Haptic Feedback makes you vibrate!

Marius Constantinescu
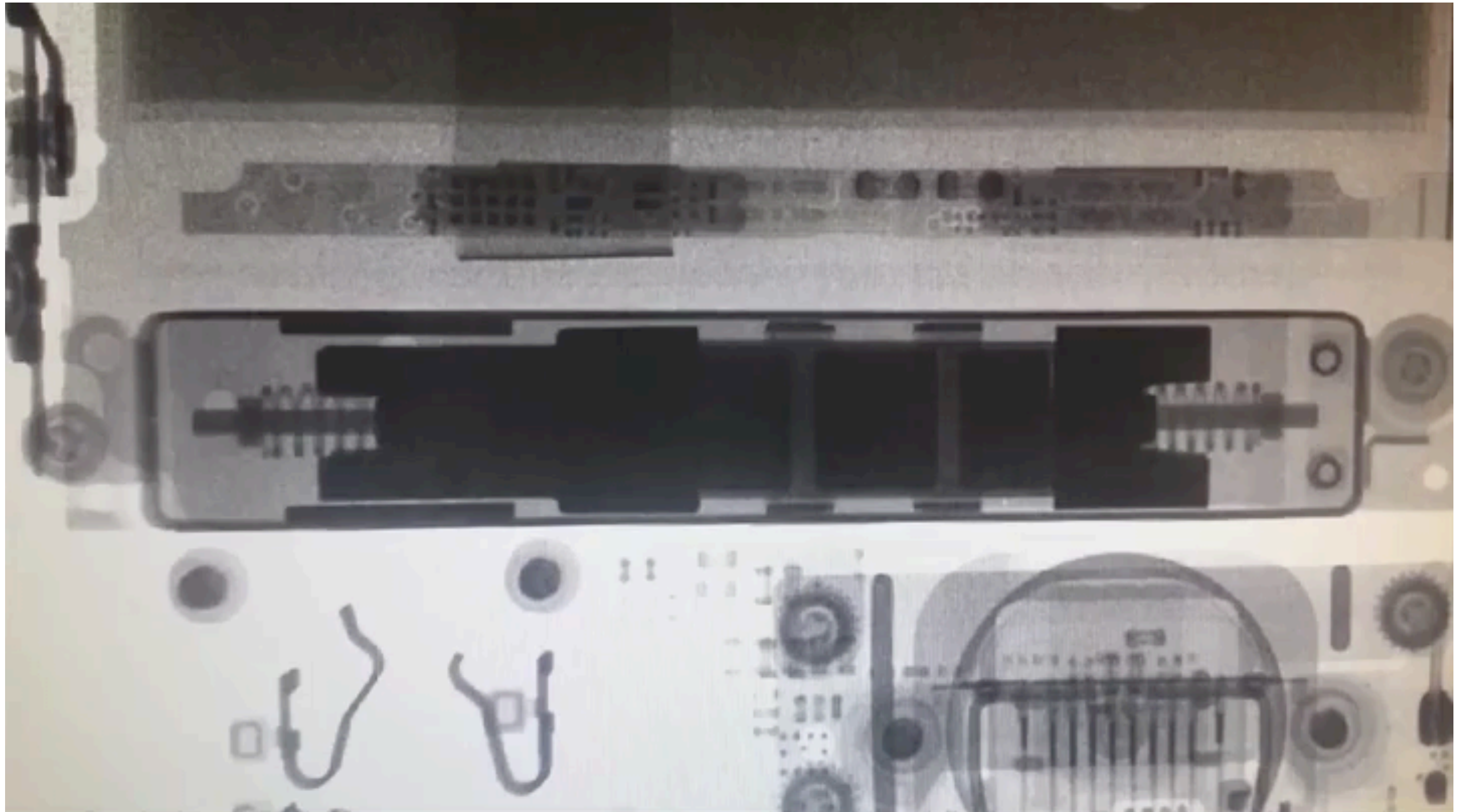
@marius_const

greener pastures

iOS Goodies

Haptic Feedback makes you vibrate!

