



## Overzicht:

In deze STEM-les past je jouw kennis van analoge uitvoer toe om de helderheid van de rode, groene en blauwe kanalen van een RGB (rood-groen-blauw) LED gemonteerd in een diffuus omhulsel te regelen. Er wordt een programma geschreven om een willekeurige reeks rode, groene en blauwe kleuren weer te geven. Je zult ervaren hoe uitgezonden licht zich vermengt om secundaire en tertiaire kleuren te maken. Als optie kan de behuizing van de sfeerlamp worden ontworpen in CAD-software en worden afgedrukt op een 3D-printer.

## Doelen:

1. Pas de principes van de kleurmenging van het uitgezonden licht toe.
2. Ontwerp en bouw een lamphuis voor de RGB (Rood-Groen-Blauw) LED.
3. Schrijf een TI-Python programma dat een willekeurige rode, groene en blauwe analoge uitvoerwaarde genereert en het mengsel gedurende één seconde op het sfeerlampje weergeeft.
4. Optioneel: Maak een 3D behuizing voor een sfeerlamp.

## Achtergrond:

Licht is ons allemaal erg bekend, maar wat is het? Licht is een golf die beweegt door een stationair kenmerk van de ruimte dat het elektromagnetische veld wordt genoemd. Licht heeft veel interessante functies die onze ogen kunnen waarnemen. Een functie is kleur, terwijl een andere helderheid is.

De kleur van het licht is te wijten aan de frequentie van de golf, dat wil zeggen hoe vaak hij op en neer trilt. De frequentie van het licht wordt gemeten met de eenheden van Hertz (Hz). De helderheid van het licht is te wijten aan hoeveel energie er in de golf zit en kan worden gemeten in Lux (Lx). Interessant is dat ons brein verschillende kleuren licht waarneemt wanneer drie primaire geëmitteerde kleuren worden gemengd met verschillende helderheden voordat ze onze ogen binnendringen.

De hersenen zullen bijvoorbeeld geel waarnemen als een lichtgolf met exact de juiste enkele frequentie in het oog komt of als de twee kleuren rood en groen licht gelijkmatig worden gemengd voordat ze in het oog komen. Op dezelfde manier zal het oog paars waarnemen wanneer een lichtgolf met exact de juiste enkele frequentie in het oog komt of wanneer felrood met dimblauw wordt gemengd. De kleuren rood, groen en blauw worden primaire kleuren genoemd wanneer het uitgestraalde licht wordt gemengd voordat het in het oog komt.

In dit project wordt een speciale lichtgevende diode gebruikt die eigenlijk drie LED's bevat. De RGB (rood-groen-blauw) LED heeft een rode, groene en blauwe diode in één plastic behuizing. Aangezien er in één geval drie LED's zijn en de drie een gemeenschappelijke kathode (-) leiden, heeft het pakketje vier pootjes. De ene is de rode anode (+), de andere is de groene anode (+), de andere is de blauwe anode (+) en de vierde is de gemeenschappelijke kathode (-) voor de drie afzonderlijke anodes.

Wanneer de RGB LED-anodes zijn verbonden met drie afzonderlijke ANALOG.OUT-pinnen op de TI-Innovator Hub en de gemeenschappelijke kathode is verbonden met een aardingspin, kan de helderheid van elke kleur onafhankelijk worden geregeld met behulp van de vaardigheid van analoge uitvoer (Vaardigheden Trainer 3). Om het licht van de drie afzonderlijke interne LED's te mengen, is een witgekleurde kap nuttig. Een kap kan worden gemaakt van wit papier of worden gemaakt met een 3D-printer.

