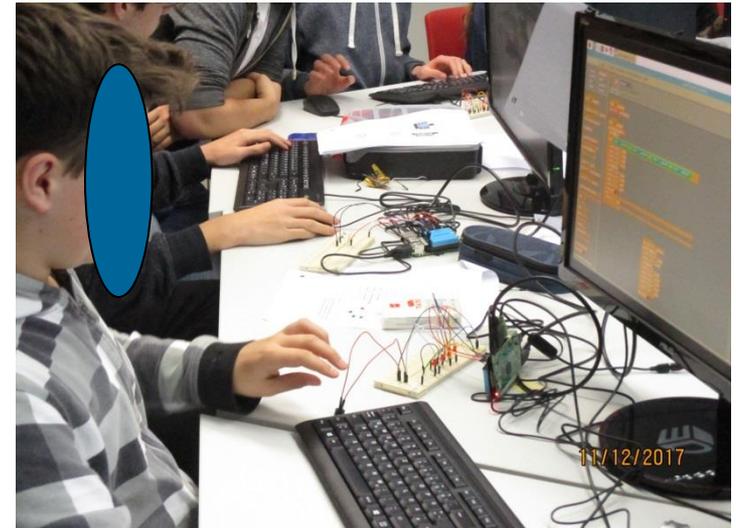


# Die MicroBerry-Lernumgebung: Ein handlungsorientiertes Konzept für den Informatikunterricht mit fächerübergreifenden MINT-Bezügen



## Gliederung

1. Rahmenbedingungen für die Konzeption / Erprobung
  - 1.1 Ziele und Angebote der MPDV-Junior-Akademie
2. Konzeption der Lernumgebung
  - 2.1 Entwicklung von Interessen
  - 2.2 Lernmotivation
  - 2.3 Schlussfolgerungen
3. Umsetzung
4. Fachübergreifende Bezüge
5. Ergebnisse der Evaluation

# 1. Rahmenbedingungen für die Konzeption / Erprobung

## Raspberry Pi – und Technische Informatik-Seminare an der PH für Studierende der Informatik (Modul 3)

- rekrutiert Studierende, die an der MPDV-Junior-Akademie unterrichten wollen
- schafft Grundlagen für z. B. wissenschaftl. Hausarbeiten

### MPDV-Junior-Akademie

- außerschulischer Lernort
- stellt Rahmenbedingungen und finanzielle Mittel zur Verfügung

### Verein Jugend@Technik e. V.

- organisiert die Seminare an der MPDV
- Bindeglied zwischen beteiligten Schulen, Hochschulen und MPDV

### Forschungsprojekte

- Entwicklung
- Begleitforschung
- Optimierung

## 1. Rahmenbedingungen für die Konzeption / Erprobung

# Die MPDV-Junior Akademie – ein außerschulischer Lernort für Mädchen und Jungen



Bildquelle: MPDV-Broschüre

## 1.1 Ziele und Angebote der MPDV-Junior-Akademie

Allgemeines Ziel ist es, durch spielerisches Lernen Freude und Spaß an MINT-Themen zu wecken. Im Speziellen soll Interesse für die Informationstechnik entwickelt werden.



Bildquelle: MPDV-Broschüre

Im Angebot sind drei verschiedene Seminare:

- Robotik (Kl. 5-7)
- Mikrocontroller (Kl. 8-10)
- Automation (Kl. 11-13)

