

# Experimente

Mehr oder weniger sinnvolle Experimente über Programmierung und Robot-Game

- [Präzise Ausgänge](#)
  - [Linear Motion - Test Aufbaute](#)
  - [Rotary Motion - Test Aufbaute](#)
  - [Fazit](#)

# Präzise Ausgänge

Ziel war es für die kommende Saison Outputs mit erhöhter Präzision anzusteuern.

# Linear Motion - Test Aufbaute

Ziel des ersten Test Aufbaus war es, den weißen Indikator präzise zu steuern. Dazu wurden Gearracks verwendet. Jedoch konnte man es durch das Getriebe erstmal nicht Präzise steuern.

Dieses Problem lösen wir software-technisch:

```
import color as col
from ..gsgr.enums import Attachment
from ..gsgr.movement import run_attachment
import motor
import time
from ..gsgr.config import cfg
display_as = "P"
color = col.RED

def run():
    run_attachment(Attachment.FRONT_RIGHT, -300, 0.5, stall=True) #1. gegen eine Blockade
fahren
    run_attachment(Attachment.FRONT_RIGHT, 500, 60, stall=True,
when_i_say_duration_i_mean_degrees=True) #2. Wieder positionieren
    run_attachment(Attachment.FRONT_RIGHT, 500, 85, stall=True,
when_i_say_duration_i_mean_degrees=True) #3. Bewegung
    time.sleep(1)
    run_attachment(Attachment.FRONT_RIGHT, 500, 85, stall=True,
when_i_say_duration_i_mean_degrees=True)
    time.sleep(1)
    run_attachment(Attachment.FRONT_RIGHT, 500, 85, stall=True,
when_i_say_duration_i_mean_degrees=True)
    time.sleep(1)
    run_attachment(Attachment.FRONT_RIGHT, -300, 45, stall=True,
when_i_say_duration_i_mean_degrees=True) #4. Kompensation des Richtungswechsels
    run_attachment(Attachment.FRONT_RIGHT, -500, 85, stall=True,
when_i_say_duration_i_mean_degrees=True) #5. Bewegung
    time.sleep(1)
    run_attachment(Attachment.FRONT_RIGHT, 300, 45, stall=True,
```

```
when_i_say_duration_i_mean_degrees=True) #6. Kompensation des Richtungswechsels
    run_attachment(Attachment.FRONT_RIGHT, 500, 85, stall=True,
when_i_say_duration_i_mean_degrees=True) #7. Bewegung
```

1. Um die Zähne des Getriebecontainers ([18947](#)) und des Getriebezahnrads ([35185](#)) an einander zu drehen
2. Um die Zähne des Getriebecontainers ([18947](#)) und des Getriebezahnrads ([35185](#)) auf die andere Seite an einander zu drehen
3. Bewegung auf erste Position
4. Bei Richtungswechsel muss um 45 Grad gedreht werden, um die Zähne des Getriebecontainers ([18947](#)) und des Getriebezahnrads ([35185](#)) auf die andere Seite an einander zu drehen
5. Bewegung auf vierte Position
6. Bei Richtungswechsel muss um 45 Grad gedreht werden, um die Zähne des Getriebecontainers ([18947](#)) und des Getriebezahnrads ([35185](#)) auf die andere Seite an einander zu drehen
7. Bewegung auf fünfte Position

Das Problem kann man auch hardware-technisch lösen:

```
import color as col
from ..gsgr.enums import Attachment
from ..gsgr.movement import run_attachment
import motor
import time
from ..gsgr.config import cfg
display_as = "P"
color = col.RED

def run():
    run_attachment(Attachment.FRONT_RIGHT, -300, 0.5, stall=True)
    run_attachment(Attachment.FRONT_RIGHT, 500, 60, stall=True,
when_i_say_duration_i_mean_degrees=True)
    run_attachment(Attachment.FRONT_RIGHT, 500, 85, stall=True,
when_i_say_duration_i_mean_degrees=True)
    time.sleep(1)
    run_attachment(Attachment.FRONT_RIGHT, 500, 85, stall=True,
when_i_say_duration_i_mean_degrees=True)
    time.sleep(1)
    run_attachment(Attachment.FRONT_RIGHT, 500, 85, stall=True,
```

```
when_i_say_duration_i_mean_degrees=True)
    time.sleep(1)
    run_attachment(Attachment.FRONT_RIGHT, -500, 85, stall=True,
when_i_say_duration_i_mean_degrees=True
```

Es muss nicht mehr der Richtungswechsel einkalkuliert werden, da durch die Spannung des Gummis die Zähne des Getriebecontainer ([18947](#)) und des Getriebezahnrads ([35185](#)) schon an einander gedrückt werden.

