

com.\_604robotics.utils

# Class XboxController

java.lang.Object  
com.\_604robotics.utils.XboxController

```
public class XboxController
extends Object
```

Wrapper joystick class for the Xbox 360 controllers.

**Author:**

Michael Smith

### Nested Class Summary

**Nested Classes**

Modifier and Type	Class and Description
static interface	<a href="#">XboxController.Axis</a> Enumeration for the available axes on the Xbox controller.
static interface	<a href="#">XboxController.Button</a> Enumeration for the available buttons on the Xbox controller.
static interface	<a href="#">XboxController.Stick</a> Enumeration for the available sticks on the Xbox controller.

### Constructor Summary

**Constructors**

Constructor and Description
<a href="#">XboxController</a> (int port) Initialize a new XboxController on the specified port.
<a href="#">XboxController</a> (Joystick joystick) Initialize a new XboxController from the underlying Joystick.

### Method Summary

**Methods**

Modifier and Type	Method and Description
double	<a href="#">getAxis</a> (int axis) Get the value of the specified axis.
boolean	<a href="#">getButton</a> (int button) Get whether or not the specified button is currently pressed.
<a href="#">Joystick</a>	<a href="#">getJoystick</a> () Gets the underlying Joystick object.
boolean	<a href="#">getStick</a> (int stick) Get whether or not there's a value reading on the stick.
boolean	<a href="#">getToggle</a> (int button) Get the toggle state of the specified button.
void	<a href="#">resetToggles</a> () Resets the toggle registry for the contrller.
void	<a href="#">setDeadband</a> (int axis, double lower, double upper) Sets the deadband for a particular axis.

**Methods inherited from class java.lang.Object**  
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Constructor Detail

### XboxController

```
public XboxController(int port)
```

Initialize a new XboxController on the specified port.

**Parameters:**

`port` - The USB port the controller is connected to.

### XboxController

```
public XboxController(Joystick joystick)
```

Initialize a new XboxController from the underlying Joystick.

**Parameters:**

`joystick` - The Joystick to overlay the XboxController interface on.

## Method Detail

### getAxis

```
public double getAxis(int axis)
```

Get the value of the specified axis.

**Parameters:**

`axis` - One of the axis values specified in XboxController.Axis.

### getStick

```
public boolean getStick(int stick)
```

Get whether or not there's a value reading on the stick.

**Parameters:**

`stick` - One of the stick values specified in XboxController.Stick.

**Returns:**

Whether or not there's a value reading on the stick.

### getButton

```
public boolean getButton(int button)
```

Get whether or not the specified button is currently pressed.

**Parameters:**

`button` - One of the button values specified in XboxController.Button.

### resetToggles

```
public void resetToggles()
```

Resets the toggle registry for the contrller.

### getToggle

```
public boolean getToggle(int button)
```

Get the toggle state of the specified button.

**Parameters:**

`button` - One of the button values specified in XboxController.Button.

## getJoystick

```
public Joystick getJoystick()
```

Gets the underlying Joystick object. What, is XboxController not good enough for you?

### Returns:

The underlying Joystick object.

## setDeadband

```
public void setDeadband(int axis,  
                        double lower,  
                        double upper)
```

Sets the deadband for a particular axis.

### Parameters:

`axis` - The axis to set the deadband for.

`lower` - The lower bound of the deadband.

`upper` - The upper bound of the deadband.

[Overview](#) [Package](#) **[Class](#)** [Tree](#) [Deprecated](#) [Index](#) [Help](#)

[Prev Class](#) [Next Class](#) [Frames](#) [No Frames](#) [All Classes](#)

Summary: [Nested](#) | [Field](#) | [Constr](#) | [Method](#)      [Detail: Field](#) | [Constr](#) | [Method](#)