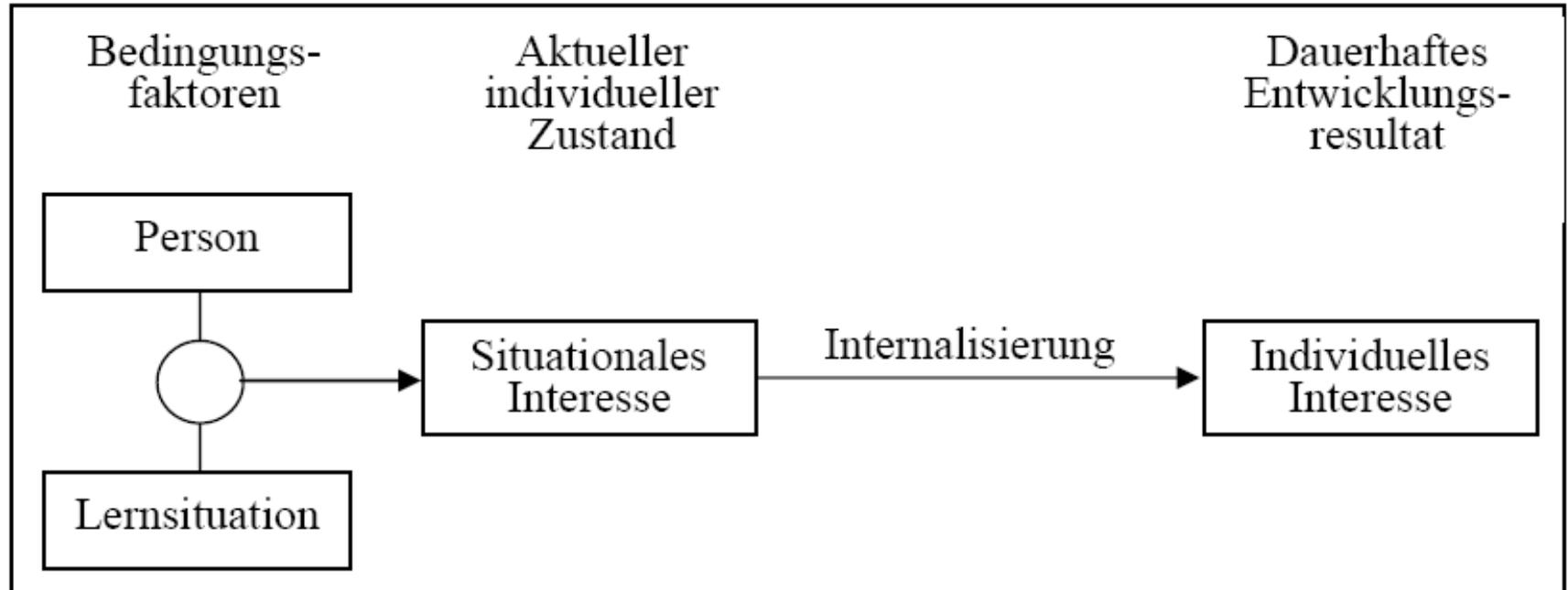
## 2. Konzeption der Lernumgebung

### **Ausgangsfrage / Zielperspektive:**

Wie sollte eine Lernumgebung gestaltet werden, um Schülerinnen und Schülern einen motivierenden und interessanten Einstieg in den informatischen Anfangsunterricht zu ermöglichen?



## 2.1 Entwicklung von Interessen



Rahmenmodell der Interessengenese (nach Krapp 1998, 191)

## 2.2 Lernmotivation

Zur Entwicklung individueller Interessen bedarf es mehrerer positiv besetzter Begegnungen mit dem Lerngegenstand (Krapp 2005).

### Wie erreicht man dies?

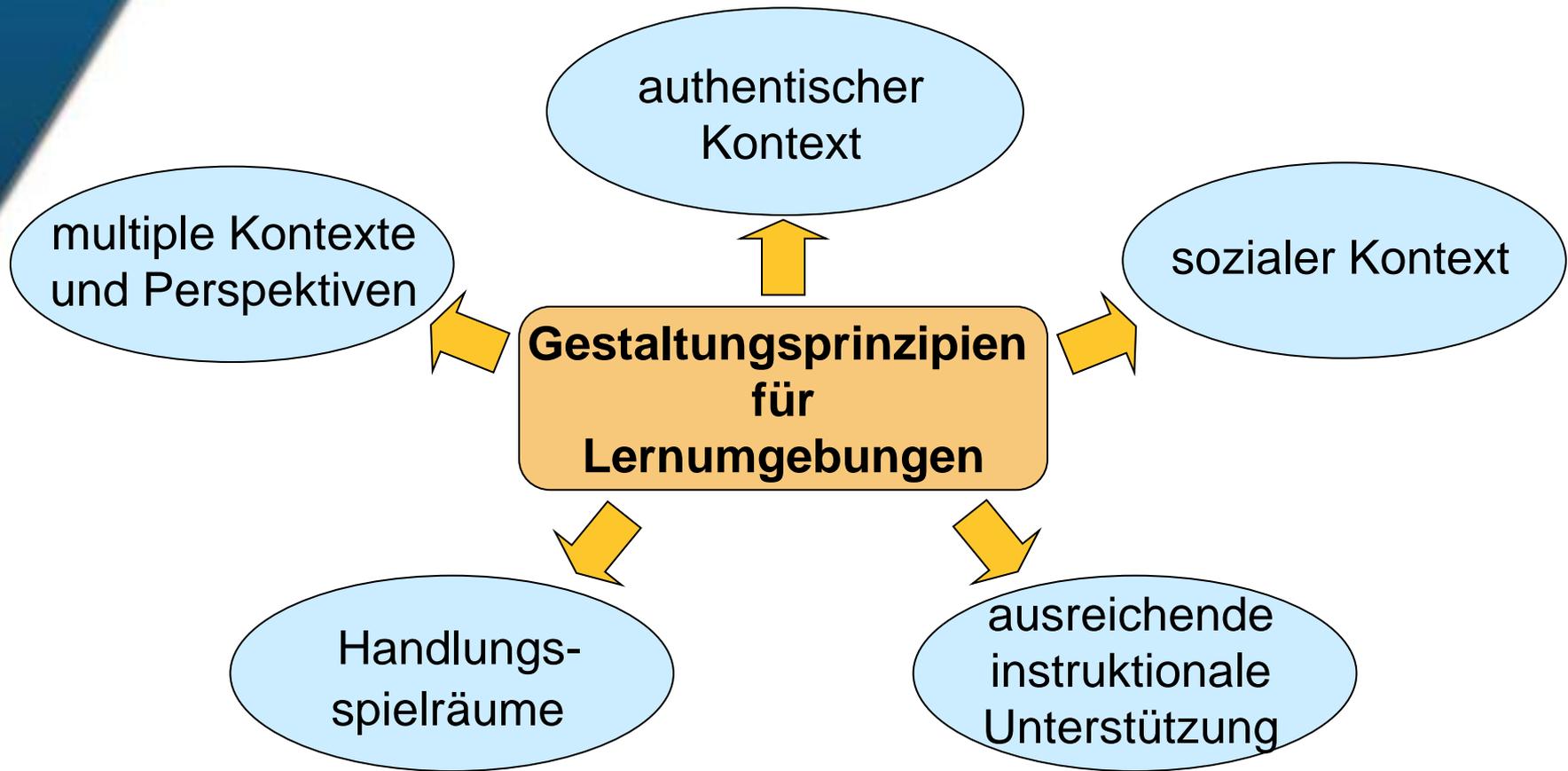
Lernumgebungen sollten hoch **motivierend** sein.

