

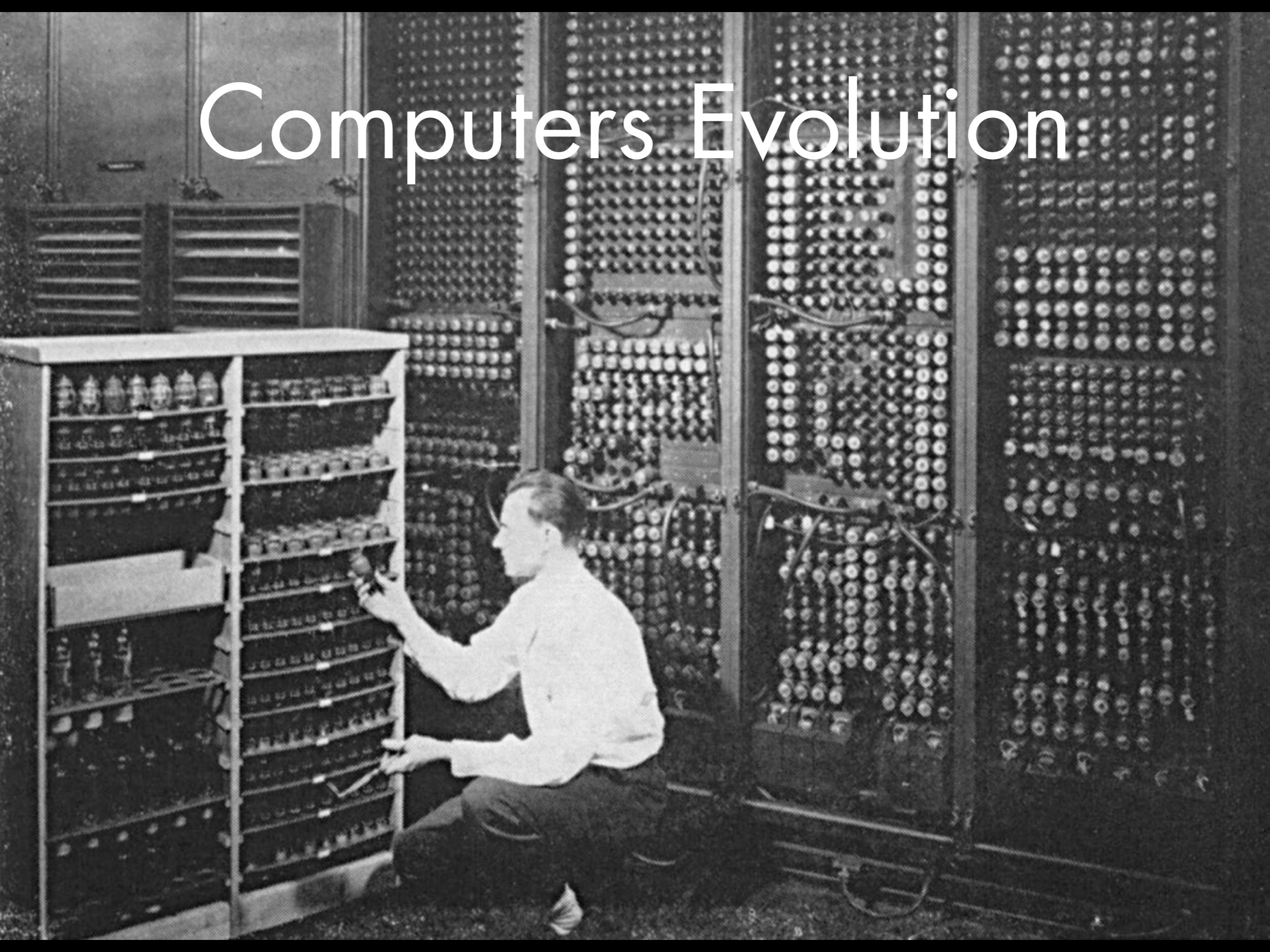
# PERVASIVE COMPUTING WORKSHOP

**Imanol Gómez**  
[imanolgomez.net](http://imanolgomez.net)  
13 Nov 2015

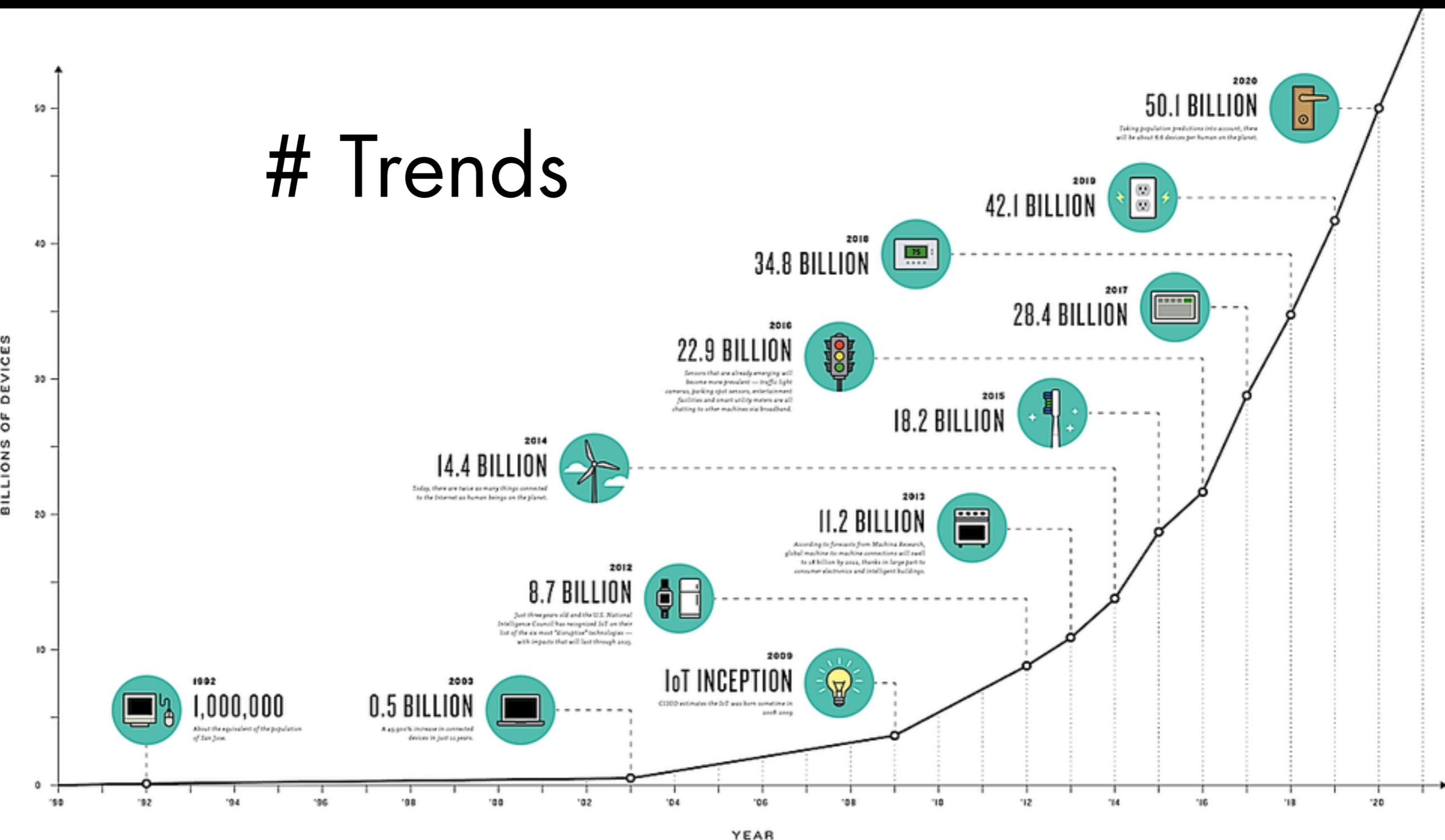
**hfg**  
OF\_MAIN

# Introduction

# Computers Evolution



# # Trends



# Ubiquitous





Paradigm

*"In theory, theory and practice are the same. In practice, they are not."*

- Albert Einstein -

# Challenges

Anything  
Any Device



Anytime  
Any Context



Anyone  
Anybody



Any Place  
Anywhere



Any Service  
Any Business



Any Path  
Any Network



# OBEY

PROPAGANDA-SERVICES

RETINAL DELIVERY



100% GUARANTEED



PROP MFG.

