

Self-Balancing Bot

Laboratory 4

Kristina Ćosić, Stefan Urach

05.02.2021

Content

1 1

2 2

# Model development

The general goal of the exercise is to use the skills gained from the previous laboratories to model the complex system of the self-balancing bot, which can be interpreted as an inverted pendulum.

## System equations

The first step of the development process is the derivation of the system equations with the free body diagram. The free body diagram of the bot can be seen as an inverted pendulum on a cart. The pendulum and the cart have one degree of freedom.

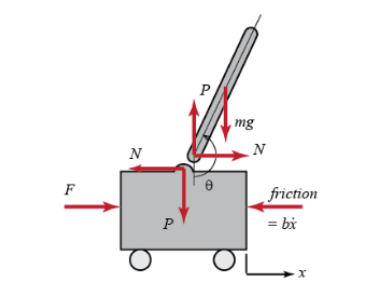


Figure - layout of the system

Table – system model items

|  |  |
| --- | --- |
| cart position | *b*: coefficient of cart friction |
| : cart velocity | *l*: length to pendulum centre of mass |
| : cart acceleration | *J*: moment of inertia of the pendulum |
| pendulum position | *F*: external force applied (motor) |
| : angular velocity | *N*: interaction force between cart and pendulum in x direction |
| : angular acceleration | *P*: interaction force between cart and pendulum in y direction |
| : mass of pendulum | *g*: gravitational constant |
| : mass of cart |  |

The equations are established using newton’s laws.

Combining the third and fourth equation into the second one results in the mathematical description of the angle behaviour.

Using the third equation in the first establishes the equation for the position behaviour.

With these equations and the system parameters given by the hardware, a model is implemented in SimuLink.

Table - parameters derived from hardware

|  |  |
| --- | --- |
| Parameter | Value |
| Pendulum mass (m) | 0,7016 kg |
| Chart mass (M) | 0,1756 kg |
| Gravitational constant (g) | 9,81 m/s2 |
| Friction coefficient (b) | 0,1 kg/s |
| Length to pendulum centre (l) | 0,11 m |
| Bot width (W) | 0,145 m |
| Inertia moment (J) | 0,001 kgm2 |

## MatLab Implementation

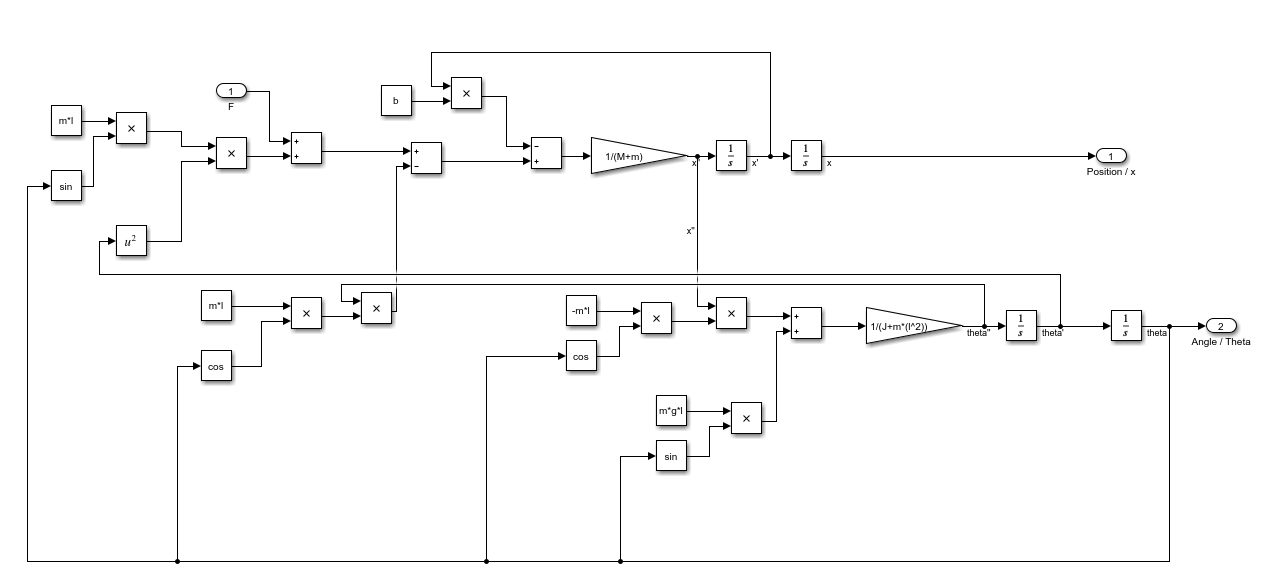


Figure - system equation as SimuLink model

# System analysis and control

## Open loop analysis

In this step the system responses are analysed. First the initial conditions with no inputs are tested.

