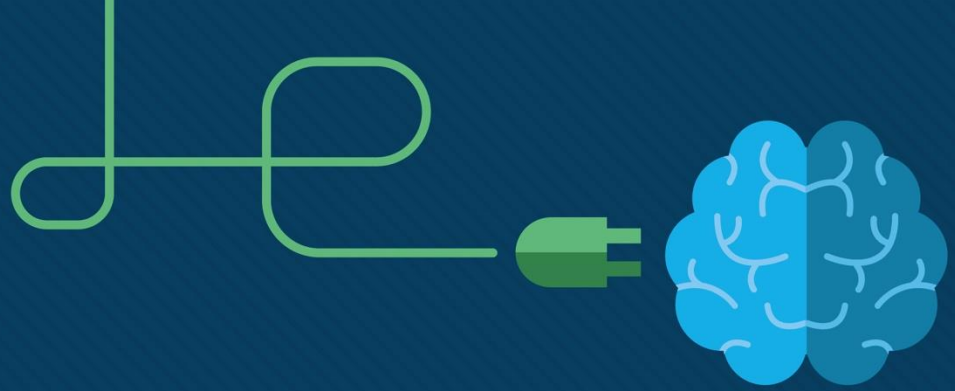


IoT Prototyping

Cisco PL-App Workshop

Eugene Morozov
Technical Manager CEE-RCIS, N&B

22 September 2017
Aachen

GFO TFE Team of Technical Managers

Karen Alderson

Echo Rantanen
US-Canada

Semyon Ovsyannikov
UKI, Europe South, North (BeNeLux)

Eugene Morozov
Europe Central, North (Nordics/Baltics)

David Lee
Greater China

Karina Butron
Mexico

LATAM

Julien Berton
France

Marc Khayat
TMENA

Jose Esquivel

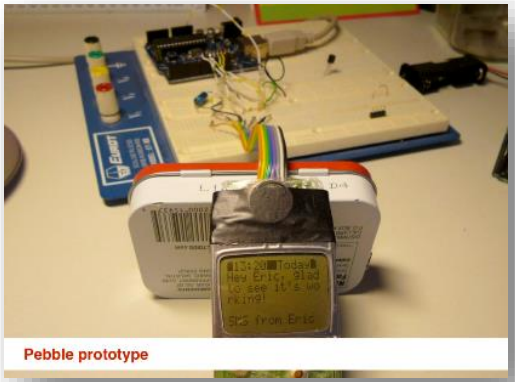
Gabriela Neira,
Associate

Serges Nanfack
Sub-Sahara

Ananth
B Shankar Rao
APAC/Japan

Eric Migicovsky was 26 years old and has a vision: creating Pebble!

What is Pebble: *"It is a smart watch, something that connects to your phone and brings some of the functionality of your phone to your wrist"*



With his team he created a prototype with pieces of Old Nokia Phone and and some open hardware.

On April, 2012 he was looking for 100.000$ to realize his idea.
Six VC and Investors put him down.

So he created this video


Привет, Kickstarter. Меня зовут Эрик.

KICKSTARTER | What is Kickstarter? | Discover great projects | Start your project | Search projects | Help | Me

Pebble: E-Paper Watch for iPhone and Android

by Pebble Technology

Home | Updates 37 | Backers 68,929 | Comments 14,752 | Palo Alto, CA | Product Design

Funded! This project successfully raised its funding goal on May 19, 2012.

68,929
backers

$10,266,845
pledged of $100,000 goal

0
seconds to go

Like 121,502 people like this. Be the first of your friends. | Tweet | Embed | http://kck.st/HumlV5

