

# Plotten von Geschwindigkeitsänderungen und Stop on Resistance

Hierbei geht es darum, die Geschwindigkeit von Motoren auszulesen und Veränderungen z.B. bei Blockaden zu erkennen.

```
import hub
import motor
import device
import time

async def loop():
    motor.run(hub.port.A, 1000) #Geschwindigkeit des Motors auf 1000 stellen
    while not hub.button.pressed(hub.button.POWER):
        print(">", device.data(hub.port.A)[0], sep="") #Übertragung der Geschwindigkeit
        time.sleep(0.2) #Damit es möglichst live, ist. Wenn zu niedrig kommt die BT-
Übertragung nicht hinterher
    while hub.button.pressed(hub.button.POWER):
        ...
```

Das ist ein einfach Aufbau, der die Geschwindigkeit des Motors ausliest. Die gemessene Geschwindigkeit ist immer 1/10 so groß, wie die eingestellte.

Damit kann man jetzt zum Beispiel für präzise Outputs ([Zum Artikel](#)) richtig nullen, ohne unterschiedlich am Ende zu stehen und zu stark zu verziehen.

```
import hub
import motor
import device
import time

def until_stalled(speed, limfuck=0.8): #Funktion: speed: Geschwindigkeit vom Motor, limfuck:
Prozent der Zielgeschwindigkeit, ab der gestoppt werden soll
```

```

lim = abs(speed) / 10 * limfuck #limfuck * 10, weil die Geschwindigkeit nur 1/10 so groß
ist
motor.run(hub.port.A, speed)
time.sleep(0.5) #Kann man noch besser machen, damit der Motor Zeit hat zu beschleunigen
while not hub.button.pressed(hub.button.POWER):
    print(">", device.data(hub.port.A)[0], sep="") #Für BT-Übertragung

    sped = device.data(hub.port.A)[0] #gemessene Geschwindigkeit
    if lim > sped > -lim: #Überprüfun, ob die Geschwingigkeit zu niedrig ist
        break
    time.sleep(0.2) #Damit nicht zu viel via BT übertragen wird, sollte man aus dem
Fenster schmeißen, wenn man keine BT-Übertragung macht

motor.run(hub.port.A, 0) Motor wieder ausschalten
return

async def loop():
    while not hub.button.pressed(hub.button.POWER): #Unendlich vor und zurück
        until_stalled(500)
        until_stalled(-500)

    while hub.button.pressed(hub.button.POWER): #Debounce
        ...

```

---

Version #5

Erstellt: 2026-06-26 20:55:35 UTC von GSG Robots

Zuletzt aktualisiert: 2026-06-28 11:50:58 UTC von Finn Hohoff