Class

GCMotion

A controller profile that supports orientation and motion.

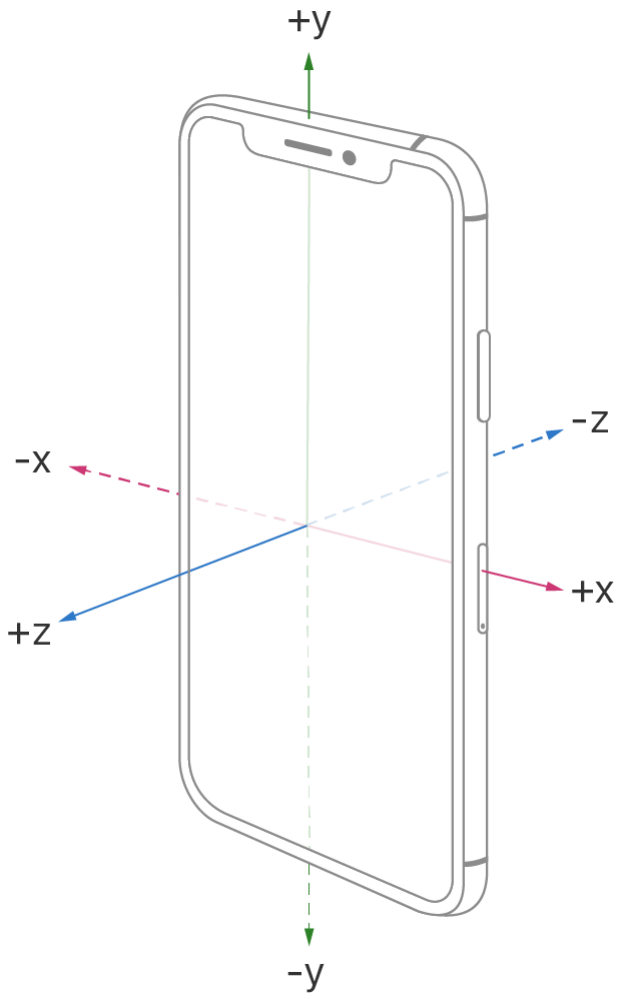
iOS 8.0+ | iPadOS 8.0+ | Mac Catalyst 13.1+ | macOS 10.10+ | tvOS 9.0+ | visionOS 1.0+

```
class GCMotion
```

Overview

The motion controller profile provides attitude and rotation data, as well as acceleration and sensor information. Use this profile to get motion input from a controller that measures acceleration and rotation rate. If the controller's `motion` property is a `GCMotion` object, the controller supports motion.

This illustration shows the direction of the x, y, and z axes of an iPhone when held upright.



Topics

Getting the Controller

`var controller: GCController?`

The controller for the profile.

Receiving a Callback When Input Values Change

`var valueChangedHandler: GCMotionValueChangedHandler?`

The block that the profile calls when an element's value changes.

`typealias GCMotionValueChangedHandler`

The signature for the block that the profile calls when an element's value changes.

Verifying Capabilities

`var hasAttitude: Bool`

A Boolean value that indicates whether the controller provides attitude data.

`var hasRotationRate: Bool`

A Boolean value that indicates whether the controller provides rotation data.

`var hasGravityAndUserAcceleration: Bool`

A Boolean value that indicates whether the controller provides gravity and user acceleration data.

~~`var hasAttitudeAndRotationRate: Bool`~~

A Boolean value that indicates whether the controller provides attitude and rotation data.

Deprecated

Accessing Attitude and Rotation Data

`var attitude: GCQuaternion`

The attitude of the controller.

`struct GCQuaternion`

A quaternion that represents a controller's measurement of attitude.

`var rotationRate: GCRotationRate`

The rotation rate of the controller.

`struct GCRotationRate`

A structure that represents rotation rates around the x, y, and z axes.

`struct GCEulerAngles`

A structure that specifies the controller's attitude as a series of rotations around the x, y, and z axes.

Accessing Gravity and Acceleration Data

`var acceleration: GCAcceleration`

The total acceleration of the controller that includes gravity and the acceleration the user applies to the controller.

`var gravity: GCAcceleration`

The gravity acceleration vector from the controller's reference frame.

`var userAcceleration: GCAcceleration`

The acceleration that the user applies to the controller.

`struct GCAcceleration`

A three-dimensional acceleration vector.

Accessing Sensor Data

`var sensorsRequireManualActivation: Bool`

A Boolean value that indicates whether the sensors that compute the motion data require manual activation.

`var sensorsActive: Bool`

A Boolean value that indicates whether the sensors that compute the motion data are active.

Setting Snapshot Values

`func setStateFrom(GCMotion)`

Copies the input values from a specified motion profile to a snapshot of a motion profile.

`func setAttitude(GCQuaternion)`

Sets the controller's attitude.

`func setRotationRate(GCRotationRate)`

Sets the controller's rotation rate.

`func setAcceleration(GCAcceleration)`

Sets the total acceleration of the controller that includes gravity and the user's acceleration.

`func setGravity(GCAcceleration)`

Sets the controller's gravity data.

`func setUserAcceleration(GCAcceleration)`

Sets the acceleration the user applies to the controller.

Relationships

Inherits From

NSObject

Conforms To

CVarArg
CustomDebugStringConvertible
CustomStringConvertible
Equatable
Hashable
NSObjectProtocol

See Also

Game controller profiles

Input

Receive controller input in the way that best integrates with the flow of your game or game engine.

`class` GCDeviceBattery

The charge level and state of a device's battery.

`class` GCDeviceHaptics

The locations of haptic actuators on a game controller.

`class` GCDeviceLight

The colored light on a device.