Controlling Game Input for Apple TV

Session 607

JJ Cwik Software Engineer
What’s New

tvOS 10

Apps may require an MFi game controller
Support for up to four MFi game controllers
Apple TV Remote app
Game Controller Framework

Overview

Adds Siri Remote and MFi game controller support to your game
Game Controller Framework

Overview

Adds Siri Remote and MFi game controller support to your game

Same API for all controllers

- Detect controllers
- Handle connection, disconnection
- Read inputs
Game Controller Framework
Overview

Adds Siri Remote and MFi game controller support to your game

Same API for all controllers

- Detect controllers
- Handle connection, disconnection
- Read inputs

Available for tvOS, iOS and macOS
Detecting Controllers

GCController represents a physical controller
Same class for all controllers
Detecting Controllers

GCController represents a physical controller
Same class for all controllers
Currently-connected controllers:

```javascript
GCController.controllers() // [GCController]
```
Detecting Controllers

GCController represents a physical controller
Same class for all controllers
Currently-connected controllers:

```javascript
GCController.controllers() // [GCController]
```

Connection and disconnection notifications:

```javascript
GCControllerDidConnectNotification
GCControllerDidDisconnectNotification
```
Siri Remote

GCController
Siri Remote

GCController

• GCMicroGamepad
Siri Remote

GCCController
  • GCMicroGamepad
  • GCMotion
GCMicroGamepad Profile

Dpad (analog or digital)
‘A’ button (digital)
‘X’ button (digital)
GCMicroGamepad Profile

Dpad (analog or digital)
‘A’ button (digital)
‘X’ button (digital)
GC Micro Gamepad Profile

- Dpad (analog or digital)
- ‘A’ button (digital)
- ‘X’ button (digital)
GCMicroGamepad Profile

Dpad (analog or digital)
‘A’ button (digital)
‘X’ button (digital)
GCMicroGamepad Profile

Dpad (analog or digital)

‘A’ button (digital)

‘X’ button (digital)
GC MicroGamepad Profile

Buttons

Polling

```swift
let isAPressed = controller.microGamepad?.buttonA.isPressed // Bool
let isXPressed = controller.microGamepad?.buttonX.isPressed // Bool
```
GCMicroGamepad Profile

Buttons

Polling

```swift
let isAPressed = controller.microGamepad?.buttonA.isPressed // Bool
let isXPressed = controller.microGamepad?.buttonX.isPressed // Bool
```

Event callbacks

```swift
controller.microGamepad?.buttonA.pressedChangedHandler = myButtonAHandler
controller.microGamepad?.buttonX.pressedChangedHandler = myButtonXHandler
```
GC Micro Gamepad Profile

GCControllerDirectionPad

Treated as four buttons
• up, down, left, right

And as two axes
• xAxis, yAxis
Polling

```swift
let dpadX = controller.microGamepad?.dpad.xAxis.value // Float, -1.0 to 1.0
let dpadY = controller.microGamepad?.dpad.yAxis.value // Float, -1.0 to 1.0
```
GC Micro Gamepad Profile
GC Controller Axis Input

Polling

```swift
let dpadX = controller.microGamepad?.dpad.xAxis.value // Float, -1.0 to 1.0
let dpadY = controller.microGamepad?.dpad.yAxis.value // Float, -1.0 to 1.0
```

Event callbacks

```swift
controller.microGamepad?.dpad.valueChangedHandler = myDpadHandler
```
Siri Remote
DPAD windowing
Siri Remote
DPAD windowing
Siri Remote

DPAD windowing

Provides windowed values (by default)
• Touch start location defines (0, 0)
• Subsequent values relative to start point
Siri Remote

DPAD windowing
Siri Remote

DPAD windowing
Siri Remote
DPAD windowing
Siri Remote

DPAD windowing
Siri Remote

DPAD windowing
Siri Remote

DPAD windowing

Opt-in to get absolute dpad values
Places (0, 0) at center of touch surface

```
controller.microGamepad?.reportsAbsoluteDpadValues = true
```
Siri Remote

