motor.

## Lernergebnisse:

- Die SchülerInnen verstehen den Unterschied zwischen Servo und DC Motoren und deren unterschiedliche Funktionen.
- Die SchülerInnen können Module so zusammenbauen, dass sie entweder den Servo oder den DC Motor für die Bewegung des Kopfes verwenden.
- Die SchülerInnen können ein einfaches sequentielles Programm erstellen, welches folgende Bewegungen beinhaltet: Entweder Servo oder Motor 1 Bewegung.
- Die SchülerInnen können die Einstellungen für die jeweiligen Befehle verändern.

## Keywords:

- Bewegung: Servo und Motor 1 Befehle; Entfernung, Winkel, Null-Position.
- Mechanisches und Code Design.

## Zusatzmaterialien:

- Challenge Karte #7.
- Grafiken zum ausdrucken: Hauptblock, DC und Servo Motoren
- *Optional: Lego™ Bausteine, Farbpapier, andere Materialien.*

# Einheit 7: Robo schaut sich um

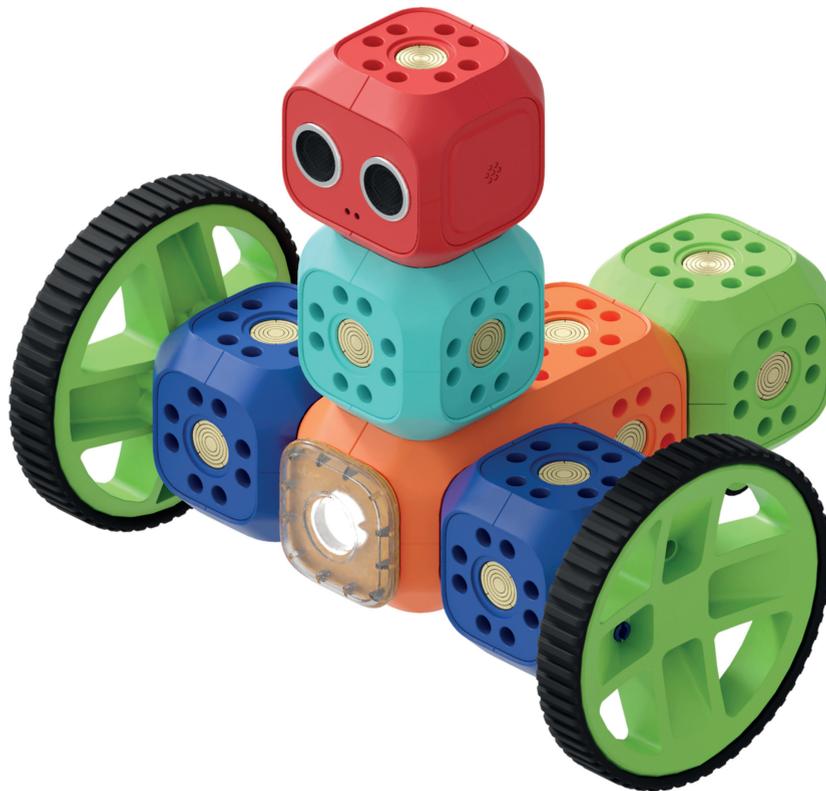
## Unterrichtsabschnitte:

