

# Der TI-Innovator Rover



Dieses Buch gehört

---

# Herzlich Willkommen,

in diesem Büchlein stehen die wichtigsten Dinge, die Du zum Programmieren und Steuern des TI-Innovator Rovers benötigst.

Viel Spaß!



# Inhaltsverzeichnis


Wichtiges.....	5
Häufig benötigte Befehle .....	6
Steuern des Rovers .....	7
Neue Datei anlegen .....	8
Befehle mit dem Menü einfügen .....	9
Programm starten .....	10
Zurück zum Programmeditor .....	11
Datei abspeichern .....	12
Datei öffnen .....	13
FOR-Schleife .....	14

# VORSICHT!

- Nicht auseinanderbauen, ALLE Teile belassen wie sie sind!
- Nicht mehr als 1 kg Last auf die Plattform!
- Nur auf dem Boden verwenden! (Absturzgefahr!)
- Nur auf harter und griffiger Oberfläche verwenden!  
(Papier unterlage gut festkleben!)
- Stifte nicht als Hebel oder zum Ziehen/Schieben des Rover verwenden!
- Während und nach der Ausführung eines Programms, den Rover nicht SCHIEBEN oder ZIEHEN!
- Im Notfall den Rover anheben und durch langes Drücken der „esc“-Taste das Programm abbrechen

# Häufig benötigte Befehle 1/2

neue Datei anlegen

 on

- Neu (1)
- Python hinzufügen (A)
- Neu (1)
- Programmname eingeben: Roverprogrammierung





Datei speichern



- Datei (1)
- Speichern (4)
- Dokumentname eingeben
- Speichern (4)



## Datei schließen

doc

→ Datei (1)

→ Schließen (3)

enter



## Datei öffnen

on

→ Eigene Dateien  
(1)

→

→ Datei auswählen (Pfeiltasten)

enter

# Häufig benötigte Befehle 2/2

im Programmierer ..



**Programm prüfen  
und zum Starten  
vorbereiten**

ctrl



R



im Programmierer ..



**Vorwärtsfahren**

\* menu - 9 - 2 - 1  
\* „rv.forward(Abstand)“



**Rückwärtsfahren**

\* menu - 9 - 2 - 2  
\* „RV BACKWARD(abstand)“



**nach links drehen**

\* menu -9 -2 3  
\* „RV LEFT(Winkel)“

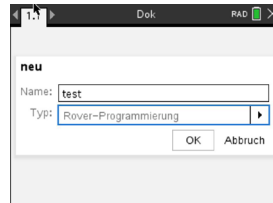
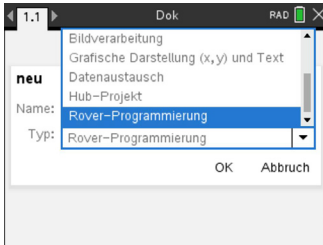
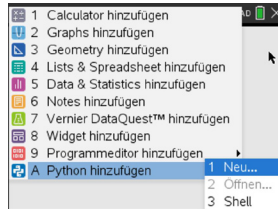


**nach rechts drehen**

\* menu - 9 - 2 - 4 „RV  
RIGHT(Winmkle)“



# Neue Datei anlegen



# Programm benennen



1.1 | Dok | RAD

Neu

Name: test1

Typ: Programm

Bibliothekszugriff: Keine

OK Abbruch

4

5

1.1 | \*Dok | RAD

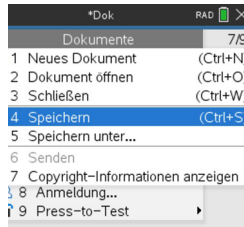
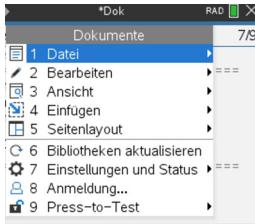
test1 1/1

```
Define test1()=
Prgm
{
}
EndPrgm
```



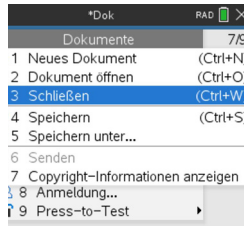
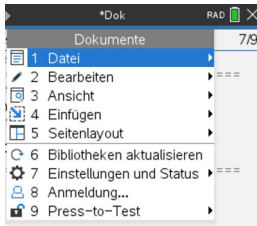
# Datei speichern & schließen

doc ▾



enter

doc ▾

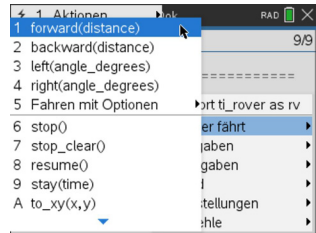
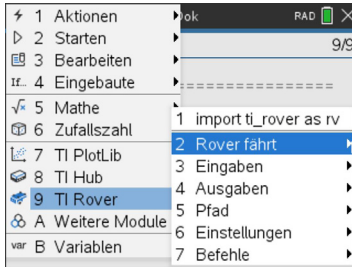


enter

# Rover fahren lassen (vorwärts / rückwärts)

Tip:  
für rückwärts  
stätt (1) die  
(2) wählen

Press:



```
1.1 *Dok CAPS RAD 9/9
[Test.py
# Rover Coding
#=====
import ti_rover as rv
from math import *
import ti_plotlib as plt
from ti_system import *
from time import *
#=====
rv.forward(Abstand)
```