### Grundbedingungen für motiviertes Lernen:

- soziale Eingebundenheit
- selbstbestimmtes Arbeiten
- Kompetenzerleben (Deci/Ryan)

### Wie sollten Lernumgebungen gestaltet sein, um motiviertes Lernen zu ermöglichen?

## 2.2 Lernmotivation



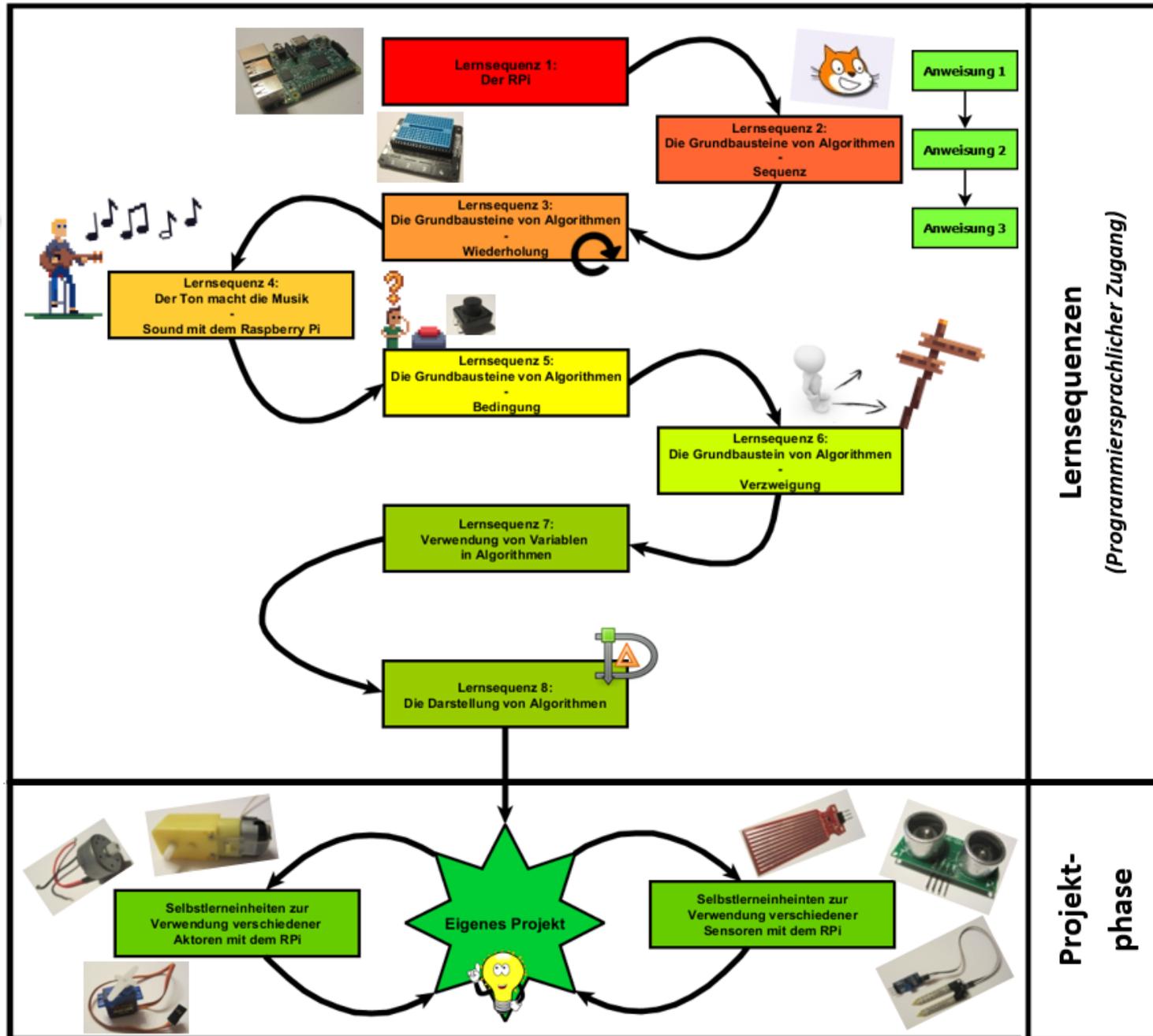
## 2.3 Schlussfolgerungen

- aktive Auseinandersetzung mit problem- und anwendungsorientierten Aufgaben
  - fächerübergreifend (Authentizität und multiple Kontexte)
  - handlungsorientiert
- ⇒ Steuerung von einfachen Sensoren und Aktoren durch selbst erstellte Programme fördert den handelnden Umgang mit dem Lerngegenstand.
- ⇒ Zusammenhänge zwischen Soft- und Hardware werden exemplarisch greif- und erfassbar.

### **Die MicroBerry-Lernumgebung:**

Ein handlungsorientiertes Konzept zur Behandlung von Algorithmen im Informatikunterricht mit Verknüpfungen zu technischen Elementen der Robotik.

## 3. Umsetzung



## 3. Umsetzung

- **Grundlagentext** zur möglichst selbständigen Erarbeitung der Grundstrukturen von Algorithmen

## Grundlagen

Jeder von uns hatte schon einmal Kontakt mit einem Programm, welches erst unter einer bestimmten Bedingung abläuft. Womöglich nur noch nicht bewusst!

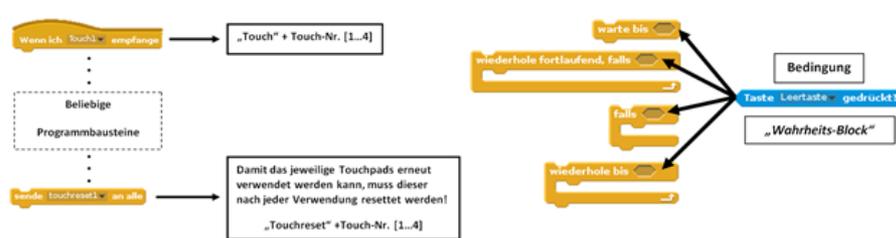
Zum Beispiel der Getränkeautomat, der das Getränk erst unter der Bedingung „Geld“ ausgibt. Oder der Fahrstuhl, der erst kommt, wenn er durch Tastendruck angefordert wird. Auch der Digitalwecker wirft einige von uns erst unter der Bedingung „Uhrzeit 6:30“ mit seinem nervigen Piepsen aus den Federn.