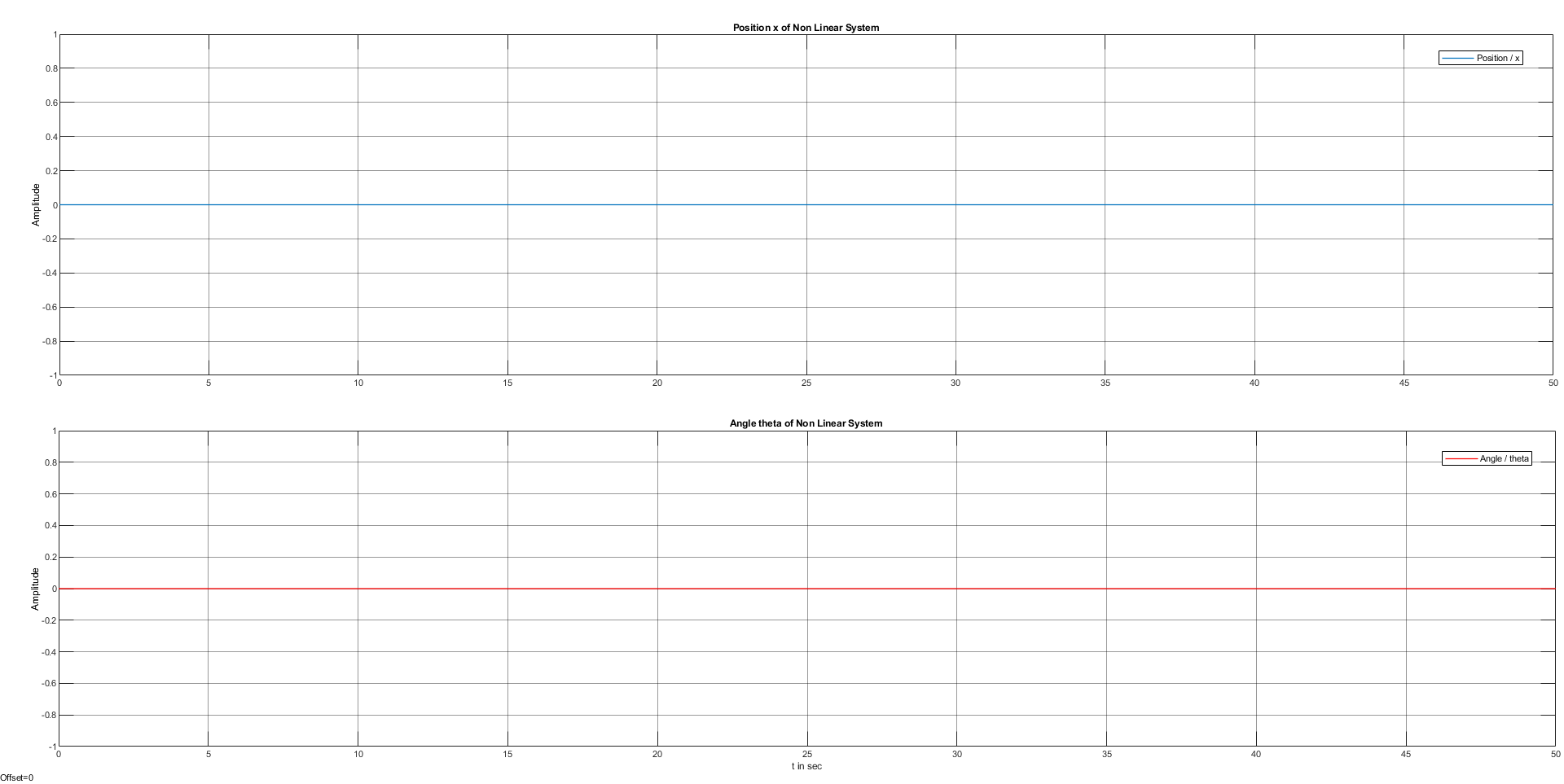


Figure - initial conditions

Figure 3 proves the systems stability, since they are both have a constant value of zero, when no force is applied.