- Einleitung**  
7 – 10 min
- 1 Fragen:** Was waren die letzten Projekte? Welche Module haben wir verwendet? Welche Funktion hatten dabei die DC Motoren? In welchen Geräten werden Motoren im echten Leben verwendet? Sind alle Motoren gleich?
  - 2 Analysieren:** Erzählen der Robo Story; Gemeinsames erarbeiten von **Problemstellung** und Lösungen zur Bewältigung.
- 
- Angeleitete Aufgabe**  
15 – 20 min
- 3 Vorbereitung:** Verbindung des Hauptblocks mit dem Tablet und Erstellung eines neuen Projektes.
  - 4 Bauen und Programmieren:** : Lernen wir unserem Robo sich umzuschauen! **Verwenden Sie die Challenge Karte #7**
    - Finden der **DC- und Servo Motoren**; Vergleich der unterschiedlichen Farben und des mechanischen Designs: Schauen sie gleich aus, oder gibt es Unterschiede?
    - Verbinden und Programmieren eines **DC Motors** – Programmieren des Motor 1 Befehls, und Analyse der Bewegung: Der Kopf wird mehrere Male im Kreis gedreht
    - Verbinden und Programmieren des **Servo Motors** – Programmieren des Servo Befehls, und Analyse der Bewegung: Der Kopf wird zu einem bestimmten Winkel gedreht
    - Abspielen desselben **Servo Befehls** (mit denselben Einstellungen) Stellen Sie Fragen: Dreht der Robo jetzt seinen Kopf? Warum nicht?
    - Anpassung des **Winkels** in den Einstellungen des **Servo Befehls** und besprechen Sie folgende Begriffe: **Grad, Winkel, Null-Position**; Was bedeutet es die Null-Position einzustellen?
    - Ausprobieren der **Zufalls Funktion** des Servo Befehls und besprechen Sie: Was bedeutet "zufällig"? Wer bestimmt die Einstellung, wenn "zufällig" eingestellt ist? Wie können wir zufälligen Bereich bestimmen, wenn die Zufall Funktion aktiviert ist?
    - **Spielerisches Ausprobieren:** Programmieren verschiedener **Servo Befehle** und Verbindungen zwischen diesen, sodass der Robo seinen Kopf dreht oder schüttelt. Ersetzen des Servo Motors mit einem DC Motor um die Unterschiede der beiden Motoren zu thematisieren.
  - 5 Zusammenfassung** Die Unterschiede zwischen den DC und den Servo Motoren, deren Einstellungen und den Auswirkungen auf die Ausführung des Robos.
- 
- Selbstständige Aufgabe**  
15 – 20 min
- 6 Gestaltung eines eigenen Projektes:** Wie kann der Servo Motor im Alltag eingesetzt werden? Programmieren deinen eigenen Robo und verwende dabei den Servo Motor, welcher für eine bestimmte Situation hilfreich ist. Ihr könnt auch Materialien verwenden um euren Robo selbst zu gestalten.
  - \* Zusatzaufgabe:** *Vermittlung der Begriffe des technischen and Code Designs anhand von Motoren. Verwendung beider DC Motoren und des Servo Motors um den Robo passend zu seinen Fähigkeiten zu bauen und zu programmieren. Besprechen Sie: Welcher der beiden Motoren ist für welche Aufgaben besser geeignet? Wie unterscheiden sich die beiden Motoren voneinander? Ist es möglich beide Motor-Befehle in einem Zustand zu programmieren? Wieso?*
- 
- Reflection & Feedback**  
7 – 10 min
- 7 Zusammenfassung:** Technische Details der DC Motoren und des Servo Motors, Servo Befehl und dessen Einstellungen (Grad, Winkel, Zufalls Funktion, Null-Position). **Feedback erhalten:** über die Schwierigkeit der Aufgaben.
  - 8 Aufräumen:** Die SchülerInnen sollen lernen auf die Geräte die sie verwenden aufzupassen und diese ordnungsgemäß aufzuräumen – RW Baukästen und Tablets.

# Einheit 8: Robo lernt Freunde kennen

**Konzepte:** Sequentielle Logik und Parallele Ausführung; RW visuell basiertes Programmieren: Code Design

**Schwierigkeitsgrad:** ★★ ★



## Story of Robo:

Heute ist ein ganz besonderer Tag für unseren Robo: Wir helfen Robo, andere Spielzeuge kennenzulernen und mit ihnen zu kommunizieren! Welche Module und Befehle wird Robo dafür benötigen?



## Unterrichtsziel:

Wir wollen Robo dabei helfen mit anderen Spielzeugen kommunizieren zu können. Wir werden dazu alle Module die wir kennengelernt haben verwenden und einen Robo bauen und programmieren, der viele verschiedene Befehle ausführen kann.