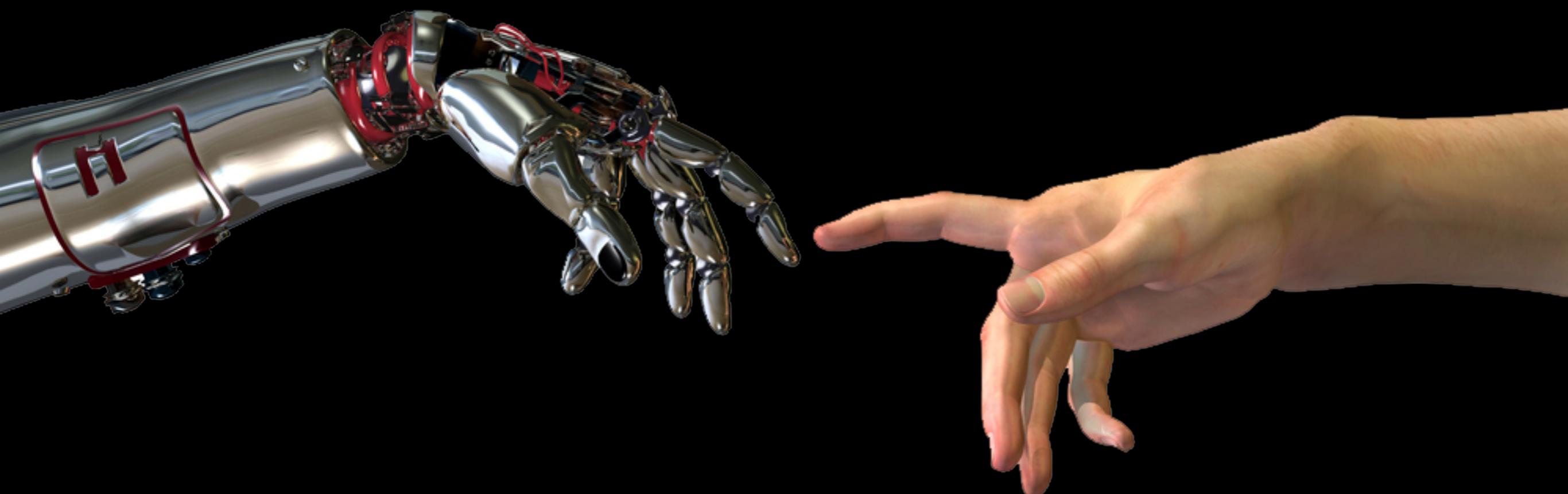
MANUFACTURING DISSENT

PERSUASION SINCE 89

# Security

SEVENFEETFOURINCHESFIVEHUNDREDTWENTYPOUNDS

# Transdisciplinary Approach



# Related Technologies



# Internet of Things

# Wearable Technology



# Ambient Intelligence



# Examples

# Products



SMART THERMOSTATS

nest



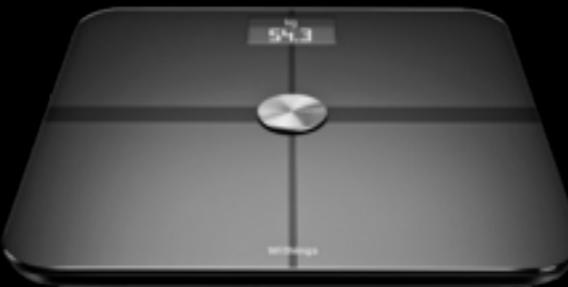
ACTIVITY TRACKERS

NIKE



SMART LIGHTS

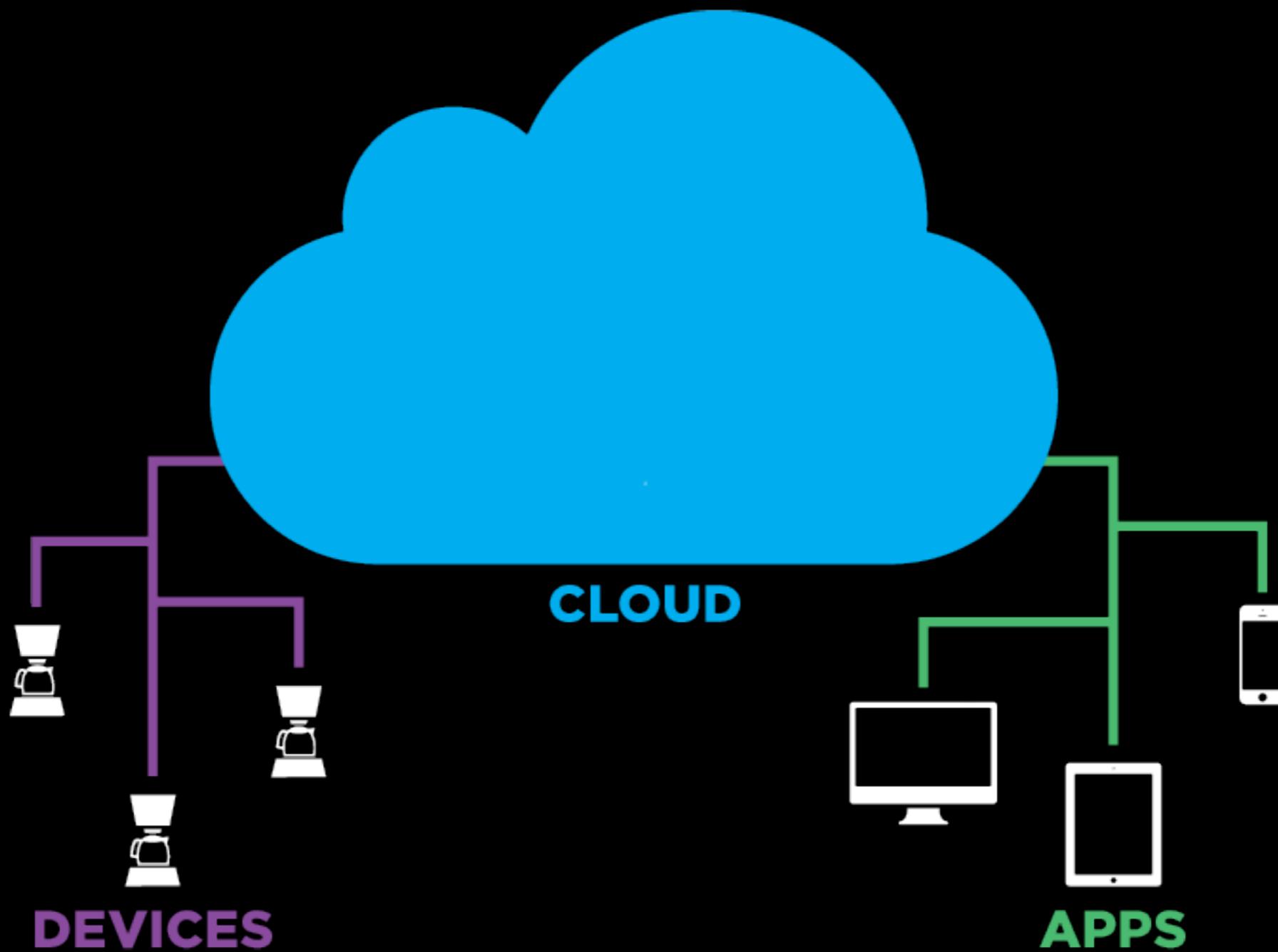
PHILIPS



CONNECTED SCALE

Withings

# How does it work?



# The Cloud



# Applications



FASHION  
LIFESTYLE

TRANSPORT  
MOBILITY/TRAFFIC

RETAIL  
INVENTORY

near-field connect for your apps  
Developer Preview Kit  
10 Bluetooth Low Energy Beacons



