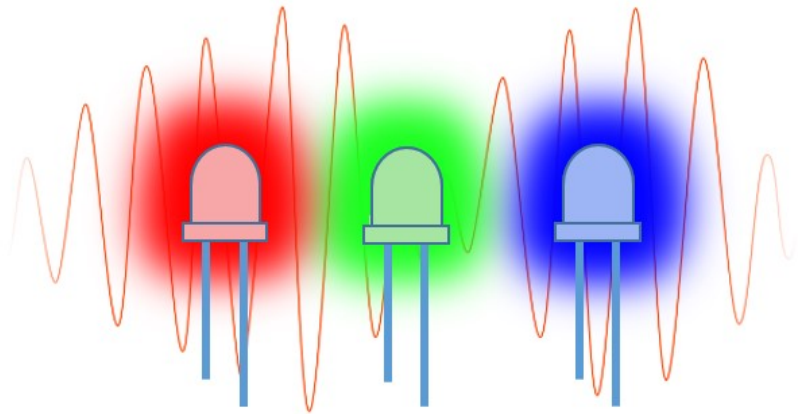


1



Variante: Leuchte nach Musik mit pegelabhängiger Pause

Dieses einfache Lichtorgelprogramm knüpft an Aufgabe 4 an.

Die Pegelwerte werden über die Variable „Lautstärke“ sofort an eine der 3 LEDs weitergegeben. Zwischen diesen Vorgängen pausiert das Programm und die Helligkeitswerte bleiben stehen.

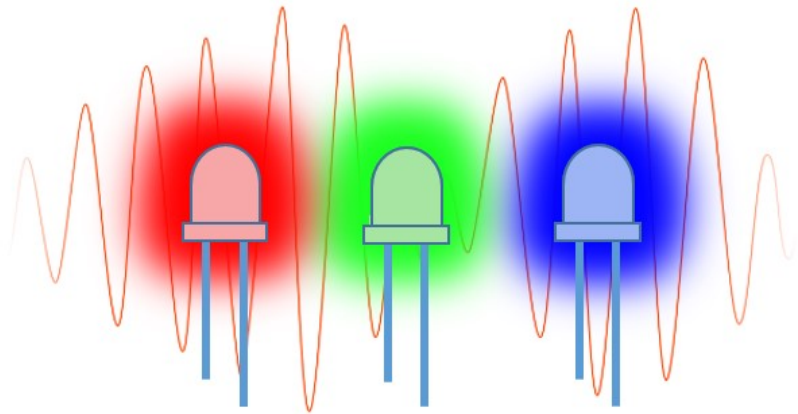
Je lauter die gemessene Musik war, desto länger dauert diese Pause.

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

- schreibe Lautstärke = lese Wert von Sensor 4 in %**
- setze LED 1 auf Lautstärke %**
- pausiere Lautstärke * 5 ms**
- schreibe Lautstärke = lese Wert von Sensor 4 in %**
- setze LED 2 auf Lautstärke %**
- pausiere Lautstärke * 5 ms**
- schreibe Lautstärke = lese Wert von Sensor 4 in %**
- setze LED 3 auf Lautstärke %**
- pausiere Lautstärke * 5 ms**

The code is set to a main loop of 100,000 iterations per second and a blink code of short 1 and long 1. The left sidebar shows the component palette with categories: Aktion, Sensor, Motor, LED, Pixel, Logik, Frage, Ablauf, Zahl, Variable, Mathe, Struktur, Funktion, Loop, and Spezial.

2



Leuchte 2 LEDs gegenfarbig

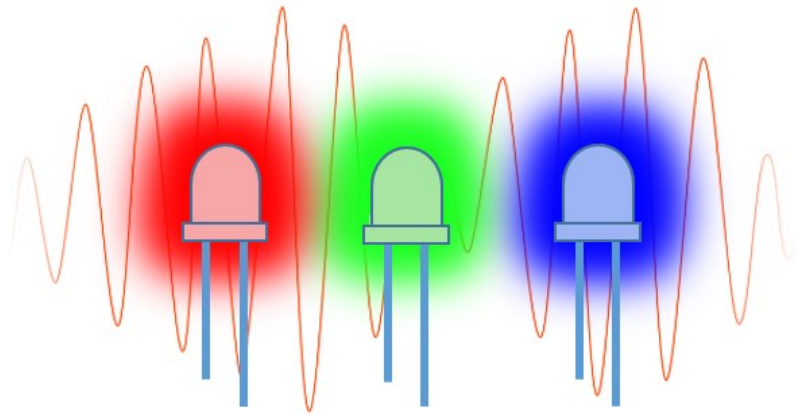
- LED1 leuchtet nach dem Pegel der Musik.
- LED3 macht genau das Gegenteil: sie leuchtet hell, wenn es still ist!

