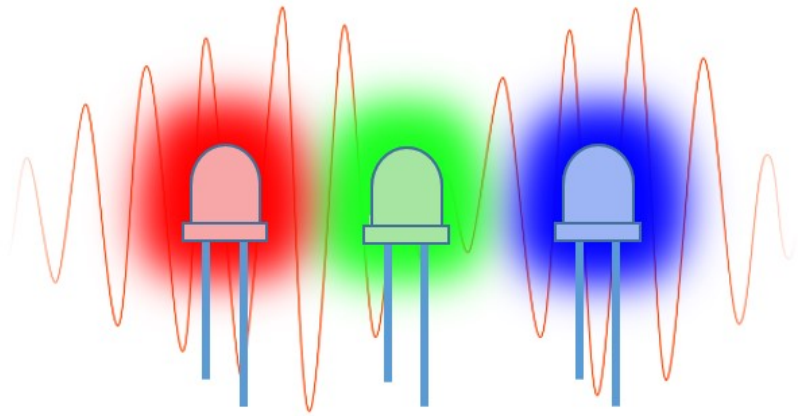


6



Aufgabe: Erzeuge RGB Farbmischung

Recherchiere im Web mit dem Keyword „online color picker“ und verstehe, wie mit Rot, Grün und Blau (RGB) Farbmischungen erzeugt werden.

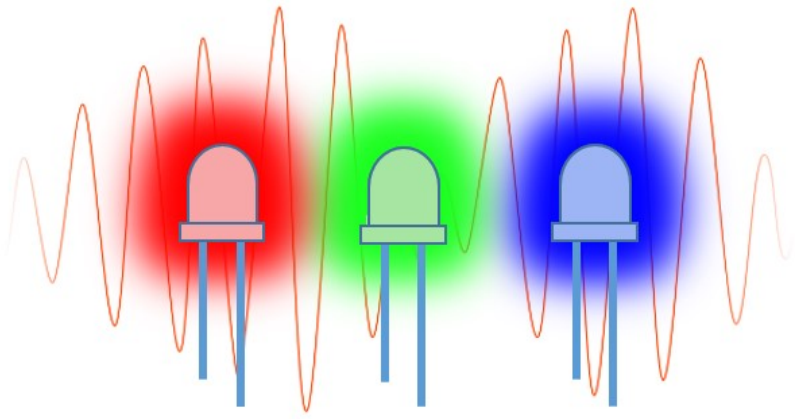
Beachte: Im Internet sind die Farben in 0 bis 255 Stufen einstellbar, während du auf deiner Platine mit 0% bis 100% arbeitest. Die Stufe 255 entspricht also 100% Helligkeit!

Stelle die Helligkeit deiner LEDs nun so ein, dass diese Farben entstehen:



A screenshot of the PGLU IDE interface. The interface is dark-themed and shows a code editor with a loop and three 'setze' blocks. The loop is labeled 'Hauptloop: 100'000 mal pro Sekunde' and has 'Blinkcode: kurz 1' and 'lang 1' settings. The three 'setze' blocks are for LED 1, LED 2, and LED 3, all set to 'auf 100 %'. The left sidebar shows various categories like 'Aktion', 'Sensor', 'Motor', 'LED', 'Pixel', 'Logik', 'Frage', 'Ablauf', 'Zahl', 'Variable', 'Mathe', 'Struktur', 'Funktion', 'Loop', and 'Spezial'. The top navigation bar has 'Grafik', 'Code', 'Simulator', and 'Programme' tabs.

7



Aufgabe: Blinke mit zufälligen Farben

Lasse deine 3 LEDs im Sekundentakt in einer zufälligen Farbe leuchten.

Nutze anstelle von festen Helligkeitswerten den Zufallsgenerator und stelle ihn so ein, dass er Werte zwischen 0 und 100 erzeugt!

Pausiere das Programm für 1000ms, wenn alle RGB-Werte definiert sind.