**LOGISTICS**  
REAL TIME



**MANUFACTURING**  
M2M



**HEALTH**  
BODY



**ENVIRONMENT**  
PREVENTION



**BUILDING**  
INFRASTRUCTURE  




**AGRICULTURE**  
CONTROL



**SECURITY**  
DETECTION

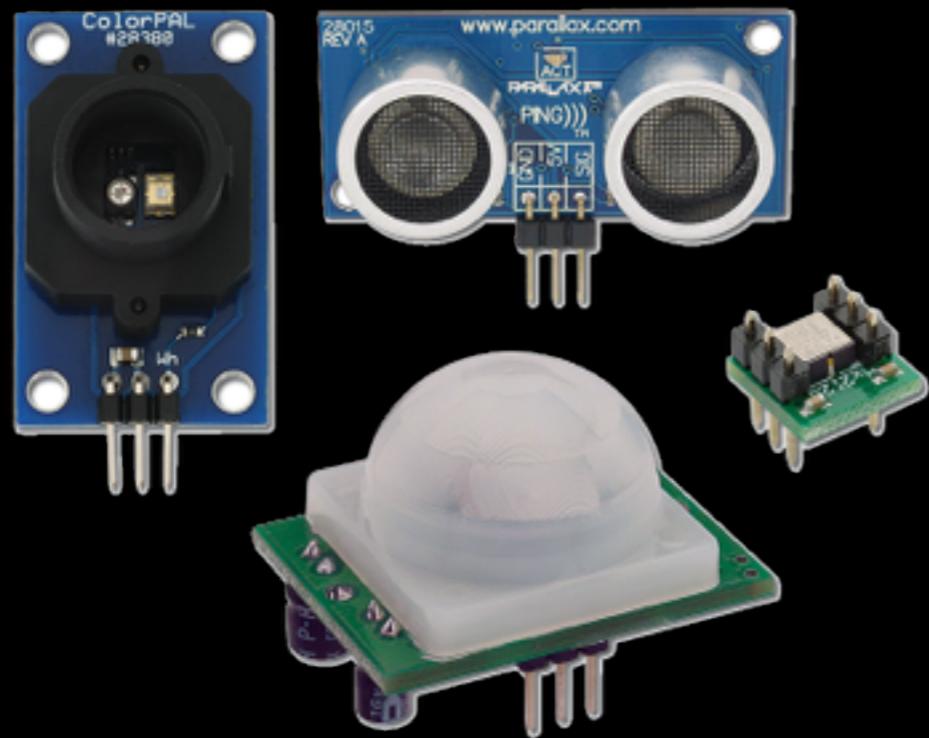
*"Technology should create calm"*

Mark Weiser, father of pervasive computing

# Prototyping

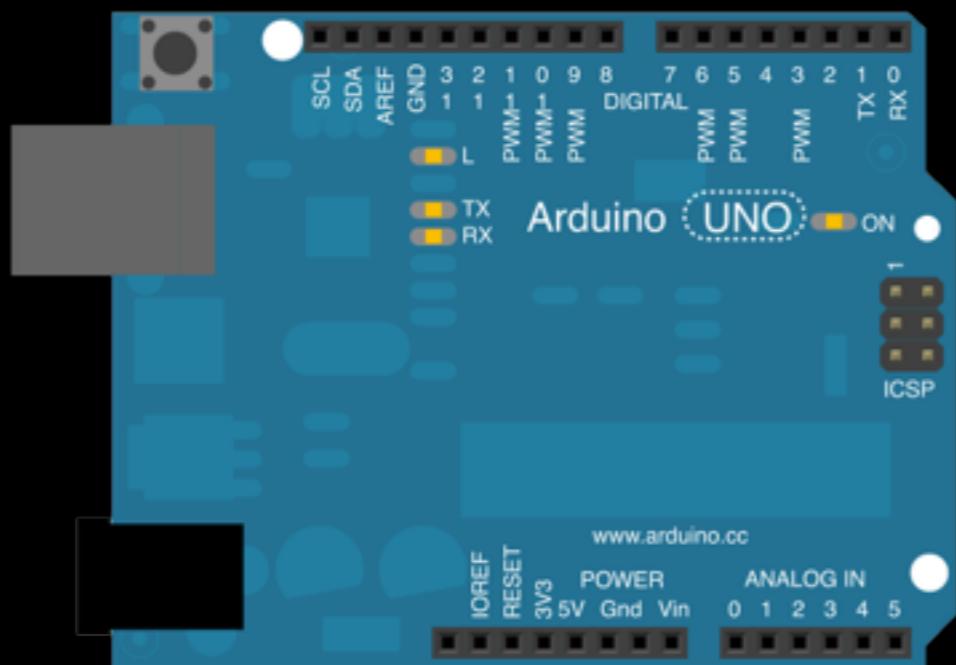
# Discussion

# Sensors



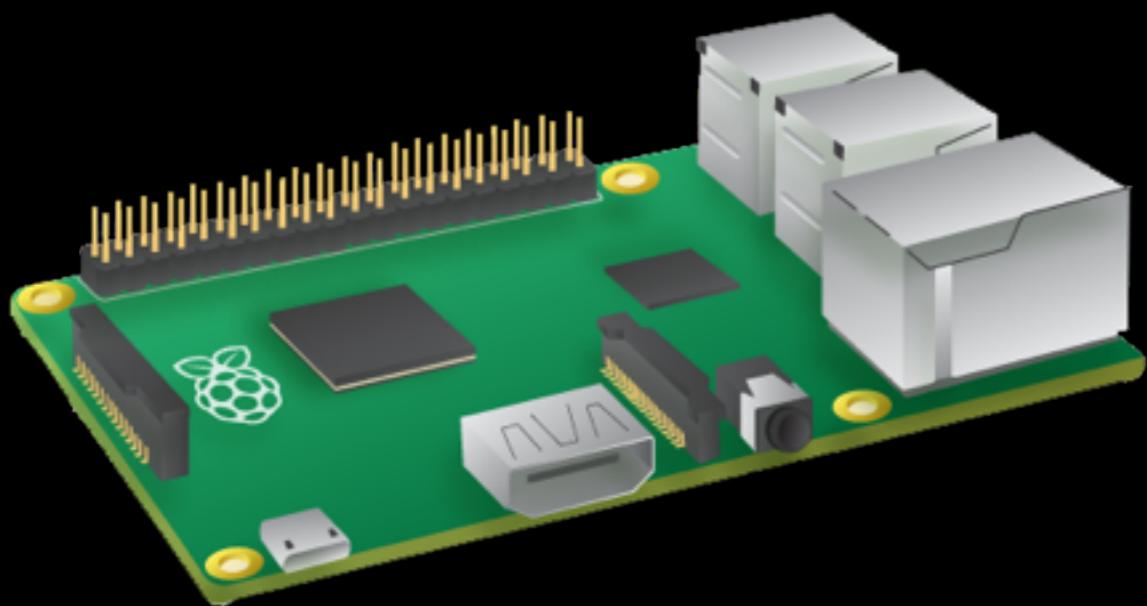
Temperature sensor  
Infrared sensor  
Microphone  
Humidity sensor  
Touch sensor  
Heartbeat sensor  
Tilt switch  
Vibration switch  
Photo resistor  
Hunt sensor  
Hit sensor  
...

# Microcontrollers



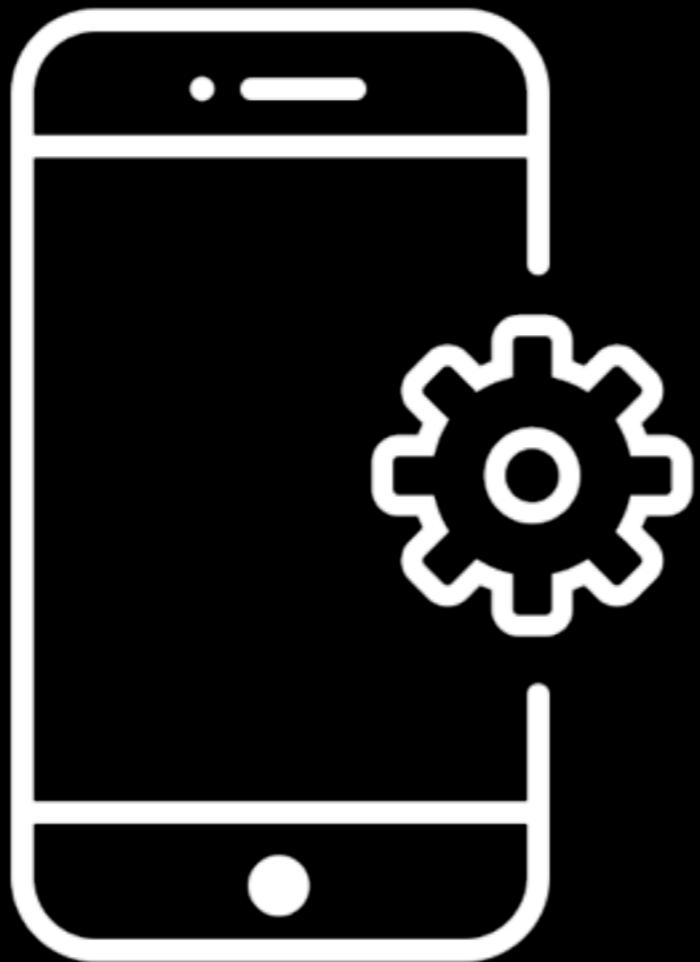
- Arduino
  - Uno, Pro, Mega, Leonardo, nano
- Teensy
- Little Bits
- MSP430 LaunchPad
- ...