**Tippt hier die Anzahl der Einheiten ein, die der Rover vorwärts fahren soll**

**1 Einheit entspricht 10 cm**

# Rover fahren lassen (links / rechts)

Startet wieder mit: Menü



```
1 Aktionen
2 Starten
3 Bearbeiten
4 Eingebaute
5 Mathe
6 Zufallszahl
7 TI PlotLib
8 TI Hub
9 TI Rover
A Weitere Module
var B Variablen
```

```
1 import ti_rover as rv
2 Rover fährt
3 Eingaben
4 Ausgaben
5 Pfad
6 Einstellungen
7 Befehle
```

```
1 forward(distance)
2 backward(distance)
3 left(angle_degrees)
4 right(angle_degrees)
5 Fahren mit Optionen
6 stop()
7 stop_clear()
8 resume()
9 stay(time)
A to_xy(x,y)
```

```
import ti_rover as rv
Rover fährt
Eingaben
Ausgaben
Pfad
Einstellungen
Befehle
```

```
1.1 *Dok
*Test.py
# Rover Coding
#-----
import ti_rover as rv
from math import *
import ti_plotlib as plt
from ti_system import *
from time import *
#-----
rv.left(Winkel
degrees
```

Winkel in Grad  
(90° = rechter Winkel)

# Fahren mit Option (5-1)

Im Menü

„Rover fährt (2)“

- 1 forward(distance)
- 2 backward(distance)
- 3 left(angle\_degrees)
- 4 right(angle\_degrees)
- 5 **Fahren mit Optionen**
- 6 stop()
- 7 stop\_clear()
- 8 resume()
- 9 stay(time)
- A to\_xy(x,y)

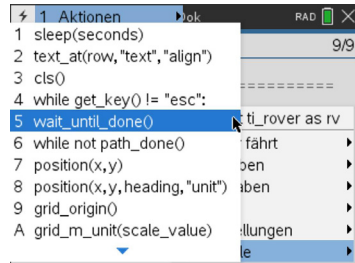
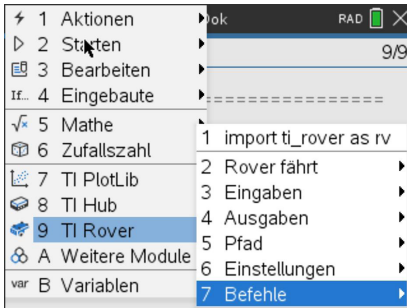


```
forward(distance)
b 1 forward_time(time)
le 2 backward_time(time)
ri 3 forward(distance, "unit")
F 4 backward(distance, "unit")
s 5 left(angle, "unit")
s 6 right(angle, "unit")
re 7 forward_time(time, speed, "rate")
s 8 backward_time(time, speed, "rate")
tc 9 forward(distance, "unit", speed, "rate")
A backward(distance, "unit", speed, "rate")
```

```
*Test.py 9/9
Rover Coding
=====
>ort ti_rover as rv
n math import *
>ort ti_plotlib as plt
n ti_system import *
n time import *
=====
forward(rv.forward_time(Zeit))
Dauer0, 1-100 s
```

Info:  
Zeit in Sekunden

# Fahren mit Option (9-7-5) Abstandmessung (aufgabenblatt)



Info  
Zeit: in Sekunden  
Geschwindigkeit:  
0.14-0.23  
Einheit: m/s

**Pfade Menü 9 -5 -2 Menü 9 -5 -9  
(warten auf Informationen)**



# Impressum

## Medieninhaber

Paris Lodron Universität Salzburg (PLUS)  
Pädagogische Hochschule Salzburg (PHS)

## Projekt

Inter-Di-Ko  
Innovation - Digitalisierung - Kommunikation  
gefördert durch die  
FFG-Forschungsförderungsgesellschaft

## Inhalte

Simon Plangg (PHS)

## Grafikdesign

Wolf Hilzensauer (PHS)

## Grafiken

Open Clipart Vectors - pixabay

## Website

[www.inter-di-ko.net](http://www.inter-di-ko.net)



