# Lab report 3: Modeling and PID Control

Haptic paddle modeling

A model for the *Haptic Paddle* was implemented in Simulink. To verify the model, it was compared to recorded data of the actual *Haptic Paddle* itself. For that purpose a signal generator was implemented in LabVIEW and the paddle position (hall sensor) was stored as well as the time. The position signal was filtered by a Lowpass filter for all Lab 3 exercises.

Position Filter:
Filter type: Butterworth Lowpass
Low cutoff frequency: 20Hz
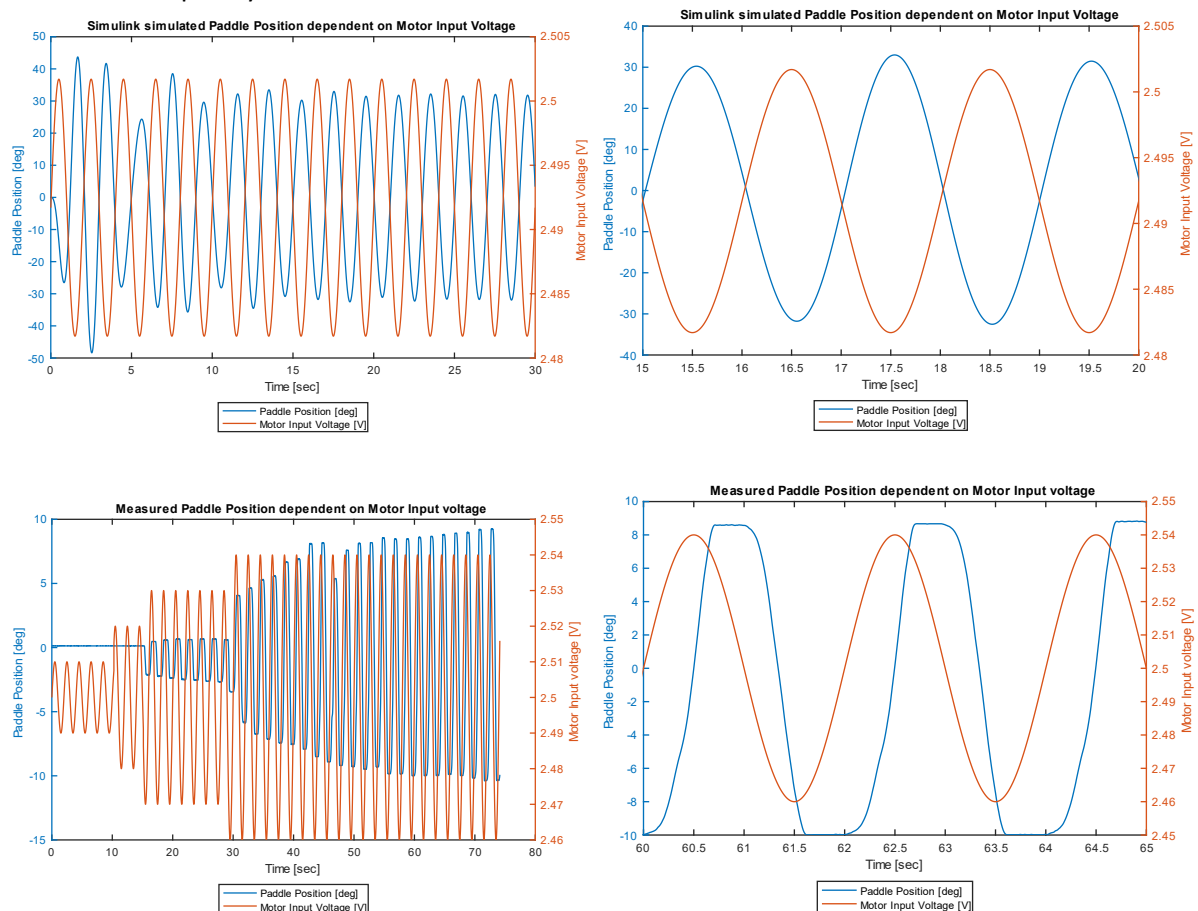
The physical paddle does need much more motor input current for a smaller amplitude in position change compared to the Simulink Model. The reason for this effect, is the static friction. The plot of our real *Haptic Paddle* also shows the unsymmetric behavior.