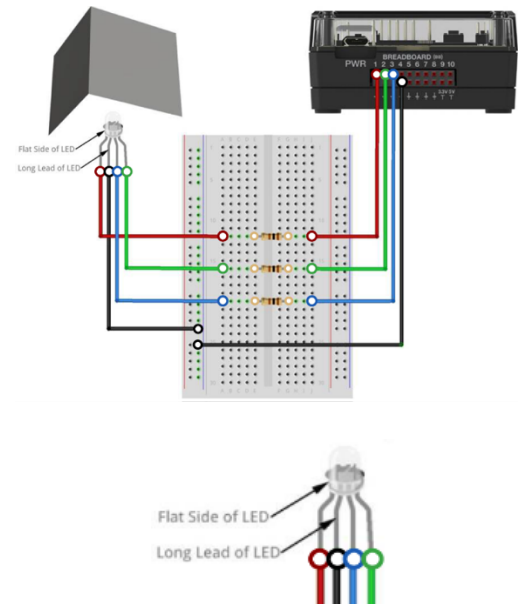


## Materialen en gereedschappen:

- TI-Nspire CX II-T Technologie
- TI-Innovator™ Hub met mini-USB-kabel
- Wit papier
- Witte 3D printer vulling (optioneel)
- TI-Innovator Breadboard Pack:
  - Breadboard
  - Man-man verbindingskabels
  - Man-vrouw verbindingskabels
  - RGB (Rood-Groen-Blauw) LED
  - 100 Ohm (bruin-zwart-bruin) weerstanden

## Bouw de hardware:

1. Plaats elke poot van de RGB-LED in een aparte man-vrouw verbindingskabel.
2. Plaats drie weerstanden van 100 Ohm (bruin-zwart-bruin) in afzonderlijke kolommen van het breadboard. Zorg ervoor dat elke weerstand de opening in het bord overbrugt, zodat de twee uiteinden elektrisch gescheiden zijn.
3. Verbind elk van de RGB-anodes met een van de 100 Ohm weerstanden met behulp een de man-vrouw verbindingskabel.
4. Gebruik een man-man verbindingskabel om het andere uiteinde van elke weerstand van 100 Ohm aan te sluiten op BB1, BB2 en BB3.
5. Verbind de gemeenschappelijke kathodepoot van de RGB-LED met de blauwe aardebus op het breadboard met behulp van de man-vrouw verbindingskabel.
6. Gebruik een man-man verbindingskabel om de blauwe aardebus van het breadboard te verbinden met een willekeurige aardepin op de TI-Innovator Hub.



## Schrijf de Software voor de TI-Nspire CX II-T:

Voorbeeldprogramma voor de TI-Nspire CX II-T:

```

from ti_hub import *
from random import *
Red=analog_out("BB 1")
Green=analog_out("BB 2")
Blue=analog_out("BB 3")
t=0
s=int(input("Hoeveel seconden? "))

```

## Programmabeschrijving

1. Verbind drie ANALOG.OUT-objecten 1 tot en met 3 met respectievelijk de breadboard-pinnen BB1, BB2 en BB3.
2. Definieert een variabele voor de duur van het programma.
3. Vraag de gebruiker om de duur en sla deze op in een variabele.



```

while t<s:
  ♦♦r=randint(0,255)
  ♦♦g=randint(0,255)
  ♦♦b=randint(0,255)
  ♦♦Red.set(r)
  ♦♦Green.set(g)
  ♦♦Blue.set(b)
  ♦♦sleep(2)
  ♦♦t+=1
Red.set(0)
Green.set(0)
Blue.set(0)

```

In Python `t+=1` betekent `t=t+1`.

