com.qualcomm.robotcore.hardware

**Class Gamepad**

* java.lang.Object
  + com.qualcomm.robotcore.hardware.Gamepad

public class **Gamepad**

extends java.lang.Object

Monitor a hardware gamepad.

The buttons, analog sticks, and triggers are represented a public member variables that can be read from or written to directly.

Analog sticks are represented as floats that range from -1.0 to +1.0. They will be 0.0 while at rest. The horizontal axis is labeled x, and the vertical axis is labeled y.

Triggers are represented as floats that range from 0.0 to 1.0. They will be at 0.0 while at rest.

Buttons are boolean values. They will be true if the button is pressed, otherwise they will be false.

The dpad is represented as 4 buttons, dpad\_up, dpad\_down, dpad\_left, and dpad\_right

* + ***Nested Class Summary***

|  |  |
| --- | --- |
| **Nested Classes** | |
| **Modifier and Type** | **Class and Description** |
| static interface | [**Gamepad.GamepadCallback**](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/hardware/Gamepad.GamepadCallback.html)  Optional callback interface for monitoring changes due to MotionEvents and KeyEvents. |

* + ***Field Summary***

|  |  |
| --- | --- |
| **Fields** | |
| **Modifier and Type** | **Field and Description** |
| boolean | [**a**](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/hardware/Gamepad.html#a)  button a |
| boolean | [**b**](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/hardware/Gamepad.html#b)  button b |
| boolean | [**back**](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/hardware/Gamepad.html#back)  button back |
| boolean | [**dpad\_down**](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/hardware/Gamepad.html#dpad_down)  dpad down |
| boolean | [**dpad\_left**](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/hardware/Gamepad.html#dpad_left)  dpad left |
| boolean | [**dpad\_right**](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/hardware/Gamepad.html#dpad_right)  dpad right |
| boolean | [**dpad\_up**](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/hardware/Gamepad.html#dpad_up)  dpad up |
| protected float | [**dpadThreshold**](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/hardware/Gamepad.html#dpadThreshold)  DPAD button will be considered pressed when the movement crosses this threshold |
| boolean | [**guide**](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/hardware/Gamepad.html#guide)  button guide - often the large button in the middle of the controller. |
| int | [**id**](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/hardware/Gamepad.html#id)  ID assigned to this gamepad by the OS. |
| static int | [**ID\_UNASSOCIATED**](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/hardware/Gamepad.html#ID_UNASSOCIATED)  A gamepad with an ID equal to ID\_UNASSOCIATED has not been associated with any device. |
| protected float | [**joystickDeadzone**](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/hardware/Gamepad.html#joystickDeadzone)  If the motion value is less than the threshold, the controller will be considered at rest |
| boolean | [**left\_bumper**](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/hardware/Gamepad.html#left_bumper)  button left bumper |
| boolean | [**left\_stick\_button**](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/hardware/Gamepad.html#left_stick_button)  left stick button |
| float | [**left\_stick\_x**](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/hardware/Gamepad.html#left_stick_x)  left analog stick horizontal axis |
| float | [**left\_stick\_y**](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/hardware/Gamepad.html#left_stick_y)  left analog stick vertical axis |
| float | [**left\_trigger**](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/hardware/Gamepad.html#left_trigger)  left trigger |
| boolean | [**right\_bumper**](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/hardware/Gamepad.html#right_bumper)  button right bumper |
| boolean | [**right\_stick\_button**](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/hardware/Gamepad.html#right_stick_button)  right stick button |
| float | [**right\_stick\_x**](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/hardware/Gamepad.html#right_stick_x)  right analog stick horizontal axis |
| float | [**right\_stick\_y**](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/hardware/Gamepad.html#right_stick_y)  right analog stick vertical axis |
| float | [**right\_trigger**](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/hardware/Gamepad.html#right_trigger)  right trigger |
| boolean | [**start**](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/hardware/Gamepad.html#start)  button start |
| long | [**timestamp**](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/hardware/Gamepad.html#timestamp)  Relative timestamp of the last time an event was detected |
| byte | [**user**](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/hardware/Gamepad.html#user)  Which user is this gamepad used by |
| boolean | [**x**](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/hardware/Gamepad.html#x)  button x |
| boolean | [**y**](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/hardware/Gamepad.html#y)  button y |