DPAD windowing
Siri Remote

DPAD windowing
Siri Remote

DPAD windowing
Siri Remote

Rotation

DPAD (x, y) values reported in portrait

- Regardless of remote orientation
- X right, Y up
Siri Remote

Rotation

DPAD (x, y) values reported in portrait

• Regardless of remote orientation
• X right, Y up
Siri Remote

Rotation

Opt-in to report \((x, y)\) based on orientation

- Portrait
- Landscape left
- Landscape right

```
controller.microGamepad?.allowsRotation = true
```
Siri Remote

Motion

GCMotion

• Gravity vector (x, y, z)
• User acceleration (x, y, z)
Siri Remote
Motion

GCMotion
• Gravity vector (x, y, z)
• User acceleration (x, y, z)
Already filtered
Siri Remote
Motion

GCMotion
• Gravity vector (x, y, z)
• User acceleration (x, y, z)

Already filtered

Fused motion data
• Accelerometer and gyroscope
Siri Remote

Motion

Not intended for

- Vigorous shaking
- Aggressive movements
Menu Button
1. Minimize app and return to Apple TV Home Screen
Menu Button

1. Minimize app and return to Apple TV Home Screen
2. Go back one level in app menus
Menu Button

1. Minimize app and return to Apple TV Home Screen
2. Go back one level in app menus
3. Pause / resume gameplay
Menu Button
Menu Button

UIKit apps
Menu Button

UIKit apps

- Child View Controller
- Child View Controller
- Root View Controller

Games

- Root View Controller
  - Main menu
  - Character selection
  - Level selection
  - Gameplay
  - Pause screen
Menu Button

GCEventViewController

• For single-view controller games
• Use as your root view controller
Menu Button

GCEventViewController

• For single-view controller games
• Use as your root view controller

```swift
public var controllerUserInteractionEnabled: Bool

• false: Stay in-game
• true: Return to Home Screen
```
Menu Button
Managing controllerUserInteractionEnabled
Menu Button

Managing `controllerUserInteractionEnabled`
Menu Button

Managing controllerUserInteractionEnabledEnabled

countrollerUserInteractionEnabledEnabled:
- true
- true
- false
- false
- false
Menu Button

Pause handler

```swift
public var controllerPausedHandler: ((GCController) -> Void)?
```

Called when Menu button pressed
Menu Button

Pause handler

```swift
public var controllerPausedHandler: ((GCController) -> Void)?
```

Called when Menu button pressed

- In submenus: Go back to previous menu
- In gameplay: Toggle pause state
MFi Game Controller
MFi Game Controller
Optional accessory

Wireless extended gamepad

- DPAD
- ABXY
- 2 thumbsticks
- 2 shoulders
- 2 triggers
- Menu
- Player indicator LEDs
MFi Game Controller
GCExtendedGamepad profile

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>dpad</td>
<td></td>
</tr>
<tr>
<td>leftThumbstick</td>
<td>GCControllerDirectionPad</td>
</tr>
<tr>
<td>rightThumbstick</td>
<td></td>
</tr>
<tr>
<td>buttonA</td>
<td>GCControllerButtonInput</td>
</tr>
<tr>
<td>buttonB</td>
<td></td>
</tr>
<tr>
<td>buttonX</td>
<td></td>
</tr>
<tr>
<td>buttonY</td>
<td></td>
</tr>
<tr>
<td>leftShoulder</td>
<td></td>
</tr>
<tr>
<td>rightShoulder</td>
<td></td>
</tr>
<tr>
<td>leftTrigger</td>
<td></td>
</tr>
<tr>
<td>rightTrigger</td>
<td></td>
</tr>
</tbody>
</table>
MFi Game Controller
Pressure-sensitive

Polling

```swift
let digital = controller.extendedGamepad?.buttonA.isPressed  // Bool
let analog = controller.extendedGamepad?.buttonA.value        // Float, 0.0 to 1.0
```
MFi Game Controller

Polling

```-swift
let digital = controller.extendedGamepad?.buttonA.isPressed // Bool
let analog = controller.extendedGamepad?.buttonA.value // Float, 0.0 to 1.0
```