Unsere Programmierungen aus den vorherigen Lernsequenzen wurden ebenfalls alle erst unter einer bestimmten Bedingung ausgeführt! Erst durch Anklicken der einzelnen Skripte bzw. durch Anklicken der grünen Fahne wurden sie gestartet. Sprich, die Skripte starteten nur wenn die besondere Bedingung „Anklicken“ erfüllt war. (WENN Bedingung erfüllt DANN...)

Natürlich können in Scratch zahlreiche verschiedene Bedingungen gestellt werden! Beispielsweise das Starten eines Skripts unter der speziellen Bedingung „Leertaste gedrückt“. Es gibt auch die Möglichkeit unser Skript an einer bestimmten Stelle solange anzuhalten, bis eine Bedingung erfüllt ist!

In dieser Lernsequenz lernen wir, welche verschiedenen Bedingungen es gibt und wie diese in der eigenen Programmierung gezielt eingesetzt werden können.

Hier sind die neuen Programmierbausteine, die du für die Lernsequenz benötigst:



In Scratch gibt es verschiedene Bausteine, die erst unter einer bestimmten Bedingung ausgeführt werden.

**Kopf-Blöcke**

Wenn ich Touch... empfangen    Wenn Taste Leertaste... gedrückt

**Denke dran!**

Wenn angeklickt  
setze AddOn... auf ExplorerHAJ  
sende gpioserveron... an alle

## Aufgaben

1 Erstelle folgende Programme, welche unter einer bestimmten Bedingung ausgeführt werden:



- Wenn x-Taste gedrückt wird, dann leuchtet die LED an „Output 4“ fünfmal schnell auf!  
(Baue die zugehörige Schaltung auf!)
- Wenn die Pfeiltaste nach oben gedrückt wird, dann leuchtet die LED an „Output 2“ dreimal langsam auf!  
(Baue die zugehörige Schaltung auf!)
- Wenn die Zahlen 1,2,3 gleichzeitig gedrückt werden, dann werden LED (blau) und LED (rot) der Zusatzplatine angeschaltet!
- Wenn das Touchpad der Zusatzplatine gedrückt wird, dann werden LED (blau) und LED (rot) der Zusatzplatine angeschaltet!



```

Ein- und Ausschalten der Touchpad-LED's
sende ledhigh... an alle → „LED“ + [1..4] + [high | low | on | off]
    
```

**Abspeichern!**

Speichere die Dateien im Netzwerkordner ab.

B Das 3-Sterne Hotel „Himbeerkuchen“ im Dorf Mik-Rocontrolla hat sich einen nagelneuen Aufzug angeschafft! Leider fehlt die Programmierung. Zurzeit fährt er die 4 Etagen des Hotels leider nicht an. Da muss eine Lösung her! Glücklicherweise lässt sich der Aufzug mit Scratch programmieren.



Die folgenden Probleme gilt es zu lösen:

- Der Fahrstuhl muss auf Anforderung die entsprechende Etage anfahren.
- Aus Sicherheitsgründen benötigt der Fahrstuhl einen separaten Not-Aus! Wird dieser betätigt stellt der Fahrstuhl unverzüglich den Betrieb ein!
- Der Fahrstuhl benötigt eine Anzeige, damit die Gäste sehen, auf welchem Stockwerk er sich zurzeit befindet.



**Abspeichern!**

Speichere die Dateien im Netzwerkordner ab.

**Tipp**

Verwende LED's, um das aktuelle Stockwerk

## Grundlagen

Jeder von uns hatte schon einmal Kontakt mit einem Programm, welches erst unter bestimmten Bedingung abließ. Womöglich nur noch nicht bewusst!

Zum Beispiel der Getränkeautomat, der das Getränk erst unter der Bedingung „Geld“ ausgibt. Oder der Fahrstuhl, der erst kommt, wenn er durch Tastendruck angefordert wird. Auch der Digitalwecker wirft einige von uns erst unter der Bedingung „Uhrzeit 6:30“ mit seinem nervigen Piepsen aus den Federn.

Unsere Programmierungen aus den vorherigen Lernsequenzen wurden ebenfalls alle erst unter einer bestimmten Bedingung ausgeführt! Erst durch den Befehl „Run Modul (F5)“ wurde das Programm gestartet. Sprich, die Skripte starteten nur wenn die besondere Bedingung „Anklicken“ oder Tastendruck „F5“ erfüllt war. (WENN Bedingung erfüllt DANN...)

Natürlich können in Python zahlreiche verschiedene Bedingungen gestellt werden! Beispielsweise das Starten eines Skripts unter der speziellen Bedingung „Leertaste gedrückt“. Es gibt auch die Möglichkeit unser Skript an einer bestimmten Stelle solange anzuhalten, bis eine Bedingung erfüllt ist!

In dieser Lernsequenz lernen wir, welche verschiedenen Bedingungen es gibt und wie diese in der eigenen Programmierung gezielt eingesetzt werden können.

### Hier sind die neuen Programmierbausteine, die du für die Lernsequenz benötigst:

**while Bedingung True:** Solange eine bestimmte Bedingung falsch (False) oder wahr (True) ist, wird der Block ausgeführt.

**if Bedingung:** Wenn Bedingung erfüllt, dann wird der Block ausgeführt.

```
def Touch1 (channel, event):
    if event == 'press':
        explorerhat.light[0].on()
    if event == 'release':
        explorerhat.light[0].off()
```

Um die Touchpads am Explorer Hat ansteuern zu können, müssen wir diese in einer Funktion definieren. Auf diese greifen wir dann im unteren Block zu.

```
explorerhat.touch[0].pressed(Touch1)
explorerhat.touch[0].released(Touch1)
```

Die Ansteuerung des Touchpads kann mit diesem Block realisiert werden.

```
x = input("Taste") == 'beliebe Taste':
    |
```

Mit diesem Block können wir Tastaturabfragen realisieren.

