

## AB5: Funkübertragung mit dem Calliope

Um Informationen von einem Calliope zu einem anderen zu übertragen, müssen beide Calliopes dieselbe Funkgruppe nutzen. Wähle dafür zusammen mit deinem Partner eine Zahl zwischen 0 und 255 und nutzt den rechts abgebildeten Block.



### Aufgabe 1: Sender - Empfänger

Programmiert eure Calliopes so, dass beim Drücken der Buttons A+B eine Textnachricht abgeschickt und auf dem anderen Calliope angezeigt wird.

- Erklärt, was es mit dem Block `receivedString` auf sich hat.
- Außer Texten können auch Zahlen oder „Wertepaare“ verschickt werden. Probiert es aus.

### Aufgabe 2: Senden, empfangen und antworten

Sendet eine Frage mit zwei Antwortmöglichkeiten (A/B bzw. Ja/Nein) an euren Partner. Dieser schickt euch seine Antwort mit den Tasten A oder B. Natürlich soll die Antwort dem Fragesteller auch angezeigt werden.

- Überlegt, wie ihr auf Fragen mit mehr als zwei Antwortalternativen antworten könntet.

### Aufgabe 3: Signalverstärker

#### a) In Zweiergruppen:

Testet, wie weit das Calliope-Funksignal reicht:

Schreibt dafür ein einfaches Programm, das ständig sendet und ein Häkchen anzeigt, wenn ein Funksignal empfangen wird.

Bewegt euch zum Testen im Raum und auf dem Flur.

#### b) In Vierergruppen:

Leitet gemeinsam ein Signal über eine möglichst weite Strecke weiter.

Sendet auch eine Antwort.

- Wieso könnte es sinnvoll sein, mehrere Funkgruppen zu verwenden?  
Überlegt euch einen geeigneten Ablauf und testet ihn.

## Projekt 1: Außenthermometer

### a) Einfaches Außenthermometer

Ein Calliope misst außen am Fenster die Temperatur.

Ein zweiter Calliope zeigt die Temperatur innen an.

### b) Höchsttemperaturanzeige

Beim Drücken auf Knopf A soll statt der aktuellen Temperatur die höchste bisher gemessene Temperatur angezeigt werden.

*Tipp:* Ihr braucht eine Variable `temp_max`, die ständig mit der aktuellen Temperatur abgeglichen und ggf. angepasst wird.

Beim Drücken auf Knopf B soll die Höchsttemperatur zurückgesetzt werden können.

## Projekt 2: Würfelspiel

Sieh dir zunächst noch einmal das Würfelspielprogramm an, das wir analysiert haben (Gruppenordner -> Programmanalyse). Nun soll ein Würfelspiel für zwei (oder mehr) Spieler entwickelt werden.

### a) Wer wirft die höhere Augenzahl?

Auf zwei Calliopes wird durch Schütteln eine zufällige Würfelzahl bestimmt, angezeigt und an den Partner gesendet. Erst wenn beide Partner gewürfelt haben, werden die Augenzahlen verglichen. Es wird ein lachender, trauriger oder neutraler Smiley ausgegeben, je nachdem, ob man die größere, die kleinere oder eine gleiche Augenzahl hatte.

➤ Hier brauchst du mehrere Variablen.

Wenn du einen Tipp brauchst, sieh auf der letzten Seite nach!

### b) Wer erreicht bei drei Würfeln die höhere Augensumme?

Die Augensumme wird erst nach dem dritten Schütteln an den Partner übermittelt (automatisch oder per Knopfdruck auf Button A).

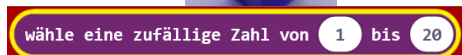
### c) Einmal würfeln mit einem 20-seitigen Würfel - Risikovariante

Statt einem normalen Würfel soll ein 20-seitiger Würfel simuliert werden.

Die Augenzahl wird angezeigt und der Spieler kann entscheiden, ob er noch einmal würfelt, um seine Augenzahl zu verbessern.

Dies darf er sooft wiederholen, wie er möchte. Würfelt er aber eine kleinere Zahl als zuvor, hat er automatisch verloren (Augenzahl = 0).

Per Knopfdruck wird die letzte Augenzahl übermittelt und wie in Aufgabenteil a) verglichen.



### Projekt 3: Quizbuzzer mit Funk

In einer der vergangenen Stunden konntet ihr einen Quizbuzzer für vier Kandidaten mit Hilfe der Pins programmieren.

Eine einfache Version findet ihr im Gruppenordner unter Vorlagen:

*mini-quizbuzzer\_vorlage.hex*

a) *Programmanalyse*

Seht euch das Programm zunächst an und erklärt die Verwendung der Variablen *beantwortet* und *erster*.

Was bewirkt das Drücken des Buttons A?

b) Entwickelt eine Funkversion des Quizbuzzers: Dabei soll ein Calliope als Quizmaster dienen und anzeigen, wer das Antwortrecht hat. Die übrigen Calliopes werden von den Kandidaten bedient und senden per Knopfdruck ein Signal, wenn der Kandidat antworten möchte.

c) Den Quizbuzzer könnt ihr auf verschiedene Arten erweitern:

- Den Kandidaten wird die Frage angezeigt.
- Sobald ein Kandidat als erster "gebuzzert" hat, wird dies auch auf den Calliopes der Kandidaten angezeigt.
- Es wird die Zeit bis zur Antwort gestoppt.
- Der Quizmaster kann einen Countdown auslösen, der dann auf den Calliopes der Kandidaten angezeigt wird.
- Der Calliope des Quizmaster und/oder der Kandidaten zählen die richtigen Antworten...
- ...

#### Weitere Ideen:

- **Diebstahlsicherung:** Ein Alarm wird ausgelöst, wenn das Funksignal zu schwach ist. Sucht in der Kategorie "Funk" nach einem passenden Block.
- **Morsen:** Überträgt einen Text per Hand in Morsezeichen, die ihr mit den Buttons A (kurz) und B (lang) sendet, und zeigt die Zeichen auf dem Partnercalliope an. Eine Tabelle über die Morsezeichen findet ihr hier: <https://de.wikipedia.org/wiki/Morsecode>
- Messt die **Schallgeschwindigkeit** mit Hilfe von zwei Calliopes, Funk und der Laufzeitmessung in Mikrosekunden. Außerdem ist es wichtig, den Abstand der beiden Calliopes möglichst genau zu kennen.
- **Schere – Stein – Papier**  
Der Ablauf ist ähnlich wie beim Würfelspiel.  
Es spielen zwei Personen gegeneinander. Mit Hilfe der Pins oder der Buttons A, B und A+B wählt jeder geheim eins der Symbole Schere, Stein oder Papier und sendet es an den Partner. Es gilt:  

Schere gewinnt gegen Papier. Papier gewinnt gegen Stein. Stein gewinnt gegen Schere.

Es können auch mehrere Runden gegeneinander gespielt werden und Punkte gezählt werden.

**Tipp zu 2a:**

Eine Möglichkeit ist die folgende: Nutze

- eine Variable für die eigene Augenzahl, (bekommt ihren Wert beim Schütteln)
- eine Variable für die Augenzahl deines Partners (bekommt ihren Wert beim Empfangen des Signals),
- eine Statusvariable, die anzeigt, ob bereits gewürfelt wurde, (ändert ihren Wert beim Schütteln)  
und
- eine Statusvariable, die anzeigt, ob bereits eine Zahl empfangen wurde (ändert ihren Wert beim Empfangen des Signals).

Eine Verzweigung im dauerhaft-Block kann abfragen, ob beide Statusvariablen gesetzt wurde, und dann die Augenzahlen vergleichen.