

A B C D E F G H I L M N O P R S T U V W X Y Z

A**A** - Static variable in interface `com._604robotics.utils.XboxController.Button`**ACCELEROMETER** - Static variable in interface `com._604robotics.robot2012.configuration.PortConfiguration.Sensors`**ACCELEROMETER_DRIVE_POWER** - Static variable in interface `com._604robotics.robot2012.configuration.ActuatorConfiguration`**ACCELEROMETER_SENSITIVITY** - Static variable in interface `com._604robotics.robot2012.configuration.SensorConfiguration`**ACCELEROMETER_UPPER_RADIANS** - Static variable in interface `com._604robotics.robot2012.configuration.SensorConfiguration`**ActuatorConfiguration** - Interface in `com._604robotics.robot2012.configuration`

Actuator polarity and power configuration.

ActuatorConfiguration.ELEVATOR - Interface in `com._604robotics.robot2012.configuration`**ActuatorConfiguration.ELEVATOR.DEADBAND** - Interface in `com._604robotics.robot2012.configuration`**ActuatorConfiguration.ELEVATOR.TOLERANCE** - Interface in `com._604robotics.robot2012.configuration`**ActuatorConfiguration.RING_LIGHT** - Interface in `com._604robotics.robot2012.configuration`**ActuatorConfiguration.SOLENOID_HOPPER** - Interface in `com._604robotics.robot2012.configuration`**ActuatorConfiguration.SOLENOID_PICKUP** - Interface in `com._604robotics.robot2012.configuration`**ActuatorConfiguration.SOLENOID_SHIFTER** - Interface in `com._604robotics.robot2012.configuration`**ActuatorConfiguration.SOLENOID_SHOOTER** - Interface in `com._604robotics.robot2012.configuration`**ActuatorConfiguration.TURRET_POSITION** - Interface in `com._604robotics.robot2012.configuration`**AIM_AND_SHOOT** - Static variable in interface `com._604robotics.robot2012.configuration.ButtonConfiguration.Manipulator`**aimAndShoot()** - Method in class `com._604robotics.robot2012.Robot2012Orange`

Aim at backboard, shoot.

AIMED - Static variable in interface `com._604robotics.robot2012.machine.TurretMachine.TurretState`**Aiming** - Class in `com._604robotics.robot2012.aiming`

Utility class for various aiming functions and such.

Aiming() - Constructor for class `com._604robotics.robot2012.aiming.Aiming`**angle** - Variable in class `com._604robotics.robot2012.physics.BallFireInfo`**angle** - Variable in class `com._604robotics.robot2012.vision.Target`

This is the angle of the target, relative to the camera.

angle_uncertainty - Variable in class `com._604robotics.robot2012.vision.Target`

This is the uncertainty of the angle of the target.

angleDeg - Variable in class `com._604robotics.robot2012.physics.ShooterAnglePick`**angleRad** - Variable in class `com._604robotics.robot2012.physics.ShooterAnglePick`**angleSlope** - Variable in class `com._604robotics.robot2012.physics.ShooterAnglePick`**AUTO_BALANCE** - Static variable in interface `com._604robotics.robot2012.configuration.ButtonConfiguration.Driver`**autonomous()** - Method in class `com._604robotics.robot2012.Robot2012Orange`

Automated drive for autonomous mode.

AutonomousConfiguration - Interface in `com._604robotics.robot2012.configuration`

Autonomous mode configuration.

B**B** - Static variable in interface `com._604robotics.utils.XboxController.Button`**Back** - Static variable in interface `com._604robotics.utils.XboxController.Button`**BACKWARD_DISTANCE** - Static variable in interface `com._604robotics.robot2012.configuration.AutonomousConfiguration`**BACKWARD_DISTANCE_SIDES** - Static variable in interface `com._604robotics.robot2012.configuration.AutonomousConfiguration`**BACKWARD_DRIVE_POWER** - Static variable in interface `com._604robotics.robot2012.configuration.AutonomousConfiguration`**Balancing** - Class in `com._604robotics.robot2012.balancing`

Utility class for automated balancing assistance.

Balancing() - Constructor for class `com._604robotics.robot2012.balancing.Balancing`**BallFireInfo** - Class in `com._604robotics.robot2012.physics`

Class representing info for firing a ball.

BallFireInfo(ShooterAnglePick, double, double) - Constructor for class `com._604robotics.robot2012.physics.BallFireInfo`

Initializes a new BallFireInfo.

begin() - Method in interface `com._604robotics.robot2012.camera.CameraInterface`

Launches the CameraInterface.

begin() - Method in class `com._604robotics.robot2012.camera.RemoteCameraTCP`
 Initializes communication.

betterVersionOfGetFiringVelocity(double, double, double) - Method in class `com._604robotics.robot2012.physics.Physics`
 This function determines the firing velocities (and time) for a given distance (horizontally, and vertically) and a vertical velocity at which the ball should enter the hoop.

betterVersionOfGetFiringVelocity(double, double) - Method in class `com._604robotics.robot2012.physics.Physics`
 This function guesses a good vertical velocity to enter the hoop, then determines the firing velocities (and time) for a given distance (horizontally, and vertically).

ButtonConfiguration - Interface in `com._604robotics.robot2012.configuration`
 Button configuration.

ButtonConfiguration.Driver - Interface in `com._604robotics.robot2012.configuration`

ButtonConfiguration.Manipulator - Interface in `com._604robotics.robot2012.configuration`

ButtonConfiguration.Manipulator.Elevator - Interface in `com._604robotics.robot2012.configuration`

C

calculate() - Method in class `com._604robotics.utils.LinearController`
 Function that performs the output calculation.

CameraInterface - Interface in `com._604robotics.robot2012.camera`
 Represents a method for obtaining processed vision data from the camera.

`com._604robotics.robot2012` - package `com._604robotics.robot2012`

`com._604robotics.robot2012.aiming` - package `com._604robotics.robot2012.aiming`

`com._604robotics.robot2012.autonomous` - package `com._604robotics.robot2012.autonomous`

`com._604robotics.robot2012.balancing` - package `com._604robotics.robot2012.balancing`

`com._604robotics.robot2012.camera` - package `com._604robotics.robot2012.camera`

`com._604robotics.robot2012.configuration` - package `com._604robotics.robot2012.configuration`

`com._604robotics.robot2012.machine` - package `com._604robotics.robot2012.machine`

`com._604robotics.robot2012.physics` - package `com._604robotics.robot2012.physics`

`com._604robotics.robot2012.rotation` - package `com._604robotics.robot2012.rotation`

`com._604robotics.robot2012.vision` - package `com._604robotics.robot2012.vision`

`com._604robotics.utils` - package `com._604robotics.utils`

CompensatingGyro - Class in `com._604robotics.utils`
 Gyro with manual compensation-setting support.

CompensatingGyro(int) - Constructor for class `com._604robotics.utils.CompensatingGyro`
 Initializes a new CompensatingGyro on the specified PWM port.

CompensatingGyro(int, int) - Constructor for class `com._604robotics.utils.CompensatingGyro`
 Initializes a new CompensatingGyro on the specified PWM port on the specified module port.

CompensatingGyro(AnalogChannel) - Constructor for class `com._604robotics.utils.CompensatingGyro`
 Initializes a new CompensatingGyro on the specified AnalogChannel.