## Aufgaben

**1** Erstelle folgende Programme, welche unter einer bestimmten Bedingung ausgeführt werden:

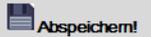


- Wenn x-Taste gedrückt wird, dann leuchtet die LED an „Output 4“ fünfmal schnell auf!  
(Baue die zugehörige Schaltung auf!)
- Wenn die Leertaste gedrückt wird, dann leuchtet die LED an „Output 2“ dreimal langsam auf!  
(Baue die zugehörige Schaltung auf!)
- Wenn eines der Touchpads der Zusatzplatine gedrückt wird, dann werden LED (blau) und LED (rot) der Zusatzplatine angeschaltet!
- Wenn eines der Touchpads der Zusatzplatine gedrückt wird, dann leuchtet die jeweilige LED des dazugehörigen Touchpads.



**Tipp**

Denke an den Doppelpunkt am Ende der Schleife und was dieser bewirkt!



**Abspeichern!**

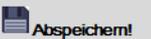
Speichere die Dateien im Netzwerkordner ab.

**B** Das 3-Sterne Hotel „Himbeerkekuchen“ im Dorf Mik-Rocontrolla hat  
**O** sich einen nagelneuen Aufzug angeschafft! Leider fehlt die  
**N** Programmierung.  
**U** Zurzeit fährt er die 4 Etagen des Hotels leider nicht an.  
**S** Da muss eine Lösung her! Glücklicherweise lässt sich der Aufzug mit Python programmieren.



Die folgenden Probleme gilt es zu lösen:

- Der Fahrstuhl muss auf Anforderung die entsprechende Etage anfahren.
- Aus Sicherheitsgründen benötigt der Fahrstuhl einen separaten Not-Aus! Wird dieser betätigt stellt der Fahrstuhl unverzüglich den Betrieb ein!
- Der Fahrstuhl benötigt eine Anzeige, damit die Gäste sehen, auf welchem Stockwerk er sich zurzeit befindet.



**Abspeichern!**

Speichere die Dateien im Netzwerkordner ab.



**Tipp**

Verwende LEDs, um das aktuelle Stockwerk

## Grundlagen

In dieser Sequenz beschäftigen wir uns näher mit der Erzeugung einzelner Töne und der Programmierung eigener Lieder!

Um verstehen zu können, wie wir mittels Lautsprecher einen Ton erzeugen, müssen wir vorher klären, was ein Ton überhaupt ist.



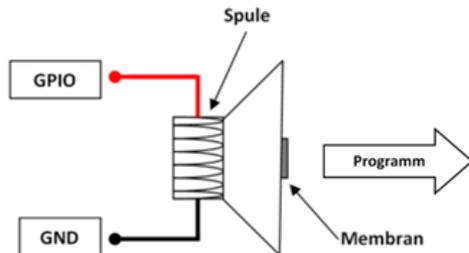
Am Beispiel der Gitarre:

Die Schwingung der Gitarrenseite wird über die Luft weitergegeben. Wenn die Schwingung das Ohr erreicht, wird diese vom Menschen als Ton wahrgenommen.

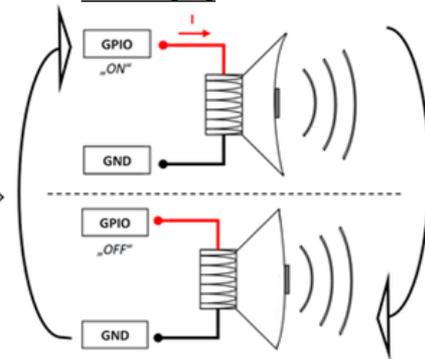
Die Erzeugung eines Tons mit einem Lautsprecher erfolgt ähnlich. Anstelle einer Saite bringen wir eine Membran zum Schwingen.

Dazu nutzen wir die Eigenschaft einer stromdurchflossenen Spule. Fließt Strom durch eine Spule wird ein Magnetfeld erzeugt. Die Membran wird durch das Magnetfeld hin zur Spule angezogen. Sobald kein Strom mehr durch die Spule fließt schwingt die Membran zurück.

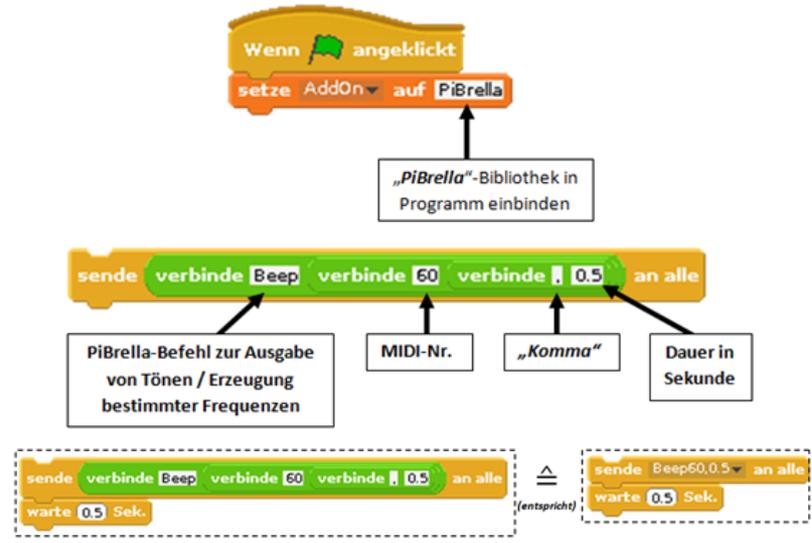
### Bestandteile und Beschriftung des Lautsprechers