# Einheit 8: Robo lernt Freunde kennen

## Module:



Main Block



RGB LED



Distance Sensor



Servo



DC Motors

x2



Wheels

x2



Small Wheel



Connectors

x4



Connector Block

## Programmiere:



Sounds



Visuals



Constant Light



Blink



Movement



Drive



Turn



Servo

## Fokus:

- **Robotik:** Outputs – Geräusche, RGB Licht, DC Motoren, Servo Motor.
- **Computerwissenschaften:** Nutzer Input; Eigenschaften der Befehle.

## Lernziele:

- Die SchülerInnen können ein Programm erstellen, in welchem sequenzielle und parallele Ausführungen durch Geräusche, visuelle Signale und Bewegungen (Fahren, Drehen) beinhaltet sind. Sie können diese Befehle miteinander verbinden.

## Lernergebnisse:

- Die SchülerInnen können einen Robo bauen, der eine Reihe bestimmter Aufgaben ausführen kann.
- Die SchülerInnen können ein Programm erstellen, welches sequenzielle und parallele Ausführungen beinhaltet. Sie verwenden dazu: Geräusche, visuelle Signale und Bewegung und können diese miteinander verbinden.
- Die SchülerInnen können die Einstellungen unterschiedlicher Befehle verändern: konstantes Licht, Blinklicht, Fahren, Drehen und Servo und berücksichtigen dabei deren Position in den jeweiligen Zuständen.

## Keywords:

- Befehl, Zustand, Verbindung, Übergang, Befehlsdauer.
- Code Design.

## Zusatzmaterialien:

- Challenge Karte #8.
- *Optional: Lego™ Bausteine, Farbpapier, andere Materialien.*

## Unterrichtsabschnitte:

Einleitung  
7 – 10 min

- 1 **Fragen:** Welche Module haben wir bereits verwendet? Was kann euer Robo alles? Wozu benötigt Robo diese Funktionen und Befehle?
- 2 **Analysieren:** Erzählen der Robo Story; Gemeinsames erarbeiten von **Problemstellung** und Lösungen zur Bewältigung.

Angeleitete Aufgabe  
15 – 20 min

- 3 **Gestaltung der Umgebung:** Verwendung unterschiedlicher Materialien um die Spielzeugstadt und deren Bewohner zu gestalten. **Verbinden** der Hauptblocks mit dem Tablet und der Robo Code App.
- 4 **Bauen und Programmieren:** Lass deinen Robo sich umschauchen! **Verwenden Sie die Challenge Karte #8.**
  - Programmieren eines **sequenziellen Codes:** Geräusche – Robo spricht mit anderen Spielzeugen; visuelle Signale – Robo verwendet verschiedene Lichter um seine Freunde zu begrüßen; und Bewegung – Robo besucht ein Spielzeug nach dem anderen um alle seine Freunde zu treffen; Robo bewegt seinen Kopf um nach seinen Freunden zu sehen.
  - **Anpassen** des **sequenziellen Codes:** Änderung der Einstellungen für einige Befehle; Ausprobieren der zufälligen Einstellungen für visuelle Signale – konstantes Licht und Blinklicht, Servo Befehle.
  - Programmieren einer **parallelen Ausführung:** Zusammenfügen mehrerer Befehle in einem Zustand – Robo führt einige Befehle gleichzeitig aus.
  - **Spielerisches Ausprobieren:** Programmieren des Robos um verschiedene Aufgaben auszuführen und mit anderen Spielzeugen zu interagieren.
- 5 **Zusammenfassung** Was ist der Unterschied zwischen einem sequentiellen Code und einer parallelen Ausführung? Wie können die verschiedenen Befehle Robo helfen um mit anderen Spielzeugen zu kommunizieren?
- \* **Zusatzaufgabe:** *Vermittlung der folgenden Inhalte zur **Zufalls Funktion:** Abrufen und erinnern, wie eine zufällige Einstellung programmiert wird. Finden aller Befehle, die eine Zufalls Funktion haben – konstantes Licht, Blinklicht, Drehen, Servo Befehle und teste diese in unterschiedlichen Kombinationen: Als sequentiellen Code und als parallele Ausführung. Wie beeinflusst die Zufalls Funktion die Programmausführungen?*