Taptic Engine

- introduced in the Apple Watch

- force touch trackpads in MacBooks

- standalone force touch trackpads

- iPhone 6S

- improved in iPhone 7 and later iPhones

Taptic Engine => Haptic feedback

Before iOS 10 / iPhone 7

```swift
import AudioToolbox

@IBAction func vibrate(_ sender: Any) {
    AudioServicesPlaySystemSound(kSystemSoundID_Vibrate)
}
```

But iPhone 6S has nice haptics for peek/pop/error

```swift
import AudioToolbox

@IBAction func vibrate(_ sender: Any) {
    AudioServicesPlaySystemSound(kSystemSoundID_Vibrate)
}
```

```swift
import AudioToolbox

@IBAction func vibrate(_ sender: Any) {
    AudioServicesPlaySystemSound(kSystemSoundID_Vibrate)
}

...

public var kSystemSoundID_Vibrate: SystemSoundID { get }
```

```swift
import AudioToolbox

@IBAction func vibrate(_ sender: Any) {
    AudioServicesPlaySystemSound(kSystemSoundID_Vibrate)
}

...

public var kSystemSoundID_Vibrate: SystemSoundID { get }

public typealias SystemSoundID = UInt32
```

```swift
// "peek"
@IBAction func oldStylePeekPressed(_ sender: Any) {
    AudioServicesPlaySystemSound(SystemSoundID(1519))
}


// "pop"
@IBAction func oldStylePopPressed(_ sender: Any) {
    AudioServicesPlaySystemSound(SystemSoundID(1520))
}


// "error"
@IBAction func oldStyleErrorPressed(_ sender: Any) {
    AudioServicesPlaySystemSound(SystemSoundID(1521))
}
```
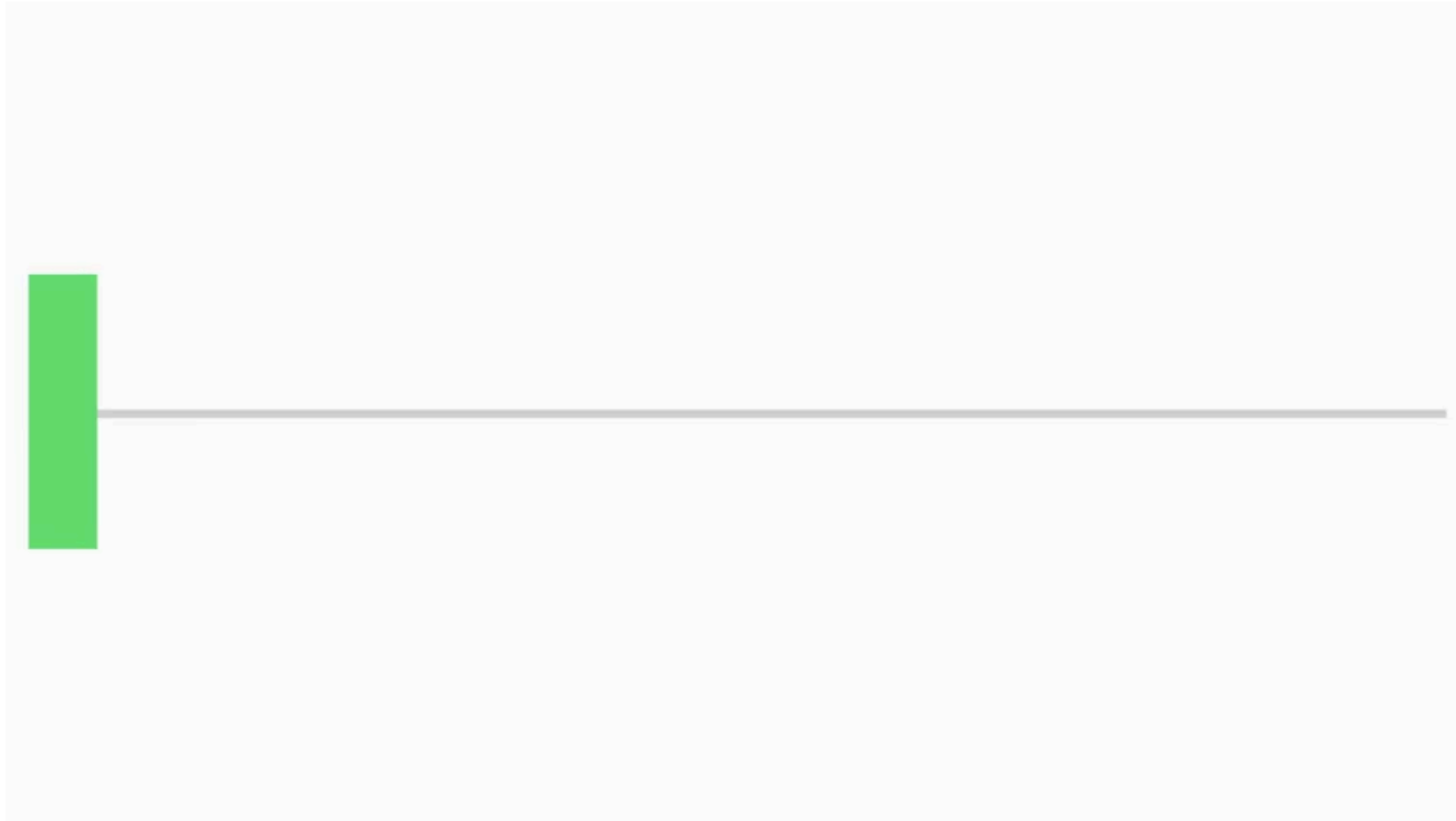
iOS 10+, iPhone 7 and later