COMPRESSOR - Static variable in interface `com._604robotics.robot2012.configuration.PortConfiguration.Pneumatics`

ConvertingPIDController - Class in `com._604robotics.utils`
 An extender of a PIDController that converts between units when getting and setting a setpoint.

ConvertingPIDController(double, double, double, PIDSource, PIDOutput) - Constructor for class `com._604robotics.utils.ConvertingPIDController`
 Allocate a PID object with the given constants for P, I, D, using a 50ms period.

ConvertingPIDController(double, double, double, PIDSource, PIDOutput, double) - Constructor for class `com._604robotics.utils.ConvertingPIDController`
 Allocate a PID object with the given constants for P, I, D

crank(int) - Method in class `com._604robotics.robot2012.machine.ElevatorMachine`

crank(int) - Method in class `com._604robotics.robot2012.machine.PickupMachine`

crank(int) - Method in class `com._604robotics.robot2012.machine.ShooterMachine`

crank(int) - Method in interface `com._604robotics.robot2012.machine.StrangeMachine`
 Causes the Machine to strive for the target state.

crank(int) - Method in class `com._604robotics.robot2012.machine.TurretMachine`

D

D - Variable in class `com._604robotics.utils.UpDownPIDController.Gains`

deadband(double, double, double, double) - Static method in class `com._604robotics.robot2012.Robot2012Orange`
 If a value is within a range, set it to a specific value.

DeadbandedSource - Class in `com._604robotics.utils`
 Implements a PIDSource, wrapping around another PIDSource, with a deadband range.

DeadbandedSource(PIDSource) - Constructor for class `com._604robotics.utils.DeadbandedSource`
 Initializes a new DeadbandedSource.

defaultAiming - Static variable in class `com._604robotics.robot2012.aiming.Aiming`

disable() - Method in class `com._604robotics.utils.VelocityController`
 Disables the VelocityController.

disabled() - Method in class `com._604robotics.robot2012.Robot2012Orange`
 The robot is disabled.

DOWN - Static variable in interface `com._604robotics.robot2012.configuration.ButtonConfiguration.Manipulator.Elevator`

Down - Static variable in interface `com._604robotics.utils.XboxController.Button.DPad`

DPAD - Static variable in interface `com._604robotics.utils.XboxController.Stick`

DRIVE - Static variable in interface `com._604robotics.robot2012.configuration.PortConfiguration.Controllers`

DualVictor - Class in `com._604robotics.utils`

Control two Victors like they're one.

DualVictor(int, int) - Constructor for class `com._604robotics.utils.DualVictor`

Initialize a DualVictor with a left and a right PWM port.

DualVictor(int, int, int, int) - Constructor for class `com._604robotics.utils.DualVictor`

Initializes a DualVictor with left and right slot and PWM port.

DualVictor(Victor, Victor) - Constructor for class `com._604robotics.utils.DualVictor`

Initializes a DualVictor with left and right slot and PWM port.

DummyRotationProvider - Class in `com._604robotics.robot2012.rotation`

Dummy implementor of a RotationProvider, for testing purposes.

DummyRotationProvider(PIDController) - Constructor for class `com._604robotics.robot2012.rotation.DummyRotationProvider`

Initializes a new DummyRotationProvider, giving it control over the specified PIDController.

E

`edu.wpi.first.wpilibj` - package `edu.wpi.first.wpilibj`

The WPI Robotics library (WPILibJ) is a set of Java classes that interfaces to the hardware in the FRC control system and your robot.

EitherTrigger - Static variable in interface `com._604robotics.utils.XboxController.Button`

ELEVATOR_A - Static variable in interface `com._604robotics.robot2012.configuration.PortConfiguration.Encoders`

ELEVATOR_B - Static variable in interface `com._604robotics.robot2012.configuration.PortConfiguration.Encoders`

ELEVATOR_LEFT - Static variable in interface `com._604robotics.robot2012.configuration.PortConfiguration.Motors`

ELEVATOR_LIMIT_SWITCH - Static variable in interface `com._604robotics.robot2012.configuration.PortConfiguration.Sensors`

ELEVATOR_POWER_MAX - Static variable in interface `com._604robotics.robot2012.configuration.ActuatorConfiguration`

ELEVATOR_POWER_MIN - Static variable in interface `com._604robotics.robot2012.configuration.ActuatorConfiguration`

ELEVATOR_RIGHT - Static variable in interface `com._604robotics.robot2012.configuration.PortConfiguration.Motors`

ElevatorMachine - Class in `com._604robotics.robot2012.machine`

Machine to control the elevator.

ElevatorMachine(PIDController, Encoder) - Constructor for class `com._604robotics.robot2012.machine.ElevatorMachine`

Initializes a new ElevatorMachine.

ElevatorMachine.ElevatorState - Interface in `com._604robotics.robot2012.machine`

Various possible states the elevator can be in.

enable() - Method in class `com._604robotics.utils.VelocityController`

Enables the VelocityController.

EncoderOffset - Class in `com._604robotics.utils`

Encoder extender that return the value of Encoder.get() when pidGet is called.

EncoderOffset(int, int, int, int, boolean) - Constructor for class `com._604robotics.utils.EncoderOffset`

Encoder constructor.

EncoderOffset(int, int, int, int) - Constructor for class `com._604robotics.utils.EncoderOffset`

Encoder constructor.

EncoderOffset(int, int, int, int, boolean, CounterBase.EncodingType) - Constructor for class `com._604robotics.utils.EncoderOffset`

Encoder constructor.

EncoderOffset(int, int, int, int, int, int, boolean) - Constructor for class `com._604robotics.utils.EncoderOffset`

Encoder constructor.

EncoderOffset(int, int, int, int, int, int) - Constructor for class `com._604robotics.utils.EncoderOffset`

Encoder constructor.

EncoderOffset(int, int, boolean) - Constructor for class `com._604robotics.utils.EncoderOffset`

Encoder constructor.

EncoderOffset(int, int) - Constructor for class `com._604robotics.utils.EncoderOffset`

Encoder constructor.

EncoderOffset(int, int, boolean, CounterBase.EncodingType) - Constructor for class `com._604robotics.utils.EncoderOffset`

Encoder constructor.

EncoderOffset(int, int, int, boolean) - Constructor for class `com._604robotics.utils.EncoderOffset`

Encoder constructor.

EncoderOffset(int, int, int) - Constructor for class `com._604robotics.utils.EncoderOffset`

Encoder constructor.

EncoderOffset(DigitalSource, DigitalSource, boolean) - Constructor for class `com._604robotics.utils.EncoderOffset`

Encoder constructor.

EncoderOffset(DigitalSource, DigitalSource) - Constructor for class `com._604robotics.utils.EncoderOffset`

Encoder constructor.

EncoderOffset(DigitalSource, DigitalSource, boolean, CounterBase.EncodingType) - Constructor for class `com._604robotics.utils.EncoderOffset`

Encoder constructor.

EncoderOffset(DigitalSource, DigitalSource, DigitalSource, boolean) - Constructor for class `com._604robotics.utils.EncoderOffset`

Encoder constructor.

EncoderOffset(DigitalSource, DigitalSource, DigitalSource) - Constructor for class `com._604robotics.utils.EncoderOffset`

Encoder constructor.