Selbstständige Aufgabe  
15 – 20 min

- 6 **Gruppenarbeit:** Gestaltet die Spielzeugstadt mit unterschiedlichen Einwohnern und deren Geschichten – einigen Aufgaben für Robo; Baut und programmiert euren Robo um die Aufgaben zu lösen. Präsentiert im Anschluss euer Projekt vor der Klasse

Reflection & Feedback  
7 – 10 min

- 7 **Feedback erhalten:** über die Schwierigkeit der Aufgaben.
- 8 **Aufräumen:** Die SchülerInnen sollen lernen auf die Geräte die sie verwenden aufzupassen und diese ordnungsgemäß aufzuräumen – RW Baukästen und Tablets.

# Einheit 9: Robo macht eine Überraschung

**Konzepte:** Technischer Entwicklungsprozess; Mechanisches und Code Design

**Schwierigkeitsgrad:** ★★ ★



## Story of Robo:

Robo freut sich darüber neue Freunde kennengelernt zu haben und möchte deshalb seine neuen Freunde überraschen.



## Unterrichtsziel:

Wir möchten Robo dabei helfen seine neuen Freunde zu überraschen. Wir werden dazu lernen, wie wir neue Designs konstruieren können und diese dann in einem Robo Projekt anwenden.

# Einheit 9: Robo macht eine Überraschung

## Module:



Main Block



DC Motors



Servo



RGB LED



Distance Sensor



Wheels



Small Wheel



Connector



Connector Flat Lego™



Connector Block

## Programmieren:



Sounds



Visuals



Constant Light



Blink



Movement



Drive



Turn



Servo

## Fokus:

- **Robotik:** Outputs – Geräusche, RGB Licht, DC Motoren, Servo Motor.
- **Computerwissenschaften:** Nutzer Input; abhängig von der Aufgabe.

## Lernziele:

- Die SchülerInnen können ein Programm erstellen, in welchem sequentielle und parallele Ausführungen durch Geräusche, visuelle Signale und Bewegungen beinhaltet werden. Sie können diese Befehle, abhängig von der Aufgabe, miteinander verbinden.

## Lernergebnisse:

- Die SchülerInnen wissen was ein technischer Entwicklungsprozess ist
- Die SchülerInnen können den einzelnen Schritten des technischen Entwicklungsprozess folgen und den Aufgaben nachgehen.
- Die SchülerInnen können Module so nutzen, dass sie einen individuellen Roboter zusammenbauen und dieser die gewünschten Aufgaben ausführen kann.
- Die SchülerInnen können ein Programm erstellen, welches sequenzielle und parallele Ausführungen beinhaltet. Sie verwenden dazu: Geräusche, visuelle Signale und Bewegung und können diese miteinander verbinden.

## Keywords:

- Technischer Entwicklungsprozess; Mechanisches und Code Design.

## Zusatzmaterialien:

- Karten zum ausdrucken mit den Ingenieur Design Prozess Schritten.
- Challenge Karte #9: Anzahl der Challenges für die SchülerInnen.
- *Optional: Lego™ Bausteine, Farbpapier, andere Materialien.*

## Unterrichtsabschnitte:

Einleitung  
7 – 10 min

- 1 Fragen:** Wie viele Projekte haben wir mit Robo gemacht? Wann haben wir Robo bei einem Problem geholfen? Was war der Grund für das Projekt? Wie haben wir das gemacht? Habt ihr schon mal ein eigenes Projekt in der Schule oder im täglichen Leben erstellt? Wie hast du es gemacht? Ist dieser Prozess ähnlich wie bei Robo? Denkt ihr, dass es einen zusammenhängenden Plan für die Erstellung eines Projekts gibt?
- 2 Analysieren:** Erzählen der Robo Story; Gemeinsames erarbeiten von **Problemstellung** und Lösungen zur Bewältigung.