* + ***Constructor Summary***

|  |
| --- |
| **Constructors** |
| **Constructor and Description** |
| [**Gamepad**](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/hardware/Gamepad.html#Gamepad--)() |
| [**Gamepad**](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/hardware/Gamepad.html#Gamepad-com.qualcomm.robotcore.hardware.Gamepad.GamepadCallback-)(**[Gamepad.GamepadCallback](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/hardware/Gamepad.GamepadCallback.html" \o "interface in com.qualcomm.robotcore.hardware)** callback) |

* + ***Method Summary***

|  |  |
| --- | --- |
| **All Methods**[**Static Methods**](javascript:show(1);)[**Instance Methods**](javascript:show(2);)[**Concrete Methods**](javascript:show(8);) | |
| **Modifier and Type** | **Method and Description** |
| boolean | [**atRest**](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/hardware/Gamepad.html#atRest--)()  Are all analog sticks and triggers in their rest position? |
| protected void | [**callCallback**](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/hardware/Gamepad.html#callCallback--)() |
| protected float | [**cleanMotionValues**](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/hardware/Gamepad.html#cleanMotionValues-float-)(float number) |
| static void | [**clearWhitelistFilter**](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/hardware/Gamepad.html#clearWhitelistFilter--)()  Clear the device whitelist filter. |
| static void | [**enableWhitelistFilter**](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/hardware/Gamepad.html#enableWhitelistFilter-int-int-)(int vendorId, int productId)  Add a whitelist filter for a specific device vendor/product ID. |
| void | [**fromByteArray**](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/hardware/Gamepad.html#fromByteArray-byte:A-)(byte[] byteArray) |
| MsgType | [**getRobocolMsgType**](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/hardware/Gamepad.html#getRobocolMsgType--)() |
| static boolean | [**isGamepadDevice**](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/hardware/Gamepad.html#isGamepadDevice-int-)(int deviceId)  Does this device ID belong to a gamepad device? |
| protected boolean | [**pressed**](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/hardware/Gamepad.html#pressed-android.view.KeyEvent-)(android.view.KeyEvent event) |
| void | [**setJoystickDeadzone**](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/hardware/Gamepad.html#setJoystickDeadzone-float-)(float deadzone)  Set the joystick deadzone. |
| byte[] | [**toByteArray**](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/hardware/Gamepad.html#toByteArray--)() |
| java.lang.String | [**toString**](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/hardware/Gamepad.html#toString--)()  Display a summary of this gamepad, including the state of all buttons, analog sticks, and triggers |
| java.lang.String | [**type**](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/hardware/Gamepad.html#type--)()  Get the type of gamepad as a String. |
| void | [**update**](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/hardware/Gamepad.html#update-android.view.KeyEvent-)(android.view.KeyEvent event)  Update the gamepad based on a KeyEvent |
| void | [**update**](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/hardware/Gamepad.html#update-android.view.MotionEvent-)(android.view.MotionEvent event)  Update the gamepad based on a MotionEvent |

* + - **Methods inherited from class java.lang.Object**

clone, equals, finalize, getClass, hashCode, notify, notifyAll, wait, wait, wait

* + ***Field Detail***
    - **ID\_UNASSOCIATED**

public static final int ID\_UNASSOCIATED

A gamepad with an ID equal to ID\_UNASSOCIATED has not been associated with any device.

**See Also:**

[Constant Field Values](http://ftckey.com/apis/ftc/constant-values.html#com.qualcomm.robotcore.hardware.Gamepad.ID_UNASSOCIATED)

* + - **left\_stick\_x**

public float left\_stick\_x

left analog stick horizontal axis

* + - **left\_stick\_y**

public float left\_stick\_y

left analog stick vertical axis

* + - **right\_stick\_x**

public float right\_stick\_x

right analog stick horizontal axis

* + - **right\_stick\_y**

public float right\_stick\_y

right analog stick vertical axis

* + - **dpad\_up**

public boolean dpad\_up

dpad up

* + - **dpad\_down**

public boolean dpad\_down

dpad down

* + - **dpad\_left**

public boolean dpad\_left

dpad left

* + - **dpad\_right**

public boolean dpad\_right

dpad right

* + - **a**

public boolean a

button a

* + - **b**

public boolean b

button b

* + - **x**

public boolean x

button x

* + - **y**

public boolean y

button y

* + - **guide**

public boolean guide

button guide - often the large button in the middle of the controller. The OS may capture this button before it is sent to the app; in which case you'll never receive it.