EncoderPIDSource - Class in `com._604robotics.utils`

Encoder extender that return the value of Encoder.get() when pidGet is called.

EncoderPIDSource(int, int, int, int, boolean) - Constructor for class `com._604robotics.utils.EncoderPIDSource`

Encoder constructor.

EncoderPIDSource(int, int, int, int, int) - Constructor for class `com._604robotics.utils.EncoderPIDSource`

Encoder constructor.

EncoderPIDSource(int, int, int, int, boolean, CounterBase.EncodingType) - Constructor for class `com._604robotics.utils.EncoderPIDSource`

Encoder constructor.
EncoderPIDSource(int, int, int, int, int, boolean) - Constructor for class com._604robotics.utils.EncoderPIDSource
 Encoder constructor.
EncoderPIDSource(int, int, int, int, int, int) - Constructor for class com._604robotics.utils.EncoderPIDSource
 Encoder constructor.
EncoderPIDSource(int, int, boolean) - Constructor for class com._604robotics.utils.EncoderPIDSource
 Encoder constructor.
EncoderPIDSource(int, int) - Constructor for class com._604robotics.utils.EncoderPIDSource
 Encoder constructor.
EncoderPIDSource(int, int, boolean, CounterBase.EncodingType) - Constructor for class com._604robotics.utils.EncoderPIDSource
 Encoder constructor.
EncoderPIDSource(int, int, int, boolean) - Constructor for class com._604robotics.utils.EncoderPIDSource
 Encoder constructor.
EncoderPIDSource(int, int, int) - Constructor for class com._604robotics.utils.EncoderPIDSource
 Encoder constructor.
EncoderPIDSource(DigitalSource, DigitalSource, boolean) - Constructor for class com._604robotics.utils.EncoderPIDSource
 Encoder constructor.
EncoderPIDSource(DigitalSource, DigitalSource) - Constructor for class com._604robotics.utils.EncoderPIDSource
 Encoder constructor.
EncoderPIDSource(DigitalSource, DigitalSource, boolean, CounterBase.EncodingType) - Constructor for class com._604robotics.utils.EncoderPIDSource
 Encoder constructor.
EncoderPIDSource(DigitalSource, DigitalSource, DigitalSource, boolean) - Constructor for class com._604robotics.utils.EncoderPIDSource
 Encoder constructor.
EncoderPIDSource(DigitalSource, DigitalSource, DigitalSource) - Constructor for class com._604robotics.utils.EncoderPIDSource
 Encoder constructor.
end() - Method in interface com._604robotics.robot2012.camera.CameraInterface
 Disables the CameraInterface.
end() - Method in class com._604robotics.robot2012.camera.RemoteCameraTCP
 Ends communication.

F

FORWARD - Static variable in interface com._604robotics.robot2012.configuration.ActuatorConfiguration.TURRET_POSITION
FORWARD - Static variable in interface com._604robotics.robot2012.configuration.ButtonConfiguration.Manipulator.Elevator
FORWARD - Static variable in interface com._604robotics.robot2012.configuration.PortConfiguration.Pneumatics.HOPPER_SOLENOID
FORWARD - Static variable in interface com._604robotics.robot2012.machine.TurretMachine.TurretState
FORWARD_DISTANCE - Static variable in interface com._604robotics.robot2012.configuration.AutonomousConfiguration
FORWARD_DRIVE_POWER - Static variable in interface com._604robotics.robot2012.configuration.AutonomousConfiguration
 frc.vision - package frc.vision

G

get() - Method in class com._604robotics.utils.DualVictor
 Checks the current power the Vectors are set to.
getActualVelocity() - Method in class com._604robotics.utils.VelocityController
 Gets the actual, current velocity.
getAnalogChannel() - Method in class edu.wpi.first.wpilibj.GyroHax
 Gets the raw AnalogChannel.
getAngle() - Method in class com._604robotics.utils.Gyro360
 Gets the angle of the gyro, constrained to 360 degrees.
getAngleAndRelXYZOfTarget(double, double, double, double, double, double, double, double) - Method in class com._604robotics.robot2012.aiming.Aiming
 Get the angle from the targets, and the relative distances of the corners of the target as perceived by the camera.
getAngleOfTarget(double, double, double, double, double, double, double, double, double, double) - Method in class com._604robotics.robot2012.aiming.Aiming
 This function gets the direction the target is facing, relative to the camera.
getAxis(int) - Method in class com._604robotics.utils.XboxController
 Get the value of the specified axis.
GetBallFiringInfo(double, double, double, double, double) - Method in class com._604robotics.robot2012.physics.Physics
 This function will determine how to fire the ball if the shooter only has 2 vertical angles.
getButton(int) - Method in class com._604robotics.utils.XboxController
 Get whether or not the specified button is currently pressed.
getDownGains() - Method in class com._604robotics.utils.UpDownPIDController
 Gets the Gains for going down.
getJoystick() - Method in class com._604robotics.utils.XboxController
 Gets the underlying Joystick object.
getRaw() - Method in class com._604robotics.utils.EncoderOffset
getRealSetpoint() - Method in class com._604robotics.utils.ConvertingPIDController
 Gets the "real" setpoint of the PIDController.
getRecordedTime() - Method in interface com._604robotics.robot2012.camera.CameraInterface
 Gets the estimated time since the last packet was received.
getRecordedTime() - Method in class com._604robotics.robot2012.camera.RemoteCameraTCP
 Records the time elapsed between reception of data packets from camera.
getRelXYZOfTarget(double, double, double, double) - Method in class com._604robotics.robot2012.aiming.Aiming
 Remember that this requires the camera to be "perfectly" flat, and the targets to be "perfectly" vertical.
getRelXYZOfTarget(Target) - Method in class com._604robotics.robot2012.aiming.Aiming
getSetpoint() - Method in class com._604robotics.utils.ConvertingPIDController
getSpeedforBalance(double) - Static method in class com._604robotics.robot2012.balancing.Balancing

Given a specific gyro reading, returns what speed you should be going at.

getSprung() - Method in class com._604robotics.utils.DualVictor
Has the victor been sprung?

getSprung() - Method in class com._604robotics.utils.SpringableDoubleSolenoid
Has the DoubleSolenoid been sprung?

getSprung() - Method in class com._604robotics.utils.SpringableRelay
Has the Relay been sprung?

getSprung() - Method in class com._604robotics.utils.SpringableVictor
Has the victor been sprung?

getStick(int) - Method in class com._604robotics.utils.XboxController
Get whether or not there's a value reading on the stick.

getSubparFiringVelocity(double, double, double) - Method in class com._604robotics.robot2012.physics.Physics
This untested function might determine the firing velocity for a given distance (horizontally, and vertically) and the angle of the shooter.

getTarget() - Method in class com._604robotics.utils.LinearController
Gets the current target.

getTargets() - Method in interface com._604robotics.robot2012.camera.CameraInterface
Returns the most recently-obtained array of Target that represents the visible targets.

getTargets() - Method in class com._604robotics.robot2012.camera.RemoteCameraTCP
Returns the last targets acquired from the remote software.

getToggle(int) - Method in class com._604robotics.utils.XboxController
Get the toggle state of the specified button.

getUpGains() - Method in class com._604robotics.utils.UpDownPIDController
Gets the Gains for going up.

getUPS() - Method in class com._604robotics.robot2012.camera.RemoteCameraTCP
Returns the number of updates received per second.

getVelocity() - Method in class com._604robotics.utils.VelocityController
Gets the current target velocity.

getX() - Method in class com._604robotics.robot2012.vision.Point3d

getY() - Method in class com._604robotics.robot2012.vision.Point3d

getZ() - Method in class com._604robotics.robot2012.vision.Point3d

Gyro360 - Class in com._604robotics.utils
Extender class to constrain the output of a Gyro to 360 degrees, looping.

