

In de tweede les van Unit 2 leer je de intensiteit van de drie kleuren van de KLEURENLED op de TI-Innovator™ Hub te variëren

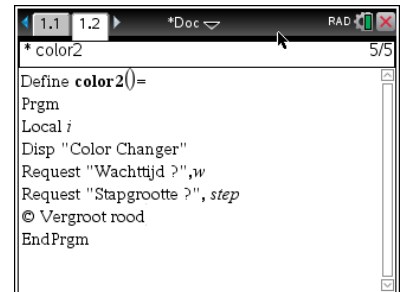
**Doel:**

- Gebruik van de **For** loops om de drie kleurkanalen van de KLEURENLED te beïnvloeden.

De rood, groen en blauw waarde (van 0 tot 255) die naar HUB gezonden worden bepalen de helderheid van ieder kanaal. Dit programma demonstreert een variatie aan kleuren uit de 16 miljoen (256<sup>3</sup>) mogelijke kleuren. Hierbij ga je gebruik maken van een aantal **For** loops.

**Creatie van een "Color Changer" programma**

1. Start een nieuw programma en noem het COLOR2.
2. Voeg het commando **Disp** toe met daarin de titel van het programma
3. Voeg **Request** toe met tussen de aanhalingstekens "Wachttijd ?"
4. Voeg dan een komma toe en de variabele **w**.
  - Deze variabele zal gebruikt worden in het commando **Wait**. Een lager getal zal een kortere wachttijd hebben en het programma zal sneller het volgende commando uitvoeren.
5. Vraag ook om de stapgrootte, zoals in het voorbeeld hiernaast. Deze variabele zal gebruikt worden in de **For** loop.



```

Define color2()=
Prgm
Local i
Disp "Color Changer"
Request "Wachttijd ?",w
Request "Stapgrootte ?", step
© Vergroot rood
EndPrgm
    
```

*Notitie: Local i voorkomt dat de variabele ook buiten het programma zichtbaar is.*

*Het commentaarsymbool © is beschikbaar via het menu **menu > Acties > opmerking toevoegen**.*

Het programma zal geleidelijk (afhankelijk van de wachttijd en de stapgrootte) de intensiteit van ROOD vergroten, daarna van GROEN, vervolgens de intensiteit van ROOD verminderen en BLAUW weer laten toenemen, dan GROEN laten afnemen en ROOD weer laten toenemen en dan BLAUW laten afnemen en als laatste weer het ROOD laten afnemen. Dit is een vrij lang programma, en je kunt het laten lopen nadat elke **For** loop compleet is om te testen.

Het is prettig dat de editor automatisch voor elk **For** commando meteen een **EndFor** commando toevoegt. Hierdoor kun je het niet vergeten.

6. Voeg een **For...EndFor** loop toe vanuit het Control menu na het commentaar



```

Define color2()=
Prgm
Local i
Disp "Color Changer"
Request "Wachttijd ?",w
Request "Stapgrootte ?", step
© Vergroot rood
For i,0,255
EndFor
    
```

**De eerste loop**

7. Voeg de details van het **For** commando in zodat de "loop"-variabele **i** loopt van 0 tot 255 met als stapgrootte **s**
8. Voeg **Send "SET COLOR** toe via het **HUB** menu.
9. Gebruik de **eval( )** functie uit het **HUB** menu om de variabele **i** te gebruiken voor de waarde van het rode kanaal. Zet het groene en blauwe kanaal op - 0.
10. Denk om de aanhalingstekens en de haakjes.
11. Voeg na het **Send** commando een **Wait** commando in met de variabele **w** dat je hebt gebruikt in het eerdere **Request** commando.



```

Define color2()=
Prgm
Local i
Disp "Color Changer"
Request "Wachttijd ?",w
Request "Stapgrootte ?", step
© Vergroot rood
For i,0,255,step
Send "SET COLOR eval(i) 0 0"
Wait w
EndFor
    
```

**Tip:** Na de voltooiing van de eerste loop kunnen de leerlingen het programma uitvoeren zodat ze de KLEURENLED steeds helderder rood kunnen zien worden. Gebruik een korte wachttijd (b.v. 0.1 s) en een grote stapgrootte (10).

12. Na de **EndFor** van deze eerste For loop kun je een **Tekst** commando gebruiken om het rode licht te bekijken.

```

1.1 1.2 *Doc RAD 0/15
* color2
Local i
Disp "Color Changer"
Request "Wachttijd ?",w
Request "Stapgrootte ?",step
@ \Vergroot rood
For i,0,255,step
  Send "SET COLOR eval(i) 0 0"
  Wait w
EndFor
Text "Druk op enter"
  
```

**Het toevoegen van de groene kleur**

We gaan nu een nieuwe loop maken om het groene licht aan de LED toe te voegen. We willen alleen het groene kanaal beïnvloeden en niet het rode kanaal. Dit kunnen we op twee manieren oplossen:

**Send "SET COLOR 255 eval(i) 0"**

(we weten ROOD is altijd aan en BLAUW is altijd uit)

of

**Send "SET COLOR.GREEN eval(i)"**

Dit laatste commando beïnvloedt alleen het GROENE kanaal en heeft geen invloed op het RODE en BLAUWE kanaal. In beide gevallen kunnen we de variabele *i* opnieuw gebruiken.