### Funktionsweise des Lautsprechers zur Tonerzeugung



## Für die Lernsequenz benötigst du folgende Programmierbausteine:

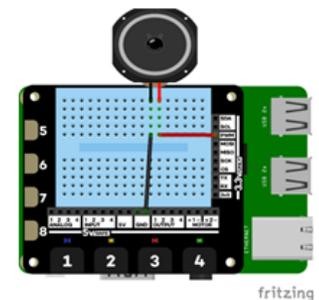
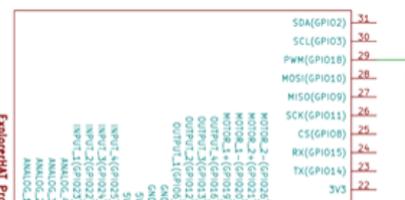


## Aufgaben

1 Erzeuge Töne mit einem Lautsprecher!



a. Baue die unten aufgeführte Schaltung auf und entwickle eine geeignete Programmierung um einen Ton zu erzeugen.



fritzing

### 3. Umsetzung

- **Grundlagentext** zur möglichst selbständigen Erarbeitung der Grundstrukturen von Algorithmen
- **Projektskript** zur Unterstützung der anschließenden Projektphase

## Roboterarm

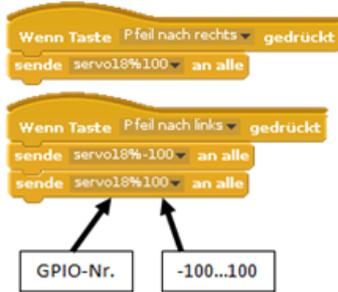
Roboter finden in der Industrie zunehmend an Bedeutung. Beispielsweise in der Fertigung von Produkten übernehmen sie Arbeitsschritte, welche ursprünglich von Menschen durchgeführt wurden. So können gefährliche Arbeitsschritte vom Roboter übernommen werden. Zudem sind Roboter nicht angewiesen auf Pausen und die Präzision bleibt auch ohne Pausen selbst nach Stunden unverändert.



In diesem Projekt soll ein solcher Roboter realisiert werden.

Baue und programmiere einen Roboterarm, welcher in der Lage ist folgendes zu leisten: automatisches Umsetzen eines Gegenstandes von Ort A zu Ort B.

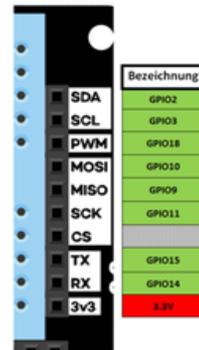
**Für diese Projektarbeit sind die folgenden Programmierbausteine wichtig:**



**Wichtige Hinweise für die Erstellung der Schaltung und Programmierung:**



Pinbelegung des Servomotors



GPIO-Nummerierung des ExplorerHAT's

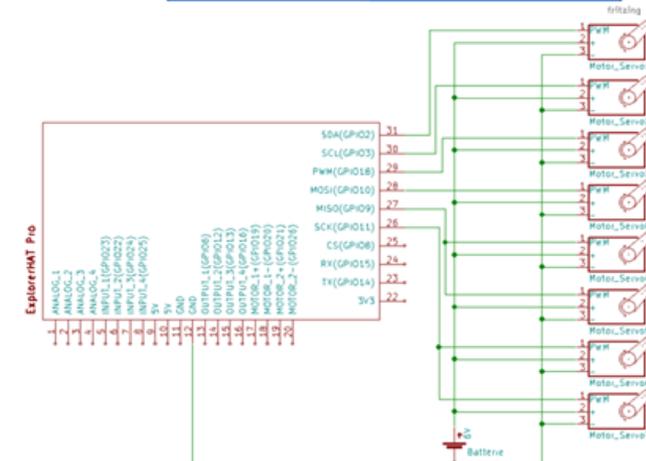
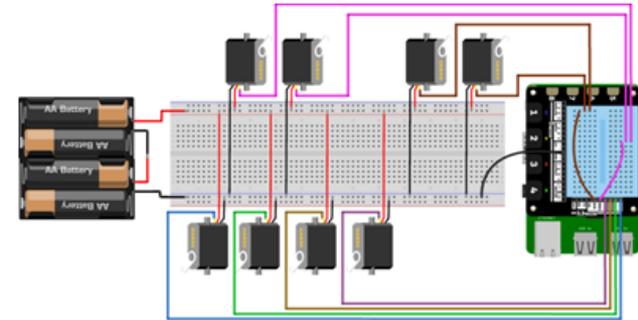
1 Baue die unten aufgeführte Schaltung auf.



### Wichtig!

Die Batteriehalterung wird als Letztes angesteckt!

Vor dem Anschließen der zusätzlichen Batterien müssen alle Anschlüsse genau überprüft werden, um einen Schaden der Geräte zu vermeiden.

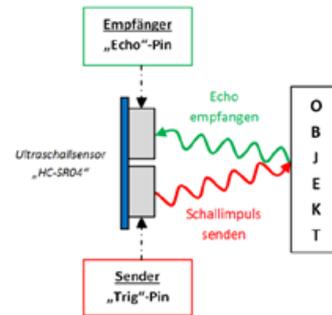


## Der Ultraschallsensor (digital)

Mithilfe des Ultraschallsensors können Entfernungen äußerst präzise ermittelt werden. Die Ermittlung einer bestimmten Entfernung erfolgt beim Ultraschallsensor wie folgt:

Zunächst wird ein Schallimpuls vom Ultraschallsensor ausgesendet. Der Schallimpuls ist ein Ton, welcher für den Menschen nicht zu hören ist. Der Schallimpuls wird ausgesendet, sobald am Trig-Pin (also dem Auslöser Pin) das entsprechende Signal ankommt. Die Ultraschallwellen breiten sich nun mit einer bestimmten Geschwindigkeit aus. Ultraschall breitet sich in der Luft und unter Normalbedingungen schätzungsweise 34,2m/s bzw. 1236 km/h aus. Also äußerst schnell.

