



In deze derde les van module 4 leer je over draaien naar een specifieke hoekrichting, timing en het werken met de COLOR LED op de Rover.

#### Doelen:

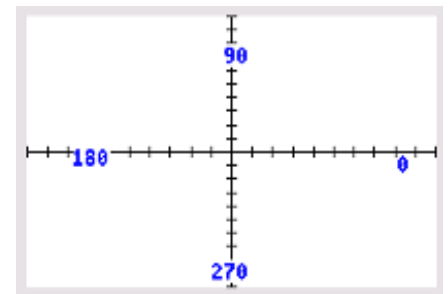
- De opdracht **TO ANGLE** gebruiken
- De **RV.COLOR** LED op de Rover gebruiken
- Timing aansturen op de rekenmachine en de Rover

Deze les gaat over drie extra mogelijkheden van de Rover:

- De opdracht **TO ANGLE** (die verschilt van **LEFT** en **RIGHT**)
- De **RV.COLOR** LED op de Rover aanzetten (gelabeld als 'Color' in de linkerbovenhoek van de voorkant, naast de indicatoren voor het batterij-niveau)
- Het synchroniseren van je programma met de bewegingen van de Rover door **Wait** te gebruiken

#### TO ANGLE

De opdracht **Send "RV TO ANGLE <getal>"** wordt gebruikt om de Rover te laten draaien naar een bepaalde koers. Wanneer je een opdracht stuurt om de Rover te verbinden, wordt de koers van de Rover ingesteld op 0 graden, wat, in de wiskundige wereld, oost is (vanuit de oorsprong 'kijkend' naar de positieve x-as). In deze wereld geldt dat noord 90 graden is, west 180 graden en is 270 graden. Zie het plaatje hier rechts.



Richtingen

Ongeacht waar de Rover op dit moment naar toe rijdt, zorgt de opdracht **Send "RV TO ANGLE 0"** dat de Rover draait in de richting waar hij heen ging toen de opdracht "CONNECT RV" werd gegeven.

De standaard hoekmaat is **DEGREES** maar je kunt ook **RADIANS** of **GRADS** specificeren (te kiezen vanuit het menu **RV Settings**).

#### Probeer dit:

- Send "RV TO ANGLE 90"**
- Wait 2**
- Send "RV TO ANGLE 180"**
- Wait 2**
- Send "RV TO ANGLE 270"**
- Wait 2**
- Send "RV TO ANGLE 360"**

```

3.1 4.1 4.2 *Rover - 1...ams RAD 8/8
rover43a
Prgm
Send "CONNECT RV"
Send "RV TO ANGLE 90"
Wait 2
Send "RV TO ANGLE 180"
Wait 2
Send "RV TO ANGLE 270"
Wait 2
Send "RV TO ANGLE 360"
EndPrgm

```

Deed de Rover wat je verwachtte dat hij deed?



#### Jouw programma synchroniseren met Rover

Programma's op de rekenmachine zijn gereed ('done') voordat de Rover stopt met bewegen. Dit is zo omdat de opdrachten voor rijden ('driving') sneller worden opgeslagen in de TI-Innovator™ Hub dan de Rover ze kan verwerken. Opdrachten om te rijden ('driving') worden opgeslagen in een wachtrij en worden verwerkt door de TI-Innovator Hub wanneer de Rover er klaar voor is.

In deze activiteit gaan we een programma schrijven om volgens een willekeurig patroon te bewegen en de **RV.COLOR** LED op de Rover te laten oplichten terwijl de Rover in beweging is. We zullen ook de opdracht **TO ANGLE** gebruiken en **eval()** opnemen om naar de passende richting te draaien.

#### Het programma inrichten:

1. Begin het programma met de opdracht **CONNECT RV**.
2. Voeg een **For**-lus toe (**menu > Control > For...EndFor**) om de Rover volgens een willekeurig patroon te laten bewegen. Je kunt altijd regels toevoegen aan de kern van de lus als je ze nodig hebt
3. Voeg, in de kern van de lus, de opdracht toe om **FORWARD** te bewegen.  
**Send "RV FORWARD 1"**
4. Voeg om de Rover te laten draaien naar een *willekeurige* richting, eerst de opdracht: **h:=randInt(0,360)** toe
5. Voeg de opdracht **TO ANGLE** toe om te draaien in de richting **h**.  
**Send "RV TO ANGLE eval(h)"**
6. Voer het programma nu uit om te zien dat:
  - De Rover beweegt volgens een willekeurig patroon.
  - Het programma ongeveer meteen gereed ('done') is, terwijl de Rover nog moet bewegen,

```

* rover43 6/24
Denne rover43
Prgm
© makes a random sketch (scribble)
Local h,i
Send "CONNECT RV"
Text "Press enter to start."
For i,1,10
|
EndFor

```

```

* rover43 9/16
Send "CONNECT RV"
Text "Press enter to start."
For i,1,10
  Send "RV FORWARD 1"
  h:=randInt(0, 360)
  Send "RV TO ANGLE eval(h)"
EndFor

```

#### RV.COLOR

Aangezien de kleurenLED op TI-Innovator Hub verborgen is binnenin de Rover hebben we een andere kleurenLED bovenop de Rover gekregen om aan te sturen. De naam van de kleurenLED van de Rover is **RV.COLOR** en werkt met behulp van dezelfde aansturing als de ingebouwde LED. Je kunt elk van de vier opdrachten gebruiken in het **menu > Hub > Rover (RV) > RV Color**, dat je hier rechts ziet.

