On Lecture 3 we talked about height control of a small quad-copter. Here is a summary of that.

Process model for the drone

Denote the height of the drone by $y(t)$.
Let $u(t)$ be the upward force generated by the propellers. One can control $u(t)$ with the remote controller.
The drone's vertical acceleration $\ddot{y}(t)$ satisfies (Newton's law)

$$m\ddot{y} = u(t) - mg.$$

We can rewrite this as

$$\ddot{y}(t) = K_p u_1(t),$$

where $K_p = 1/m$ and $u_1(t) = u(t) - mg$. (On the lecture I used the variable name v, but that looks like the variable name for a velocity, so I prefer u_1). After Laplace transformation we get $s^2 Y(s) = K_p U_1(s)$ (assuming zero initial conditions). So the uncontrolled process, the open loop system, is

$$Y(s) = G_p(s) U_1(s), \quad \text{with } G_p(s) = \frac{K_p}{s^2}.$$

P-control

Let's assume we want the drone to reach a certain reference height r. We hence want the height error $e(t) = r(t) - y(t)$ to become zero. If we control u_1 using a simple proportional controller based on the height error we have $u_1(t) = Ke(t)$, i.e. using Laplace transforms

$$U_1(s) = G_r E(s), \quad \text{with } G_r(s) = K.$$

The standard formula for the closed loop system is (we did this calculation on Lecture 2. It will be useful to memorize it...)

$$Y(s) = \frac{G_p(s) G_r(s)}{1 + G_p(s) G_r(s)} R(s)$$

This gives the closed loop system

$$Y(s) = \frac{K_p K}{s^2 + K_p K} R(s).$$

If we calculate the closed loop poles (solve $s^2 + K_p K = 0$) we see that

$$s = \pm i\sqrt{K_p K}$$

This means that the closed looop poles lie on the imaginary axis, whatever value of the controller gain K we chose. The P-controller fails to stabilize the system. We need a more advanced control algorithm.
If one tries the P-controller in practice and does a step response (reference value r changes from 0 to 1) the height will oscillate and $y(t)$ does not stabilize. This is because the closed loop system with this controller is on the border of instability (the poles are on the stability boundary).

PD-control

If we add a derivative part to the controller we have $u_1(t) = Ke + K_D\dot{e}$. This means that we look also on the derivative of the error, i.e. the controller takes the velocity of the drone into account.

$$U_1(s) = G_r(s)E(s), \quad \text{with } G_r(s) = K + K_D s.$$

With the PD-controller we now get

$$Y(s) = \frac{G_p(s)G_r(s)}{1 + G_p(s)G_r(s)} R(s) = \frac{K_p(K + K_D s)}{s^2 + K_p K_D s + K_p K} R(s).$$

With $K > 0, K_D > 0$ the poles, given by $s^2 + K_p K_D s + K_p K = 0$, are in the left half plane and the closed loop system is stable. After a step change in reference value r the height stabilizes at the new level. It can be seen that K_D introduces damping (similar to friction in an oscillating "mass connected to spring" system).