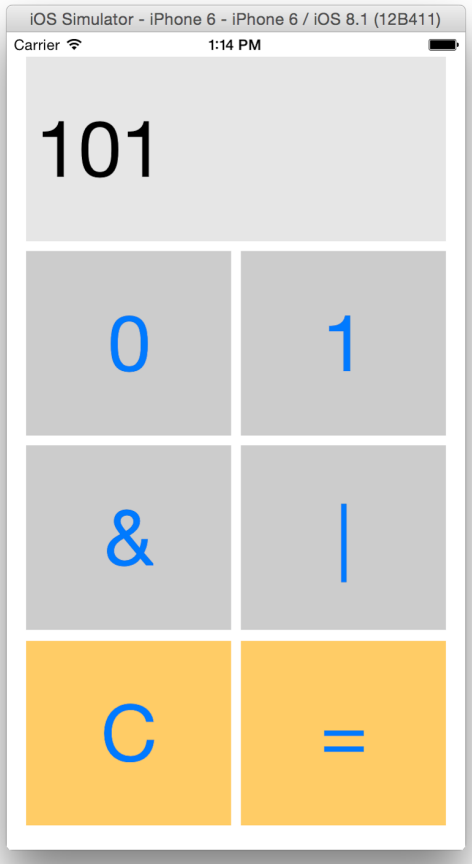
Let’s build an iPhone app: BinaryCalculator

Goals

- Build single-view iPhone app
- Explore Swift and relevant technolgies
- Automated testing (unit, functional)

Calculator

Note: Source available in [BinaryCalculator GitHub repo]



- Unsigned binary number (e.g., “101” is 5)
- Bitwise and, or operations
- *Buffer* holds the current number, *register* holds the previous number

Reference

- MVC
    - *Model*: plain old Swift objects (alternatively, Core Data)
    - *View*: storyboard (alternatively, XIBs or programatically)
    - *Controller*: subclass of UI`ViewController`
- *Storyboard*: develop application layout in a single file
- *Auto Layout*: constraint-based layout
- *Playground*: interactive Swift worksheet (interpretted on save)
- *Swift*: successor language to Objective-C, released in 2014
- *XCTest*: XCode Testing framework.

	C	**Objective-C, Swift**
Language APIs	Core Foundation (e.g., `CFStringRef`)	Foundation (e.g., `NSString`)

	OS X	**iOS**
Frameworks collection	Cocoa	Cocoa Touch
UI Framework	AppKit (e.g., `NSButton`)	UIKit (e.g., `UIButton`)

Errata

Here's how to define a custom unary operator:

```
// Can use prefix too
postfix operator +++ {}

postfix func +++ (i:Int) -> Int {
    return i + 2
}

var i = 0
i+++   // i = 2
```