Project by
Pebble Technology
Palo Alto, CA
Contact me

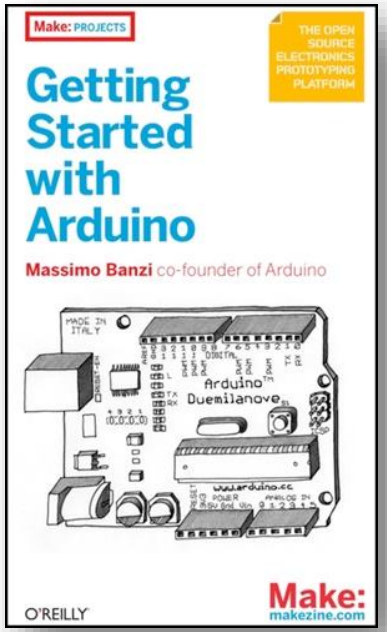
First created · 28 backed

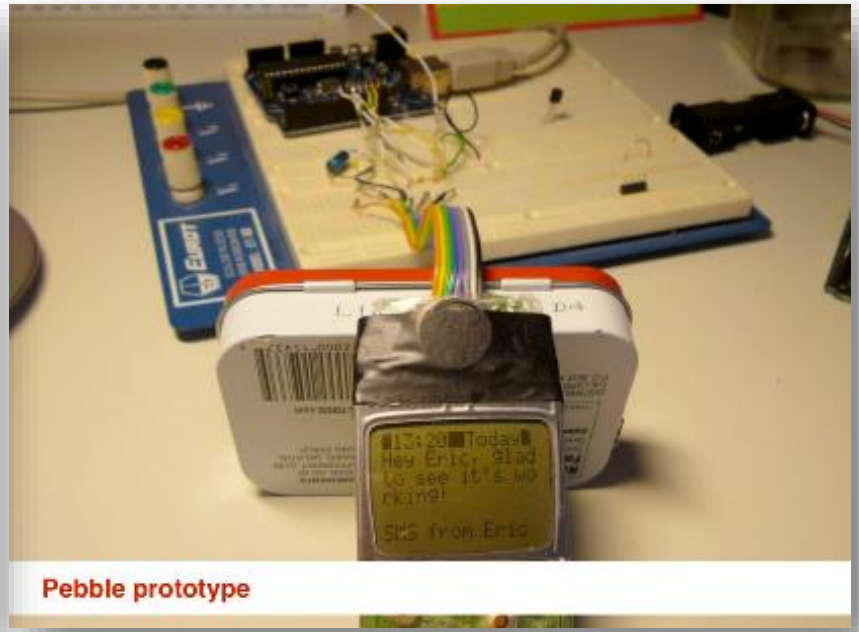
Eric Migicovsky 720 friends

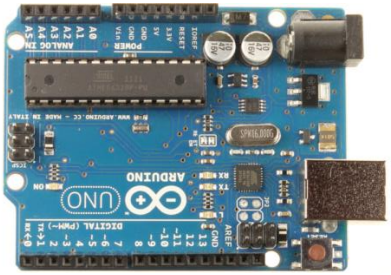
Website: getpebble.com

Arduino

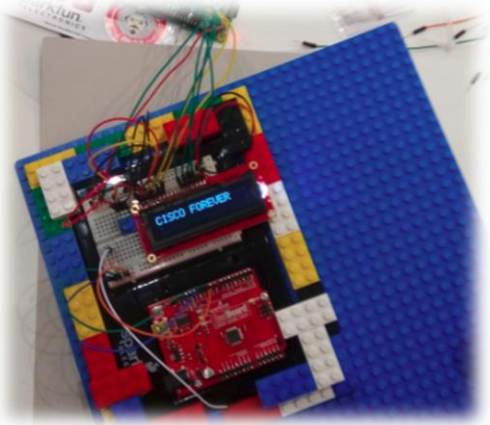
The prototype of Pebble was done using an Open Hardware project named “Arduino”





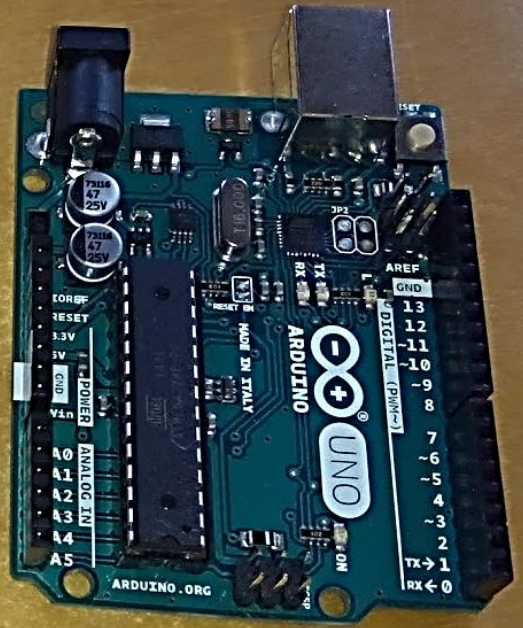


What can you create with Arduino?