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

```
Hauptloop: 100'000 mal pro Sekunde | Blinkcode: kurz 1 | lang 1 |
schreibe Pegel = lese Wert von Sensor 4 in %
setze LED 1 auf Pegel %
setze LED 3 auf 100 - Pegel %
pausiere Pegel + 5 ms
```

The interface includes a sidebar with categories: Aktion, Sensor, Motor, LED, Pixel, Logik, Frage, Ablauf, Zahl, Variable, Mathe, Struktur, Funktion, Loop, and Spezial. The main workspace shows the code block with a 'Pegel' variable and a 'Sensor 4' input. The 'pausiere' block is set to 5 ms.

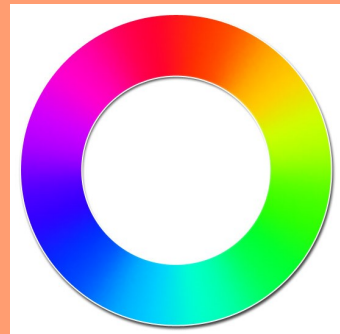
3



Leuchte mit allen 3 LEDs nach dem Farbkreis

Dieses Programm besteht aus Überblendungen zwischen jeweils zwei Grundfarben des RGB-Farbkreises.

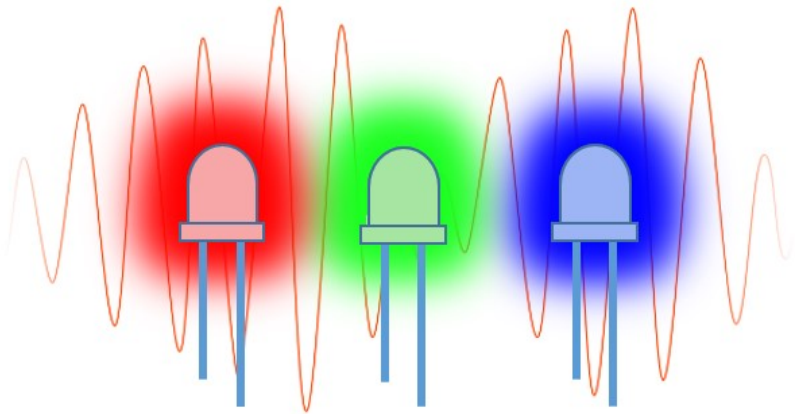
Damit können $100 \times 100 \times 100 = 1\text{Mio}$ Farbtönen dargestellt werden.



The screenshot shows a block-based programming environment with the following structure:

- Panel:** Grafik, Code, Simulator, Programme
- Left Sidebar:** Aktion, Sensor, Motor, LED, Pixel, Logik, Frage, Ablauf, Zahl, Variable, Mathe, Struktur, Funktion, Loop, Spezial
- Main Code Area:**
 - Vor Hauptloop: 1x**
 - setze LED 1 auf 100 %
 - Einblenden mit: aktuelle LED**
 - zähle Helligkeit von 0 bis 100 alle 0.1
 - setze aktuelle LED auf Helligkeit %
 - pausiere 1 ms
 - Ausblenden mit: aktuelle LED**
 - zähle Helligkeit von 100 bis 0 alle 0.1
 - setze aktuelle LED auf Helligkeit %
 - pausiere 1 ms
 - Hauptloop: 100'000 mal pro Sekunde**
 - Blinkcode: kurz 1 lang 1
 - Einblenden mit: aktuelle LED = LED 2
 - Ausblenden mit: aktuelle LED = LED 1
 - Einblenden mit: aktuelle LED = LED 3
 - Ausblenden mit: aktuelle LED = LED 2
 - Einblenden mit: aktuelle LED = LED 1
 - Ausblenden mit: aktuelle LED = LED 3

4



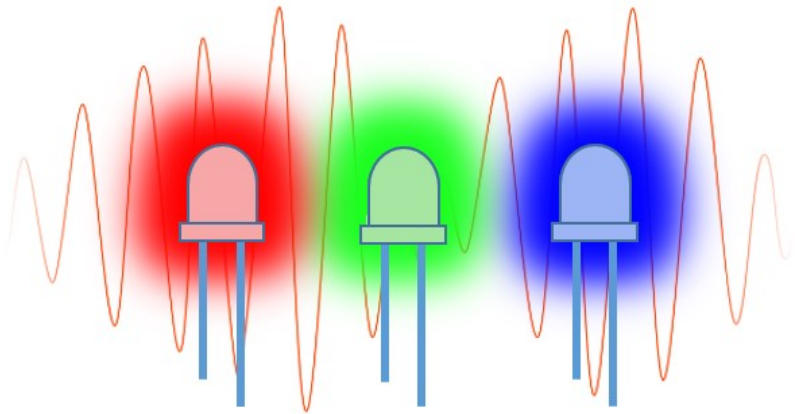
Farbkreis nach Musik

Die LEDs leuchten wie in Variante 3 nach dem RGB-Farbkreis. Ihre Helligkeit wird aber laufend dem Pegel eines Musiksignals aus Sensor 4 angepasst!