Gyro360(int) - Constructor for class com._604robotics.utils.Gyro360
Initializes a new Gyro360 on the specified PWM port.

Gyro360(int, int) - Constructor for class com._604robotics.utils.Gyro360
Initializes a new Gyro360 on the specified PWM port on the specified module port.

Gyro360(AnalogChannel) - Constructor for class com._604robotics.utils.Gyro360
Initializes a new Gyro360 on the specified AnalogChannel.

GYRO_BALANCE - Static variable in interface com._604robotics.robot2012.configuration.PortConfiguration.Sensors

GYRO_DRIFT - Static variable in interface com._604robotics.robot2012.configuration.SensorConfiguration

GYRO_HEADING - Static variable in interface com._604robotics.robot2012.configuration.PortConfiguration.Sensors

GYRO_RESET - Static variable in interface com._604robotics.robot2012.configuration.ButtonConfiguration.Driver

GyroHax - Class in edu.wpi.first.wpilibj
Extender class for the Gyro class that exposes the underlying AnalogChannel.

GyroHax(int) - Constructor for class edu.wpi.first.wpilibj.GyroHax
Initializes a new GyroHax on the specified PWM port.

GyroHax(int, int) - Constructor for class edu.wpi.first.wpilibj.GyroHax
Initializes a new GyroHax on the specified PWM port on the specified module port.

GyroHax(AnalogChannel) - Constructor for class edu.wpi.first.wpilibj.GyroHax
Initializes a new GyroHax on the specified AnalogChannel.

H

h - Variable in class frc.vision.Target

HIGH - Static variable in interface com._604robotics.robot2012.configuration.ActuatorConfiguration.ELEVATOR.DEADBAND

HIGH - Static variable in interface com._604robotics.robot2012.configuration.ActuatorConfiguration.ELEVATOR

HIGH - Static variable in interface com._604robotics.robot2012.configuration.ActuatorConfiguration.ELEVATOR.TOLERANCE

HIGH - Static variable in interface com._604robotics.robot2012.machine.ElevatorMachine.ElevatorState

HIGH_GEAR - Static variable in interface com._604robotics.robot2012.configuration.ActuatorConfiguration.SOLENOID_SHIFTER

HIGH_GEAR - Static variable in interface com._604robotics.robot2012.configuration.PortConfiguration.Pneumatics.SHIFTER_SOLENOID

HOPPER - Static variable in interface com._604robotics.robot2012.configuration.PortConfiguration.Motors

HOPPER_POWER - Static variable in interface com._604robotics.robot2012.configuration.ActuatorConfiguration

HOPPER_POWER_REVERSE - Static variable in interface com._604robotics.robot2012.configuration.ActuatorConfiguration

horizontalAngle - Variable in class com._604robotics.robot2012.physics.BallFireInfo

I

I - Variable in class com._604robotics.utils.UpDownPIDController.Gains

IN - Static variable in interface com._604robotics.robot2012.configuration.ActuatorConfiguration.SOLENOID_PICKUP

IN - Static variable in interface `com._604robotics.robot2012.configuration.PortConfiguration.Pneumatics.PICKUP_SOLENOID`

IN - Static variable in interface `com._604robotics.robot2012.machine.PickupMachine.PickupState`

isEnabled() - Method in class `com._604robotics.utils.VelocityController`

Is the VelocityController currently enabled?

isInRange(double, double, double) - Static method in class `com._604robotics.robot2012.Robot2012Orange`

Figures out if a value is within a specific range.

L

LB - Static variable in interface `com._604robotics.utils.XboxController.Button`

LEFT - Static variable in interface `com._604robotics.robot2012.configuration.ActuatorConfiguration.TURRET_POSITION`

LEFT - Static variable in interface `com._604robotics.robot2012.configuration.ButtonConfiguration.Manipulator.Elevator`

LEFT - Static variable in interface `com._604robotics.robot2012.machine.TurretMachine.TurretState`

Left - Static variable in interface `com._604robotics.utils.XboxController.Button.DPad`

LEFT_A - Static variable in interface `com._604robotics.robot2012.configuration.PortConfiguration.Encoders.Drive`

LEFT_B - Static variable in interface `com._604robotics.robot2012.configuration.PortConfiguration.Encoders.Drive`

LEFT_DRIVE - Static variable in interface `com._604robotics.robot2012.configuration.PortConfiguration.Motors`

LEFT_DRIVE_INCHES_PER_CLICK - Static variable in interface `com._604robotics.robot2012.configuration.SensorConfiguration.Encoders`

LEFT_STICK - Static variable in interface `com._604robotics.utils.XboxController.Stick`

LEFT_STICK_X - Static variable in interface `com._604robotics.utils.XboxController.Axis`

LEFT_STICK_Y - Static variable in interface `com._604robotics.utils.XboxController.Axis`

LeftStick - Static variable in interface `com._604robotics.utils.XboxController.Button`

LinearController - Class in `com._604robotics.utils`

This class implements a controller with a horizontal segment, a linear segment, and finally a coasting segment.

LinearController(PIDSource, PIDOutput, double, double, double, double) - Constructor for class `com._604robotics.utils.LinearController`

Initializes a new LinearController.

LOW - Static variable in interface `com._604robotics.robot2012.configuration.ActuatorConfiguration.ELEVATOR.DEADBAND`

LOW - Static variable in interface `com._604robotics.robot2012.configuration.ActuatorConfiguration.ELEVATOR`

LOW - Static variable in interface `com._604robotics.robot2012.configuration.ActuatorConfiguration.ELEVATOR.TOLERANCE`

LOW - Static variable in interface `com._604robotics.robot2012.machine.ElevatorMachine.ElevatorState`

LOW_GEAR - Static variable in interface `com._604robotics.robot2012.configuration.ActuatorConfiguration.SOLENOID_SHIFTER`

LOW_GEAR - Static variable in interface `com._604robotics.robot2012.configuration.PortConfiguration.Pneumatics.SHIFTER_SOLENOID`

LOWER_ANGLE - Static variable in interface `com._604robotics.robot2012.configuration.ActuatorConfiguration.SOLENOID_SHOOTER`

LOWER_ANGLE - Static variable in interface `com._604robotics.robot2012.configuration.PortConfiguration.Pneumatics.SHOOTER_SOLENOID`

LT - Static variable in interface `com._604robotics.utils.XboxController.Button`

M

MANIPULATOR - Static variable in interface `com._604robotics.robot2012.configuration.PortConfiguration.Controllers`

MEDIUM - Static variable in interface `com._604robotics.robot2012.configuration.ActuatorConfiguration.ELEVATOR`

MEDIUM - Static variable in interface `com._604robotics.robot2012.machine.ElevatorMachine.ElevatorState`

MEDIUM_LOWER - Static variable in interface `com._604robotics.robot2012.configuration.ActuatorConfiguration.ELEVATOR.DEADBAND`

MEDIUM_LOWER - Static variable in interface `com._604robotics.robot2012.configuration.ActuatorConfiguration.ELEVATOR.TOLERANCE`

MEDIUM_UPPER - Static variable in interface `com._604robotics.robot2012.configuration.ActuatorConfiguration.ELEVATOR.DEADBAND`

MEDIUM_UPPER - Static variable in interface `com._604robotics.robot2012.configuration.ActuatorConfiguration.ELEVATOR.TOLERANCE`

N

NaiveRotationProvider - Class in `com._604robotics.robot2012.rotation`

A naive implementation of a RotationProvider.