Now the value of pendulum position should be initialised with 5.

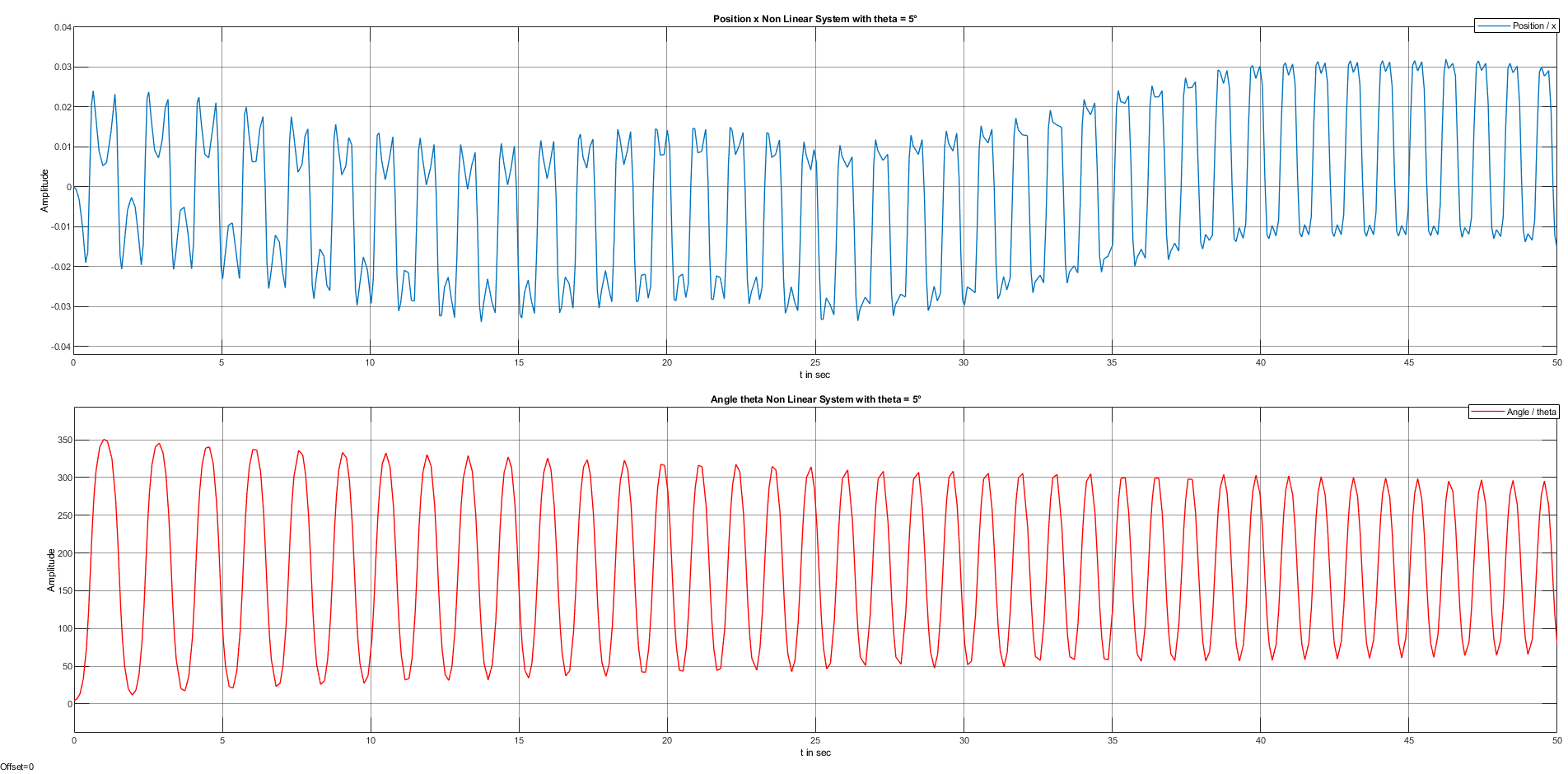


Figure - system resoponse

A diversion from the initial zero value, would suggest an uneven platform for the system, which the system tries to balance out. Since the values oscillate, the system is clearly no longer stable.

