

# Arduino programmieren mit ArduBlock



**Verfasser:** Sascha Ratner

**Kurz-Info:**

Informatischer Inhalt: Grundlagen der Programmierung und der technischen Informatik

Jahrgangsstufe: Unterstufe

Vorwissen: keinerlei Vorkenntnisse notwendig

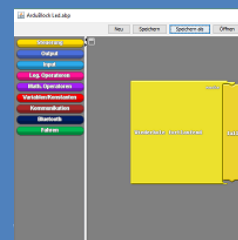
# KURZINFORMATION FÜR DIE LEHRKRAFT

**Titel:** *Arduino programmieren mit ArduBlock*

**Schulstufe:** *Unterstufe*

**optimale Jahrgangsstufe:** *Klasse 5 bis 7*

**Themenbereiche:** *grafische Programmierung, Kontrollstrukturen, Variablenkonzepte, einfache elektrische Schaltungen, einfache Komponenten (Widerstände, LEDs)*



## EINORDNUNG IN GESETZLICHE RAHMENBEDINGUNGEN

**Bildungsstandards der GI:** *Modellieren und Implementieren, Begründen und Bewerten, Strukturieren und Vernetzen, Kommunizieren und Kooperieren, Algorithmen und Informatiksysteme*

**Einbindung in den Unterricht:** Das Modul kann als Einstieg in die Programmierung genutzt werden, da allgemeine Kontrollstrukturen und Variablenkonzepte thematisiert werden. Aufgrund der zusätzlichen Herausforderungen durch die elektrischen Schaltungen empfiehlt es sich bereits im Vorfeld erste Erfahrungen mit einer grafischen Programmierumgebung (z. B. Scratch) gesammelt zu haben. In diesem Modul wird die grafische Programmieroberfläche ArduBlock genutzt, die es erlaubt Programme mithilfe von vorgefertigten Befehlsbausteinen in Form von Puzzleteilen zusammensetzen. Außerdem kann das Modul als Ergänzung zum Fach Physik oder Naturwissenschaften im Bereich „elektrische Schaltungen“ eingesetzt werden. Die Schülerinnen und Schüler bauen selbstständig einfache Schaltungen auf und lernen auch die physikalischen Hintergründe kennen.

**Vorkenntnisse:** keine (bzw. erste Erfahrungen mit grafischer Programmierung von Vorteil)

**Dauer:** *4 Stunden*

**Inhaltsbeschreibung:** Die Schülerinnen und Schüler lernen wie infrarotes Licht beim Einparken helfen kann, wie man damit eine Türe automatisch öffnet oder wie man eine automatische Rollladensteuerung realisiert. Sie bauen die dafür benötigten Schaltungen selbst und schreiben die entsprechenden Programme. Im Gegensatz zum verwandten Modul „Informatik Enlightened“ wird in diesem Modul der Klassenstufe entsprechend auf textuelle Programmierung verzichtet. Stattdessen nutzen die Schülerinnen und Schüler die grafische Programmieroberfläche ArduBlock und können so schnell und ohne Programmierkenntnisse die entsprechenden Programme zusammensetzen.

## INHALT

Kurzinformation für die Lehrkraft.....	2
Lernziele .....	4
Fachliche Analyse .....	4
Einordnung in gesetzliche Rahmenbedingungen .....	4
Lehrplan Informatik .....	4
Bildungsstandards der GI .....	4
Benötigte unterrichtliche Voraussetzungen .....	5
Einbettung in den Schulunterricht .....	5
Didaktische/Methodische Schwerpunktsetzung.....	5
Verlaufsplan des Moduls.....	7
Quellenverzeichnis .....	9
Abbildungsverzeichnis .....	9

## LERNZIELE

Die Schülerinnen und Schüler..

- erläutern die Funktion der grundlegenden Programmierkonzepte: Schleifen und Kontrollstrukturen.
- können die Funktionsweise von Programmen in eigenen Wort wiedergeben.
- implementieren mithilfe der grafischen Programmieroberfläche ArduBlock eigene Programme für den Arduino.
- planen und bauen einfache elektrische Schaltungen auf.
- präsentieren ihre Ergebnisse.

