Google

Simplifying IoT with Firebase

Mike McDonald

Engineer @ Google/Firebase

@asciimike, @firebase

IoT development requires system level thought

Hardware
Interacting with the physical world

Cloud
Bridging the gap between devices

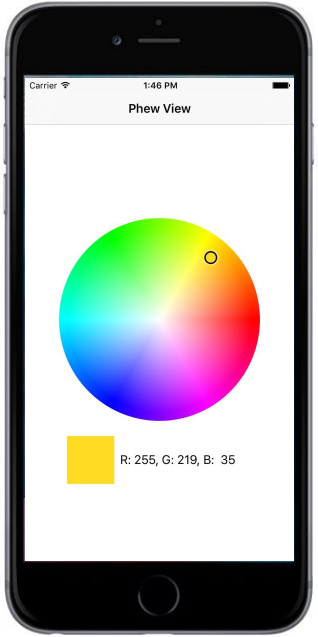
Mobile
Building compelling user experiences

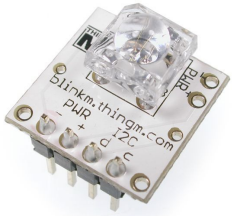
It’s like trying to build a stool

How can we make hardware development as fast and easy as software development?

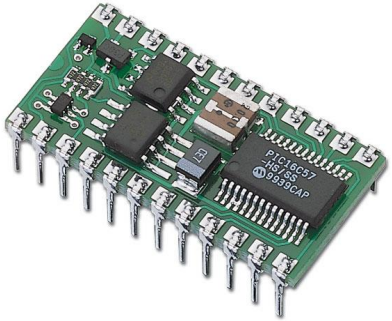
Let’s give ourselves something to work towards...

Introducing “The Phew”





Picking hardware is hard

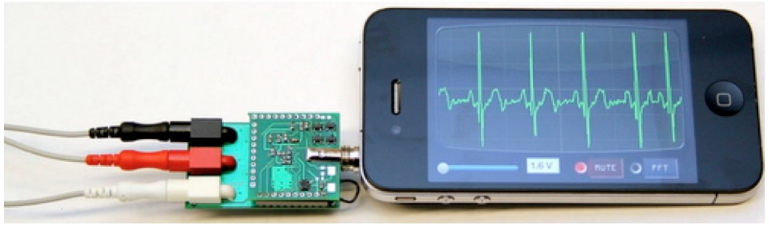


Talking to hardware: Lightning!

Made for iPod iPhone iPad

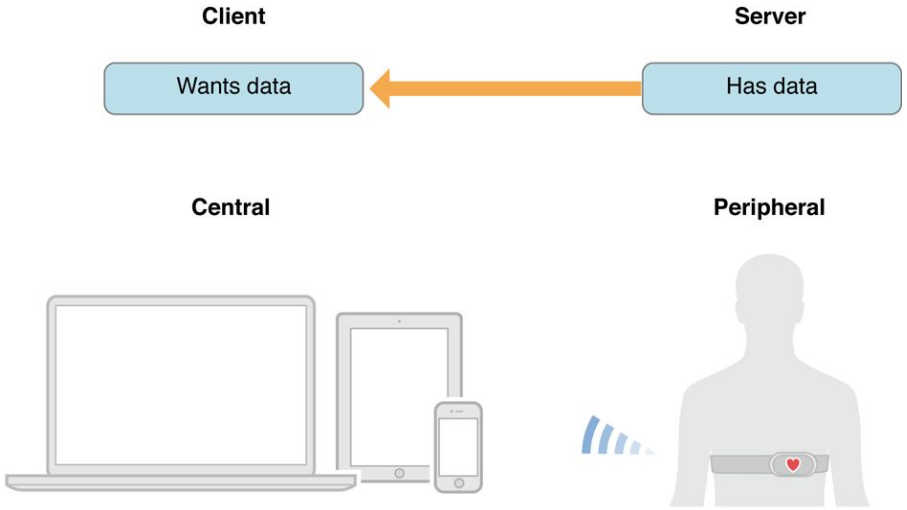
- MFi program allows access to the Lightning port
- Most commonly used for things like game controllers (and… that's about it)
- ExternalAccessory.framework

Talking to hardware: Headphone Jack



- Outputs audio signal through the headphone jack to a microcontroller, which can then interact with sensors
- Easily ported cross platform
- HiJack: project from U of M
- Thermodo: commercial temp sensor

Talking to hardware: Bluetooth



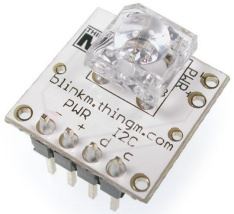
- CoreBluetooth is your new best friend
- Great for low power wireless, audio transmission
- There are *lots* of different bluetooth protocols: A2DP, HID, etc. which can be somewhat confusing
- Beacons are a great offshoot of this technology (iBeacon, Eddystone)