Sobald die Ultraschallwellen auf ein Objekt auftreffen, werden diese Wellen reflektiert. Es entsteht ein sogenanntes "Echo". Der Empfänger des Ultraschallsensors kann dieses Echo über den "Echo"-Pin wiederum messen. Anhand der verstrichenen Zeit, welche die Ultraschallwellen vom Zeitpunkt ihres Aussendens bis zurück zum Empfänger benötigen, lässt sich die Entfernung genau berechnen. Diese Berechnung kann durch ein Programm automatisch erledigt werden.



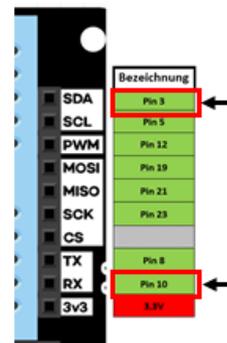
### Projektaufgabe:

Lerne den digitalen Ultraschallsensor näher kennen und setze ihn in einem eigenen Projekt sinnvoll ein.

Für diese Projektarbeit sind die folgenden Programmierbausteine wichtig:

setze Entfernung\_in\_cm auf Wert von Sensor ultra3

sende ultra 3,10 an alle

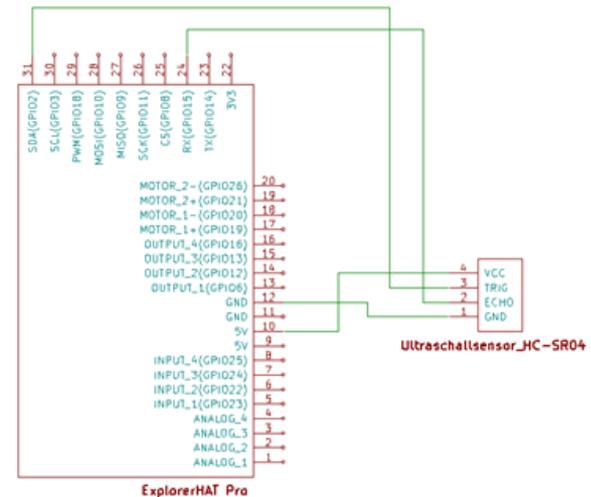
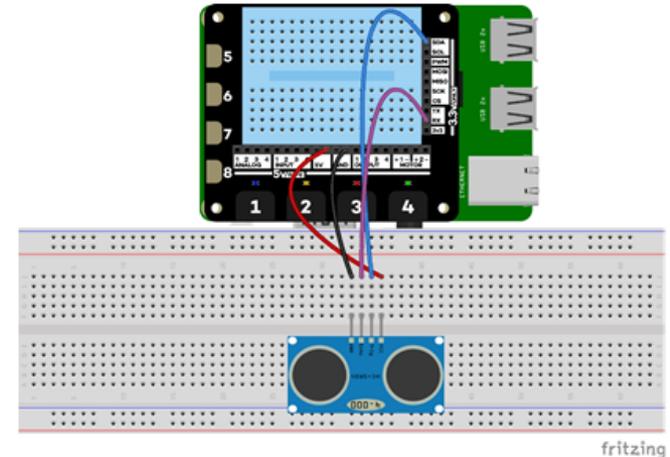


Befehl „ultra“:  
Hiermit werden in regelmäßigen Abständen Ultraschallwellen losgeschickt und empfangen

Pin-Nr., an welchem Trigger-Pin des US-Sensors angeschlossen ist

Pin-Nr., an welchem Echo-Pin des US-Sensors angeschlossen ist

- 1 Verbinde zunächst den digitalen Ultraschallsensor mit dem Raspberry Pi. Verwende hierzu folgende Schaltung:



### 3. Umsetzung

- **Grundlagentext** zur möglichst selbständigen Erarbeitung der Grundstrukturen von Algorithmen
- **Projektskript** zur Unterstützung der anschließenden Projektphase
- **Sortimentskasten**
- **Vernetzung** über VNC



## 4. Fachübergreifende Bezüge

### 4.1 Technik

- Steuern und Regeln einfacher Sensoren und Aktoren
- Aspekte der Elektrotechnik/Elektronik:  
Schaltkreise, Funktionsweise elektronischer Bauteile,  
Vor-, Pullup- Pulldown-Widerstände,  
Transistorschaltungen ...

## 4. Fachübergreifende Bezüge

### 4.2 Physik

- E-Lehre
- Akustik (Tonerzeugung, Ultraschallsensor)

## 4. Fachübergreifende Bezüge

### 4.3 Mathematik

- Binärcode

File menu: Datei Bearbeiten Veröffentlichen! Hilfe

Sprite1  
x: -95 y: -73 Richtung: 90

Skripte Kostüme Klänge

**Zähler**

AddOn ExplorerHAT

A3 0 A2 0 A1 1 A0 0

Neues Objekt: [Icons]

Sprite1

Bühne

The image shows a software interface for controlling a MicroBerry device. The left side features a script editor with a sprite named 'Sprite1' at coordinates (-95, -73) facing right. The script is organized into four columns, each starting with a 'Wenn angeklickt' (When clicked) event block. Each column contains a 'wiederhole fortlaufend' (Repeat loop) block with a 'falls' (If) condition and a 'sende' (Send) block. The conditions are A0=1, A1=1, A2=1, and A3=1 respectively. The 'sende' blocks are labeled 'output1high', 'output2high', 'output3high', and 'output4high'. Below the loops, each column has a 'setze' (Set) block followed by a 'warte' (Wait) block. The 'setze' blocks are for A3, A2, A1, and A0. The 'warte' blocks are all set to 1 second. The right side shows a stage with a digital display showing the number '8'. Above the display are four indicator lights labeled A3, A2, A1, and A0, with values 0, 0, 1, and 0. The A1 light is currently lit. The stage also shows a small sprite character and a 'Neues Objekt' (New Object) menu with three icons.