# Mini Computers



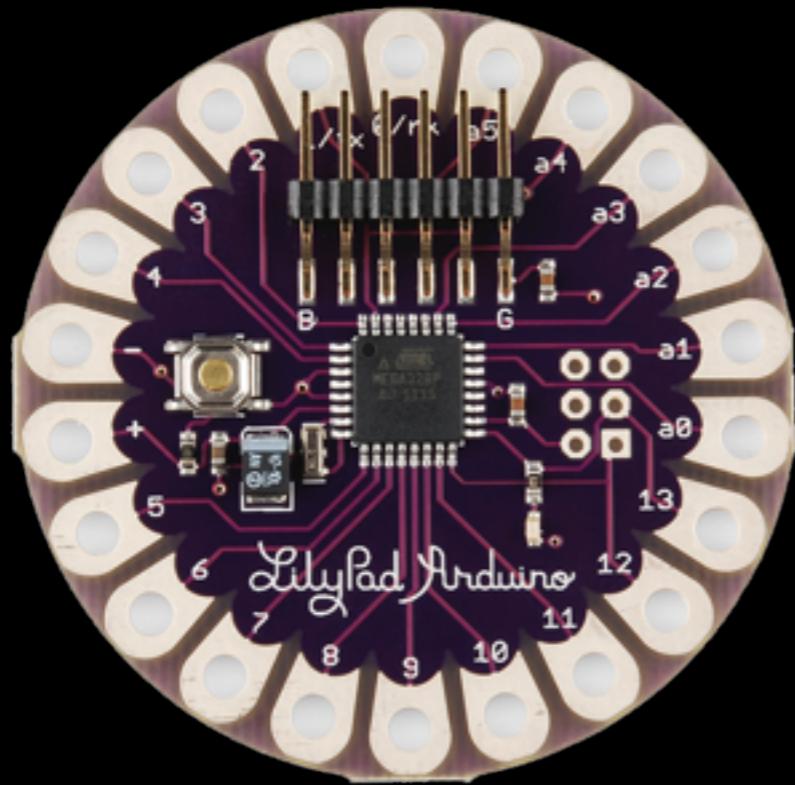
- Raspberry Pi
- Odroid
- BeagleBone
- pcDuino
- Intel Edison
- Acadia
- ...

# Smartphone



- Motion Sensors
- Environmental Sensors
- Position sensors
- Camera
- Microphone
- Touchscreen
- WiFi
- Bluetooth
- ...

# Wearables



- Arduino Lilypad
- Arduino Gemma
- Teensy
- Metawear

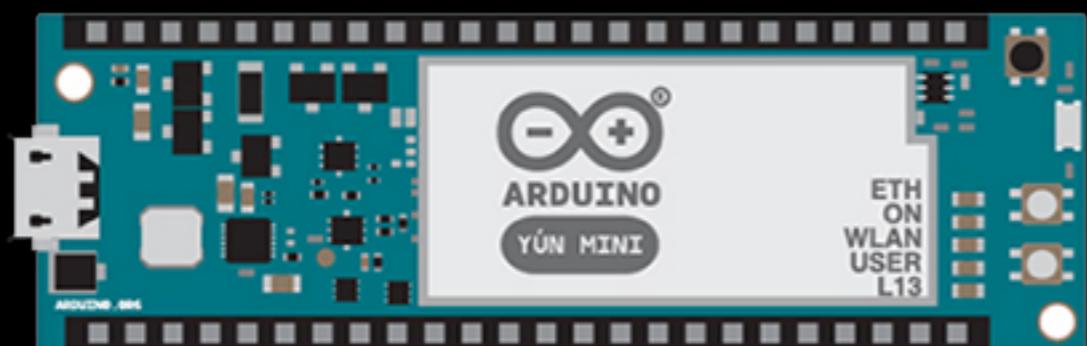
# Internet of Things



- Particle
- lotivity
- Arduino Yún
- PanStamp
- ...

# Workshop

# Arduino Yún Mini

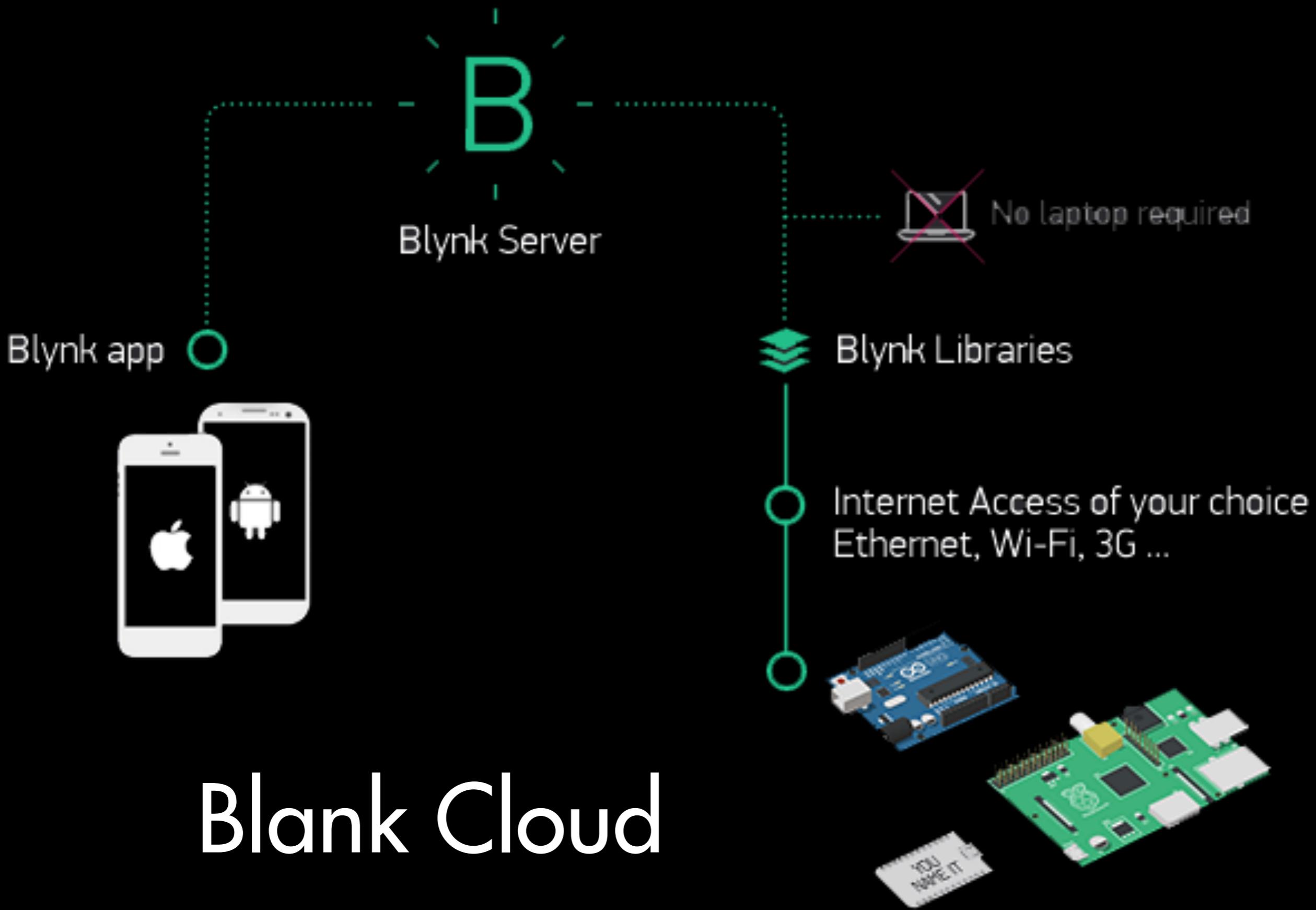


- Arduino with WiFi built in
- Linux with Python preinstalled
- <https://www.arduino.cc/en/Main/ArduinoBoardYun>
- Download Software: <https://www.arduino.cc/en/Main/Software>

# Blynk



- iOS and Android apps to control Arduino.
- <http://www.blynk.cc/>
- Getting Started: <http://www.blynk.cc/getting-started/>