In het voorbeeld hiernaast gebruiken we de eerste methode.

13. Voeg het **Wait** commando toe binnen de loop body en gebruik de variabele *w*.
14. Voeg weer het **Text** commando toe na **EndFor** om de nieuwe kleur te bewonderen. Welke kleur is dat?

```

1.1 1.2 *Doc RAD 13/15
* color2
Send "SET COLOR eval(i) 0 0"
Wait w
EndFor
Text "Druk op enter"
[]
For i,0,255,step
  Send "SET COLOR 255 eval(i) 0"
  Wait w
EndFor
EndPrgm
  
```

Nu willen we de bijdrage van ROOD verminderen zodat we alleen groen overhouden.

Om een aflopende **For** loop te krijgen, start je met het hoogste nummer, dan het laagste nummer en gebruik je een negatieve stapgrootte:

**For i, 255, 0, -Step**

Deze loop start met 255 en trekt daar bij elke stap van de loop **step** vanaf totdat de variabele *i* kleiner is dan 0 wanneer de loop stopt. Gebruik de knop en niet het minteken voor aftrekken. Dat veroorzaakt een error.

# 10 Minutes of Code

## TI-NSPIRE CX MET DE TI-INNOVATOR™ HUB

We willen alleen het RODE kanaal beïnvloeden daarom gebruiken we COLOR.RED in het **Send** commando. De rest van deze loop is hetzelfde als de eerste twee loops die we hebben gemaakt. Het voorbeeld toont alleen de *keywords*.

Voltooi zelf de rest van deze commando's. Lukt dat niet, kijk dan bij de volgende stap.

Hier is het complete stuk code dat de bijdrage van ROOD geleidelijk laat afnemen.

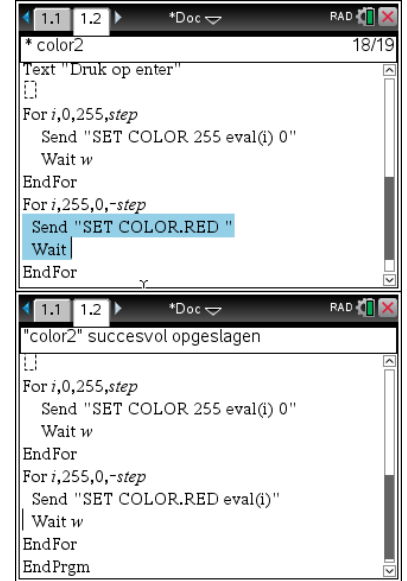
Aan het einde van deze loop moet het licht helder groen zijn.

15. Voeg zelf een loop toe om BLAUW toe te voegen.
16. Voeg dan een loop toe om GROEN te verwijderen.
17. Voeg een loop toe om ROOD weer toe te voegen.
  - Welke kleur zie je aan het einde van deze loop?
18. Voeg een loop toe die het BLAUW verwijdert.
19. Tenslotte, voeg een loop toe die ook het ROOD verwijdert.
  - Wat is de kleur van de LED aan het einde van het programma?
  - Wat gebeurt er als alle drie de kanalen de waarde van 0 hebben?

**Tip:** Als ROOD en BLAUW aan zijn dan krijgen we magenta (paars). Als BLAUW en GROEN aan zijn dan krijgen we Cyaan. Aan het einde van het programma kan het zo zijn dat de LED niet uit is omdat de laatste waarde die naar de LED gestuurd is mogelijk niet gelijk is aan 0. Om er zeker van te zijn dat hij uit is kun je nog send "SET COLOR 0 0 0" naar de HUB sturen.

## UNIT 2: OEFENBLAD 2

### DOCENTENHANDLEIDING



```
*color2 18/19
Text "Druk op enter"
[]
For i,0,255,step
  Send "SET COLOR 255 eval(i) 0"
  Wait w
EndFor
For i,255,0,-step
  Send "SET COLOR.RED "
  Wait
EndFor
[]
*color2" succesvol opgeslagen
[]
For i,0,255,step
  Send "SET COLOR 255 eval(i) 0"
  Wait w
EndFor
For i,255,0,-step
  Send "SET COLOR.RED eval(i)"
  Wait w
EndFor
EndPrgm
```