## FACHLICHE ANALYSE

Folgende fachliche Bereiche werden thematisiert:

- grafische Programmierung ([https://de.wikipedia.org/wiki/Visuelle\\_Programmiersprache](https://de.wikipedia.org/wiki/Visuelle_Programmiersprache))
- Schleifen ([https://de.wikipedia.org/wiki/Schleife\\_\(Programmierung\)](https://de.wikipedia.org/wiki/Schleife_(Programmierung)))
- elektrische Schaltungen ([https://de.wikipedia.org/wiki/Elektrische\\_Schaltung](https://de.wikipedia.org/wiki/Elektrische_Schaltung))
- Mikrocontroller (<https://de.wikipedia.org/wiki/Mikrocontroller>)

Weiterführende Links zur Programmierumgebung ArduBlock und zum Arduino:

- <http://blog.ardublock.com/>
- <https://www.arduino.cc/>
- [https://de.wikipedia.org/wiki/Arduino\\_\(Plattform\)](https://de.wikipedia.org/wiki/Arduino_(Plattform))

## EINORDNUNG IN GESETZLICHE RAHMENBEDINGUNGEN

### LEHRPLAN INFORMATIK

- Das Modul knüpft an die Kompetenzbereiche **Implementieren** und **Modellieren** an. Wenn die Schülerinnen und Schüler bisher keine Programmierkenntnisse erworben haben, können sie zu Beginn und während des Moduls Grundlagen der grafischen Programmierung erwerben. Hierbei werden einfache Kontrollstrukturen, Methoden und Variablen erläutert, die die Schülerinnen und Schüler nachher anhand von beispielhaften Modellen (z. B. das Farbthermometer) selbst anwenden.
- Im Bereich der Inhaltsfelder kann in diesem Modul sowohl an **Informatiksysteme** als auch an **Informatik, Mensch und Gesellschaft** angeknüpft werden. Die Begriffe „analog“ und „digital“ bzw. „digitale und analoge“ Anschlüsse werden erläutert, sodass die Schülerinnen und Schüler diese anwenden können und den Unterschied erkennen. Zusätzlich tragen alltagsnahe Beispiele (Einparkhilfe, Türsteuerung) dazu bei, den Schülerinnen und Schülern zu vermitteln, dass ihnen Informatik nahezu überall begegnet.

### BILDUNGSSTANDARDS DER GI

- Dieses Modul ist für Schülerinnen und Schüler der Unter- und Mittelstufe geeignet (dabei bietet sich das verwandte Modul „Informatik Enlightened“ mit textueller Programmierung zur späteren Vertiefung an).
- Das Modul knüpft hierbei an die folgenden Inhaltsbereiche an:

- **Informatik, Mensch und Gesellschaft**
- **Algorithmen**
- **Informatiksysteme** (hier insb. Mikrocontroller)
- Außerdem umfasst es die folgenden Prozessbereiche:
  - **Modellieren und Implementieren**
  - **Begründen und Bewerten**
  - **Strukturieren und Vernetzen** (hier insb. Verknüpfung zu anderen Fächern wie Physik durch den Bau von elektrischen Schaltungen – fächerübergreifendes Lernen)
  - **Kommunizieren und Kooperieren**

## BENÖTIGTE UNTERRICHTLICHE VORAUSSETZUNGEN

- Grundsätzlich werden für dieses Modul keine Vorkenntnisse benötigt, da in den dazugehörigen Materialien alle nötigen Informationen vorhanden sind.
- Vorwissen in grafischer Programmierung (z. B. mit Scratch) ist dennoch von Vorteil.
- Kenntnisse über Stromkreise und einfache elektrische Bauteile und Schaltungen sind hilfreich.

## EINBETTUNG IN DEN SCHULUNTERRICHT

Dieses Modul kann als Einstieg in die Programmierung genutzt werden, da wichtige Grundlagen der Programmierung wie Kontrollstrukturen vermittelt werden. Aufgrund der zusätzlichen Herausforderungen durch die elektrischen Schaltungen empfiehlt es sich bereits im Vorfeld erste Erfahrungen mit einer grafischen Programmierumgebung (z. B. Scratch) gesammelt zu haben. Durch den Einsatz der grafischen Programmieroberfläche ArduBlock eignet es sich besonders für Schülerinnen und Schüler in der Unterstufe und schafft so eine solide Grundlage, um in Zukunft auf textuelle Programmierung umzusteigen. Darüber hinaus bietet es sich als praxisorientierte Vertiefung für grafische Programmiersprachen an, z. B. im Anschluss an eine Scratch-Unterrichtseinheit.