PID Controller

A PID controller for the *Haptic Paddle* was implemented. First the controller parameters were tuned to meet the minimum requirements for a step response (see below). After tuning the PID controller was fed with two different signals to verify its function. For verification the paddle movements were recorded (motor input voltage, position, time and video). For further development of our PID controller the D-parameter was changed to act on the process variable instead of the error (set setpoint weighting).

Minimum requirements for a step response:

Step:	Start: 0°
	End: 10°
Error band:	0.4° ($\pm$0.2°)
Overshoot max:	40% (4°)
Rise time (t_{90}):	100ms
Settling time max:	350ms

We used the Ziegler-Nichols (ZN) method introduced in the lecture to tune the PID parameters. First, we increased proportional gain until stability boundary is reached, recorded the current K_P as the ultimate gain K_u. Then we measured the oscillation period T_u. Finally, we used them to calculate K_P, K_I and K_D.

We have K_u (ultimate gain) as 0.026 and the T_u (oscillation period) as 0.2 sec.

K_P	0.60 x K_u	0.0156
K_I	1.2 x K_u / T_u	0.156
K_D	3 x K_u x T_u / 40	0.00039
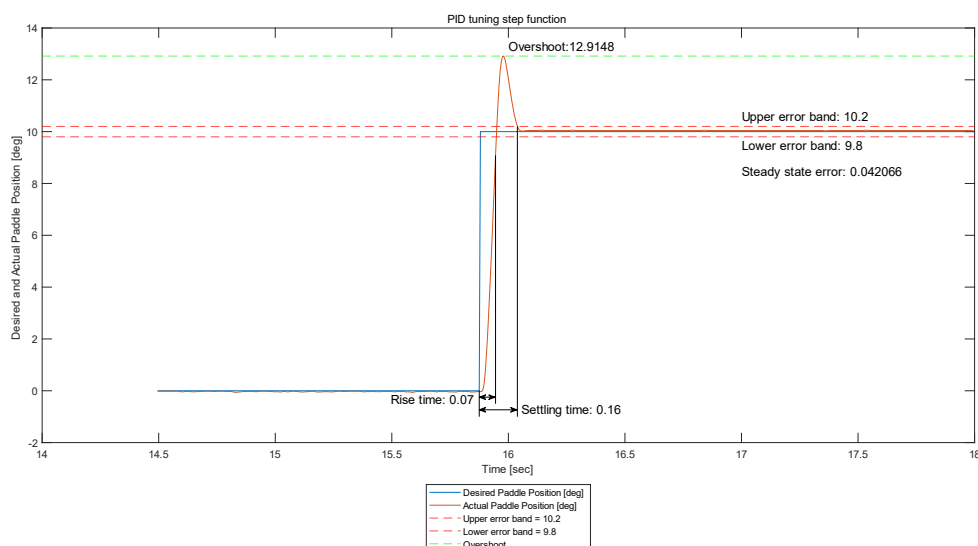
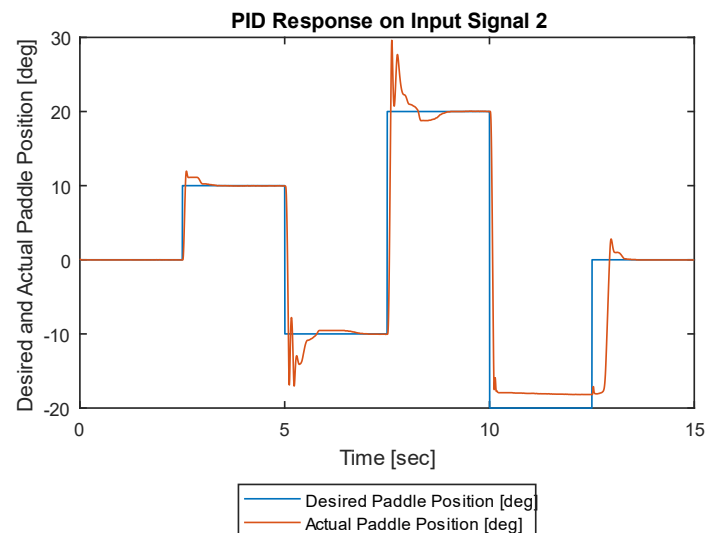
Settling time: 160ms
Overshoot: 2.9148 degree
Steady state error: 0.042066 degree
Rise time: 70ms