Event callbacks

```-swift
controller.extendedGamepad?.buttonA.pressedChangedHandler = myButtonAHandler // digital
controller.extendedGamepad?.buttonA.valueChangedHandler = myButtonAHandler // analog
```
Supported Controller Types

tvOS

Siri Remote
Supported Controller Types

tvOS

Siri Remote

Siri Remote and MFi game controllers
Supported Controller Types

tvOS

Siri Remote
Siri Remote and MFi game controllers
MFi game controllers
Supported Controller Types

tvOS

Specify in Game Controllers capability in Xcode
Automatically updates Info.plist
Needed for App Review
Supported Controller Types

App Store

- Badging
- May warn if MFi game controller has not been paired
Requiring MFi Game Controllers

Design considerations

Game may launch with no game controllers connected
Requiring MFi Game Controllers

Design considerations

Game may launch with no game controllers connected
Provide feedback to player to connect a game controller
Requiring MFi Game Controllers

Design considerations

Game may launch with no game controllers connected
Provide feedback to player to connect a game controller
Gracefully handle game controllers that disconnect
Requiring MFi Game Controllers

Design considerations

Game may launch with no game controllers connected
Provide feedback to player to connect a game controller
Gracefully handle game controllers that disconnect
See “Designing for tvOS” (session 802)
Apple TV Remote App
Apple TV Remote App
GC Micro Gamepad, GCMotion
Apple TV Remote App
Coalescing

controllers[0] <-> Siri Remote
controllers[1] <-> Apple TV Remote app
controllers[n]

NEW
Apple TV Remote App
Coalescing

controller[0] -> Siri Remote
controller[1] -> Apple TV Remote app
controller[n]
Apple TV Remote App

Separate remotes

controllers[0]

controllers[1]

controllers[n]

Siri Remote

Apple TV Remote app
Apple TV Remote App
Separate remotes

Opt-in to separate Siri Remote and Apple TV Remote app

GC Supports Multiple Micro Gamepads YES in Info.plist
Apple TV Remote App
Design considerations

Game Controller mode is in landscape
Apple TV Remote App

Design considerations

Game Controller mode is in landscape
Dpad values automatically rotated 90 degrees
Apple TV Remote App

**Design considerations**

Game Controller mode is in landscape

Dpad values automatically rotated 90 degrees

Don’t rotate dpad values in-game, instead set

`allowsRotation = true`
Apple TV Remote App

Design considerations

Game Controller mode separates ‘A’ Button from touchpad surface
Apple TV Remote App

Design considerations

Game Controller mode separates ‘A’ Button from touchpad surface
Can still simultaneously use both
Apple TV Remote App
Design considerations

Game Controller mode separates ‘A’ Button from touchpad surface
Can still simultaneously use both
‘A’ Button presses don’t implicitly provide touchpad DPAD values
Multiple Controllers
Multiple Controllers

One Siri Remote
Up to four MFi game controllers
Apple TV Remote app
Single-player game
Single-player game
Single-player game
Single-player game
Multiple Controllers

Single-player best practices

Allow user to seamlessly switch to a different controller
Multiple Controllers

Single-player best practices

Allow user to seamlessly switch to a different controller

Treat intentional input (buttons, DPAD, thumbsticks, touch surface) as user intent
Multiple Controllers

Single-player best practices

Allow user to seamlessly switch to a different controller
Treat intentional input (buttons, DPAD, thumbsticks, touch surface) as user intent
Take motion from the last controller to register intentional input
Summary

Siri Remote
Menu button
MFi game controllers
Apple TV Remote app
Multiple controllers
## Related Sessions

<table>
<thead>
<tr>
<th>Session</th>
<th>Location</th>
<th>Date/Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mastering UIKit on tvOS</td>
<td>Presidio</td>
<td>Wednesday 10:00AM</td>
</tr>
<tr>
<td>Designing for tvOS</td>
<td>Presidio</td>
<td>Tuesday 4:00PM</td>
</tr>
</tbody>
</table>
## Labs

<table>
<thead>
<tr>
<th>Lab</th>
<th>Location</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Game Controllers Lab</td>
<td>Graphics, Games, and Media Lab A</td>
<td>Thursday 9:00AM</td>
</tr>
<tr>
<td>tvOS Lab</td>
<td>Frameworks Lab D</td>
<td>Thursday 9:00AM</td>
</tr>
</tbody>
</table>