# Repository

<https://github.com/lmanolGo/PervasiveMediaWorkshop>

# Steps Arduino

1. Install Arduino Software

<http://www.arduino.org/software>

2. Download Blynk Library

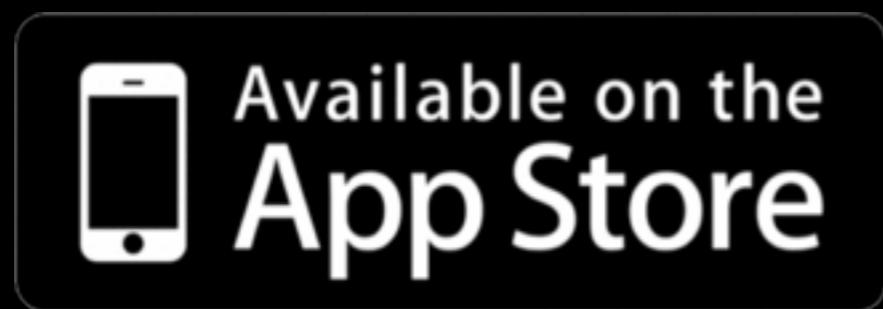
[https://github.com/blynkkk/blynk-library/releases/  
latest](https://github.com/blynkkk/blynk-library/releases/latest)

3. Install Library

<http://www.arduino.cc/en/guide/libraries>

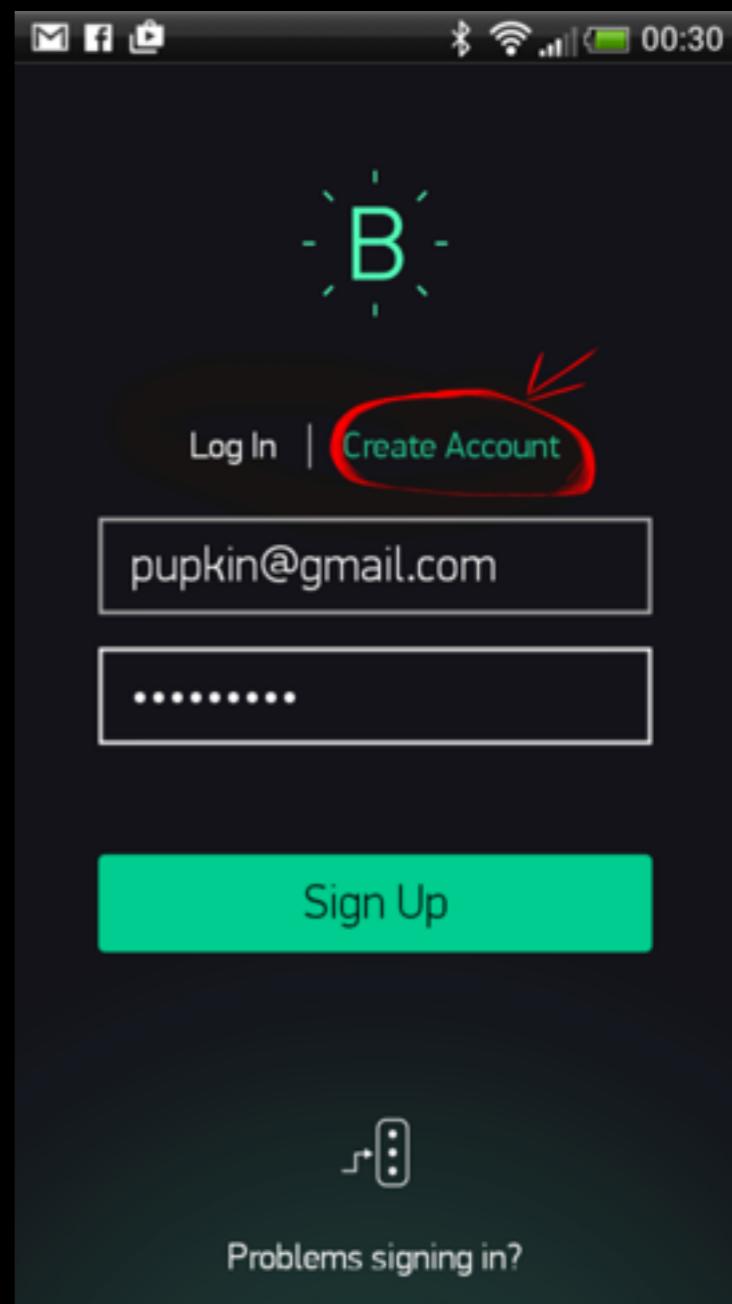
# Steps Blynk

- 1 . Install Blynk Apps for iOS or Android.