## Proportional control feedback loop

To analyse the impact of a proportional control on the system, a gain block is added and a closed feedback loop is implemented into the model.

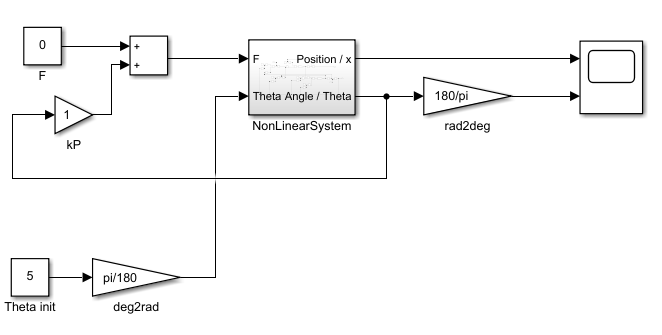


Figure - Non-linear System with P-Control

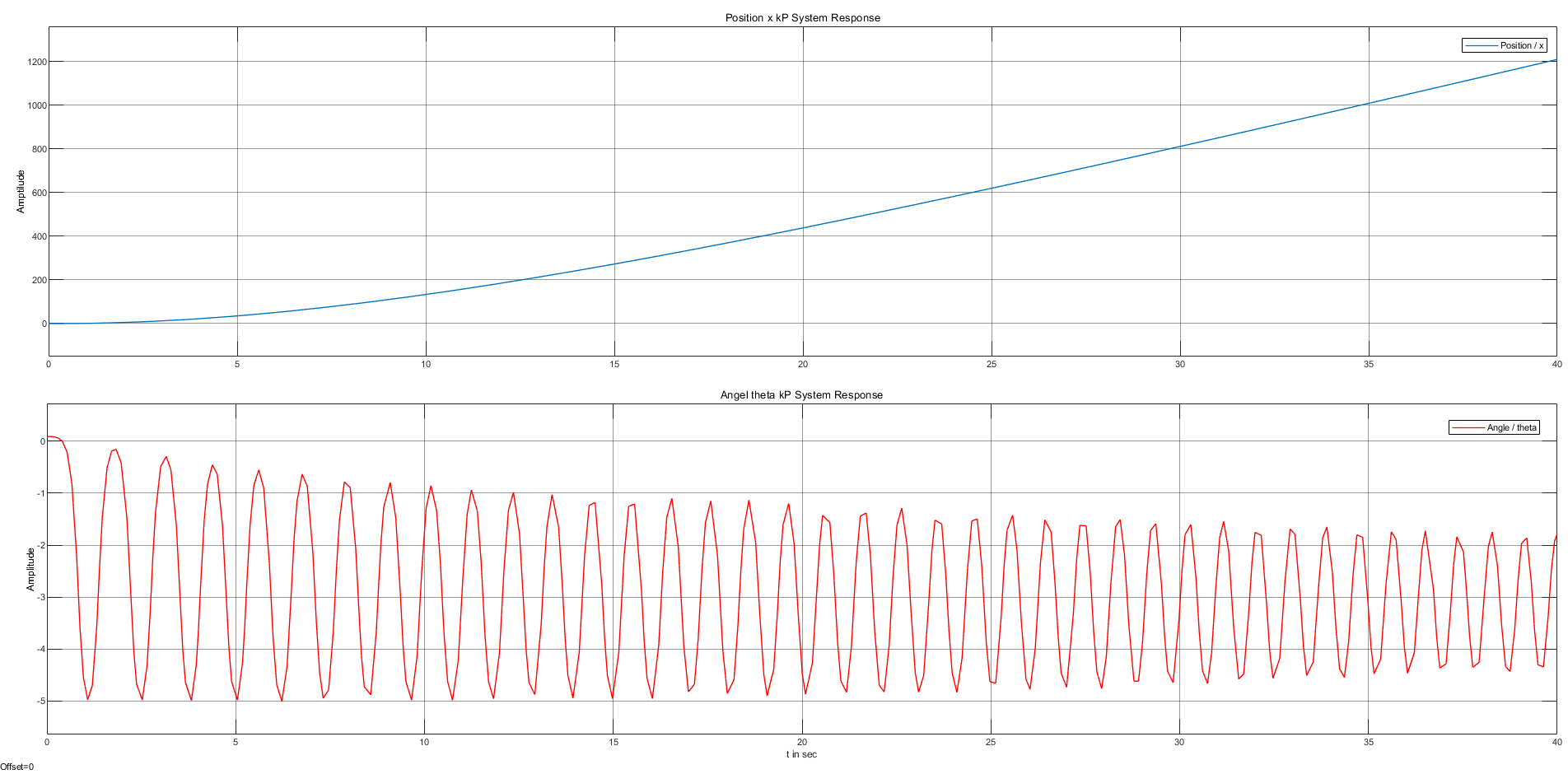


Figure - Simulation with kp=1

For a small kP value the system stays unstable, which is represented by the oscilation of the angle and the exponential growth of the position value.