NaiveRotationProvider(PIDController, CameraInterface, Encoder) - Constructor for class `com._604robotics.robot2012.rotation.NaiveRotationProvider`

Initializes a new NaiveRotationProvider, giving it control over the specified PIDController.

O

OFF - Static variable in interface `com._604robotics.robot2012.configuration.ActuatorConfiguration.RING_LIGHT`

OKAY_TO_TURN - Static variable in interface `com._604robotics.robot2012.configuration.ActuatorConfiguration.ELEVATOR`

ON - Static variable in interface `com._604robotics.robot2012.configuration.ActuatorConfiguration.RING_LIGHT`

onTarget() - Method in class `com._604robotics.utils.LinearController`

Are we there yet?

operatorControl() - Method in class `com._604robotics.robot2012.Robot2012Orange`

Operator-controlled drive for Teleop mode.

OUT - Static variable in interface `com._604robotics.robot2012.configuration.ActuatorConfiguration.SOLENOID_PICKUP`

OUT - Static variable in interface `com._604robotics.robot2012.configuration.PortConfiguration.Pneumatics.PICKUP_SOLENOID`

OUT - Static variable in interface `com._604robotics.robot2012.machine.PickupMachine.PickupState`

P

P - Variable in class `com._604robotics.utils.UpDownPIDController.Gains`

Physics - Class in `com._604robotics.robot2012.physics`

Used for determining launch velocities of the ball.

Physics() - Constructor for class `com._604robotics.robot2012.physics.Physics`

PICKUP - Static variable in interface `com._604robotics.robot2012.configuration.ButtonConfiguration.Manipulator`

PICKUP - Static variable in interface `com._604robotics.robot2012.configuration.PortConfiguration.Motors`

PICKUP_OKAY - Static variable in interface `com._604robotics.robot2012.machine.ElevatorMachine.ElevatorState`

PICKUP_POWER - Static variable in interface `com._604robotics.robot2012.configuration.ActuatorConfiguration`

PickupMachine - Class in `com._604robotics.robot2012.machine`

Machine to control the pneumatic pickup.

PickupMachine(DoubleSolenoid) - Constructor for class `com._604robotics.robot2012.machine.PickupMachine`

Initializes a new PickupMachine.

PickupMachine.PickupState - Interface in `com._604robotics.robot2012.machine`

Possible states the pickup could be in.

PIDDriveEncoderDifference - Class in `com._604robotics.robot2012.autonomous`

This class implements a PIDSource, based on the difference of values between two encoders.

PIDDriveEncoderDifference(Encoder, Encoder) - Constructor for class `com._604robotics.robot2012.autonomous.PIDDriveEncoderDifference`

Initializes a new PIDDriveEncoderDifference, based on the given encoders.

PIDDriveEncoderOutput - Class in `com._604robotics.robot2012.autonomous`

This class implements the default PIDOutput class provided in the WPILib API.

PIDDriveEncoderOutput(RobotDrive, boolean) - Constructor for class `com._604robotics.robot2012.autonomous.PIDDriveEncoderOutput`

Initializes a new PIDDriveEncoderOutput.

PIDDriveEncoderOutput(RobotDrive) - Constructor for class `com._604robotics.robot2012.autonomous.PIDDriveEncoderOutput`

Initializes a new PIDDriveEncoderOutput.

PIDDriveGyro - Class in `com._604robotics.robot2012.autonomous`

Driving shim for the gyro-based PID-turning controller thing.

PIDDriveGyro(RobotDrive) - Constructor for class `com._604robotics.robot2012.autonomous.PIDDriveGyro`

Initializes a new PIDDriveGyro, based on the given RobotDrive.

pidGet() - Method in class `com._604robotics.robot2012.autonomous.PIDDriveEncoderDifference`

Gets the difference between the two encoder values, as an output to a PID controller.

pidGet() - Method in class `com._604robotics.utils.DeadbandedSource`

Hooks into PIDSource - gets the value to send to the PIDController.

pidGet() - Method in class `com._604robotics.utils.EncoderPIDSource`

Hooks into the PIDSource interface.

pidGet() - Method in class `com._604robotics.utils.Gyro360`

Implements the pidGet() function in the type PIDSource, allowing this class to be used as such.

pidWrite(double) - Method in class `com._604robotics.robot2012.autonomous.PIDDriveEncoderOutput`

Robot will drive with the configured power, and swerve determined by the encoder readings.

pidWrite(double) - Method in class `com._604robotics.robot2012.autonomous.PIDDriveGyro`

Writes the output from the PIDController to the RobotDrive, in the form of a turn value.

pidWrite(double) - Method in class `com._604robotics.utils.DualVictor`

Function to hook into the PIDController.

pidWrite(double) - Method in class `com._604robotics.utils.SpringableVictor`

Function to hook into the PIDController.

Point2d - Class in `com._604robotics.robot2012.aiming`

Represents a single point on the 2D plane.

Point2d(double, double) - Constructor for class `com._604robotics.robot2012.aiming.Point2d`

Initializes a new Point2d.

Point3d - Class in `com._604robotics.robot2012.aiming`

Represents a single point in 3D space.

Point3d() - Constructor for class `com._604robotics.robot2012.aiming.Point3d`

Initializes a new Point3d.

Point3d(double, double, double) - Constructor for class `com._604robotics.robot2012.aiming.Point3d`

Initializes a new Point3d.

Point3d - Class in `com._604robotics.robot2012.vision`

This represents a point in 3d space

Point3d(double, double, double) - Constructor for class `com._604robotics.robot2012.vision.Point3d`

PointAndAngle3d - Class in `com._604robotics.robot2012.aiming`

A class to hold a 3d point.

PointAndAngle3d(double, double, double, double) - Constructor for class `com._604robotics.robot2012.aiming.PointAndAngle3d`

Initializes variables for the point.

PointAndAngle3d(Point3d, double) - Constructor for class `com._604robotics.robot2012.aiming.PointAndAngle3d`

Initializes variables for the point.

PortConfiguration - Interface in `com._604robotics.robot2012.configuration`

Port configuration.

