

## Handreichung PhC

### 1. Kurzvorstellung

Arduino-Spiel: Binärzahlen mit Hilfe einer Fernbedienung richtig erkennen

### 2. Einordnung in die Lehrpläne

Gym	Klassenstufe 10: <ul style="list-style-type: none"> <li>• Lernbereich 1: Algorithmen             <ul style="list-style-type: none"> <li>○ „Kennen des Prinzips der Modularisierung - Nutzen von Unterprogrammen und Bibliotheken“</li> </ul> </li> <li>• Lernbereich 3: Komplexaufgabe             <ul style="list-style-type: none"> <li>○ Leistungsschwache SuS vielleicht so beschäftigen</li> </ul> </li> </ul>
OS	Klassenstufe 9: <ul style="list-style-type: none"> <li>• Lernbereich 2: Daten darstellen: Informatikprojekte             <ul style="list-style-type: none"> <li>○ Durchführung eines Projektes</li> </ul> </li> </ul>

### 3. Lernziele

#### Kognitive Lernziele:

- Die Schülerinnen und Schüler üben den Umgang mit Zufallszahlen, if-Verzweigungen und Schleifen.
- Die Schülerinnen und Schüler erkennen, dass es keine wirklichen Zufallszahlen in Computern geben kann.
- Die Schülerinnen und Schüler verwenden den Modulo Operator zur Umrechnung Binärzahlen im Computer.
- Die Schülerinnen und Schüler lernen kennen, wie sie Bibliotheken in der Arduino IDE einbinden können.
- Die Schülerinnen und Schüler lernen den Umgang mit dem Serieller Monitor in der Arduino IDE kennen.
- Die Schülerinnen und Schüler untersuchen die Funktionsweise von Infrarotfernbedienungen.

#### Psychomotorische Lernziele:

- Die Schülerinnen und Schüler üben das Verbinden von Bauteilen mit Kabeln und das Stecken von Einzelteilen auf einem Steckbrett.
- Die Schülerinnen und Schüler üben den Umgang mit Maus und Tastatur, um Code zu schreiben.

#### Affektives Lernziel:

- Die Schülerinnen und Schüler erkennen den engen Zusammenhang von Informatik zum Alltag am Beispiel der Funktionsweise der Fernbedienung.



## 4. Voraussetzungen

### Fachliche Voraussetzungen:

- Kenntnis der Binärzahlen und deren Umrechnung
- Grundlagen der Programmierung von einfachen Algorithmen
  - Umgang mit if-Verzweigungen und Schleifen in der Arduino IDE
  - Umgang mit Bibliotheken in Arduino
  - Umgang mit dem seriellen Monitor in Arduino
  - Funktion des Modulo-Operators %

### Technische und materielle Voraussetzungen:

- Computer mit Internetzugang, USB-Anschluss und Arduino IDE
- Arduino Mega oder Mega 2560 (oder andere)
- Steckbrett, Kabel, Widerstände, LEDs, Taster
- Batterien und Batteriefach
- Verbindungskabel USB-A zu USB-B
- Infrarot Empfänger VS1838B und Infrarotfernbedienung

## 5. Kurzdarstellung

Geplant ist ein Spiel zu erstellen, in welchem der Computer eine zufällige Zahl erzeugt und über vier LEDs in Binär anzeigt. Die Schülerinnen und Schüler senden ihre Lösung mit einer Fernbedienung über einen Infrarotsensor zurück.

Es bietet sich an das Projekt in einzelne Abschnitte zu gliedern und einzelne Aufgaben schrittweise mit den Schülerinnen und Schülern zu bearbeiten.

### Aufgabe 1:

Erzeuge eine Zufallszahl und bringe die richtigen LEDs zum Leuchten.

- LEDs mit Vorwiderständen auf dem Steckbrett anbringen
- LEDs mit dem Arduino belegen und die entsprechenden Pins im Programm ansteuern
- Taster aufstecken und die entsprechenden Pins korrekt ansteuern
- Bei jedem Tastendruck eine Zufallszahl mit dem Befehl „random(1,10);“ erzeugen
- Zusätzlich die Notwendigkeit von dem Befehl „randomSeed(millis());“ erklären (damit wird der Seed, welchen das Programm als Bezugspunkt für die Zufallszahlgeneration verwendet, immer auf die Millisekunden seit Start des Programms gesetzt)
- Umrechnung der zufälligen Dezimalzahl in Binärzahl mit Modulo Operator und Ansteuerung der entsprechenden Pins/ LEDs

## Aufgabe 2:

Fernbedienung in Betrieb nehmen.