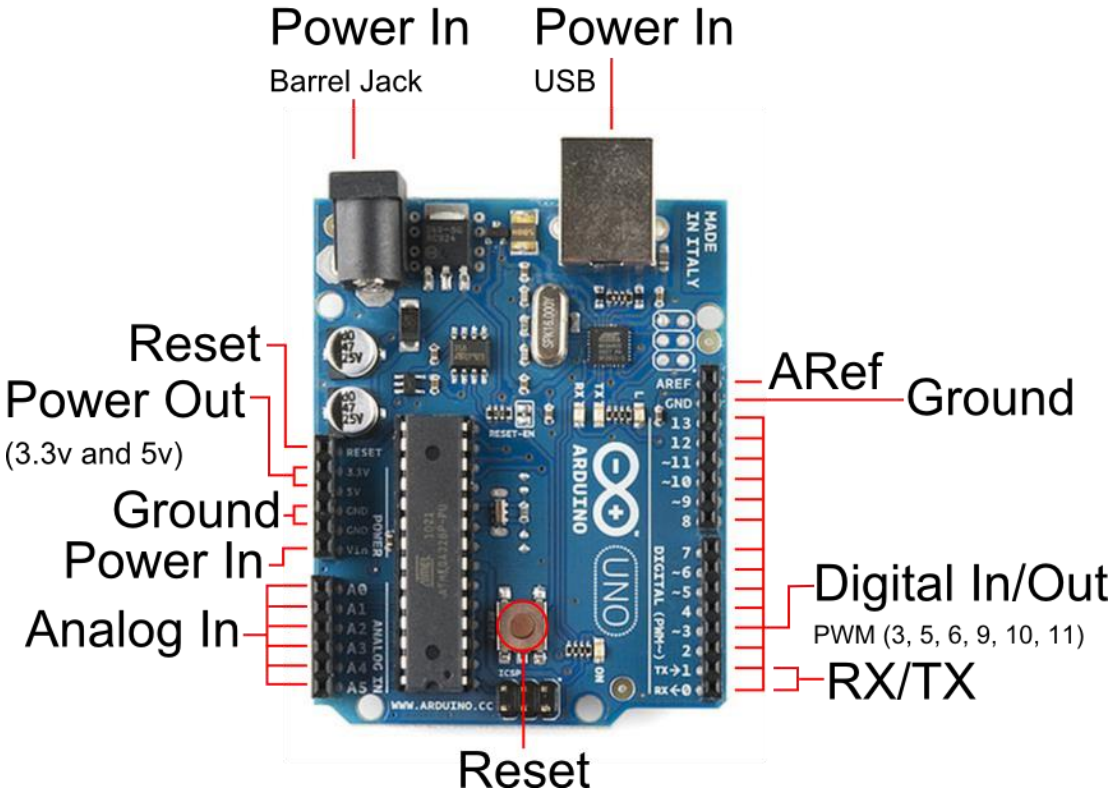


Prototyping

Arduino


ARDUINO
UNO
MADE IN ITALY
DIGITAL (PWM~)
POWER
ANALOG IN
AREF
GND
ARDUINO.ORG

Arduino pins