UIFeedbackGenerator

- **UISelectionFeedbackGenerator**

- **UIImpactFeedbackGenerator**

- **UINotificationFeedbackGenerator**

Selection

Impact

- enhance a visual experience

Impact - Light

Impact - Medium

Impact - Heavy

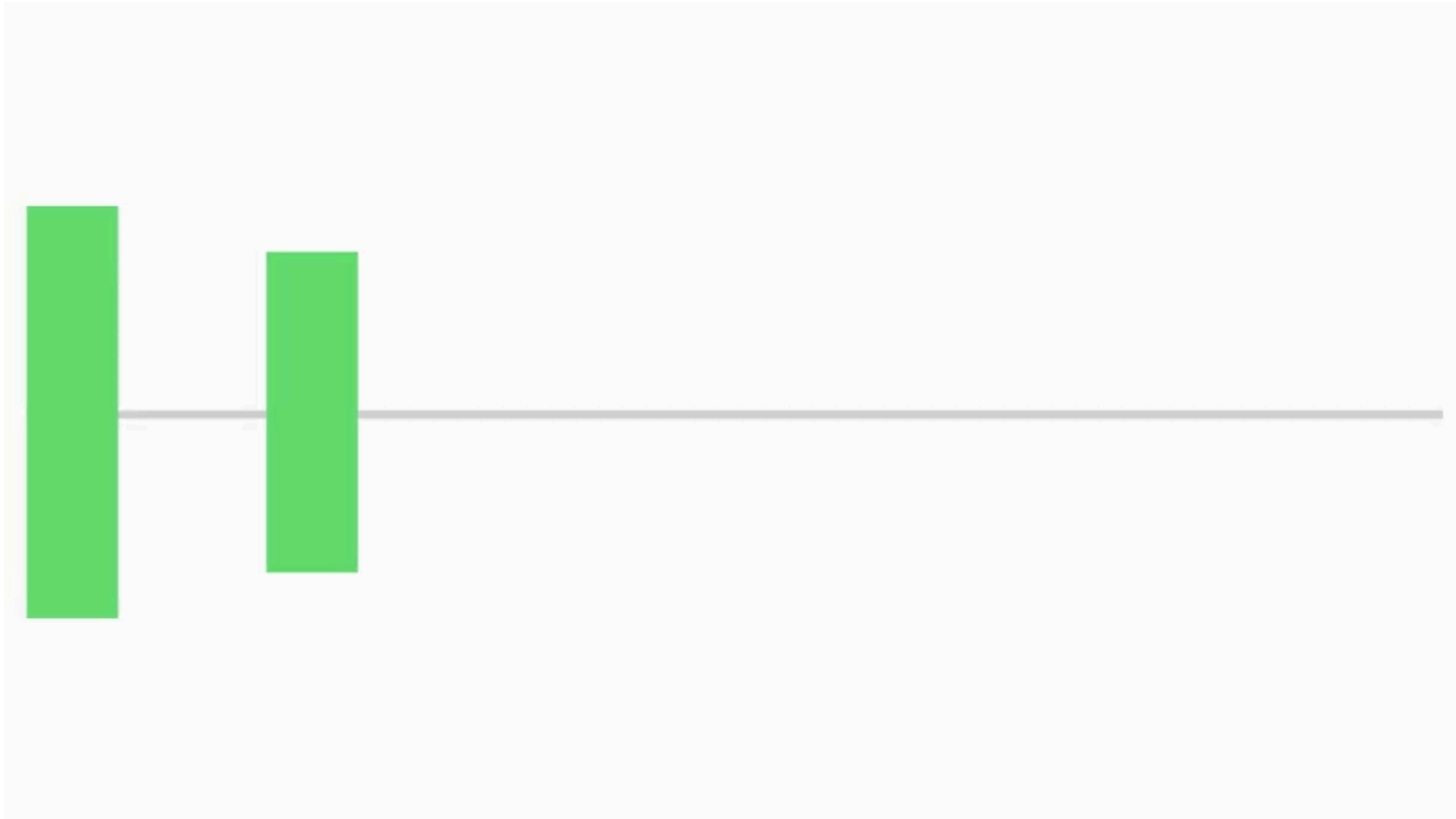# Notification

- a task has completed

Notification - Success

Notification - Warning

Notification - Failure

Implementation

Selection

```swift
let selectionFeedbackGenerator = UISelectionFeedbackGenerator()
selectionFeedbackGenerator.selectionChanged()
```

Impact

```
let lightImpactFeedbackGenerator = UIImpactFeedbackGenerator(style: .light)
lightImpactFeedbackGenerator.impactOccurred()

let mediumImpactFeedbackGenerator = UIImpactFeedbackGenerator(style: .medium)
mediumImpactFeedbackGenerator.impactOccurred()

let heavyImpactFeedbackGenerator = UIImpactFeedbackGenerator(style: .heavy)
heavyImpactFeedbackGenerator.impactOccurred()
```

Notification

```swift
let successNotificationFeedbackGenerator = UINotificationFeedbackGenerator()
successNotificationFeedbackGenerator.notificationOccurred(.success)

let warningNotificationFeedbackGenerator = UINotificationFeedbackGenerator()