- Bibliothek einbinden:
  - Bibliothek „IRremote“ kann über den Bibliotheksverwalter oder über GitHub (<https://github.com/Arduino-IRremote/Arduino-IRremote>) installiert werden
  - ACHTUNG: nur Versionen 2.x.x. funktionieren, da die Befehle ab Version 3 verändert wurden
  - Befehl im Quelltext: `#include <IRremote.h>`
- Infrarotempfänger auf dem Steckbrett anbringen und richtig verbinden und folgenden Code mit der richtigen Pin-Nummer übernehmen:

```
int RECV_PIN = 13;
IRrecv irrecv(RECV_PIN);
decode_results results;
```

- Infrarotsignal mit dem Infrarotempfänger auslesen und auf dem seriellen Monitor mit folgenden Befehlen anzeigen lassen:

```
if(irrecv.decode(&results)){
    Serial.println(results.value, DEC);
    irrecv.resume();
}
```

- Infrarot-Codierung der Fernbedienung der Ziffern null bis neun abspeichern

## Aufgabe 3:

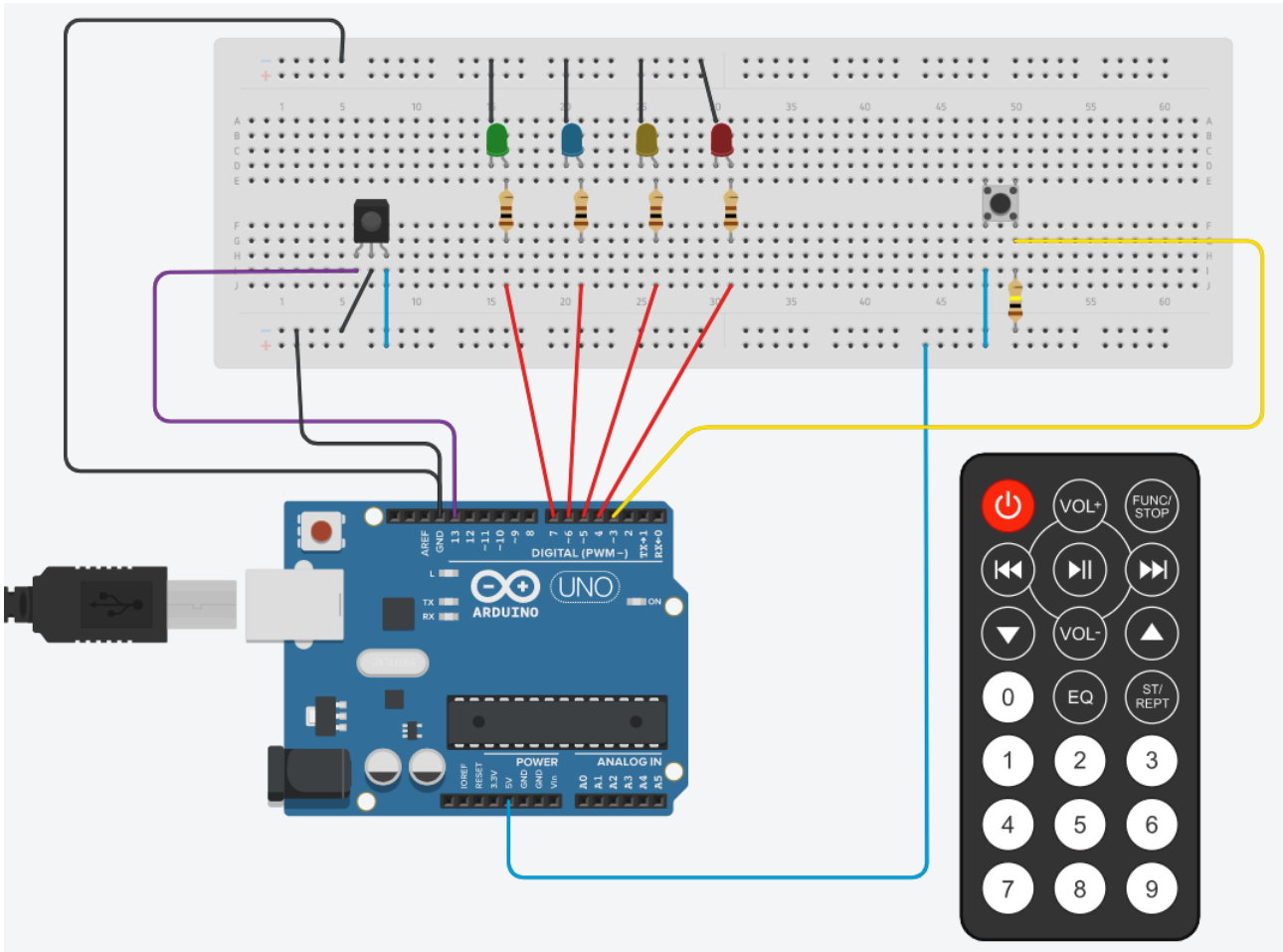
Die Logik des Spiels programmieren.