**Send "SET RV.COLOR 255 255 255"** geeft een wit licht.



# 10 minuten programmeren

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

### MODULE 4: OEFENBLAD 3

### DOCENTENHANDLEIDING

#### RV.COLOR toevoegen aan je programma

7. Voeg een **RV.COLOR** opdracht toe aan de kern van de lus onmiddellijk voor de opdracht **FORWARD 1**. De kleurwaarden kun je zelf bepalen.
8. Voer het programma opnieuw uit om te zien wat er gebeurt. Merk op dat de LED bijna meteen oplicht en aan blijft.

```

* rover43 12/15
send CONNECT RV
Text "Press enter to start."
For i,1,10
  Send "SET RV.COLOR 128 246 100"
  Send "RV FORWARD 1"
h:=randInt(0,360)
Send "RV TO ANGLE eval(h)"

```

Laten we nu proberen om de LED *alleen* te laten oplichten als de Rover vooruit beweegt. Hiervoor is het nodig dat we de rekenmachine laten wachten (**Wait**) totdat elk gedeelte is voltooid en dat we dan de LED uitschakelen terwijl de Rover aan het draaien is.

We moeten een **Wait** –opdracht toevoegen aan ons programma om aan te sturen wanneer de LED aan en uit wordt geschakeld.

**Docenten Tip:** Dit is een goed moment om te experimenteren als je de tijd hebt. Anders dan bij de opdrachten om te rijden ('drive'), wordt de kleur-opdracht ('COLOR') verwerkt zodra deze is geaccepteerd door de TI-Innovator Hub, dus de timing van de beweging en van de LED zijn niet gesynchroniseerd. Dit wordt overgelaten aan de programmeurs.

9. Hoe lang doet de Rover erover om **FORWARD 1** te bewegen? Ongeveer 1 seconde?  
Voeg een opdracht **Wait 1** toe na de opdracht **FORWARD 1**. Je kunt de waarde aanpassen als deze niet geschikt is voor jouw omgeving. Bedenk dat **Wait** te vinden is in het **menu > Hub** menu.
10. Voer het programma uit om het te testen.

```

* rover43 9/16
send CONNECT RV
Text "Press enter to start."
For i,1,10
  Send "SET RV.COLOR 128 246 100"
  Send "RV FORWARD 1"
  Wait 1
h:=randInt(0,360)
Send "RV TO ANGLE eval(h)"

```

Merk op dat de LED nog steeds aan staat. Wij moeten de LED uitzetten nadat de Rover klaar is met bewegen.

11. Hoe langt doet de Rover over een draai? Ongeveer 1 seconde? Voeg een opdracht **Wait 1** toe.
12. Schakel na de opdracht **Wait 1** de LED uit door toe te voegen: **Send "SET RV.COLOR 0 0 0"**.
13. Test je programma. Schakelde de Rover het licht op het goede moment uit?

```

* rover43 9/16
send CONNECT RV
Text "Press enter to start."
For i,1,10
  Send "SET RV.COLOR 128 246 100"
  Send "RV FORWARD 1"
  Wait 1
  Send "SET RV.COLOR 0 0 0"
h:=randInt(0,360)
Send "RV TO ANGLE eval(h)"

```



## 10 minuten programmeren

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

We moeten ook wachten, terwijl de Rover naar een nieuwe richting draait, om te controleren of het licht is uitgeschakeld.

14. Voeg een opdracht **Wait** toe *na* de opdracht **TO ANGLE**. Deze **Wait** zou lang genoeg moeten zijn om elke draai tussen 0 en 360 graden aan te kunnen. .

#### Uitbreiding

Kan je zorgen voor een *andere* kleur bij elke hoek? Hint: Gebruik **eval(iets)**, en neem de waarde van **h** daarin op, maar wees zorgvuldig met het toegestane bereik van waarden.

Extra uitdaging:

Maak van de wachttijd bij de draai een functie die afhankelijk is van de hoek die gedraaid gaat worden door de draairichting van de Rover te observeren wanneer **TO ANGLE** wordt gebruikt.

**Docenten Tip:** Opdrachten voor het rijden worden onmiddellijk naar TI-Innovator Hub gezonden en worden daar in een wachtrij geplaatst totdat de Rover klaar is voor elk van deze deze opdrachten.

De **Wait**-opdrachten zijn instructies *voor de rekenmachine* om te wachten terwijl de Rover de huidige taak afmaakt. Grote waarden voor **Wait** werken wel, maar het zal lijken alsof de Rover pauzeert tijdens de route totdat hij de volgende opdracht ontvangt.

Voor de uitbreiding kunnen de leerlingen een combinatie gebruiken van *i*, de lus-variabele en *h*, de richting, om verschillende kleuren te produceren. Denk eraan dat het bereik van de waarden voor r, g en b van 0 tot en met 255 loopt.

## MODULE 4: OEFENBLAD 3

### DOCENTENHANDLEIDING

```
* rover43 12/13
For i,1,10
  Send "SET RV.COLOR 128 246 100"
  Send "RV FORWARD 1"
  Wait 1
  Send "SET RV.COLOR 0 0 0"
  h:=randInt(0,360)
  Send "RV TO ANGLE eval(h)"
  Wait 3
EndFor
EndPrgm
```