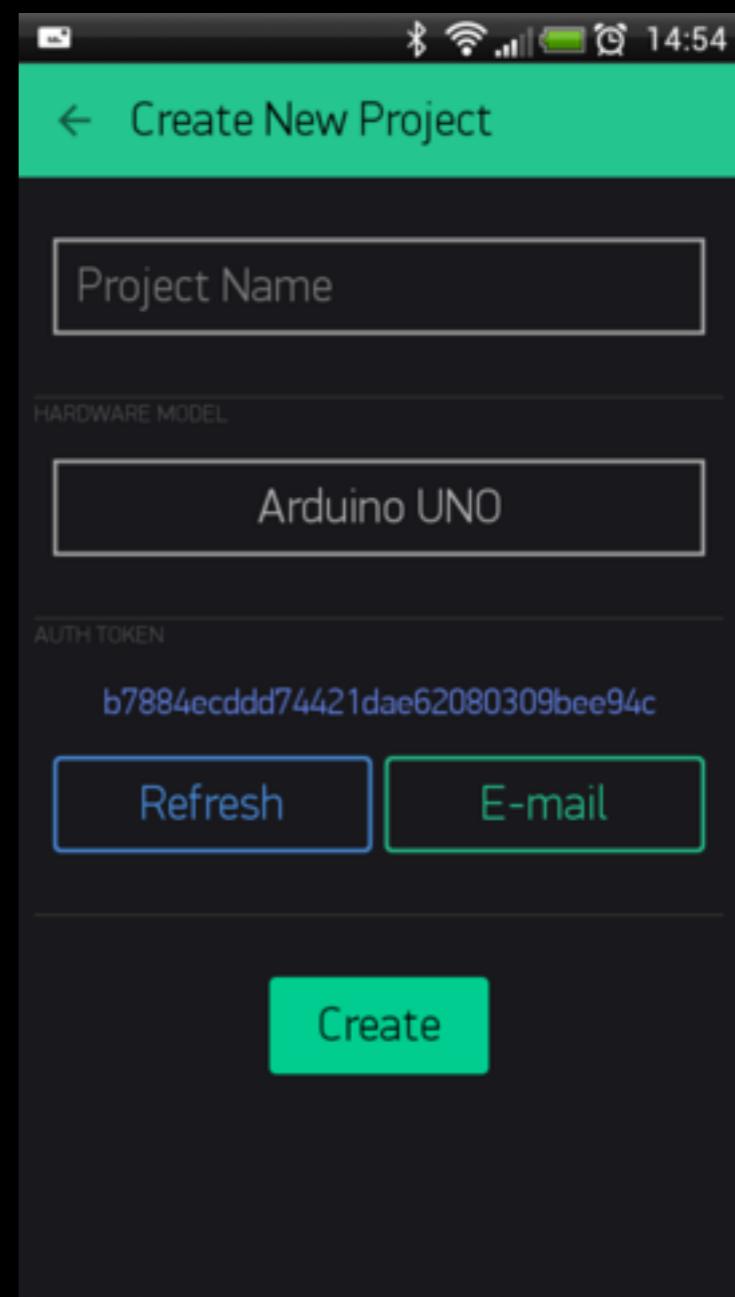
# Steps Blynk

## 2 . Create Blynk Account



# Steps Blynk

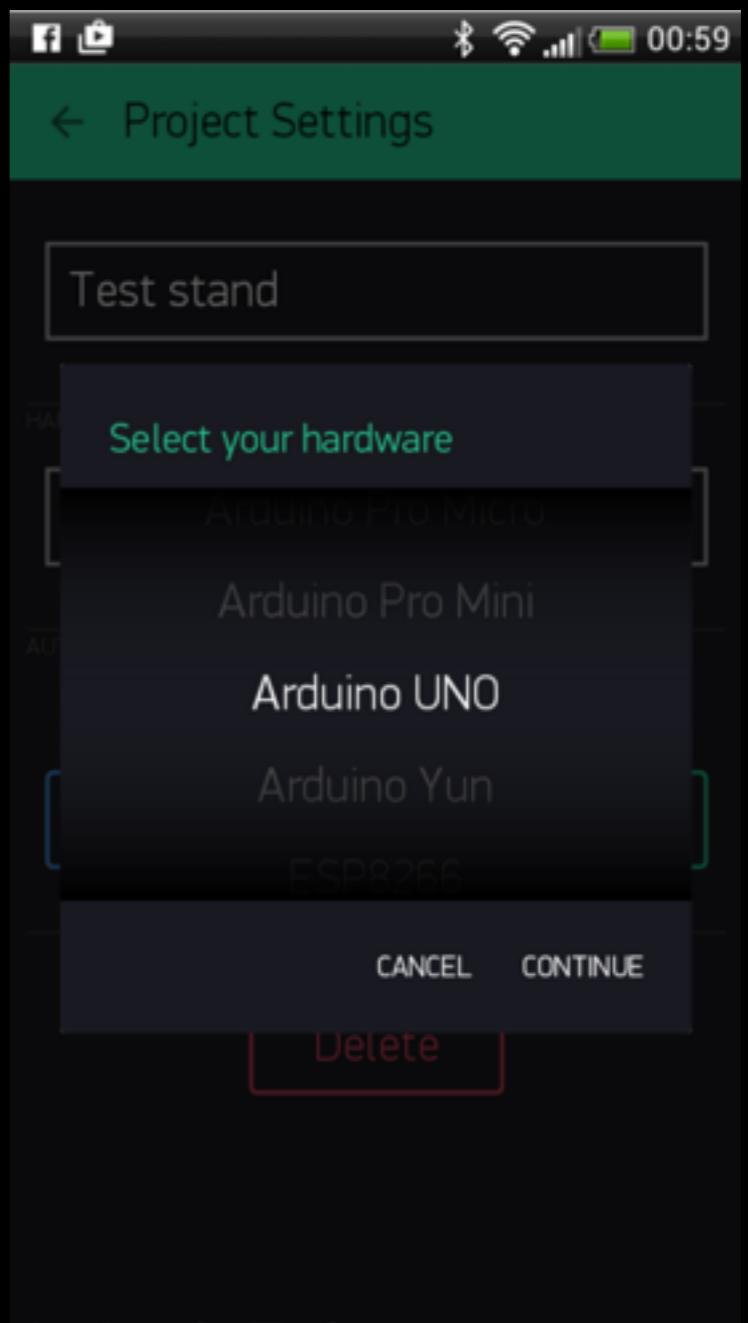
## 3 . Create new project



# Steps Blynk

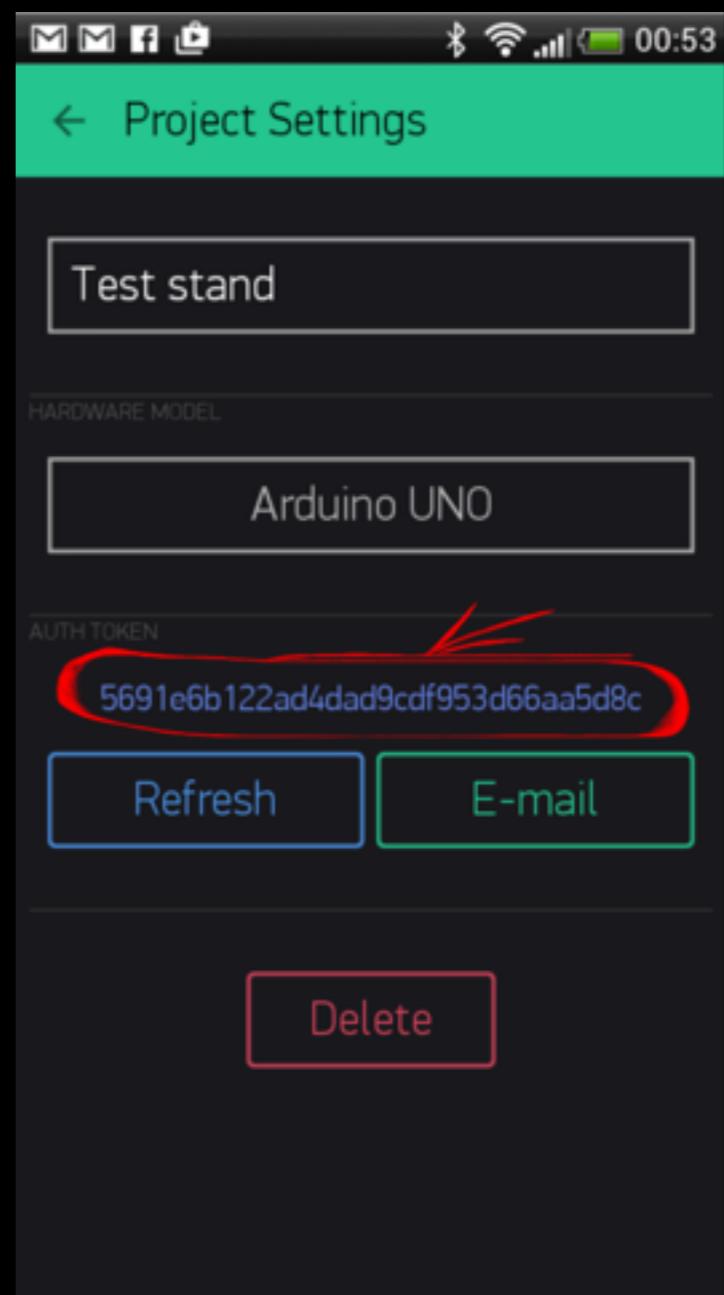
4 . Choose your hardware

1. Arduino Yun



# Steps Blynk

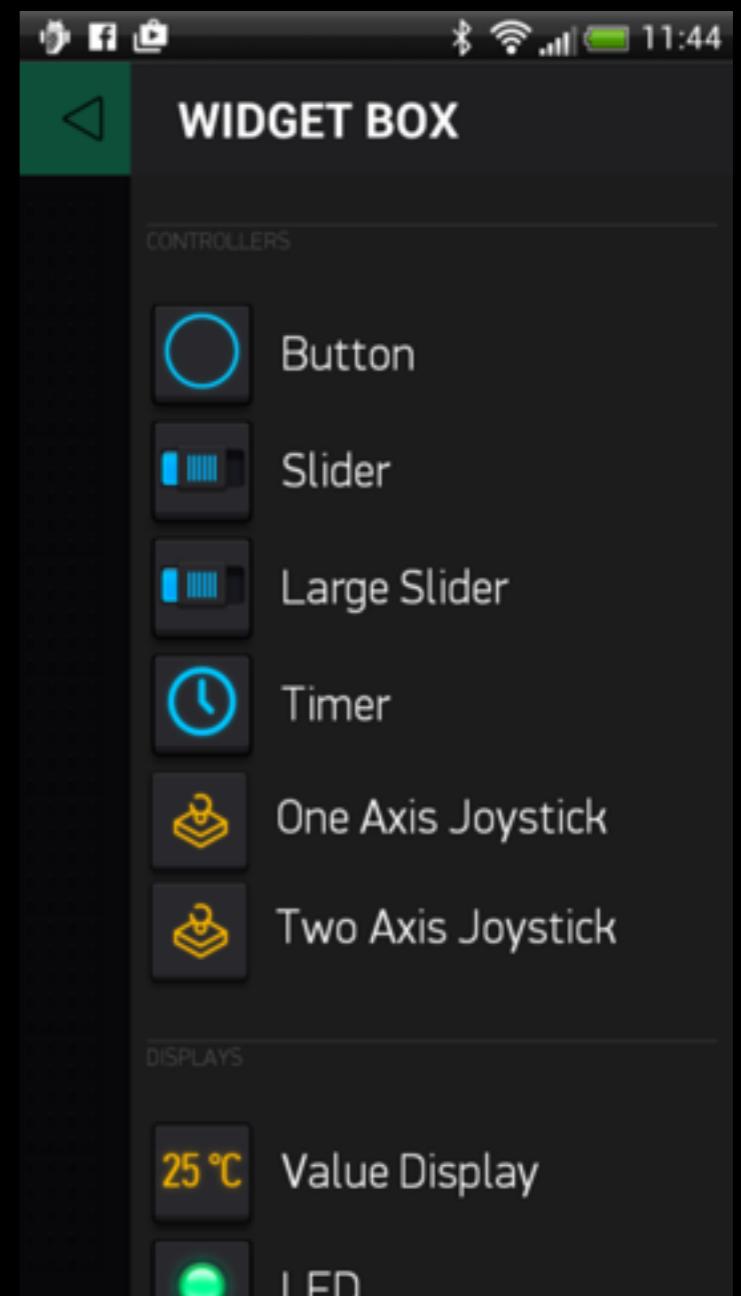
## 5 . Auth Token



# Steps Blynk

## 7 . Add a Widget

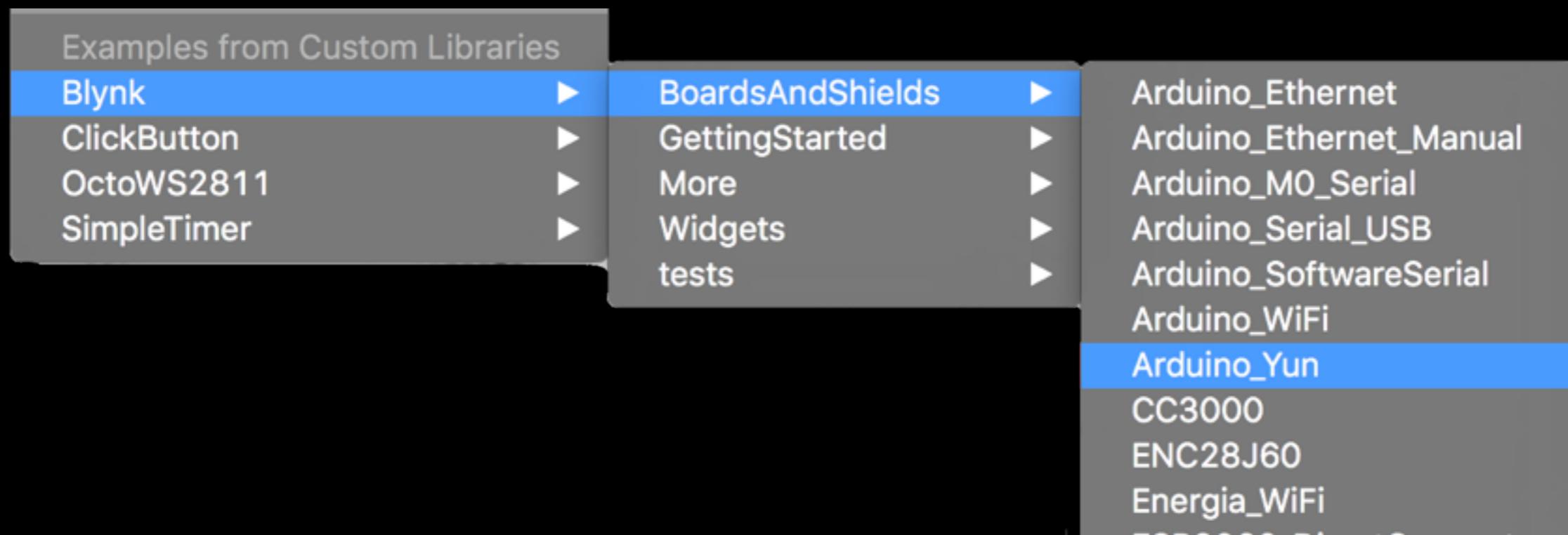
1. Drag-n-Drop Tap and hold the Button Widget to drag it to the new position.
2. Widget Settings Each Widget has it's own settings. Tap on the widget to get to them . Set D13



# Steps Code

1 . Open the Arduino Blynk Arduino Yun Example

1. File >> Examples >> Blynk >> BoardsAndShields  
>> Arduino\_Yun



# Steps Code

## 2 . Create new sketch

1. Copy example code

2. Auth Token



The screenshot shows the Arduino Yun software interface with the title bar "Arduino\_Yun | Arduino 1.6.6". The main window displays the Blynk example code. The code includes comments explaining the Blynk platform, its features, and how to use it with the Arduino Yun Bridge. It defines the BLYNK\_PRINT macro to point to the Serial port, includes the `<Bridge.h>` and `<BlynkSimpleYun.h>` headers, and sets an authentication token. The `setup()` and `loop()` functions are also shown.

