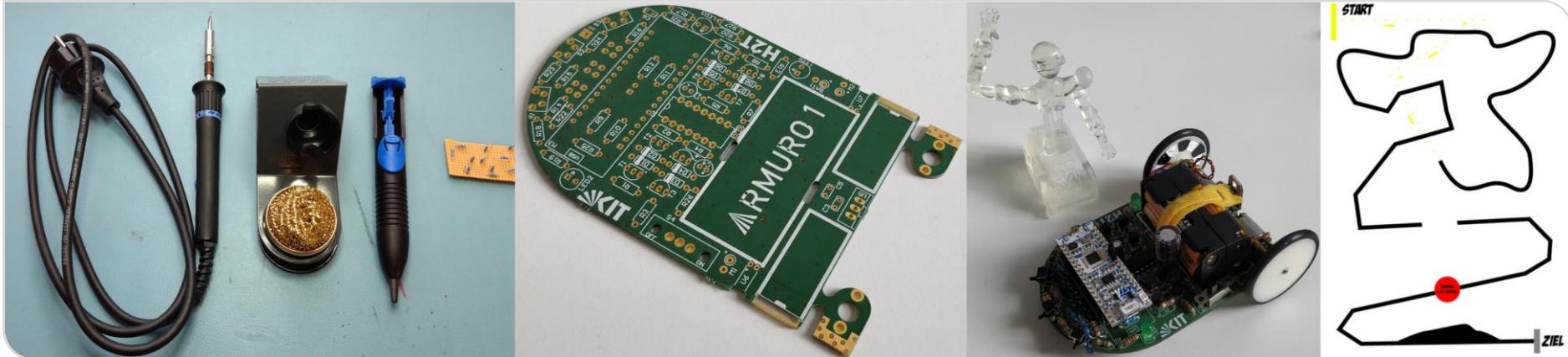


# Basispraktikum Mobile Roboter im SoSe 2024

Kolloquium 07 am 19.06.2024



## Frage 1.1

Entwirf einen P-Regler zum Linienfolgen für den Fall, dass der Armuro bereits auf der Linie positioniert ist als Blockschaltbild und programmiere den Regler als kooperativen Task. Der Regler hat als Eingänge die Helligkeitswerte der Liniensensoren und als Ausgabe die beiden PWM-Werte.

```
//Aufgabe 1.1
int speedAdjustment = 5;
void followLine(int speed)
{
    int left = adc[5] - LINE_SENSOR_LEFT_OFFSET;
    int middle = adc[0] - LINE_SENSOR_MIDDLE_OFFSET;
    int right = adc[2] - LINE_SENSOR_RIGHT_OFFSET;

    //falls alle werte ~ 0 sind, dann ist linie verloren worden -> funktion hier aufrufen zur behandlung
    int lr = left - right;

    setSpeedLeft(speed - speedAdjustment * lr);
    setSpeedRight(speed + speedAdjustment * lr);
}
```

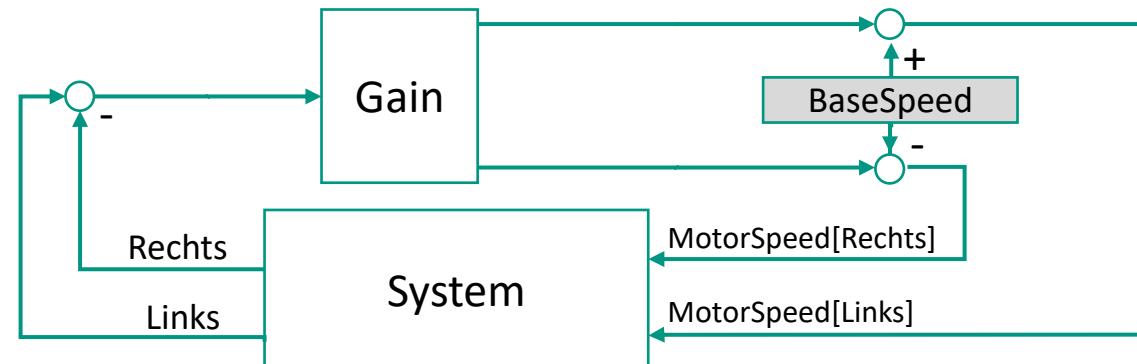
Marlon Lisak

## Frage 1.1

Entwirf einen P-Regler zum Linienfolgen für den Fall, dass der Armuro bereits auf der Linie positioniert ist als Blockschaltbild und programmiere den Regler als kooperativen Task. Der Regler hat als Eingänge die Helligkeitswerte der Liniensensoren und als Ausgabe die beiden PWM-Werte.

### ■ Regler (Linie zwischen den äußeren Sensoren):

- $Diff = Links - Rechts$
- $MotorSpeed[Links] = BaseSpeed + Diff * Gain$
- $MotorSpeed[Rechts] = BaseSpeed - Diff * Gain$