PortConfiguration.Controllers - Interface in `com._604robotics.robot2012.configuration`

PortConfiguration.Controllers - Interface in `com._604robotics.robot2012.configuration`

PortConfiguration.Encoders - Interface in `com._604robotics.robot2012.configuration`

PortConfiguration.Encoders.Drive - Interface in `com._604robotics.robot2012.configuration`

PortConfiguration.Motors - Interface in `com._604robotics.robot2012.configuration`

PortConfiguration.Pneumatics - Interface in `com._604robotics.robot2012.configuration`

PortConfiguration.Pneumatics.HOPPER_SOLENOID - Interface in `com._604robotics.robot2012.configuration`

PortConfiguration.Pneumatics.PICKUP_SOLENOID - Interface in `com._604robotics.robot2012.configuration`

PortConfiguration.Pneumatics.SHIFTER_SOLENOID - Interface in `com._604robotics.robot2012.configuration`

PortConfiguration.Pneumatics.SHOOTER_SOLENOID - Interface in `com._604robotics.robot2012.configuration`

PortConfiguration.Relays - Interface in `com._604robotics.robot2012.configuration`

PortConfiguration.Sensors - Interface in `com._604robotics.robot2012.configuration`

PRESSURE_SWITCH - Static variable in interface `com._604robotics.robot2012.configuration.PortConfiguration.Pneumatics`

PUSH - Static variable in interface `com._604robotics.robot2012.configuration.ActuatorConfiguration.SOLENOID_HOPPER`

R

RB - Static variable in interface `com._604robotics.utils.XboxController.Button`

refreshGains() - Method in class `com._604robotics.utils.UpDownPIDController`
Updates the gains for the current direction.

REGULAR - Static variable in interface `com._604robotics.robot2012.configuration.ActuatorConfiguration.SOLENOID_HOPPER`

reload() - Method in class `com._604robotics.utils.DualVictor`

If the Victor has been sprung, unsprung it; if not, set the output to 0.

reload() - Method in class `com._604robotics.utils.SpringableDoubleSolenoid`

If the DoubleSolenoid has been sprung, unsprung it; if not, set the output to the default output.

reload() - Method in class `com._604robotics.utils.SpringableRelay`

If the Relay has been sprung, unsprung it; if not, set the output to the default output.

reload() - Method in class `com._604robotics.utils.SpringableVictor`

If the Victor has been sprung, unsprung it; if not, set the output to 0.

RemoteCameraTCP - Class in `com._604robotics.robot2012.camera`

Implements a CameraInterface that draws data from a TCP connection.

RemoteCameraTCP() - Constructor for class `com._604robotics.robot2012.camera.RemoteCameraTCP`

reset() - Method in class `com._604robotics.utils.EncoderOffset`

Resets the Encoder.

resetToggles() - Method in class `com._604robotics.utils.XboxController`

Resets the toggle registry for the contrller.

REVERSE - Static variable in interface `com._604robotics.robot2012.configuration.PortConfiguration.Pneumatics.HOPPER_SOLENOID`

RIGHT - Static variable in interface `com._604robotics.robot2012.configuration.ActuatorConfiguration.TURRET_POSITION`

RIGHT - Static variable in interface `com._604robotics.robot2012.configuration.ButtonConfiguration.Manipulator.Elevator`

RIGHT - Static variable in interface `com._604robotics.robot2012.machine.TurretMachine.TurretState`

Right - Static variable in interface `com._604robotics.utils.XboxController.Button.DPad`

RIGHT_A - Static variable in interface `com._604robotics.robot2012.configuration.PortConfiguration.Encoders.Drive`

RIGHT_B - Static variable in interface `com._604robotics.robot2012.configuration.PortConfiguration.Encoders.Drive`

RIGHT_DRIVE - Static variable in interface `com._604robotics.robot2012.configuration.PortConfiguration.Motors`

RIGHT_DRIVE_INCHES_PER_CLICK - Static variable in interface `com._604robotics.robot2012.configuration.SensorConfiguration.Encoders`

RIGHT_STICK - Static variable in interface `com._604robotics.utils.XboxController.Stick`

RIGHT_STICK_X - Static variable in interface `com._604robotics.utils.XboxController.Axis`

RIGHT_STICK_Y - Static variable in interface `com._604robotics.utils.XboxController.Axis`

RightStick - Static variable in interface `com._604robotics.utils.XboxController.Button`

RING_LIGHT_DIRECTION - Static variable in interface `com._604robotics.robot2012.configuration.PortConfiguration.Relays`

RING_LIGHT_PORT - Static variable in interface `com._604robotics.robot2012.configuration.PortConfiguration.Relays`

Robot2012Orange - Class in `com._604robotics.robot2012`

Main class for the 2012 robot code.

Robot2012Orange() - Constructor for class `com._604robotics.robot2012.Robot2012Orange`

Constructor.

robotinit() - Method in class `com._604robotics.robot2012.Robot2012Orange`

Initializes the robot on startup.

RotationProvider - Interface in `com._604robotics.robot2012.rotation`

Based on external feedback, aims the turret at the target.

RT - Static variable in interface `com._604robotics.utils.XboxController.Button`

S

SensorConfiguration - Interface in `com._604robotics.robot2012.configuration`

Sensor configuration.

SensorConfiguration.Encoders - Interface in `com._604robotics.robot2012.configuration`

set(double) - Method in class `com._604robotics.utils.DualVictor`

Sets the power of the Victors.

set(DoubleSolenoid.Value) - Method in class `com._604robotics.utils.SpringableDoubleSolenoid`

Sets the direction of the DoubleSolenoid.

set(Relay.Value) - Method in class `com._604robotics.utils.SpringableRelay`

Sets the direction of the Relay.

set(double) - Method in class `com._604robotics.utils.SpringableVictor`

Sets the power of the Victor.

setAccumulatorCenter(int) - Method in class `com._604robotics.utils.CompensatingGyro`

Manually sets the center for the accumulator.

setAngleGains(double, double, double) - Method in class `com._604robotics.utils.VelocityController`

Based on gyro angles TODO - javadoc

setCoastingRange(double, double) - Method in class `com._604robotics.utils.LinearController`

Updates the coasting values.

setController(PIDController) - Method in class `com._604robotics.utils.DeadbandedSource`

Sets the PIDController the source is fed into.

setController(PIDController) - Method in class `com._604robotics.utils.DualVictor`

Sets the PIDController for this DualVictor, if there is one.

setController(PIDController) - Method in class `com._604robotics.utils.SpringableVictor`

Sets the PIDController for this Victor, if there is one.

setConversionFactor(double) - Method in class `com._604robotics.utils.ConvertingPIDController`

Sets the factor to use when doing conversion on `setSetpoint` and `getSetpoint`.

setDeadband(double, double) - Method in class `com._604robotics.utils.DeadbandedSource`

Sets the range for the deadband.

setDeadband(double, double) - Method in class `com._604robotics.utils.DualVictor`

Sets the deadband for the DualVictor.

setDeadband(int, double, double) - Method in class `com._604robotics.utils.XboxController`

Sets the deadband for a particular axis.

setDefaultPosition(double) - Method in class `com._604robotics.robot2012.rotation.DummyRotationProvider`

setDefaultPosition(double) - Method in class `com._604robotics.robot2012.rotation.NaiveRotationProvider`

setDefaultPosition(double) - Method in interface `com._604robotics.robot2012.rotation.RotationProvider`

Sets the "default" position, if no targets can be located.

setDefaultPosition(double) - Method in class `com._604robotics.robot2012.rotation.SlightlySmarterRotationProvider`

setDefaultPosition(double) - Method in class `com._604robotics.robot2012.rotation.SlowbroRotationProvider`

setDownGains(UpDownPIDController.Gains) - Method in class `com._604robotics.utils.UpDownPIDController`

Sets the gains for going down.

setGains(double, double, double) - Method in class `com._604robotics.utils.VelocityController`