## 4. Fachübergreifende Bezüge

### 4.3 Mathematik

- Binärcode
- Boolesche Algebra
- Funktionale Zusammenhänge

The screenshot displays the MicroBerry learning environment, which is a Scratch-like interface for programming a microcontroller. It is divided into several sections:

- Left Panel:** Contains navigation buttons for 'erung', 'n', 'atoren', and 'blen'.
- Top Panel:** Shows the object name 'Objekt1' with a lock icon and coordinates 'x: 190 y: 27' and 'Richtung: 90'. Below this are tabs for 'Skripte', 'Kostüme', and 'Klänge'.
- Script Area:** Contains three event-driven scripts starting with 'Wenn Flagge angeklickt':
  - Script 1:** 'wiederhole fortlaufend' loop containing 'sende update an alle' and 'setze Ultraschallsensor auf Wert von Sensor ADC1'.
  - Script 2:** 'senke Stift ab', 'setze Stiftfarbe auf' (black), 'wiederhole fortlaufend' loop with 'ändere x um 1', 'setze y auf Wert von Sensor ADC1 \* 10', and 'warte 0.1 Sek.'.
  - Script 3:** 'wiederhole fortlaufend' loop with 'senke Stift ab', 'falls wird Rand berührt?' (if true, 'wische Malspuren weg', 'hebe Stift an', 'gehe zu x: -150 y: 0'), and another 'wiederhole fortlaufend' loop with 'wiederhole 250 mal', 'füge Ultraschallsensor zu Abstandswerte hinzu', 'warte 1 Sek.', and 'entferne alles aus Abstandswerte'.
- Right Panel (Stage):** Titled 'Bsp\_Analoge\_A...', it shows a graph of 'Abstandswerte' with a cat character. A table below the graph displays data:

Abstandswerte
233 2.971
234 2.775
237 2.762
238 2.745

A box with the number '4' is overlaid on the right side of the stage. Below the stage are icons for a pencil, a star, and a question mark, and a 'Bühne' (Stage) label.

## 4. Fachübergreifende Bezüge

### 4.3 Mathematik

- Binärcode
- Boolesche Algebra
- Funktionale Zusammenhänge
- Geometrie im Raum

- Variablen



## 5. Ergebnisse der Evaluation

Tab. 7.3 Ausprägung der Lernmotivation

	Interesse	Intrinsische Motivation	Identifizierte Motivation	Introjierte Motivation	Externale Motivation	Amotivation
Mittelwert	3,5645	4,2930	3,9328	1,9516	,4597	,6559
Median	3,6667	4,6667	4,0000	2,0000	,0000	,3333
N	62	62	62	62	62	62
Standardabweichung	,96871	,77099	,85542	1,61375	,67325	,84923

Tab. 7.4 Bedingungen für motiviertes Handeln

	Relevanz	Instruktionsqualität	Autonomie	Kompetenz	Überforderung	Soziale Eingebundenheit
Mittelwert	3,5188	4,0618	3,9147	3,9978	1,0188	4,2696
Median	3,6667	4,0000	4,0714	4,0833	,6667	4,3333
N	62	62	62	62	62	62
Standardabweichung	,93899	,66502	,71145	,66322	,90440	,61546

## 5. Ergebnisse der Evaluation

Tab. 7.6 gut gefallen hat mir:

Kategorie	Unterkategorie		Anzahl Nennungen
Kursleiter/Lehrkräfte	Hilfe gegeben	6	18
	Gute Erklärung	5	
	waren nett	5	
	Durchführung	1	
	Hohe Erwartungen	1	
Methode	Selbständiges Arbeiten	6	15
	Selbst erfinden/ experimentieren	2	
	Freiraum zum Üben und Testen und der Projektwahl	2	
	Freies Arbeiten	1	
	Aufgaben	1	
	Fragen waren möglich	1	
	Viele Möglichkeiten	1	
	Theorie	1	
Praktische Arbeit	Programmieren/Scratch	6	13
	Arbeiten mit dem RPi	4	
	Bauen mit elektronischen Bauelementen	3	

## 5. Ergebnisse der Evaluation

Projekt			10
Alles			9
Persönliches Empfinden	Hat Spaß gemacht	4	9
	Interesse geweckt	2	
	Neue Sachen kennengelernt	1	
	Zeit ging schnell vorbei	1	
	Erfolg durch Zielerreichung	1	
Soziale Aspekte	Entspannte Atmosphäre	1	2
	Stimmung war gut	1	

## 5. Ergebnisse der Evaluation

Tab. 7.7 nicht gut gefallen hat mir:

Kategorie	Unterkategorie		Anzahl Nennungen
Zeitaspekte	Wenig Zeit für Projekt	2	9
	Zu wenig Zeit	2	
	Zeit ging zu schnell um	2	
	Pausen	2	
	Nur 2 Tage Zeit	1	
Nichts / alles war gut			8
Funktionsweise	Das es auch mal nicht funktionierte	2	7
	Roboterarm nicht funktionierte	2	
	Projekt nicht funktionierte	1	
	Kabelsalat	1	
	7-Segment-Anzeige funktionierte nicht	1	
Inhaltliche Aspekte	Binärsystem	1	3
	Theorie	1	
	Scratch	1	
Soziale Aspekte	Teampartner	1	1
Sonstiges	Fragebogen am Ende	3	4
	Das man den Mikrocontroller am Ende nicht mit nach Hause bekam	1	

**Vielen Dank für Ihre Aufmerksamkeit!**