- Gibt es einen Punktestand?
- Wie erkenne ich die richtige Zahl?
  - Infrarot-Code && Zahl stimmen überein
- Was soll passieren, wenn die richtige Zahl erkannt wurde?
  - Im Beispiel: LEDs blinken und die Punktezahl wird um 1 erhöht
- Was soll passieren, wenn die Zahl nicht richtig erkannt wurde?
  - Im Beispiel: die rechte LED (rot) blinkt

Eine Simulation in Tinkercad findet sich hier:

[https://www.tinkercad.com/things/aCyoZQCHUXT?sharecode=Dz8xIM2RpKIQPOK\\_yEtn7PW428PoOjgQXVlfyze3Ku4](https://www.tinkercad.com/things/aCyoZQCHUXT?sharecode=Dz8xIM2RpKIQPOK_yEtn7PW428PoOjgQXVlfyze3Ku4)

Die Fernbedienung funktioniert nicht, da die Simulation eine aktuelle Version der IRremote-Bibliothek verwendet.



### Eine mögliche Lösung des Quellcodes:

```
#include<IRremote.h>

int RECV_PIN = 13;
IRrecv irrecv(RECV_PIN);
decode_results results;

// Achtung: Pinbelegung muss übernommen werden
int L3 = 7;
int L2 = 6;
int L1 = 5;
int L0 = 4;
int Taster = 3;

int zufallszahl;
int temp;
int temp1 = 0;
int score = 0;
bool test;

void setup() {
    Serial.begin(9600);
    irrecv.enableIRIn();

    pinMode(L0, OUTPUT);
    pinMode(L1, OUTPUT);
    pinMode(L2, OUTPUT);
    pinMode(L3, OUTPUT);
    pinMode(Taster, INPUT);
}

void loop() {

    if (digitalRead(Taster) == 1){
        test = true;
        Serial.println("Neue Zahl generiert.....");
```

```

randomSeed(millis());
zufallszahl = random(1,10);
Serial.println(zufallszahl);
delay(500);

// LED's belegen: -----
if(zufallszahl%2 == 1){
    digitalWrite(L0, HIGH);
    temp = zufallszahl/2;
}
else{
    digitalWrite(L0, LOW);
    temp = zufallszahl/2;
}
//-----
if(temp%2 == 1){
    digitalWrite(L1, HIGH);
    temp = temp/2;
}
else{
    digitalWrite(L1, LOW);
    temp = temp/2;
}
//-----
if(temp%2 == 1){
    digitalWrite(L2, HIGH);
    temp = temp/2;
}
else{
    digitalWrite(L2, LOW);
    temp = temp/2;
}
//-----
if(temp%2 == 1){
    digitalWrite(L3, HIGH);
    //temp = temp/2;
}

```

```

    }
    else{
        digitalWrite(L3, LOW);
        //temp = temp/2;
    }
}

// Fernbedienung auslesen:-----
if(irrecv.decode(&results)){
    Serial.println("result");
    Serial.println(results.value, DEC);

    if (results.value == 16769055 & zufallszahl == 0){
        // 0 nicht möglich, weil da sieht man die neue Zahl nicht
    }
    if (results.value == 16738455 & zufallszahl == 1){
        if(test){
            score = score + 1;
            test = false;
        }
        for(int i = 0; i<5; i++){
            digitalWrite(L0, HIGH);
            digitalWrite(L1, HIGH);
            digitalWrite(L2, HIGH);
            digitalWrite(L3, HIGH);
            delay(100);
            digitalWrite(L0, LOW);
            digitalWrite(L1, LOW);
            digitalWrite(L2, LOW);
            digitalWrite(L3, LOW);
            delay(100);
        }
    }
    else if (results.value == 16750695 & zufallszahl == 2){
        if(test){