Table – variation of kP

|  |  |  |  |
| --- | --- | --- | --- |
| kP | Oscillation period [s] | Stable time [s] | Stable |
| 1 | 2.982 | 0.825 | No |
| 3 | 1.146 | 0.883 | No |
| 5 | 1.021 | 0.947 | No |
| 8 | 0.786 | 1.501 | No |
| 10 | 0.732 | 7.463 | No |
| 12 | 0.683 | 24.747 | No |
| 15 | 0.591 | 57.745 | No |

Just the proportional constant stabilises the system only for a certain period, proving it as insufficient for this system.

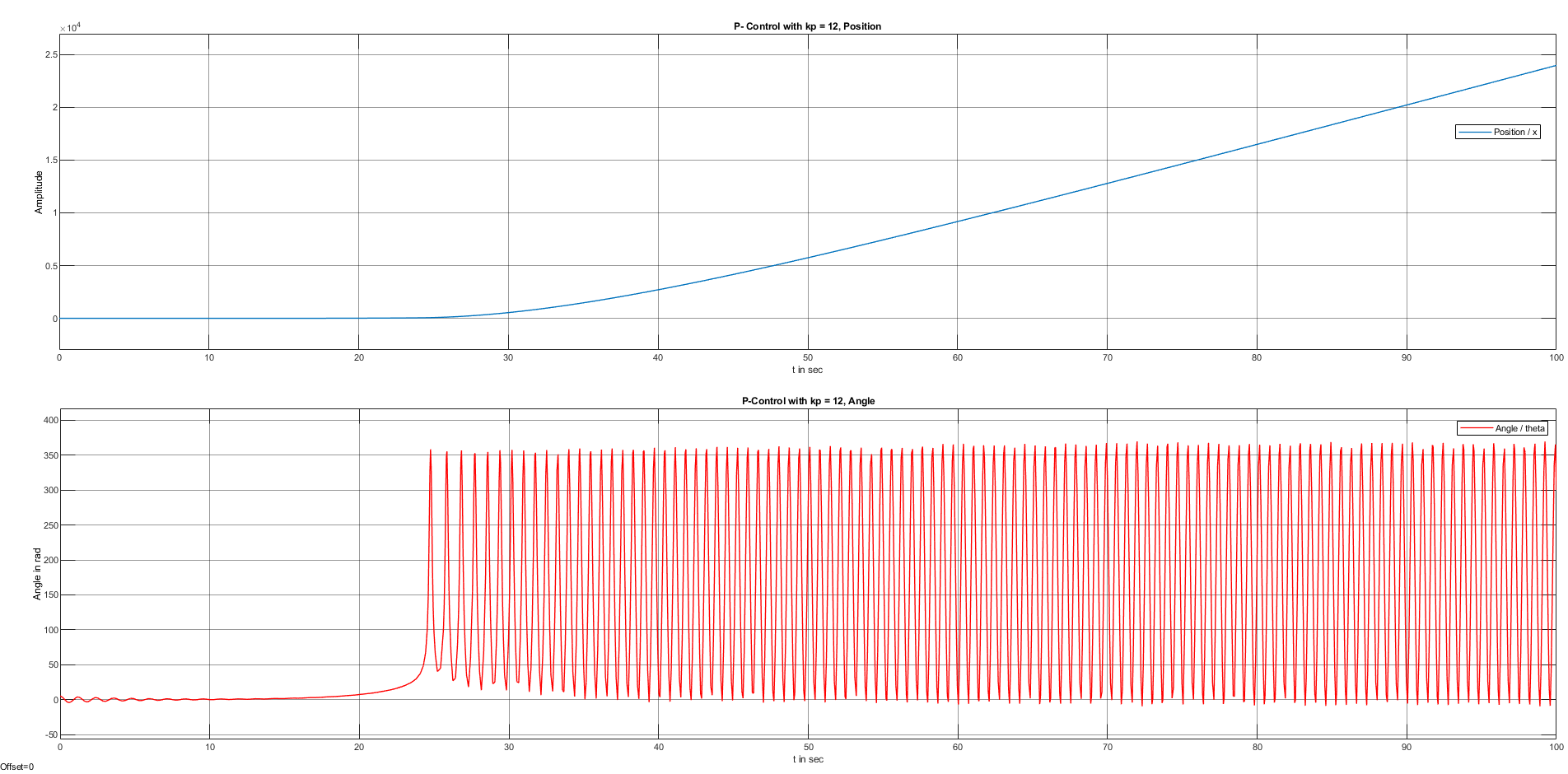


Figure - system with kP=12

# Linearization

The restriction to linear time invariant systems makes it necessary to linearize the system equations. The linearization process encompasses the elimination of sine and cosine operators in the equations. This is made possible by the fact that the range of capabilities of the system is restricted, since the operation radius of the pendulum is well under 360°. The system only needs a few degrees to left and right of the centre.