Talking to hardware: Internet (WiFi, ethernet)

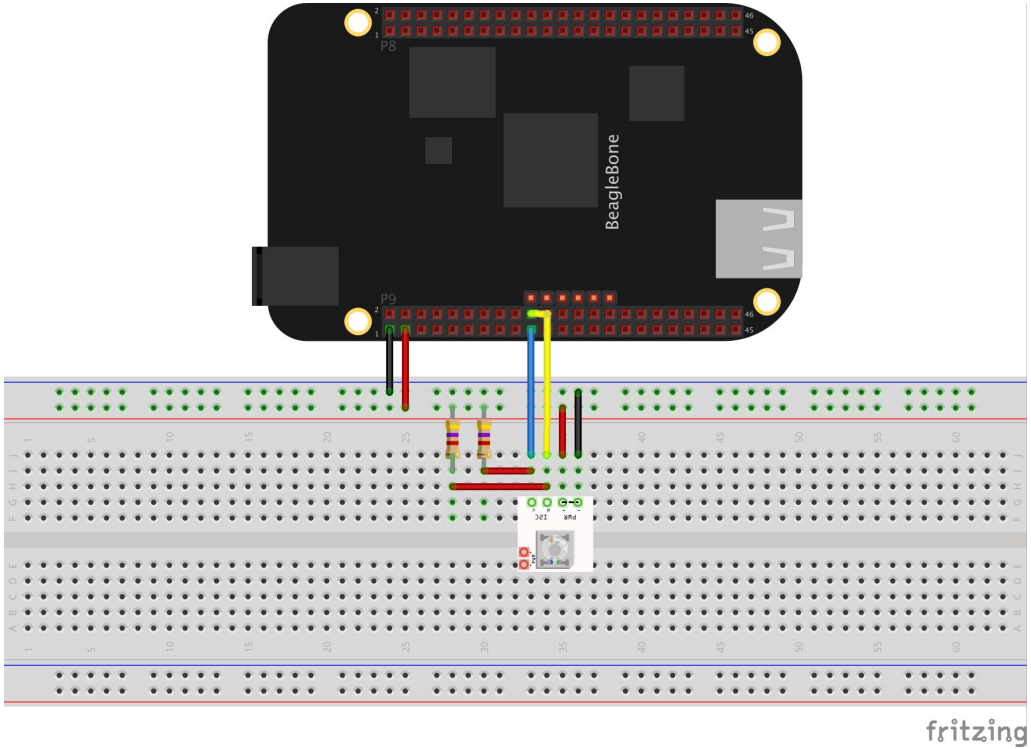
- Easy for mobile: **it's just another API**
- Robust, high throughput, secure*
- But, if your Internet goes out...

*with HTTPS...

Let’s see the hardware!



Hardware schematic



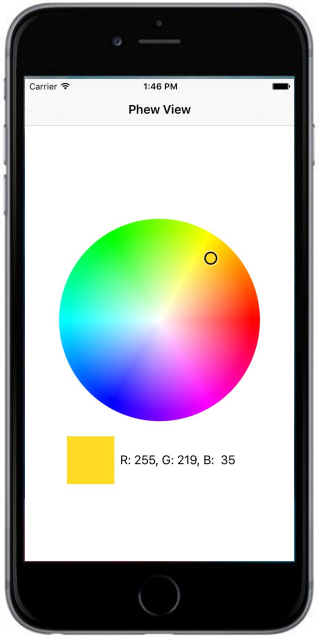
Firebase glues devices together

```
// Firebase is JSON
"your-firebase": {
  "devices": {
    "phew_bulb": {
      "a": 255,
      "b": 255,
      "g": 255,
      "r": 255
    }
  }
}
```

```objc
// Access your data by going directly to that path
Firebase *ref = [[Firebase alloc] initWithUrl:@“https://your-firebase.firebaseio.com”];
Firebase *bulbRef = [ref childByAppendingPath:@“devices/phew_bulb”];
[bulbRef setValue:@{/* R, G, B, A */}];
[bulbRef observeEventType:FEventTypeValue withBlock:^(FDataSnapshot *snapshot) {
  NSDictionary *dict = snapshot.value;
  // dict = @{/* R, G, B, A */}
}];
```

```
// And in Swift...
let ref = Firebase(url: @“https://your-firebase.firebaseio.com”)
let bulbRef = ref.childByAppendingPath(@“devices/phew_bulb”)
bulbRef.setValue([/* R, G, B, A */])
bulbRef.observeEventType(.Value, withBlock: { snapshot -> Void in
  let dict = snapshot.value as! Dictionary<String, Double>
  // dict = [/* R, G, B, A */]
}
```

Let’s build an app!



Questions?

Appetize: https://goo.gl/oWUNvx
Twitter: @asciimike, @firebase, @googledevs
Github: mcdonamp/phew (code & slides)