* + - **start**

public boolean start

button start

* + - **back**

public boolean back

button back

* + - **left\_bumper**

public boolean left\_bumper

button left bumper

* + - **right\_bumper**

public boolean right\_bumper

button right bumper

* + - **left\_stick\_button**

public boolean left\_stick\_button

left stick button

* + - **right\_stick\_button**

public boolean right\_stick\_button

right stick button

* + - **left\_trigger**

public float left\_trigger

left trigger

* + - **right\_trigger**

public float right\_trigger

right trigger

* + - **user**

public byte user

Which user is this gamepad used by

* + - **id**

public int id

ID assigned to this gamepad by the OS. This value can change each time the device is plugged in

* + - **timestamp**

public long timestamp

Relative timestamp of the last time an event was detected

* + - **dpadThreshold**

protected float dpadThreshold

DPAD button will be considered pressed when the movement crosses this threshold

* + - **joystickDeadzone**

protected float joystickDeadzone

If the motion value is less than the threshold, the controller will be considered at rest

* + ***Constructor Detail***
    - **Gamepad**

public Gamepad()

* + - **Gamepad**

public Gamepad([Gamepad.GamepadCallback](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/hardware/Gamepad.GamepadCallback.html" \o "interface in com.qualcomm.robotcore.hardware) callback)

* + ***Method Detail***
    - **setJoystickDeadzone**

public void setJoystickDeadzone(float deadzone)

Set the joystick deadzone. Must be between 0 and 1.

**Parameters:**

deadzone - amount of joystick deadzone

* + - **update**

public void update(android.view.MotionEvent event)

Update the gamepad based on a MotionEvent

**Parameters:**

event - motion event

* + - **update**

public void update(android.view.KeyEvent event)

Update the gamepad based on a KeyEvent

**Parameters:**

event - key event

* + - **getRobocolMsgType**

public MsgType getRobocolMsgType()

* + - **toByteArray**
    - public byte[] toByteArray()

throws [RobotCoreException](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/exception/RobotCoreException.html)

**Throws:**

[RobotCoreException](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/exception/RobotCoreException.html)

* + - **fromByteArray**
    - public void fromByteArray(byte[] byteArray)

throws [RobotCoreException](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/exception/RobotCoreException.html)

**Throws:**

[RobotCoreException](http://ftckey.com/apis/ftc/com/qualcomm/robotcore/exception/RobotCoreException.html)

* + - **atRest**

public boolean atRest()

Are all analog sticks and triggers in their rest position?

**Returns:**

true if all analog sticks and triggers are at rest; otherwise false

* + - **type**

public java.lang.String type()

Get the type of gamepad as a String. This method defaults to "Standard".

**Returns:**

gamepad type

* + - **toString**

public java.lang.String toString()

Display a summary of this gamepad, including the state of all buttons, analog sticks, and triggers

**Overrides:**

toString in class java.lang.Object

**Returns:**

a summary

* + - **cleanMotionValues**

protected float cleanMotionValues(float number)

* + - **pressed**

protected boolean pressed(android.view.KeyEvent event)

* + - **callCallback**

protected void callCallback()

* + - **enableWhitelistFilter**
    - public static void enableWhitelistFilter(int vendorId,

int productId)

Add a whitelist filter for a specific device vendor/product ID.

This adds a whitelist to the gamepad detection method. If a device has been added to the whitelist, then only devices that match the given vendor ID and product ID will be considered gamepads. This method can be called multiple times to add multiple devices to the whitelist.

If no whitelist entries have been added, then the default OS detection methods will be used.

**Parameters:**

vendorId - the vendor ID

productId - the product ID

* + - **clearWhitelistFilter**

public static void clearWhitelistFilter()

Clear the device whitelist filter.

* + - **isGamepadDevice**

public static boolean isGamepadDevice(int deviceId)

Does this device ID belong to a gamepad device?

**Parameters:**

deviceId -

**Returns:**

true, if gamepad device; false otherwise