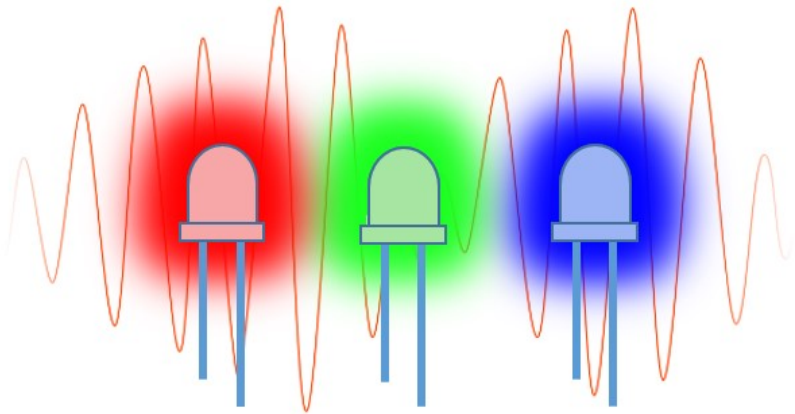
The screenshot shows the PGLU programming environment with the following code block:

```
Hauptloop: 100'000 mal pro Sekunde Blinkcode: kurz 1 lang 1
```

The code block contains three parallel actions:

- setze LED 2 auf % (connected to a random number generator block: ganzzahliger Zufallswert zwischen 0 bis 100)
- setze LED 1 auf % (connected to a random number generator block: ganzzahliger Zufallswert zwischen 0 bis 100)
- setze LED 3 auf % (connected to a random number generator block: ganzzahliger Zufallswert zwischen 0 bis 100)

Below these actions is a pause block: pausiere 1000 ms.



Aufgabe: Blende die leuchtende LED1 langsam aus

Definiere ab jetzt die Helligkeitswerte einer LED nicht mehr mit einer festen Zahl, sondern mit einer Variable.

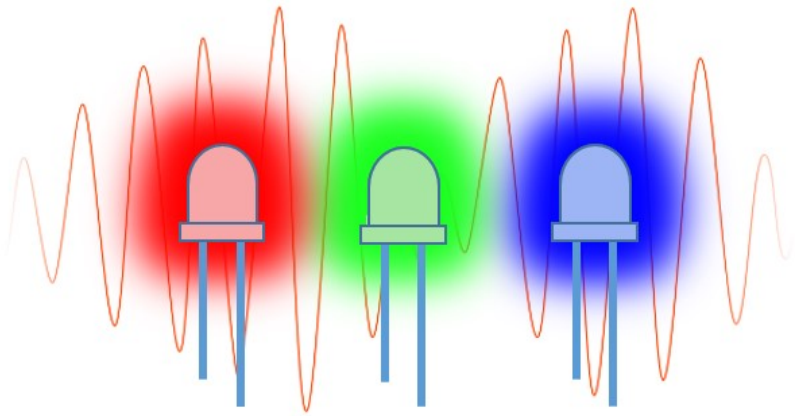
Eine Variable ist ein Wort, welches mit wechselnden Zahlen „gefüllt“ werden kann.

- Setze vor Beginn des Loops die Helligkeit von „Rot“ auf 100%
- Stelle diese Frage: Ist „Rot“ grösser als 0%? Dann vermindere „Rot“ um 1%!
- Setze nach dieser Frage LED1 auf den aktuellen Wert von „Rot“
- Pausiere das Programm für 10ms
- Beginne wieder von vorne (Das macht der Hauptloop von sich aus)

The screenshot shows the PGLU programming environment with the following components:

- Top Bar:** PGLU logo and tabs for Grafik, Code, Simulator, and Programme.
- Left Panel:** A vertical menu with categories: Aktion, Sensor, Motor, LED, Pixel, Logik, Frage, Ablauf, Zahl, Variable, Mathe, Struktur, Funktion, Loop, and Spezial.
- Main Workspace:**
 - Vor Hauptloop: 1x:** A red block containing a 'schreibe rot = 100' block.
 - Hauptloop:** A blue loop block labeled 'Hauptloop: 100'000 mal pro Sekunde' with 'Blinkcode: kurz 1 lang 1'.
 - Inside the loop:**
 - A 'schreibe rot = rot - 1' block (yellow).
 - A 'wenn rot > 0' block (blue) with a 'setze LED 1 auf rot %' block (orange) connected to its 'wenn' block.
 - A 'pausiere 10 ms' block (blue).
- Right Panel:** A vertical toolbar with a trash icon and zoom controls (+, -, and a target icon).