# Rotary Motion - Test Aufbaute

Ziel des ersten Test Aufbaus war es, den weißen Indikator präzise zu drehen. Dazu wurde eine simple [90°-Übersetzung](#) verwendet. Jedoch konnte man es durch das Getriebe erstmal nicht Präzise drehen.

Dieses Problem lösen wir software-technisch:

```
import color as col
from ..gsgr.enums import Attachment
from ..gsgr.movement import run_attachment
import motor
import time
from ..gsgr.config import cfg
display_as = "P"
color = col.RED

def run():
    run_attachment(Attachment.FRONT_RIGHT, -300, 0.5, stall=True) #1. gegen eine Blockade
    drehen
    run_attachment(Attachment.FRONT_RIGHT, 300, 60, stall=True,
when_i_say_duration_i_mean_degrees=True) #2. Wieder positionieren
    run_attachment(Attachment.FRONT_RIGHT, 500, 90, stall=True,
when_i_say_duration_i_mean_degrees=True) #3. erste Bewegung
    time.sleep(1)
    run_attachment(Attachment.FRONT_RIGHT, 500, 180, stall=True,
when_i_say_duration_i_mean_degrees=True)
    time.sleep(1)
    run_attachment(Attachment.FRONT_RIGHT, 300, -45, stall=True,
when_i_say_duration_i_mean_degrees=True) #4. Kompensation des Richtungswechsels
    run_attachment(Attachment.FRONT_RIGHT, 500, -90, stall=True,
when_i_say_duration_i_mean_degrees=True) #5. dritte Bewegung
    time.sleep(1)
    run_attachment(Attachment.FRONT_RIGHT, 500, -90, stall=True,
when_i_say_duration_i_mean_degrees=True)
```

1. Um die Zähne des Getriebecontainers ([18947](#)) und des Getriebezahnrads ([35185](#)) an einander zu drehen
2. Um die Zähne des Getriebecontainers ([18947](#)) und des Getriebezahnrads ([35185](#)) auf die andere Seite an einander zu drehen
3. Bewegung auf erste Position
4. Bei Richtungswechsel muss um 45 Grad gedreht werden, um die Zähne des Getriebecontainers ([18947](#)) und des Getriebezahnrads ([35185](#)) auf die andere Seite an einander zu drehen
5. Bewegung auf dritte Position

Das Problem kann man auch hardware-technisch lösen:

```
import color as col
from ..gsgr.enums import Attachment
from ..gsgr.movement import run_attachment
import motor
import time
from ..gsgr.config import cfg
display_as = "P"
color = col.RED

def run():

    run_attachment(Attachment.FRONT_RIGHT, -300, 0.5, stall=True)
    run_attachment(Attachment.FRONT_RIGHT, 300, 90, stall=True,
when_i_say_duration_i_mean_degrees=True)
    run_attachment(Attachment.FRONT_RIGHT, 100, 90, stall=True,
when_i_say_duration_i_mean_degrees=True)
    time.sleep(1)
    run_attachment(Attachment.FRONT_RIGHT, 1000, 180, stall=True,
when_i_say_duration_i_mean_degrees=True)
    time.sleep(1)
    run_attachment(Attachment.FRONT_RIGHT, 1000, -100, stall=True,
when_i_say_duration_i_mean_degrees=True)
    time.sleep(1)
    run_attachment(Attachment.FRONT_RIGHT, 1000, -90, stall=True,
when_i_say_duration_i_mean_degrees=True)
```

Es muss nicht mehr der Richtungswechsel einkalkuliert werden, da durch die Spannung des Gummis die Zähne des Getriebecontainer ([18947](#)) und des Getriebezahnrads ([35185](#)) schon an einander gedrückt werden.

Präzise Ausgänge

# Fazit

Verwendungszwecke:

- Drehungen, die nur in eine Richtung gehen
- Linear Motion
- Bei Drehungen mit weniger Kraft
- Wenn man präzise Bewegungen braucht

Probleme:

- Gummi-Spannung ist unterschiedlich und oft groß

Rating:     von 5