```
/*
 * Blynk is a platform with iOS and Android apps to control
 * Arduino, Raspberry Pi and the likes over the Internet.
 * You can easily build graphic interfaces for all your
 * projects by simply dragging and dropping widgets.
 *
 * Downloads, docs, tutorials: http://www.blynk.cc
 * Blynk community: http://community.blynk.cc
 * Social networks: http://www.fb.com/blynkapp
 * http://twitter.com/blynk\_app
 *
 * Blynk library is licensed under MIT license
 * This example code is in public domain.
 *
 */
/*
 * This example shows how to use Arduino Yun Bridge
 * to connect your project to Blynk.
 * Feel free to apply it to any other example. It's simple!
 *
 */

#define BLYNK_PRINT Serial // Comment this out to disable prints and save space
#include <Bridge.h>
#include <BlynkSimpleYun.h>

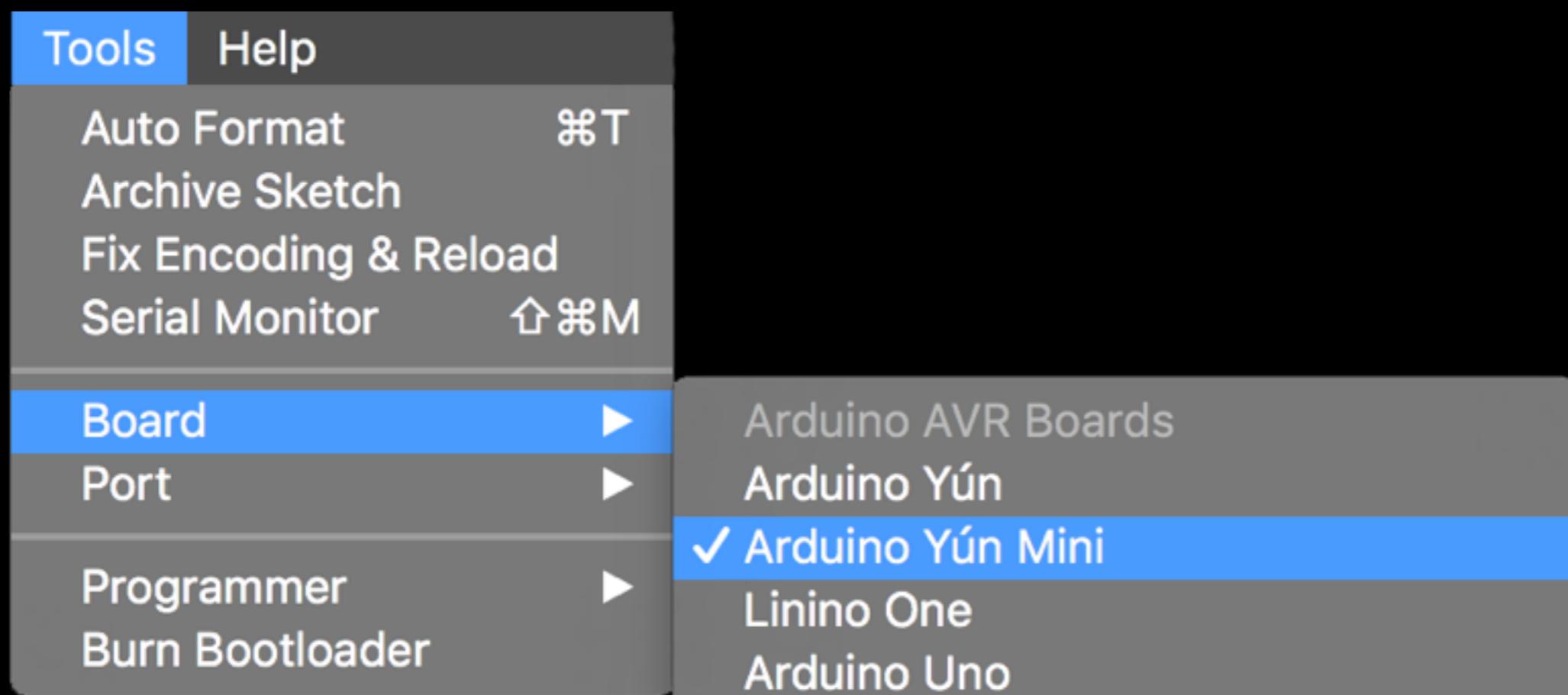
// You should get Auth Token in the Blynk App.
// Go to the Project Settings (nut icon).
char auth[] = "YourAuthToken";

void setup()
{
    Serial.begin(9600);
    Blynk.begin(auth);
    // Or specify server using one of those commands:
    //Blynk.begin(auth, "server.org", 8442);
    //Blynk.begin(auth, server_ip, port);
}

void loop()
{
    Blynk.run();
}
```