## Transfer Function Derivation

To linearize the equations in question, certain assumptions must be made.

For a small degree range, cosine stays relatively close to one and sine stays close to its relative value, which means that they can be replaced with the specific value. As for the angle derivative, since the angle itself will be fairly small and as such the derivative will be too, the value can be assumed to be zero.

## SimuLink implementation

Now the linearized version of the model is put into the SimuLink model and compared to the previous, non-linear version.

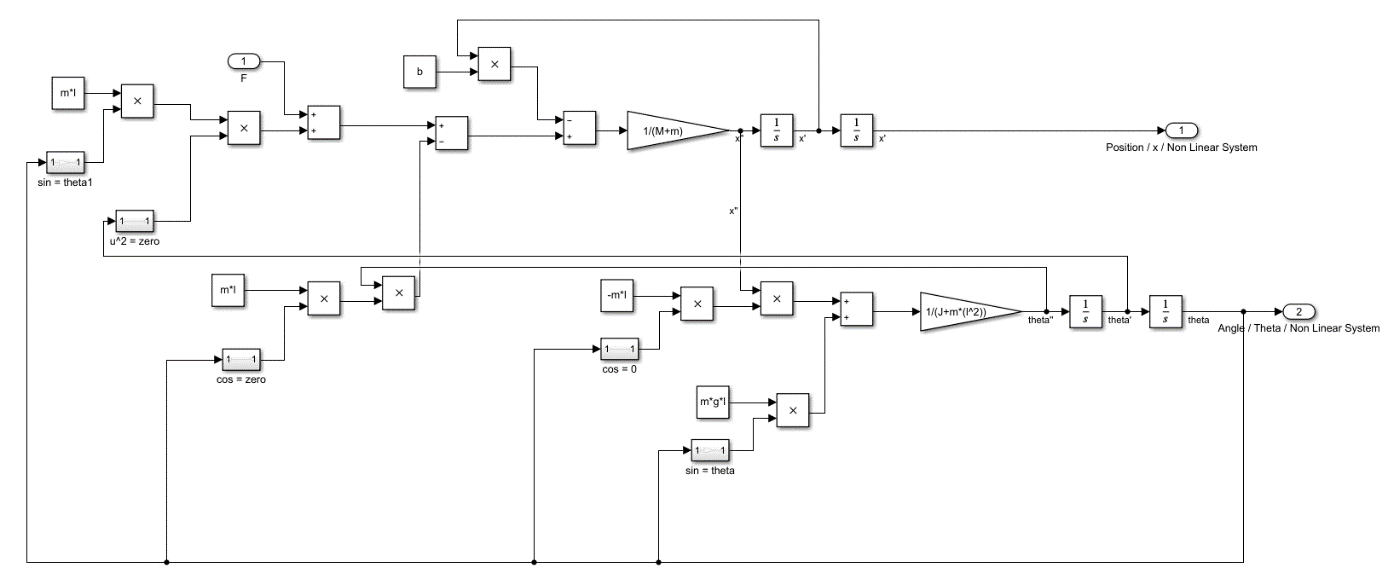


Figure - linearized model

# PID Control

# Discretization of the linear model

# Resources

## MatLab 2020a

* SimuLink

## Arduino

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung05.02.2021,

Kristina Ćosić Stefan Urach