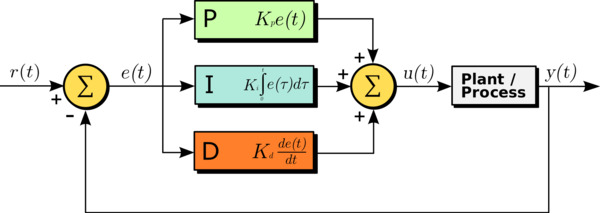
Pan/Tilt Tracking Code

The Code runs 4 processes simultaneously, ObjectCenter, which calculates the position of the object in the camera. The output is sent to PID. The PID portion of the code calculates the new angle for the motors to go to, which set\_servos() function reads and sends forward. The servos themselves aren’t run with degrees however, but with microseconds, which setServoAngle() function translates for ease of use.

Moving of the servos is done one at a time. First comes panning which is followed by tilting.

The Code runs a PID, Proportional-Integral-Derivative, for error term calculation to compensate for errors. PID is a common feedback loop in automation, and its output is used as input for other processes.

PID itself is a simple loop, diagrammed here:

It can also be written as an equation:

The P, I and D in PID are as follows:

**P (proportional):** If the *current* error is large, the output will be proportionally large to cause a significant correction.

**I (integral):** *Historical* values of the error are integrated over time. Less significant corrections are made to reduce the error. If the error is eliminated, this term won’t grow.

**D (derivative):** This term anticipates the *future*. In effect, it is a dampening method. If either P or I will cause a value to overshoot (i.e. a servo was turned past an object or a steering wheel was turned too far), D will dampen the effect before it gets to the output.

Within the ObjectCenter() process the robot’s main function is located. It detects the position of the object relative to robot as well as the aforementioned “in camera” calculation. If the robot is in predefined position relative to target, the rest of the functions of the device fires up.