Außerdem bekommen die Schülerinnen und Schüler durch alltägliche Beispiele direkt einen lebensnahen Bezug zur Programmierung und lernen wie vielfältig Informatik sein kann. Das Modul eignet sich auch zur Einbettung in den Physikunterricht und somit zum fächerübergreifenden Lernen, da die Schülerinnen und Schüler keinerlei Programmierkenntnisse benötigen und selbstständig einfache Stromkreise aufbauen und diese anschließend in praktischen Beispielen nutzen.

## DIDAKTISCHE/METHODISCHE SCHWERPUNKTSETZUNG

### **Beschreibung des Modulablaufs**

#### *Einstieg*

Zu Beginn des Moduls findet die Begrüßung und Vorstellung des Moduls statt. Die Schülerinnen und Schüler lernen den Mikrocontroller kennen und erhalten einen groben Überblick über den Modulablauf und -aufbau. Anschließend bilden die Schülerinnen und Schüler 2er-Teams, um sich gegenseitig zu unterstützen und gleichzeitig die Relevanz von Teamarbeit in der Informatik zu erleben.

## *Station 0*

Die Station 0 dient als Einstieg in das Modul und soll die benötigten Grundlagen für das Modul vermitteln. Die Schülerinnen und Schüler lernen den Arduino und seine Anschlüsse kennen und stecken auf einem Steckbrett ihre ersten Schaltungen, um eine LED zum Leuchten zu bringen. Im Anschluss lernen die Schülerinnen und Schüler wie sie mithilfe der grafischen Programmieroberfläche ArduBlock den Mikrocontroller steuern und die LED zum Blinken bringen können. So sammeln sie erste Erfahrungen in der grafischen Programmierung und erlernen wichtige Befehle (z. B. Variablen, Kontrollstrukturen), die sie später noch vertiefen werden.

## *Sicherung*

Um sicherzugehen, dass auch alle Schülerinnen und Schüler die Grundlagen verstanden haben, findet danach noch eine Sicherungsphase statt. Dort stellen einige Teams ihre bisherigen Ergebnisse vor und diskutieren diese.

## *Auswahl zwischen Station 1-4*

Im Anschluss an Station 0 und die dazugehörige Sicherungsphase dürfen die Schülerinnen und Schüler zwischen vier verschiedenen Stationen mit unterschiedlichen Schwierigkeitsgraden wählen. Zur Auswahl stehen hierbei folgende Stationen:

Station 1: Die Schülerinnen und Schüler sollen hier mithilfe eines Helligkeitssensor und eines Servo-Motors eine automatische Rollladensteuerung bauen, indem sie zunächst die Helligkeit messen und dann den Servo-Motor entsprechend anschalten, um einen Rollladen zu schließen.

Station 2: Bei dieser Station wird eine Einparkhilfe nachgebaut, wobei mithilfe eines Distanzmessers die Distanz zum Objekt (z. B. der Wand) gemessen werden soll und anschließend über den Piezo-Signalgeber Töne passend zur jeweiligen Entfernung ausgegeben werden sollen.

Station 3: In Station 3 bauen die Schülerinnen und Schüler eine Lichtschranke mithilfe einer IR-LED und IR-Photodiode und programmieren diese anschließend so, dass eine LED angeht, wenn die Lichtschranke unterbrochen wird.

Station 4: Hier geht es darum ein Farbthermometer zu bauen. Dafür wird zunächst die Temperatur mithilfe eines Temperatursensors gemessen und daraufhin die RGB-LED in einer zur Temperatur passenden Farbe eingeschaltet.

## *Präsentation*

Zum Abschluss präsentiert möglichst jedes Team seine Lösung und bewertet ihren eigenen Fortschritt. Dadurch bekommen die Schülerinnen und Schüler auch einen Einblick in die anderen Projekte und können ihr Ergebnis mit den anderen Schülerinnen und Schüler vergleichen.