Reconfigures the gains on the PIDController.

setHorizontalRange(double, double) - Method in class `com._604robotics.utils.LinearController`

Updates the horizontal values.

setLeftInversion(boolean) - Method in class `com._604robotics.utils.DualVictor`

Sets the inversion for the "left" Victor.

setOffset(int) - Method in class `com._604robotics.utils.EncoderOffset`

Sets the offset value for the Encoder.

setRealSetpoint(double) - Method in class `com._604robotics.utils.ConvertingPIDController`

Sets the "real" setpoint of the PIDController.

setRightInversion(boolean) - Method in class `com._604robotics.utils.DualVictor`

Sets the inversion for the "right" Victor.

setSafetyEnabled(boolean) - Method in class `com._604robotics.utils.DualVictor`

Sets whether or not safety is enabled.

setSetpoint(double) - Method in class `com._604robotics.utils.ConvertingPIDController`

setSetpoint(double) - Method in class `com._604robotics.utils.UpDownPIDController`

Sets the setpoint to go to.

setShooterSpeed(double) - Method in class `com._604robotics.robot2012.machine.ShooterMachine`

Sets the shooter speed to use when, well, shooting.

setTarget(double) - Method in class `com._604robotics.utils.LinearController`

Sets the current target.

setTurretSidewaysPosition(double) - Method in class `com._604robotics.robot2012.machine.TurretMachine`

Sets the position to use as "SIDEWAYS".

setUpGains(UpDownPIDController.Gains) - Method in class `com._604robotics.utils.UpDownPIDController`

Sets the gains for going up.

setVelocity(double) - Method in class `com._604robotics.utils.VelocityController`

Sets the target velocity.

setX(double) - Method in class `com._604robotics.robot2012.vision.Point3d`

Sets the X value of this Point

setY(double) - Method in class `com._604robotics.robot2012.vision.Point3d`

Sets the Y value of this Point

setZ(double) - Method in class `com._604robotics.robot2012.vision.Point3d`

Sets the Z value of this Point

SHIFT - Static variable in interface `com._604robotics.robot2012.configuration.ButtonConfiguration.Driver`

SHOOTER_LEFT - Static variable in interface `com._604robotics.robot2012.configuration.PortConfiguration.Motors`

SHOOTER_RIGHT - Static variable in interface `com._604robotics.robot2012.configuration.PortConfiguration.Motors`

ShooterAnglePick - Class in `com._604robotics.robot2012.physics`

Enum-ish thing of angles to shoot at.

ShooterAnglePick(double) - Constructor for class `com._604robotics.robot2012.physics.ShooterAnglePick`

Initializes a new ShooterAnglePick.

shooterAnglePickBottom - Static variable in class `com._604robotics.robot2012.physics.ShooterAnglePick`

shooterAnglePickTop - Static variable in class `com._604robotics.robot2012.physics.ShooterAnglePick`

ShooterMachine - Class in `com._604robotics.robot2012.machine`

Machine to control the shooter/hopper system during firing.

ShooterMachine(DualVictor, Victor) - Constructor for class `com._604robotics.robot2012.machine.ShooterMachine`

Initializes a new ShooterMachine.

ShooterMachine.ShooterState - Interface in `com._604robotics.robot2012.machine`

The possible states the shooter could be in.

SHOOTING - Static variable in interface `com._604robotics.robot2012.machine.ShooterMachine.ShooterState`

SIDEWAYS - Static variable in interface `com._604robotics.robot2012.machine.TurretMachine.TurretState`

SlightlySmarterRotationProvider - Class in `com._604robotics.robot2012.rotation`

A slightly smarter implementation of a rotation provider, which tries to account for network delay, etc.

SlightlySmarterRotationProvider(PIDController, CameraInterface, Encoder) - Constructor for class

`com._604robotics.robot2012.rotation.SlightlySmarterRotationProvider`

Initializes a new SlightlySmarterRotationProvider.

SlowbroRotationProvider - Class in `com._604robotics.robot2012.rotation`

Implements a slow-er-ish, but more robust-ish, RotationProvider.

SlowbroRotationProvider(ConvertingPIDController, CameraInterface, Encoder) - Constructor for class

`com._604robotics.robot2012.rotation.SlowbroRotationProvider`

Initializes a new SlowbroRotationProvider.

speed - Variable in class `com._604robotics.robot2012.physics.BallFireInfo`

spring() - Method in class `com._604robotics.utils.DualVictor`

Springs the victor.

spring() - Method in class `com._604robotics.utils.SpringableDoubleSolenoid`

Springs the DoubleSolenoid.

spring() - Method in class `com._604robotics.utils.SpringableRelay`

Springs the Relay.

spring() - Method in class `com._604robotics.utils.SpringableVictor`

Springs the victor.

SpringableDoubleSolenoid - Class in `com._604robotics.utils`

Extender of a DoubleSolenoid providing an easier control flow.

SpringableDoubleSolenoid(int, int, DoubleSolenoid.Value) - Constructor for class `com._604robotics.utils.SpringableDoubleSolenoid`

Initializes a new SpringableDoubleSolenoid.

SpringableDoubleSolenoid(int, int, int, DoubleSolenoid.Value) - Constructor for class `com._604robotics.utils.SpringableDoubleSolenoid`

Initializes a new SpringableDoubleSolenoid.

SpringableRelay - Class in `com._604robotics.utils`

Extender of a Relay providing an easier control flow.

SpringableRelay(int, int, Relay.Direction, Relay.Value) - Constructor for class `com._604robotics.utils.SpringableRelay`

Initializes a new SpringableRelay.

SpringableRelay(int, Relay.Direction, Relay.Value) - Constructor for class `com._604robotics.utils.SpringableRelay`

Initializes a new SpringableRelay.

SpringableRelay(int, int, Relay.Value) - Constructor for class `com._604robotics.utils.SpringableRelay`

Initializes a new SpringableRelay.

SpringableRelay(int, Relay.Value) - Constructor for class `com._604robotics.utils.SpringableRelay`

Initializes a new SpringableRelay.

SpringableVictor - Class in `com._604robotics.utils`

Extender of a Victor providing an easier control flow.

SpringableVictor(int) - Constructor for class `com._604robotics.utils.SpringableVictor`

Initializes a new SpringableVictor on the given PWM port.

SpringableVictor(int, int) - Constructor for class `com._604robotics.utils.SpringableVictor`

Initializes a new SpringableVictor on the given module slot and PWM port.

Start - Static variable in interface `com._604robotics.utils.XboxController.Button`

StrangeMachine - Interface in `com._604robotics.robot2012.machine`

State manager for various components of the robot.

T

Target - Class in `com._604robotics.robot2012.vision`

Represents a target.

Target(double, double, double, double) - Constructor for class `com._604robotics.robot2012.vision.Target`

Target(double, double, double, double, double, double, double, double) - Constructor for class `com._604robotics.robot2012.vision.Target`

Target(Point3d, double) - Constructor for class `com._604robotics.robot2012.vision.Target`

Target() - Constructor for class `com._604robotics.robot2012.vision.Target`

Target - Class in `frc.vision`

An Object to hold target parameters.

Target() - Constructor for class `frc.vision.Target`

Blank constructor.