# Steps Code

## 3 . Locate the Arduino Board

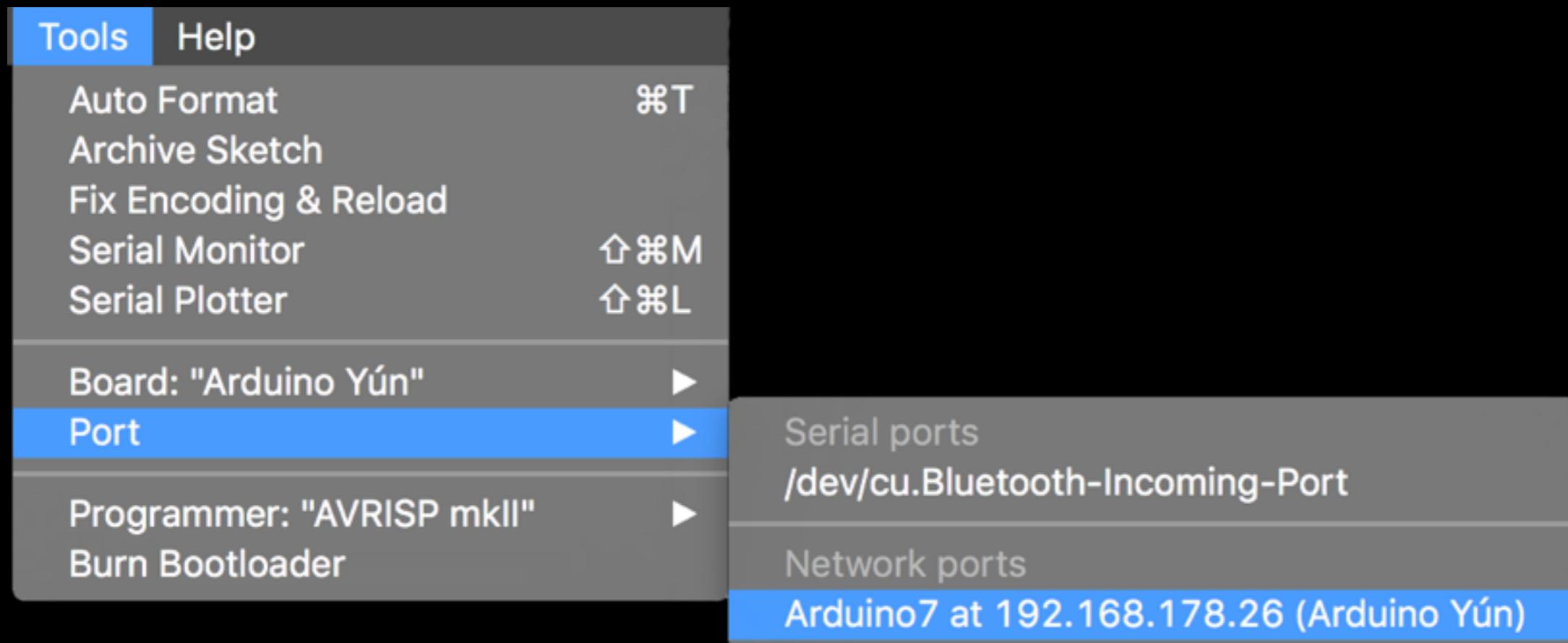


# Steps Code

4 . Locate your Arduino.

Name = "ArduinoN"

Password = "doghunter"



# Steps Code

# 5 . Upload Sketch

Arduino\_Yun | Arduino 1.6.6

Arduino\_Yun

```
* Blynk is a platform with iOS and Android apps to control
* Arduino, Raspberry Pi and the likes over the Internet.
* You can easily build graphic interfaces for all your
* projects by simply dragging and dropping widgets.
*
* Downloads, docs, tutorials: http://www.blynk.cc
* Blynk community: http://community.blynk.cc
* Social networks: http://www.fb.com/blynkkapp
* http://twitter.com/blynk\_app
*
* Blynk library is licensed under MIT license
* This example code is in public domain.
*
*****
* This example shows how to use Arduino Yun Bridge
* to connect your project to Blynk.
* Feel free to apply it to any other example. It's simple!
*
****/
```

```
#define BLYNK_PRINT Serial // Comment this out to disable prints and save space
#include <Bridge.h>
#include <BlynkSimpleYun.h>

// You should get Auth Token in the Blynk App.
// Go to the Project Settings (nut icon).
char auth[] = "YourAuthToken";

void setup()
{
    Serial.begin(9600);
    Blynk.begin(auth);
    // Or specify server using one of those commands:
    //Blynk.begin(auth, "server.org", 8442);
    //Blynk.begin(auth, server_ip, port);
```

Uploading...

```
avrduke: 1000 data erase data from input file more:
avrduke: input file @xFB contains 1 bytes
avrduke: reading on-chip efuse data:
```

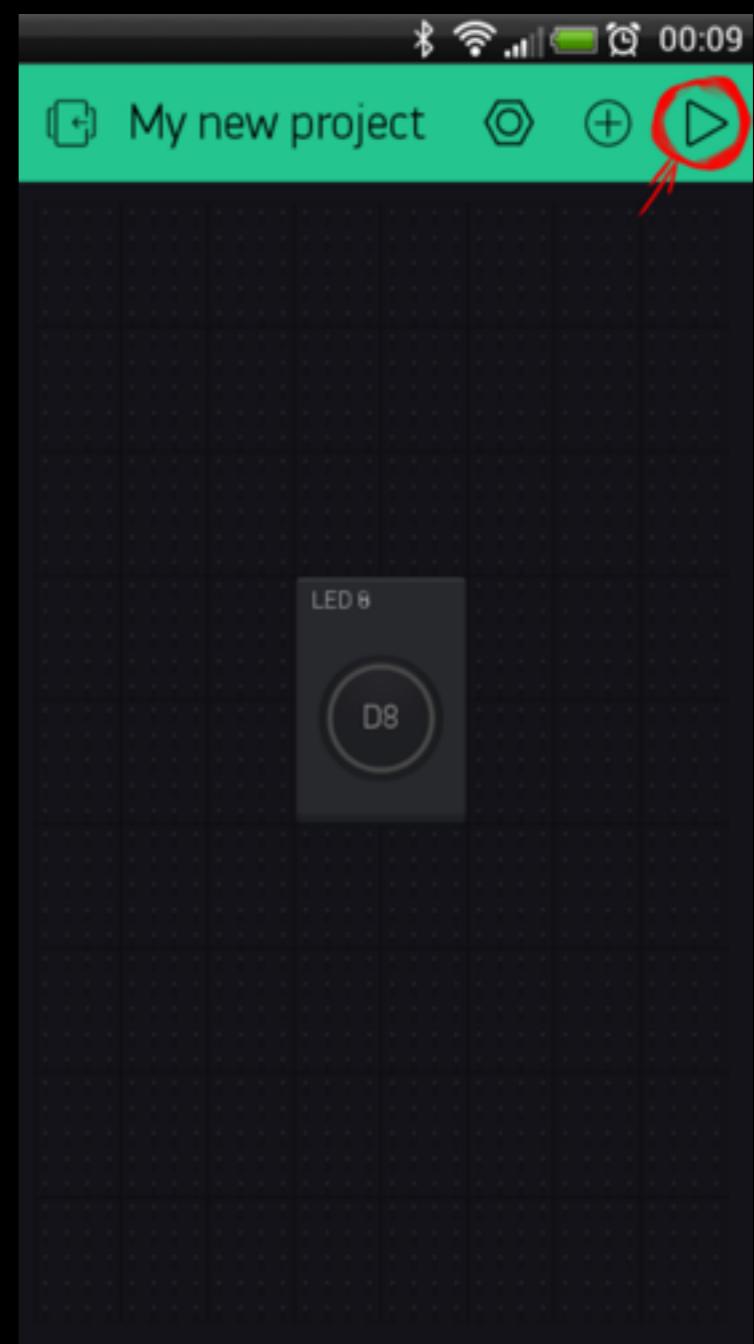
Reading I ████████████████████ | 100% 0.00s

```
avrduke: verifying ...
avrduke: 1 bytes of efuse verified
avrduke: reading input file "/tmp/sketch.hex"
avrduke: writing Flash (32748 bytes):
```

Writing I ████████████████████

# Steps Code

6 . Run the Blynk project



# Questions?

Imanol Gómez

[imanolgomez.net](http://imanolgomez.net)

[yo@imanolgomez.net](mailto:yo@imanolgomez.net)