Angeleitete Aufgabe  
15 – 20 min

- 3** Vermittlung der Inhalte zum Ingenieur Design Prozess. Fragen Sie Ihre SchülerInnen, wie sie ein Projekt zu den jeweiligen Aufgaben gestalten würden; Jeder Schritt wird gemeinsam mit den SchülerInnen bestimmt und die Karten für die jeweiligen Schritte an die Tafel gehängt. Vermittlung folgender Begriffe: **technisches und Code Design, Technischer Entwicklungsprozess** **Verwenden Sie die Challenge Karte #9 und die Karten zum ausdrucken mit den einzelnen Schritte**
  - **Schritt 1. Identifizieren einer Problemstellung:** Was ist das Problem oder die Idee? Definieren des Zweckes eines bestimmten Robo-Projekts. Sage es laut oder notiere es dir und erkläre den Grund für die Problemstellung.
  - **Schritt 2. Brainstorm: Welche Lösungen gibt es?** Brainstrome so viele Lösungen wie möglich, aber bewerte sie nicht. In diesem Schritt müssen die Lösungen nicht unbedingt gut sein.
  - **Schritt 3. Evaluieren und wähle eine Lösung: Was wäre wenn...?** Überlege welche Vor- und Nachteile die einzelnen Lösungen haben. Wähle die beste Lösung.
  - **Schritt 4. Skizziere und Plane: Was wird benötigt?** Mache dir eine Skizze und entscheide, welche Robo Module und anderen Materialien du für dein Projekt benötigst.
  - **Schritt 5. Arbeite an deiner Lösung: baue, programmiere, teste, wiederhole!** Arbeite an deiner Idee und probiere sie aus. Wenn die erste Lösung nicht funktioniert, überlege was der Grund dafür sein könnte. Es ist wichtig, es so lange zu versuchen, bis das Problem gelöst ist. Verliere nicht die Motivation, nicht alle Probleme sind leicht zu lösen.
  - **Schritt 6. Der Feinschliff: Ist alles fertig und bereit?** Sobald du die beste Lösung für dein Projekt gefunden hast, wird es abgeschlossen: Erstellen einer Umgebung für den Robo; Überprüfung, ob alles für eine Präsentation bereit ist.
  - **Schritt 7. Präsentation der Lösung:** Präsentiere dein Projekt vor der Klasse und frag nach Feedback.
  - **Schritt 8. Reflektiere: Wie war das Projekt für dich?** Überlege dir, sobald das Problem gelöst ist: Was hat funktioniert? Was nicht? Was kannst du beim nächsten Mal anders machen?

Selbstständige Aufgabe  
15 – 20 min

- 4 Üben des** Ingenieur Design Prozess : Besprecht die Robo Story und durchläuft alle Schritte in Bezug auf die Situation. Plant dadurch die eigenständige Aufgabe.
- 5 Einzel- oder Gruppenarbeit:** Gestaltet die Spielzeugstadt mit unterschiedlichen Einwohnern und deren Geschichten – einigen Aufgaben für Robo; Baut und programmiert euren Robo um die Aufgaben zu lösen. Präsentiert im Anschluss euer Projekt vor der Klasse
- 6 Präsentation:** Die SchülerInnen präsentieren ihre Projekte im Klassenzimmer, geben und erhalten konstruktives Feedback.

# Einheit 10: Was macht dein Robo?

**Konzepte:** Technischer Entwicklungsprozess, Mechanisches und Code Design

**Schwierigkeitsgrad:** ★★ ★



## Story of Robo:

Jetzt ist es an der Zeit selbst zu entscheiden, welchen Robo ihr bauen möchtet. Baut und programmiert euren eigenen Roboter und verwendet dabei alle Module, Verbindungsstücke und Befehle, die ihr bis jetzt gelernt habt. Denkt daran einen Robo zu erstellen, der uns im Alltag helfen kann. Erklärt am Ende wieso ihr den Robo auf eure Art und Weise zusammengebaut und programmiert habt.



## Unterrichtsziel:

Um deine Robo in dein eigenes Projekt zu verwandeln, folge den Schritten des Ingenieur Design Prozesses (Konstruktionsprozesses)..