Target(int, int, int, int) - Constructor for class `frc.vision.Target`

test(int) - Method in class `com._604robotics.robot2012.machine.ElevatorMachine`

test(int) - Method in class `com._604robotics.robot2012.machine.PickupMachine`

test(int) - Method in class `com._604robotics.robot2012.machine.ShooterMachine`

test(int) - Method in interface `com._604robotics.robot2012.machine.StrangeMachine`
Tests if the Machine has yet attained the target state.

test(int) - Method in class `com._604robotics.robot2012.machine.TurretMachine`

TOGGLE_ANGLE - Static variable in interface `com._604robotics.robot2012.configuration.ButtonConfiguration.Manipulator`

TOGGLE_HEIGHT - Static variable in interface `com._604robotics.robot2012.configuration.ButtonConfiguration.Manipulator`

TOGGLE_LIGHT - Static variable in interface `com._604robotics.robot2012.configuration.ButtonConfiguration.Manipulator`

TOGGLE_PICKUP - Static variable in interface `com._604robotics.robot2012.configuration.ButtonConfiguration.Driver`

TOLERANCE - Static variable in interface `com._604robotics.robot2012.configuration.ActuatorConfiguration.TURRET_POSITION`

toString() - Method in class `com._604robotics.robot2012.vision.Target`

TURRET_CALIBRATION_OFFSET - Static variable in interface `com._604robotics.robot2012.configuration.SensorConfiguration`

TURRET_DEGREES_PER_CLICK - Static variable in interface `com._604robotics.robot2012.configuration.SensorConfiguration.Encoders`

TURRET_OKAY - Static variable in interface `com._604robotics.robot2012.machine.ElevatorMachine.ElevatorState`

TURRET_ROTATION - Static variable in interface `com._604robotics.robot2012.configuration.PortConfiguration.Motors`

TURRET_ROTATION_A - Static variable in interface `com._604robotics.robot2012.configuration.PortConfiguration.Encoders`

TURRET_ROTATION_B - Static variable in interface `com._604robotics.robot2012.configuration.PortConfiguration.Encoders`

TURRET_ROTATION_POWER_MAX - Static variable in interface `com._604robotics.robot2012.configuration.ActuatorConfiguration`

TURRET_ROTATION_POWER_MIN - Static variable in interface `com._604robotics.robot2012.configuration.ActuatorConfiguration`

TurretMachine - Class in `com._604robotics.robot2012.machine`
Machine to control the turret.

TurretMachine(PIDController, RotationProvider, Encoder) - Constructor for class `com._604robotics.robot2012.machine.TurretMachine`
Initializes a new TurretMachine.

TurretMachine.TurretState - Interface in `com._604robotics.robot2012.machine`
The possible states the turret could be in.

U

Up - Static variable in interface `com._604robotics.utils.XboxController.Button.DPad`

update() - Method in class `com._604robotics.robot2012.rotation.DummyRotationProvider`

update() - Method in class `com._604robotics.robot2012.rotation.NaiveRotationProvider`

update() - Method in interface `com._604robotics.robot2012.rotation.RotationProvider`
Updates the aiming of the turret.

update() - Method in class `com._604robotics.robot2012.rotation.SlightlySmarterRotationProvider`

update() - Method in class `com._604robotics.robot2012.rotation.SlowbroRotationProvider`

update() - Method in class `com._604robotics.utils.LinearController`
Updates the PIDOutput based on the latest data.

UpDownPIDController - Class in `com._604robotics.utils`
A PIDController with different gains for up and down.

UpDownPIDController(UpDownPIDController.Gains, UpDownPIDController.Gains, PIDSource, PIDOutput) - Constructor for class `com._604robotics.utils.UpDownPIDController`
Initializes a new UpDownPIDController.

UpDownPIDController.Gains - Class in `com._604robotics.utils`
A structure containing the P, I, and D gains.

UpDownPIDController.Gains(double, double, double) - Constructor for class `com._604robotics.utils.UpDownPIDController.Gains`

UPPER_ANGLE - Static variable in interface `com._604robotics.robot2012.configuration.ActuatorConfiguration.SOLENOID_SHOOTER`

UPPER_ANGLE - Static variable in interface `com._604robotics.robot2012.configuration.PortConfiguration.Pneumatics.SHOOTER_SOLENOID`

V

VelocityController - Class in `com._604robotics.utils`
Class for controlling a motor's velocity, rather than its power directly.

VelocityController(double, double, double, Encoder, Encoder, RobotDrive, Gyro) - Constructor for class `com._604robotics.utils.VelocityController`
Initializes a new VelocityController.

velToPow(double) - Static method in class `com._604robotics.robot2012.physics.Physics`
Returns an approximation of the power the shooter should be spun at

W

w - Variable in class `frc.vision.Target`

X

x - Variable in class `com._604robotics.robot2012.aiming.Point3d`

x - Variable in class `com._604robotics.robot2012.vision.Point3d`

x - Variable in class `com._604robotics.robot2012.vision.Point3d`
the x value

x - Variable in class `com._604robotics.robot2012.vision.Target`
x, y, and z represent the 3-d position of the target x will be positive when the target appears to be right of the center of the camera.

X - Static variable in interface `com._604robotics.utils.XboxController.Button`

x1 - Variable in class `frc.vision.Target`

x_uncertainty - Variable in class `com._604robotics.robot2012.vision.Target`
These are the uncertainties of the x, y, and z positions of the target.

XboxController - Class in `com._604robotics.utils`
Wrapper joystick class for the Xbox 360 controllers.

XboxController(int) - Constructor for class `com._604robotics.utils.XboxController`
Initialize a new XboxController on the specified port.

XboxController(Joystick) - Constructor for class `com._604robotics.utils.XboxController`
Initialize a new XboxController from the underlying Joystick.

XboxController.Axis - Interface in `com._604robotics.utils`
Enumeration for the available axes on the Xbox controller.

XboxController.Button - Interface in `com._604robotics.utils`
Enumeration for the available buttons on the Xbox controller.

XboxController.Button.DPad - Interface in `com._604robotics.utils`

XboxController.Stick - Interface in `com._604robotics.utils`
Enumeration for the available sticks on the Xbox controller.

Y

y - Variable in class `com._604robotics.robot2012.aiming.Point3d`

y - Variable in class `com._604robotics.robot2012.vision.Point3d`
the y value

y - Variable in class `com._604robotics.robot2012.vision.Target`
x, y, and z represent the 3-d position of the target x will be positive when the target appears to be right of the center of the camera.

Y - Static variable in interface `com._604robotics.utils.XboxController.Button`

y1 - Variable in class `frc.vision.Target`

y_uncertainty - Variable in class `com._604robotics.robot2012.vision.Target`
These are the uncertainties of the x, y, and z positions of the target.

Z

z - Variable in class `com._604robotics.robot2012.aiming.Point3d`

z - Variable in class `com._604robotics.robot2012.vision.Point3d`
the z value

z - Variable in class `com._604robotics.robot2012.vision.Target`
x, y, and z represent the 3-d position of the target x will be positive when the target appears to be right of the center of the camera.

z_uncertainty - Variable in class `com._604robotics.robot2012.vision.Target`
These are the uncertainties of the x, y, and z positions of the target.

ABCDEFGHIJKLMNOPQRSTUVWXYZ

Overview Package Class Tree Deprecated **Index** Help

Prev Next Frames No Frames All Classes