



In dieser Übung werden wir mit den BRIGHTNESS-Werten die COLOR LED steuern.

Lernziele:

- Den Lichtsensor ablesen und entweder die Helligkeit der COLOR LED oder den Lautsprecher steuern
- Mit Formeln die BRIGHTNESS-Werte in COLOR - Werte umwandeln

Wir werden ein Programm schreiben, das auf die Helligkeit im Raum reagiert. Je heller es im Raum ist, desto heller soll die rote LED leuchten. Der Trick besteht hier darin, die BRIGHTNESS- (Helligkeits-)Werte in entsprechende COLOR- (Farb-)Werte umzuwandeln.

BRIGHTNESS (Helligkeit) B reicht von 0 bis 100.

COLOR (Farbe) C (in allen drei Kanälen) variiert von 0 bis 255.

Wie können wir **B**-Werte in **C**-Werte konvertieren?

Hinweis: Antwort: $2.55 \cdot B \rightarrow C$ ist die Umrechnungsformel, aber allgemein ist diese Form der Umrechnung eine schöne Anwendung der Steigung einer Geraden zwischen (0,0) und (100,255).

$$(255 - 0) / (100 - 0) \rightarrow m: m \cdot B \rightarrow C.$$

Wenn wir mit dem Ton arbeiten, dann beschreiben die Punkte (0,100) und (55,880) einen vernünftigen Bereich von Audiofrequenzen. Wenn wir aber nur mit Musiknoten arbeiten wollen, dann repräsentiert (15,75) ungefähr die „mittleren“ 60 Klaviertasten. Dann müssen wir diesen konvertierten Wert in $2^{(C/12)}$ einsetzen, um die richtige Note zu erhalten. Da gibt's eine MENGE Mathematik und das ist das Schöne am Programmieren!!

Der Beginn des Programms:

1. Beginne ein neues Programm und nenne es *hell3*.
2. Füge ein **Disp**, und den Text „HELL zu FARBE“ zwischen Anführungszeichen an.
3. Deklariere **b** und **c** als **lokale** Variable wie gezeigt.
4. Füge ein **While...EndWhile**-Schleife zum Ablesen der Helligkeit an mit: **Send** „READ BRIGHTNESS“ und speichere die Variable mit **Get b**.

```

* hell3
Define hell3()=
Prgm
Local b
Disp "HELL zu FARBE"
b:=2
While b>1
Send "READ BRIGHTNESS"
Get b
Disp b
EndWhile

```

Die Umwandlung der Werte heben wir uns auf.

5. Verwende die Variable **c**, um den Farb-(COLOR-)Wert zu repräsentieren, den wir dann an alle drei Farbkanäle der COLOR LED senden. Der Umrechnungsfaktor ist 2,55, daher: **c:=2.55*b**.
6. Überprüfe diese Formel mit den beiden gegebenen Wertepaaren: für b=0 wird $c=2.55 \cdot 0=0$; und für b=100 ergibt sich $c=2.55 \cdot 100=255$.
7. Füge eine **Send** „SET COLOR“-Anweisung vor dem **End** der Schleife ein. Diese Anweisung wird die Helligkeit der COLOR LED regeln.

```

* hell3
Prgm
Local b,c
Disp "HELL zu FARBE"
b:=2
While b>1
Send "READ BRIGHTNESS"
Get b
Disp b
c:=2.55*b
Send "SET COLOR "
EndWhile

```



10 Minuten Coding

TI-NSPIRE™ CX WITH THE TI-INNOVATOR™ HUB

8. Zum Abschluss vervollständige die SET COLOR-Anweisung mit dreimaligem **eval(c)** (je einmal für die drei Farbkanäle).
 - Wenn drei Farbkanäle den gleichen Wert haben, dann leuchtet die LED in Weiß und die Helligkeit der LED hängt von diesem Wert ab.
9. Schließe den TI-Innovator™ Hub an und führe das Programm aus.
10. Ändere Umgebungshelligkeit indem du den Sensor an verschiedene Gegenstände führst. Beobachte dabei die Intensität der LED am Hub.

Du möchtest vielleicht weitere **Disp**-Anweisungen ins Programm einbauen, um die Werte für **b** und **c** sichtbar zu machen.

Hinweis: Wenn sich die COLOR LED ausschaltet, erscheint sie als grünes Licht und zeigt damit den „Stand By“-Zustand.

Aber halt! Da stimmt etwas nicht! Es wäre doch sinnvoller, dass bei schwächer werdender Raumbelichtung die LED stärker aufleuchtet! Wie kannst Du die Wirkungsweise umkehren?

Eine weitere Herausforderung: Wie kann man das Programm so abändern, dass unterschiedliche Helligkeitswerte der Umgebung unterschiedliche Farben an der LED verursachen?

Hinweis: Nimm $c:=2.55*(100-b)$ um die Wirkungsweise umzukehren.

LEKTION 3: ÜBUNG 3

LEHRERINFORMATION

```
* hell3 5/10
Prgm
Local b,c
Disp "HELL zu FARBE"
b:=2
While b>1
Send "READ BRIGHTNESS"
Get b
Disp b
c:=2.55*b
Send "SET COLOR eval(c) eval(c) eval(c)"
```