

Kapitel 6: Använda TI-Innovator Hub och TI-Innovator Rover

Övning 3 Styra en robot

I denna tredje övning i kapitel 6 kommer du att lära dig hur man ansluter robotbilen TI-Innovator™ Rover med hjälp av biblioteket `ti_rover`.

Syfte:

- Upptäcka `TI_rover`-modulen
- Skriva och använda ett skript för att använda TI-Innovator Rover och dess styrdon.
- Använda en öppen loop och en villkorsinstruktion

I den här övningen kommer du att skapa ett skript som ger ROVER möjlighet att köra en bana som markeras av belysning av RGB-dioden, så länge avståndet (mätt med RANGER-sensorn) respekterar en gräns skriven i en villkorsinstruktion.

Avancera en sträcka på 2 m

Så länge rörelsen inte stoppas av användaren

`a` ← ett avstånd till ett uppmätt objekt

om `a < 0.2`

visa då en röd färg och stoppa

annars visa en grön färg och fortsätt

Stoppa, visa en blå färg

Vänta 1 s

Stänga av dioden

Slå på den blå lysdioden för att markera slutet på spelet.

- Starta ett nytt skript och namnge det KAP6OVN3
- Importera `ti-rover`-biblioteket från `modul`-menyn
- Bekräfta genom att trycka på `[enter]`.



```
FILE MANAGER
time,ti_system,ti_rover
Select Program Type
1:Blank Program
2:Math Calculations
3:Random Simulation
4:Plotting (x,y) & Text
5:Data Sharing
6:Hub Project
7:Rover
8:TI STEM Project Helpers...

Esc
```

```
EDITOR: KAP6OVN3
PROGRAM LINE 0005
#TI-Rover
from time import *
from ti_system import *
import ti_rover as rv

Fns... a A # Tools Run Files
```

- Rensa skärmen med hjälp av **instruktionen disp_clr()** som finns i **ti_system**-menyn.
- Fortfarande i **ti_system** väljer du **disp_cursor()**-instruktionen med värdet 0 för att inte visa markören.
- Be ROVER att gå vidare. Måttenheten på avståndet lämnas åt ditt val. Standardavståndet är 0,1 m. **rv.forward(20)** gör att robotfordonet går framåt 2 m. Instruktionen **rv.forward()** finns i **modulmenyn**. Välj sedan **7: ti_rover** och slutligen **2: forward(distance)** i i **Drive**-menyn.
- Ange sedan början av en öppen loop som finns i **modulmenyn** och sedan i **ti_system**-biblioteket.

Lärarkommentar: Många av instruktionerna i **ti_system**-biblioteket finns också i **ti_rover**-bibliotekets meny under Commands.

```

EDITOR: KAP60VN3
PROGRAM LINE 0007
#TI-Rover
from time import *
from ti_system import *
import ti_rover as rv
disp_clr()
disp_cursor(0)
-
    
```

```

EDITOR: KAP60VN3
PROGRAM LINE 0010
#TI-Rover
from time import *
from ti_system import *
import ti_rover as rv
disp_clr()
disp_cursor(0)
disp_at(6, "[clear] för att stann
a", "center")
rv.forward(20)
while not escape():
**
-
    
```

- Skapa en variabel **a** till vilken avståndet uppmätt med RANGER tilldelas. För att göra detta så skriv bokstaven **a** och lämna sedan markören i slutet av den här bokstaven. Skriv sedan instruktionen **rv.ranger_measurement()** som finns i **modulmenyn** sedan **7: ti_rover** sedan **I / O** (ingångar-utgångar); **1: Ingångar** och slutligen **1: rv.ranger_measurement()**. Måttenheten är meter.
- Skapa nu villkorssatsen. Om det uppmätta avståndet är mindre än 20 cm stannar roboten och RGB-lysdioden lyser rött. Instruktionen **rv.color()** finns i **ti_rover**-biblioteket i **I/O-menyn** och sedan **2: Outputs**. **rv.stop()** är en körinstruktion och är därför placerad under motsvarande meny (**Drive**). Annars lyser RGB-dioden grönt och roboten fortsätter sin väg tills den inställda sträckan nås. Instruktionen **rv.resume()** avslutar bearbetningen av åtgärderna i kön.
- I slutet av slingan:
 - Roboten stannar. **rv.stop()**.
 - Skärmen raderas.
 - Lysdioden visar en blå färg.
 - Det är 1 s fördröjning innan lysdioden slocknar.

```

EDITOR: KAP60VN3
PROGRAM LINE 0019
while not escape():
**a=rv.ranger_measurement()
**if a<0.2:
***rv.color_rgb(255,0,0)
***rv.stop()
**else:
***rv.color_rgb(0,255,0)
***rv.resume()
rv.stop()
disp_clr()
rv.color_rgb(0,0,255)_
sleep(1)
rv.color_rgb(0,0,0)
-
    
```