## **Begründung des Methoden- und Medieneinsatzes**

Im InfoSphere - Schülerlabor Informatik ist eines der wichtigsten Prinzipien, dass die Schülerinnen und Schüler selbstständig arbeiten um informatische Probleme zu lösen. Dabei sollen sie in Teams möglichst ohne direkte Anleitung durch eine Lehrperson arbeiten und sich gegenseitig unterstützen. Um dies zu gewährleisten wurden für dieses Modul Arbeitsblätter entwickelt, die die Schülerinnen und Schüler durch die einzelnen Phasen führen. Dabei ist die Rolle der Betreuerinnen und Betreuer eine rein beratende. Um trotzdem den zeitlichen Rahmen und vor allem das

Erfolgserebnis der Teilnehmerinnen und Teilnehmer gewährleisten zu können, wird auf eine kurze Einstiegspräsentation der grafischen Programmieroberfläche ArduBlock und des Arduino-Mikrocontrollers zurückgegriffen.

### Didaktische Prinzipien nach Baumann und Hubwieser

- **Prinzip der Differenzierung:** Die Aufteilung der Arbeit während des Moduls in einzelne Stationen dient dazu grundlegendes Wissen mit Station 0 aufzubauen und mit den weiteren Stationen dieses erworbene Wissen zu vertiefen.
- **Prinzip der Lebensnähe und Aktualität:** Alle weiterführenden Stationen sind aktuelle Beispiele aus dem Alltag der Schülerinnen und Schüler. So modellieren sie beispielsweise in Station 2 eine Einparkhilfe, welche die meisten Kinder selber schon mal im Auto der Eltern erlebt haben.
- **Prinzip der Zielvorstellung:** Vielen Schülerinnen und Schülern fehlt das Verständnis, wieso sie manches überhaupt lernen müssen. Die alltagsnahen Beispiele verdeutlichen den Schülerinnen und Schüler, wozu sie dieses Modul und das Wissen, welches durch das Modul vermittelt wird, brauchen.
- **Prinzip des individuellen Lerntempos:** Sobald ein Team mit Station 0 oder einer weiteren Station fertig ist, hat es die Wahl sich direkt im Anschluss eine weitere Station zur Bearbeitung auszusuchen. Dies fördert zum einen die leistungsstarken und die schnellen Schülerinnen und Schüler und zum anderen auch die leistungsschwächeren, die auch wenn sie was länger brauchen, genügend Zeit haben, ihre Stationen zu beenden.

### VERLAUFSPLAN DES MODULS

Zeit	Phase	Inhalt	Medium	Sozialform
0:00-0:05	Begrüßung	<i>Begrüßung der Schülerinnen und Schüler im Schülerlabor</i>		
0:05-0:20	Einstieg	<i>Vorstellung des Moduls, des Mikrocontrollers und der Programmierumgebung ArduBlock (Präsentation/Video)</i>	Smartboard oder Beamer, PC	LV
0:20-1:30	Erarbeitung von Programmiergrundlagen und elektrotechnischen Grundlagen; Einführung in die Arbeit mit dem Arduino und dem Mikrocontroller	<i>Erarbeitung von Station 0 (eine LED zum Leuchten und zum Blinken bringen mit Hilfe des Arduinos und elektrotechnischer Bauteile)</i>	Arbeitsblätter zu Station 0, PC, InfoSphere-Kit	PA

1:30- 1:50	Frühstückspause			
1:50- 2:00	Sicherung	<i>Vorstellung der Ergebnisse der Partnerarbeit (oder GA) zwecks Bildung gemeinsamer Wissensbasis (ca. 2 oder 3 Teams)</i>	Smartboard, Tafel oder Beamer, Laptop, InfoSphere-Kit	SV
2:00- 3:30	Vertiefen des Grundlagenwissens und Erarbeitung mit neuen elektrotechnischen Bauteile (Pause nach Bedarf)	<i>Auswahl zwischen Station 1 bis 4</i>	Arbeitsblätter zu Station 1 bis 4, Laptop, InfoSphere-Kit	PA
3:30- 3:45	Präsentationen	<i>Präsentation von mindestens einem Team pro Station</i>	Smartboard, Tafel oder Beamer, Laptop, InfoSphere-Kit	SV
3:45- 3:55	Nachevaluation	<i>Nachevaluation</i>	Laptop	
3:55- 4:00	Verabschiedung	<i>Verabschiedung der Schülerinnen und Schüler aus dem InfoSphere</i>		



## QUELLENVERZEICHNIS

- <http://blog.ardublock.com/>
- <http://www.informatikstandards.de/index.htm?section=standards>

## ABBILDUNGSVERZEICHNIS

- Alle Grafiken entstammen der Software ArduBlock, welche unter der GNU General Public License veröffentlicht wurde.