PID response on signal 1 and signal 2. D-part acts on error

The first plot is the signal 1 response for the whole 10-second period, the second plot is the signal 1 response for the zoom-in 5-second period.
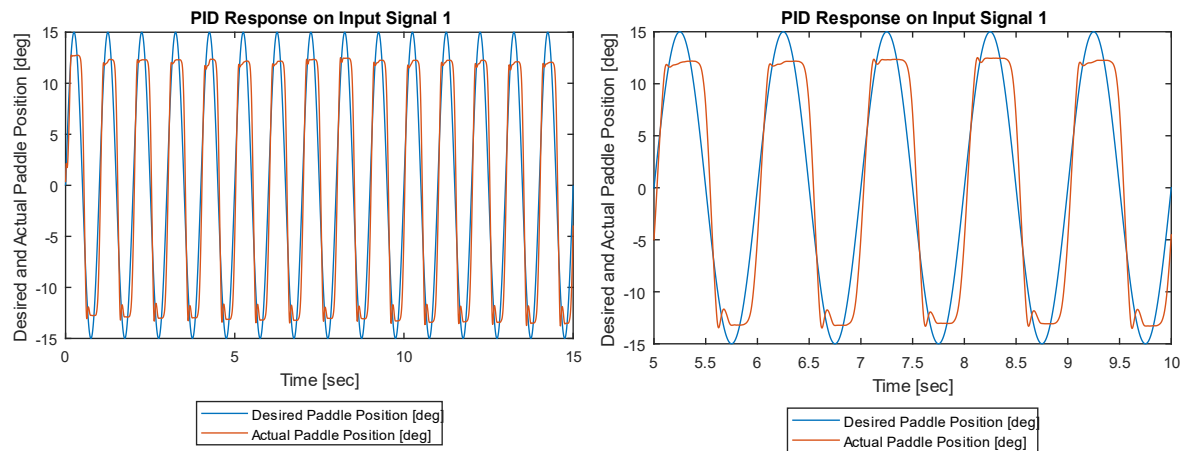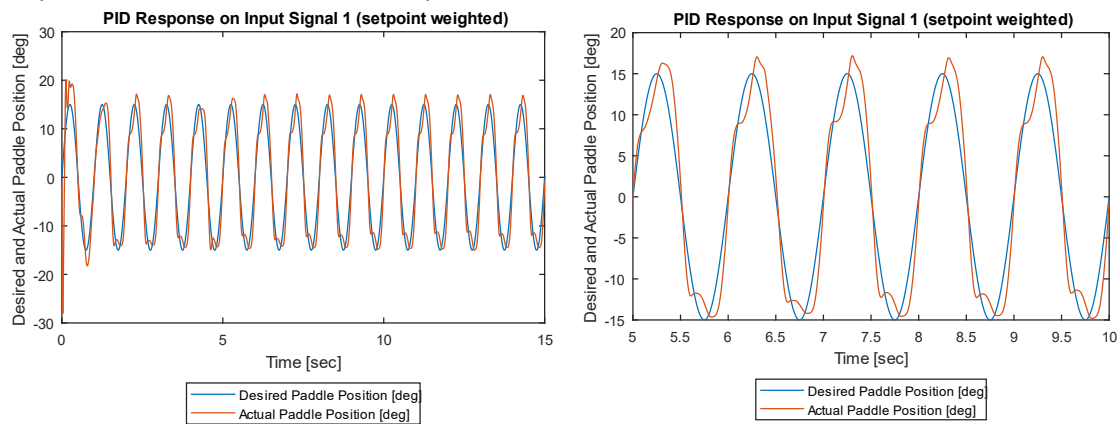

The third plot is the signal 2 response.

PID response on signal 1 and signal 2. D-part acts on process variable

The first plot is the signal 1 response for the whole 10-second period, the second plot is the signal 1 response for the zoom-in 5-second period.