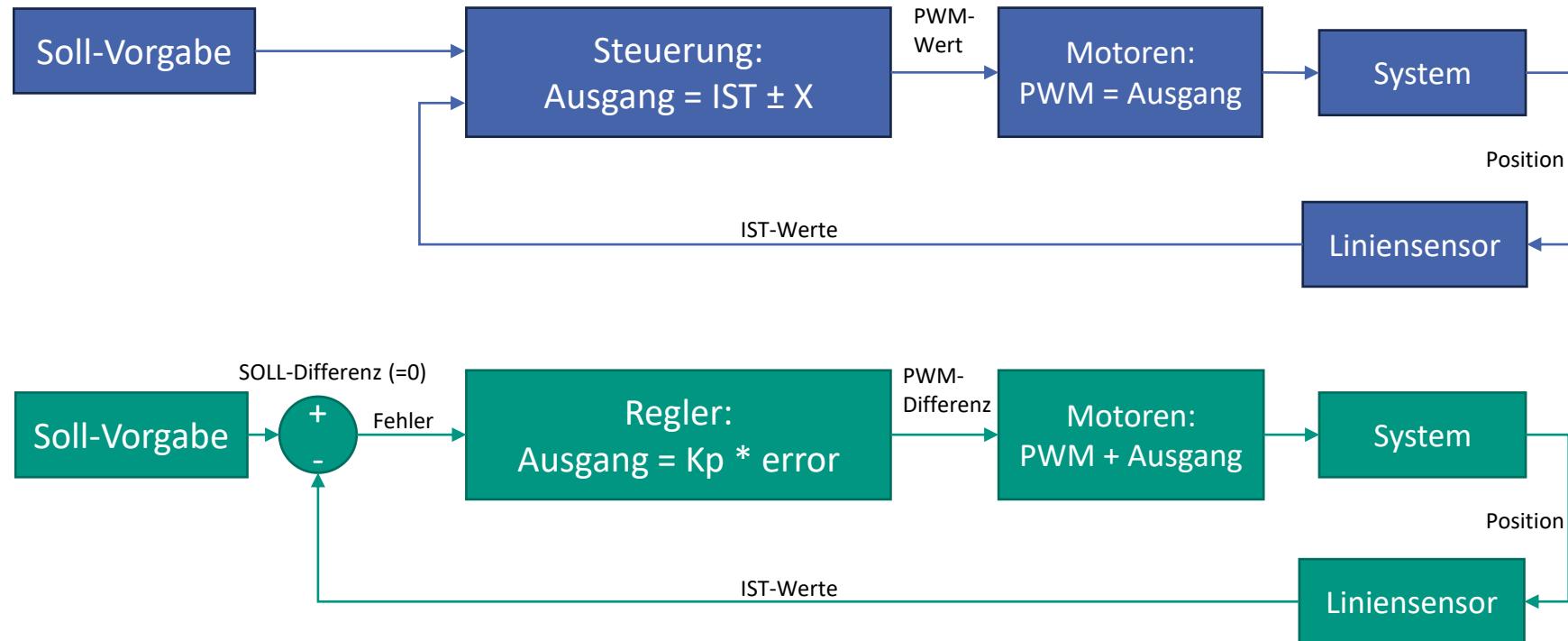
## Frage 1.2

Entwerfen Sie einen Algorithmus zum Detektieren und Beheben des Falls, dass die Linie verloren wurde.

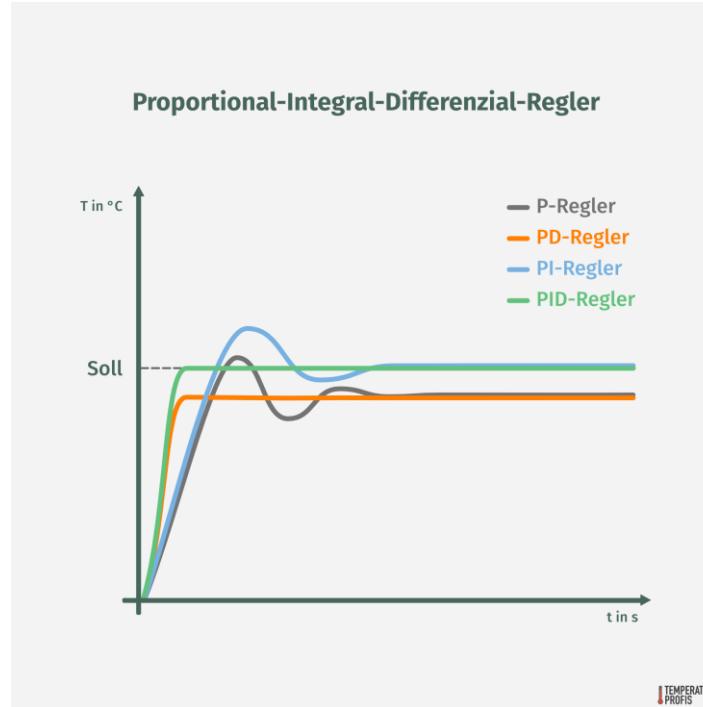
```
1 | If (left != dark && right != dark && middle != dark) {  
2 |   turnLeftRightAndSample(last_dark);  
3 |   //last_dark gibt die Richtung  
4 |   // an, in der zuletzt die Linie gesehen wurde  
5 |   //Line-gap  
6 |   if (no_line) {  
7 |     turnBackCenter();  
8 |     //wieder zurück zur ursprünglichen Orientierung  
9 |     drive5cmForward();  
10|    //parallel schauen ob schwarz wieder auftaucht  
11|    searchForLineLeftRight();  
12|  }  
13|}
```

*Beispiel*

# Steuerung vs. Regelung



# Regler-Typen (PID)



Quelle: <https://temperatur-profis.de/temperaturregler/reglertypen-pid/>

■ P-Anteil:  $K_P \cdot (SOLL - IST)$

■ D-Anteil:  $K_D \cdot \frac{d}{dt} (SOLL - IST)$

■ I-Anteil:  $K_I \cdot \int_0^t (SOLL - IST) d\tau$

# Fragen?

# Anmerkungen Code Abgaben

- Benennung der Dateien für Abgabe einhalten!

*code\_0X\_vorname\_nachname.c/.h*

oder

*code\_0X\_N\_m\_vorname\_nachname.c/.h*

- **Code innerhalb der Datei mittels Kommentares markieren!**

**// Aufgabe N.m**

- User-Code Bereiche beachten!