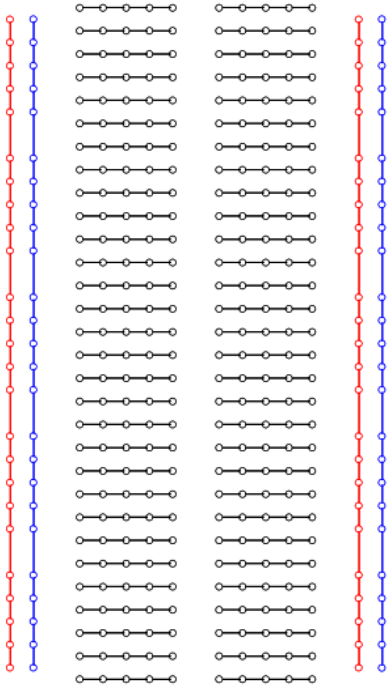
Arduino Options

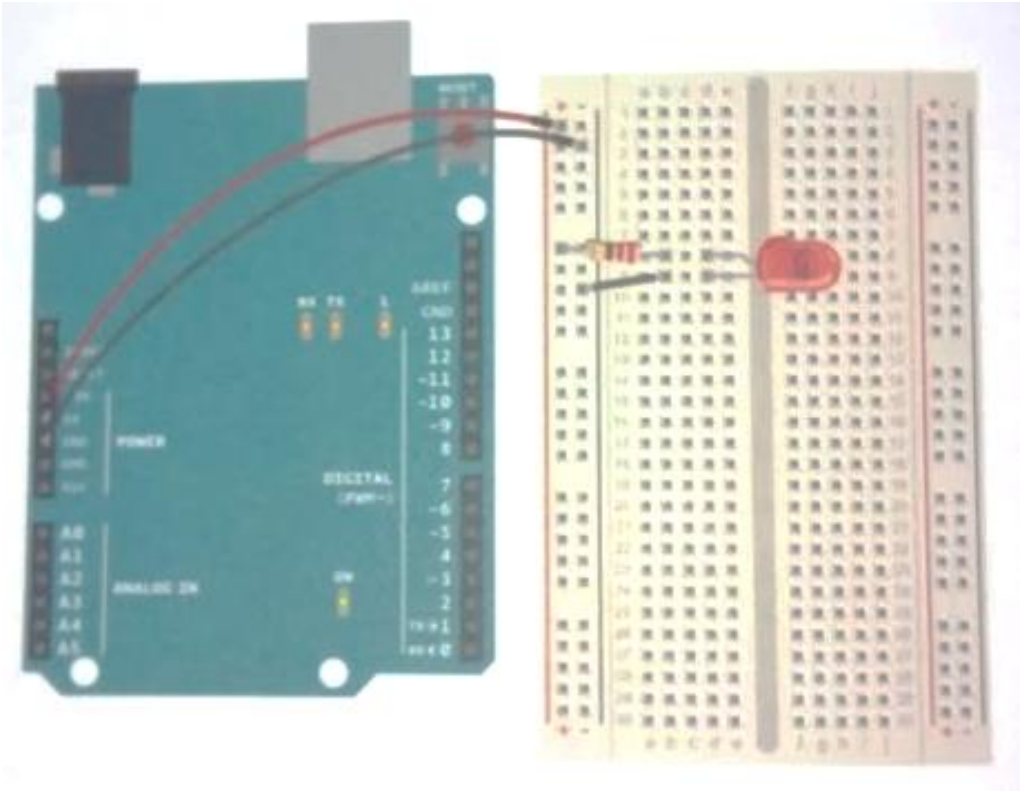
- Arduino Original 20 EUR+VAT
- Arduino clones from $5
- Arduino Starter Kit 80 EUR+VAT