9



Aufgabe: Leuchte nach Musik und blende nach Peak aus

Dieses Programm baut auf Aufgabe 8 auf:

Setze jedoch die Variable „Rot“ immer wieder auf 100%, wenn der Mikrofonpegel grösser als 80% ist.

Dies geschieht mit dieser Frage: Ist der Wert von Sensor 4 grösser als 80%? Dann setze „Rot“ auf 100%!

Teste im Simulator und klicke auf S4 um den Musikpegel zu simulieren. Teste auf der Platine und justiere die Empfindlichkeit des Mikrofons richtig!

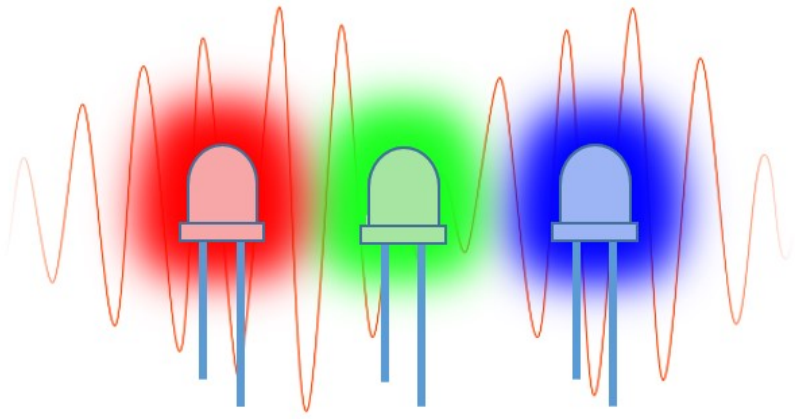
The screenshot shows the PGLU programming environment with the following code blocks:

- Hauptloop:** 100'000 mal pro Sekunde, Blinkcode: kurz 1, lang 1
- Wenn:** lese Wert von Sensor 4 in % > 80
- Wenn:** rot > 0
- Schreibe:** rot = 100
- Schreibe:** rot = rot - 1
- Setze:** LED 1 auf rot %
- Pausiere:** 10 ms

Yellow arrows point from the code blocks to the left-hand menu:

- Wenn (Sensor 4) to Sensor
- Wenn (rot) to Variable
- Schreibe (rot = 100) to Variable
- Schreibe (rot = rot - 1) to Variable
- Setze (LED 1) to LED
- Pausiere (10 ms) to Spezial

10



Aufgabe: Leuchte zufallsfarbig nach Musik und blende nach Peak aus

Dieses Programm ist eine Erweiterung der Aufgaben 8 und 9:

- Wenn der Musikpegel 80% überschreitet, berechne eine Zufallsfarbmischung
- Lasse die 3 LEDs nach dieser Zufallsfarbe leuchten
- Wenn die Helligkeit grösser als 0% ist, vermindere die Helligkeit jeder LED um 1% pro Durchlauf
- Pausiere das Programm für 10ms

The screenshot shows the PGLU programming environment with the following code blocks:

- Hauptloop:** 100'000 mal pro Sekunde, Blinkcode: kurz 1, lang 1
- LED:** setze LED 1 auf rot %, setze LED 2 auf grün %, setze LED 3 auf blau %
- Logik:** wenn lese Wert von Sensor 4 in % > 80, dann pausiere 10 ms
- Frage:** wenn rot > 0, dann schreibe rot = ganzzahliger Zufallswert zwischen 0 bis 100
- Frage:** wenn grün > 0, dann schreibe grün = ganzzahliger Zufallswert zwischen 0 bis 100
- Frage:** wenn blau > 0, dann schreibe blau = blau - 1
- Frage:** wenn blau > 10, dann schreibe rot = rot - 1
- Frage:** wenn blau > 10, dann schreibe grün = grün - 1