The screenshot shows the PGLU software interface with the 'Code' tab selected. The program is structured as follows:

- Vor Hauptloop: 1x**
 - setze LED 1 auf 100 %
- Hauptloop: 100'000 mal pro Sekunde** (Blinkcode: kurz 1, lang 1)
 - Einblenden mit: aktuelle LED = LED 2
 - Ausblenden mit: aktuelle LED = LED 1
 - Einblenden mit: aktuelle LED = LED 3
 - Ausblenden mit: aktuelle LED = LED 2
 - Einblenden mit: aktuelle LED = LED 1
 - Ausblenden mit: aktuelle LED = LED 3
- Name: Einblenden mit: aktuelle LED**
 - zähle Helligkeit von 0 bis 1 alle 0.001
 - setze aktuelle LED auf Helligkeit * lese Wert von Sensor 4 in %
 - pausiere 1 ms
- Name: Ausblenden mit: aktuelle LED**
 - zähle Helligkeit von 1 bis 0 alle 0.001
 - setze aktuelle LED auf Helligkeit * lese Wert von Sensor 4 in %
 - pausiere 1 ms

5



Klassische Pegelanzeige

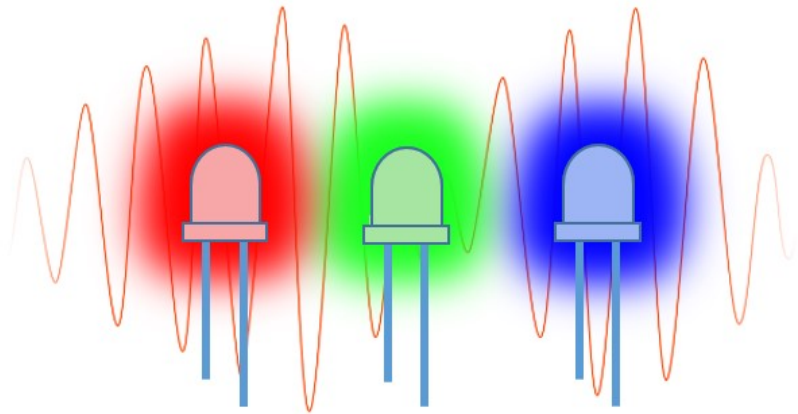
Die Musik bestimmt die Position der leuchtenden LED und nicht deren Helligkeit. Leise Musik=LED1, normale Musik=LED2, laute Musik=LED3

Erweitere die Pegelanzeige auf 4 LEDs!

The screenshot shows the PGLU software interface with a block-based code editor. The code is written in German and implements a 3-LED level indicator. The main loop is set to 100,000 iterations per second. The code reads a value from 'Sensor 4' and compares it to three ranges: 0-33 for LED 1, 34-66 for LED 2, and 67-100 for LED 3. Each range is handled by a 'wenn' (if) block that sets the corresponding LED to 'EIN' (ON) and a 'sonst' (else) block that sets it to 'AUS' (OFF). A 'pausiere' (wait) block is used to delay the update by a certain number of milliseconds.

```
Hauptloop: 100'000 mal pro Sekunde Blinkcode: kurz 1 lang 1
schreibe Pegel = lese Wert von Sensor 4 in %
wenn Pegel > 0 und Pegel < 33
  setze LED 1 auf EIN
sonst
  setze LED 1 auf AUS
wenn Pegel > 34 und Pegel < 66
  setze LED 2 auf EIN
sonst
  setze LED 2 auf AUS
wenn Pegel > 67 und Pegel < 100
  setze LED 3 auf EIN
sonst
  setze LED 3 auf AUS
pausiere Pegel ms
```

6



Einfaches Lauflicht

Ein klassisches Lauflicht über 4 LEDs

The screenshot shows the PGLU programming environment. The interface includes a top menu with 'Grafik', 'Code', 'Simulator', and 'Programme'. On the left, there is a sidebar with various categories: 'Aktion' (Sensor, Motor, LED, Pixel), 'Logik' (Frage, Ablauf), 'Zahl', 'Variable' (Mathe), 'Struktur', 'Funktion', 'Loop', and 'Spezial'. The main workspace displays a block-based program for a 4-LED running light. The program starts with a 'Vor Hauptloop: 1x' block containing a 'schreibe' block to set 'Leuchtdauer pro LED in ms' to 200. The main loop is labeled 'Hauptloop: 100'000 mal pro Sekunde' and 'Blinkcode: kurz 1 lang 1'. The code sequence is: 1. 'setze LED 1 auf EIN', 2. 'pauriere Leuchtdauer pro LED in ms ms', 3. 'setze LED 1 auf AUS', 4. 'setze LED 2 auf EIN', 5. 'pauriere Leuchtdauer pro LED in ms ms', 6. 'setze LED 2 auf AUS', 7. 'setze LED 3 auf EIN', 8. 'pauriere Leuchtdauer pro LED in ms ms', 9. 'setze LED 3 auf AUS', 10. 'setze LED 4 auf EIN', 11. 'pauriere Leuchtdauer pro LED in ms ms', 12. 'setze LED 4 auf AUS'. The interface also features a trash icon at the bottom right.