warningNotificationFeedbackGenerator.notificationOccurred(.warning)

let errorNotificationFeedbackGenerator = UINotificationFeedbackGenerator()
errorNotificationFeedbackGenerator.notificationOccurred(.error)
```

Gotchas

- `UIFeedbackGenerator` may have a little latency

- `prepare()`

- deallocate when done

When to implement

- don't overuse

- users can disable it
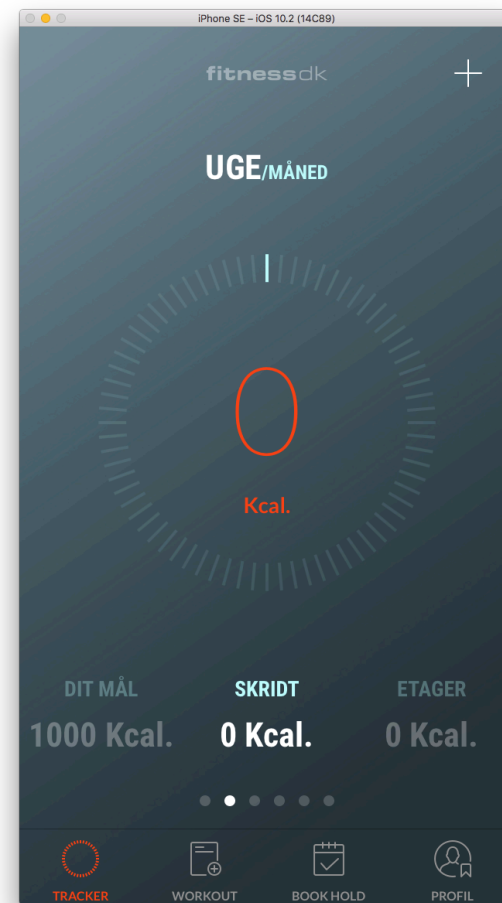
- use common sense

What Apple does

- home button

- `UIDatePicker, UIPickerView`

- `UIRefreshControl`

- `UIScrollView` zoom

- iMessage effects (i.e. fireworks, laser)

Advice

- haptic feedback alone is weird

- use it together with some animations or sounds

Usecases

- fitness dk - selection change

Usecases

- scanner app: barcode / QR

- games: collisions, etc

Conclusions

- 7 different types of haptic feedback

- use them to enhance an experience

- don't overdo it

Resources

- iOS Human Interface Guidelines

- UIFeedbackGenerator API Reference

- https://github.com/mariusc/Vibrate

- https://engineering.nodesagency.com/articles/iOS/Haptic-feedback-makes-you-vibrate/

- Make Your iOS App 'Feel' Better—A Guide Over Taptic Engine & Haptic Feedback

🙇