The third plot is the signal 2 response
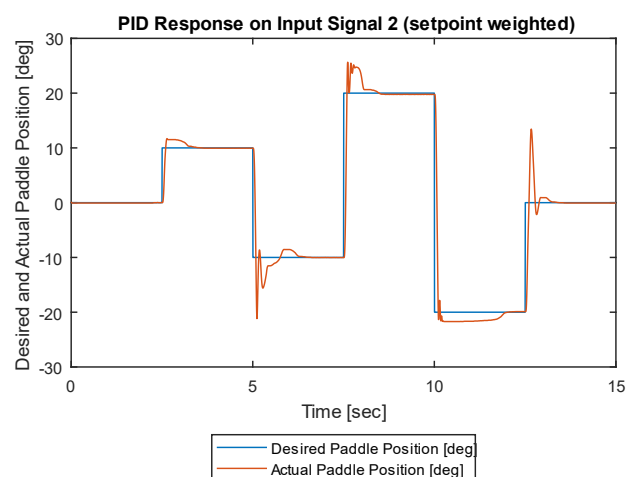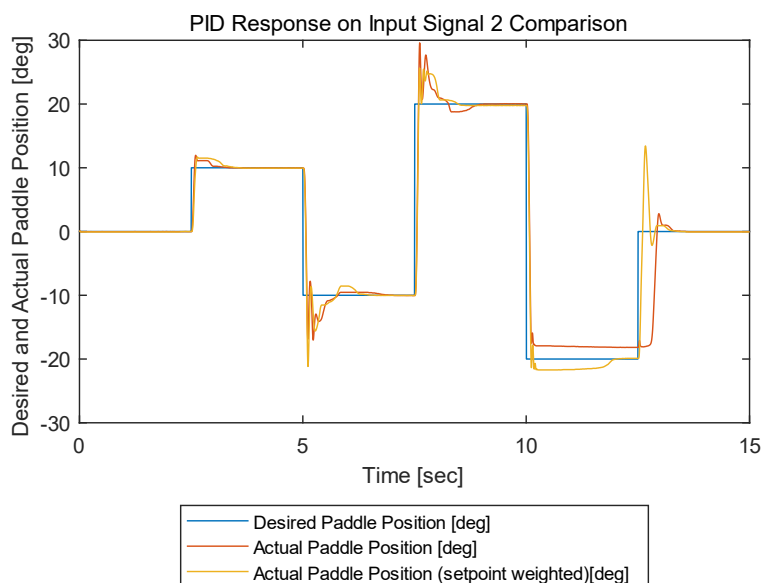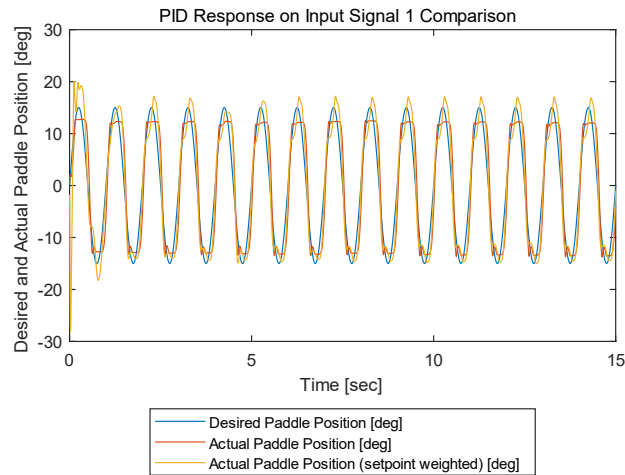

Comparison, observation, and interpretation is on the last page.

Comparison, observation, and interpretation

The plots for all three signals (desired trajectory, actual trajectory with D calculated with error term, actual trajectory with D calculate with setpoint weighting) are included. The first plot is for signal 1 and the second plot is for signal 2.

Sinusoidal input: If the D part is implemented using the error term, the paddle trajectory cannot follow the desired trajectory well when the amplitude reaches around 12 degrees.
Steps input: If the D part is implemented using the error term, the paddle cannot jump to -20 degrees when the desired position changes from 20 degrees to -20 degrees.
Interpretation: If we are using the error term for the D part, the error will be so big when the sinusoidal input is rising or falling (the slope is high), so the PID output reaches the motor's saturation and it cannot provide enough torque to move the paddle to its desired position. The same thing happens when there is a step input, when the step is so big that the output of the PID control reaches the motor's saturation, the motor cannot provide enough torque for the paddle to move as the desired trajectory.