# Einheit 10: Was macht dein Robo?

## Module:



Main Block



DC Motors



Servo



RGB LED



Distance Sensor



Wheels



Small Wheel



Connector



Connector Flat Lego™



Connector Block

## Programmieren:



Sounds



Visuals



Constant Light



Blink



Movement



Drive



Turn



Servo

## Fokus:

- **Robotik:** Outputs – Geräusche, RGB Licht, DC Motoren, Servo Motor.
- **Computerwissenschaften:** Nutzer Input; abhängig von der Aufgabenstellung.

## Lernziele:

- Die SchülerInnen können ein einfaches Programm mit parallelen Befehlen erstellen welches Geräusche, visuelle Signale (Konstantes Licht, Blinklicht) und Bewegung (Motorbefehl 1 & 2, Fahren, Drehen) beinhaltet.

## Lernergebnisse:

- Die SchülerInnen können eine Geschichte und einen Grund finden um einen speziellen Robo zu bauen.
- Die SchülerInnen kennen die Technischer Entwicklungsprozess und können den Schritten folgen um die Aufgabe zu erfüllen.
- Die SchülerInnen können eine eigene Robo Story erstellen, ein eigenes Robo-Projekt bauen und anschließend programmieren.
- Die SchülerInnen können mit verschiedenen Materialien ein Umfeld für den Robo gestalten.
- Die SchülerInnen können ihr eigenes Projekt vor der Klasse präsentieren und konstruktives Feedback zu anderen Projekten geben.

## Keywords:

- Projekt, Technischer Entwicklungsprozess; Mechanisches und Code Design.

## Zusatzmaterialien:

- Karten zum ausdrucken mit den Ingenieur Design Prozess Schritten.
- Challenge Karte #10: Ideen für SchülerInnen-Projekte.
- *Optional: Lego™ Bausteine, Farbpapier, andere Materialien.*

# Einheit 10: Was macht dein Robo?

## Unterrichtsabschnitte:

**Einleitung**  
7 – 10 min

- 1 Fragen:** erinnert ihr euch an das letzte Projekt? und an den konstruktions Design Prozess, den wir beim letzten Mal kennengelernt haben?
- 2 Analysieren:** Erzählen der Robo Story; Gemeinsames erarbeiten von **Problemstellung** und Lösungen zur Bewältigung.

**Angeleitete Aufgabe**  
15 – 20 min

- 3** Planung der einzelnen Schritt für eine eigenständige Aufgabe, gemeinsam mit den SchülerInnen. Abrufen der einzelnen Schritte des Ingenieur Design Prozesses und aufhängen der jeweiligen Karten an der Tafel.
  - **Schritt 1. Identifizieren einer Problemstellung**
  - **Schritt 2. Brainstorm: Welche Lösungen gibt es?**
  - **Schritt 3. Evaluieren und wähle eine Lösung: Was wäre wenn...?**
  - **Schritt 4. Skizziere und Plane: Was wird benötigt?**
  - **Schritt 5. Arbeite an deiner Lösung: Baue, Programmier, Teste, Wiederhole!**
  - **Schritt 6. Der Feinschliff: Ist alles fertig und bereit?**
  - **Schritt 7. Präsentation der Lösung**
  - **Schritt 8. Reflektiere: Wie war das Projekt für dich?**

**Selbstständige Aufgabe**  
15 – 20 min

- 4 Gestaltung des eigenen Projektes:** Die SchülerInnen gestalten mithilfe der Karten an der Tafel ihre eigenen Robo-Projekte. Sie können dabei alleine, in Paaren und in Kleingruppen arbeiten und Zusatzmaterialien verwenden um die Umgebung und den Roboter individuell zu gestalten. Die SchülerInnen können auch miteinander kooperieren und ein gemeinsames Projekt mit mehreren Robotern gestalten.
- 5 Präsentation:** Die SchülerInnen präsentieren ihre Projekte im Klassenzimmer, geben und erhalten konstruktives Feedback.

**Reflection & Feedback**  
7 – 10 min

- 7 Feedback geben und erhalten:** über die Schwierigkeit der Aufgaben.
- 8 Aufräumen:** Die SchülerInnen sollen lernen auf die Geräte die sie verwenden aufzupassen und diese ordnungsgemäß aufzuräumen – RW Baukästen und Tablets.