```

```

        score = score + 1;
        test = false;
    }
    for(int i = 0; i<5; i++){
        digitalWrite(L0, HIGH);
        digitalWrite(L1, HIGH);
        digitalWrite(L2, HIGH);
        digitalWrite(L3, HIGH);
        delay(100);
        digitalWrite(L0, LOW);
        digitalWrite(L1, LOW);
        digitalWrite(L2, LOW);
        digitalWrite(L3, LOW);
        delay(100);
    }
}
else if (results.value == 16756815 & zufallszahl == 3){
    if(test){
        score = score + 1;
        test = false;
    }
    for(int i = 0; i<5; i++){
        digitalWrite(L0, HIGH);
        digitalWrite(L1, HIGH);
        digitalWrite(L2, HIGH);
        digitalWrite(L3, HIGH);
        delay(100);
        digitalWrite(L0, LOW);
        digitalWrite(L1, LOW);
        digitalWrite(L2, LOW);
        digitalWrite(L3, LOW);
        delay(100);
    }
}
else if (results.value == 16724175 & zufallszahl == 4){
    if(test){

```

```

        score = score + 1;
        test = false;
    }
    for(int i = 0; i<5; i++){
        digitalWrite(L0, HIGH);
        digitalWrite(L1, HIGH);
        digitalWrite(L2, HIGH);
        digitalWrite(L3, HIGH);
        delay(100);
        digitalWrite(L0, LOW);
        digitalWrite(L1, LOW);
        digitalWrite(L2, LOW);
        digitalWrite(L3, LOW);
        delay(100);
    }
}
else if (results.value == 16718055 & zufallszahl == 5){
    if(test){
        score = score + 1;
        test = false;
    }
    for(int i = 0; i<5; i++){
        digitalWrite(L0, HIGH);
        digitalWrite(L1, HIGH);
        digitalWrite(L2, HIGH);
        digitalWrite(L3, HIGH);
        delay(100);
        digitalWrite(L0, LOW);
        digitalWrite(L1, LOW);
        digitalWrite(L2, LOW);
        digitalWrite(L3, LOW);
        delay(100);
    }
}
else if (results.value == 16743045 & zufallszahl == 6){
    if(test){

```

```

        score = score + 1;
        test = false;
    }
    for(int i = 0; i<5; i++){
        digitalWrite(L0, HIGH);
        digitalWrite(L1, HIGH);
        digitalWrite(L2, HIGH);
        digitalWrite(L3, HIGH);
        delay(100);
        digitalWrite(L0, LOW);
        digitalWrite(L1, LOW);
        digitalWrite(L2, LOW);
        digitalWrite(L3, LOW);
        delay(100);
    }
}
else if (results.value == 16716015 & zufallszahl == 7){
    if(test){
        score = score + 1;
        test = false;
    }
    for(int i = 0; i<5; i++){
        digitalWrite(L0, HIGH);
        digitalWrite(L1, HIGH);
        digitalWrite(L2, HIGH);
        digitalWrite(L3, HIGH);
        delay(100);
        digitalWrite(L0, LOW);
        digitalWrite(L1, LOW);
        digitalWrite(L2, LOW);
        digitalWrite(L3, LOW);
        delay(100);
    }
}
else if (results.value == 16726215 & zufallszahl == 8){
    if(test){

```

```

        score = score + 1;
        test = false;
    }
    for(int i = 0; i<5; i++){
        digitalWrite(L0, HIGH);
        digitalWrite(L1, HIGH);
        digitalWrite(L2, HIGH);
        digitalWrite(L3, HIGH);
        delay(100);
        digitalWrite(L0, LOW);
        digitalWrite(L1, LOW);
        digitalWrite(L2, LOW);
        digitalWrite(L3, LOW);
        delay(100);
    }
}
else if (results.value == 16734885 & zufallszahl == 9){
    if(test){
        score = score + 1;
        test = false;
    }
    for(int i = 0; i<5; i++){
        digitalWrite(L0, HIGH);
        digitalWrite(L1, HIGH);
        digitalWrite(L2, HIGH);
        digitalWrite(L3, HIGH);
        delay(100);
        digitalWrite(L0, LOW);
        digitalWrite(L1, LOW);
        digitalWrite(L2, LOW);
        digitalWrite(L3, LOW);
        delay(100);
    }
}
else{
    if(digitalRead(L0)){

```

```
        temp1 = 1;
    }
    for(int i = 0; i<5; i++){
        digitalWrite(L0, HIGH);
        delay(100);
        digitalWrite(L0, LOW);
        delay(100);
    }
    if(temp1 == 1){
        digitalWrite(L0, HIGH);
        temp1 = 0;
    }
}
Serial.println("score: ");
Serial.println(score);
Serial.println("-----");

irrecv.resume();
}
}
```