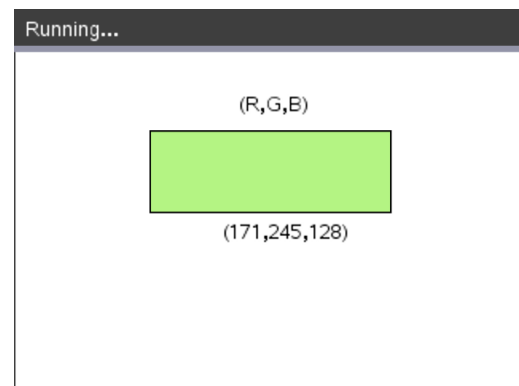
```

from ti_hub import *
from random import *
from ti_draw import *
:
use_buffer()
while t<s:
  ♦♦clear()
  ♦♦r=randint(0,255)
  ♦♦g=randint(0,255)
  ♦♦b=randint(0,255)
  ♦♦Red.set(r)
  ♦♦Green.set(g)
  ♦♦Blue.set(b)
  ♦♦set_color(0,0,0)
  ♦♦draw_text(140,40,"(R,G,B)")
  ♦♦draw_text(130,120,"({},{,})".format(r,g,b,))
  ♦♦draw_rect(84,49,151,51)
  ♦♦set_color(r,g,b)
  ♦♦fill_rect(85,50,150,50)
  ♦♦paint_buffer()
  ♦♦sleep(2)
  ♦♦t+=1
Red.set(0)
Green.set(0)
Blue.set(0)

```

4. Gebruik een while-instructie om een lus uit te voeren voor de duur van het programma.
5. Creëer voor elke lus een nieuwe willekeurige rode, groene en blauwe helderheidswaarde in het bereik van 0 tot 255 en sla die waarden op in de variabelen r, g en b.
6. Stel de analoge uitgang voor elk kleurkanaal in op de juiste willekeurige helderheidswaarde.
7. Geef de willekeurige helderheidswaarde voor de gebruiker weer.
8. Wacht 1 seconde of langer.
9. Verhoog de variabele van de while-lus.
10. Verlaat de while-lus wanneer de timerindex groter is dan de duur.
11. Doe de RGB-LEDs uit.

Naast de RGB-LED aansturen, kan je gebruikmakend van TI Draw modules ook de RGB-kleur grafisch weergeven d.m.v. de code hiernaast toegevoegd.



Zorg ervoor dat als je een programma gebruikmakend van de module TI Draw runt in TI-Nspire CX software, het document in de Handheld Preview mode staat.

Run het programma ook eens zonder de commando's `use_buffer()` en `paint_buffer()`; bv Comment beide lijnen (CTRL/Cmd + T).



## Extra voor Experts:

Schrijf een programma dat alle kleuren van de regenboog van rood naar violet weergeeft.

```
from ti_hub import *
from ti_draw import *
Red=analog_out("BB 1")
Green=analog_out("BB 2")
Blue=analog_out("BB 3")
red=[255,255,255,0,0,75,143]
green=[0,127,255,255,0,0,0]
blue=[0,0,0,0,255,130,255]
set_color(0,0,0)
draw_text(117,40,"Rainbow Colors")
for i in range(7):
    ♦♦Red.set(red[i])
    ♦♦Green.set(green[i])
    ♦♦Blue.set(blue[i])
    ♦♦set_color(red[i],green[i],blue[i])
    ♦♦fill_rect(85,50+10*i,150,10)
    ♦♦sleep(2)
Red.set(0)
Green.set(0)
Blue.set(0)
```

RGB (255, 0, 0)
RGB (255, 127, 0)
RGB (255, 255, 0)
RGB (0, 255, 0)
RGB (0, 0, 255)
RGB (75, 0, 130)
RGB (143, 0, 255)





## Ingebouwde RGB-LED TI-Innovator Hub:

Met de code `color.rgb(.....,.....)` kan je de ingebouwde RGB-LED aansturen als volgt:

### Code voor de TI-Nspire CX II-T:

```
from ti_hub import *
from random import *
t=0
s=int(input("Hoeveel seconden? "))
While t<s:
  ♦♦ r=randint(0,255)
  ♦♦ g=randint(0,255)
  ♦♦ b=randint(0,255)
  ♦♦ color.rgb(r,g,b)
  ♦♦ sleep(2)
  ♦♦ t+=1
color.rgb(0,0,0)
```