

PIDSubsystems for built-in PID control

If a mechanism uses a sensor for feedback then most often a PID controller will be used to control the motor speed or position. Examples of subsystems that might use PID control are: elevators with potentiometers to track the height, shooters with encoders to measure the speed, wrists with potentiometers to measure the joint angle, etc.

There is a `PIDController` class built into WPILib, but to simplify its use for command based programs there is a `PIDSubsystem`. A `PIDSubsystem` is a normal subsystem with the `PIDController` built in and exposes the required methods for operation.

A PIDSubsystem to control the angle of a wrist joint

In this example you can see the basic elements of a `PIDSubsystem` for the wrist joint:

Java

```
package org.usfirst.frc.team1.robot.subsystems;
import edu.wpi.first.wpilibj.*;
import edu.wpi.first.wpilibj.command.PIDSubsystem;
import org.usfirst.frc.team1.robot.RobotMap;

public class Wrist extends PIDSubsystem { // This system extends PIDSubsystem

    Victor motor = RobotMap.wristMotor;
    AnalogInput pot = RobotMap.wristPot();

    public Wrist() {
        super("Wrist", 2.0, 0.0, 0.0); // The constructor passes a name for the
        subsystem and the P, I and D constants that are used when computing the motor output
        setAbsoluteTolerance(0.05);
        getPIDController().setContinuous(false);
    }
}
```

PIDSubsystems for built-in PID control

```
public void initDefaultCommand() {  
    }  
  
    protected double returnPIDInput() {  
        return pot.getAverageVoltage(); // returns the sensor value that is providing  
the feedback for the system  
    }  
  
    protected void usePIDOutput(double output) {  
        motor.pidWrite(output); // this is where the computed output value from the  
PIDController is applied to the motor  
    }  
}
```