Breadboard

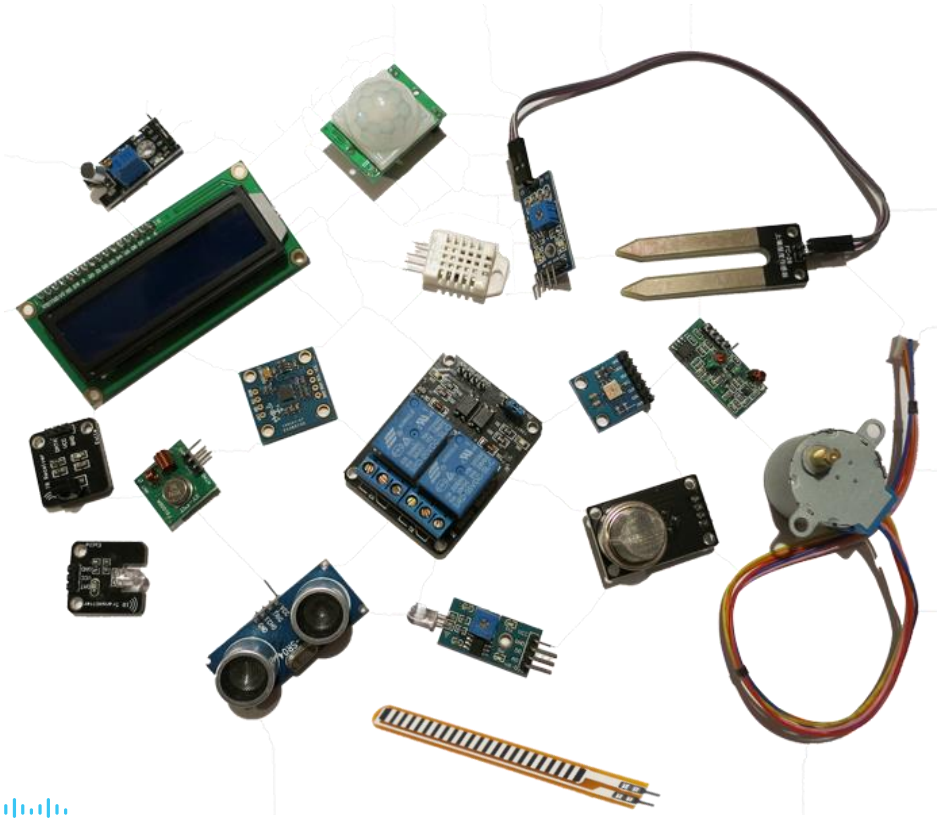

a b c d e
f g h i j
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30

Breadboard





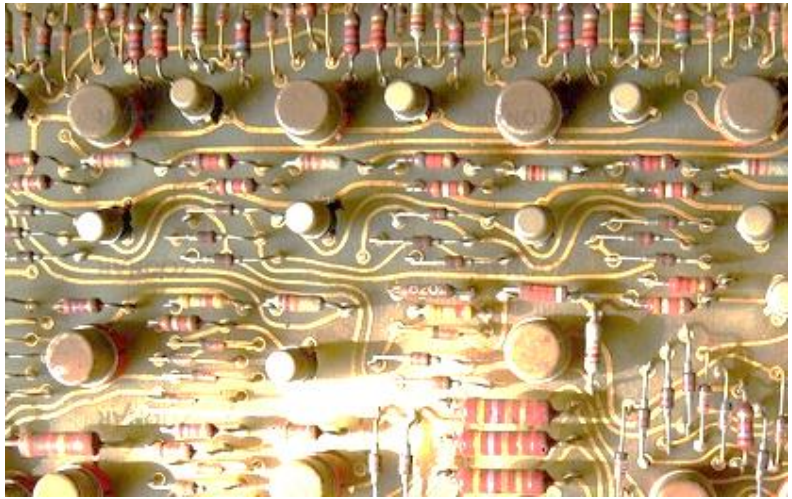
Sensors and Actuators



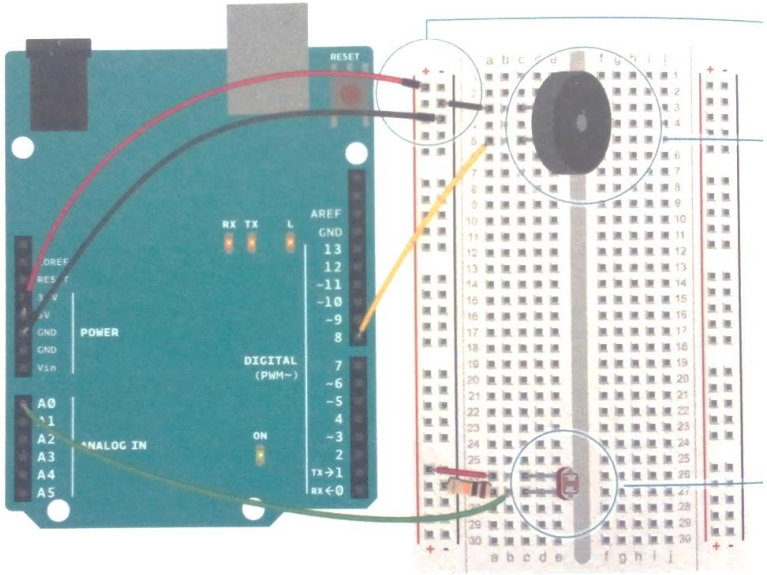
- Photo
- Temperature
- Humidity
- Moisture
- Distance
- Movement and bend
- Push and tilt

- Light
- Motor
- Servo
- Sound
- LCD screen
- Hydro and Pneumo

Amateur Electronics 2.0



vs



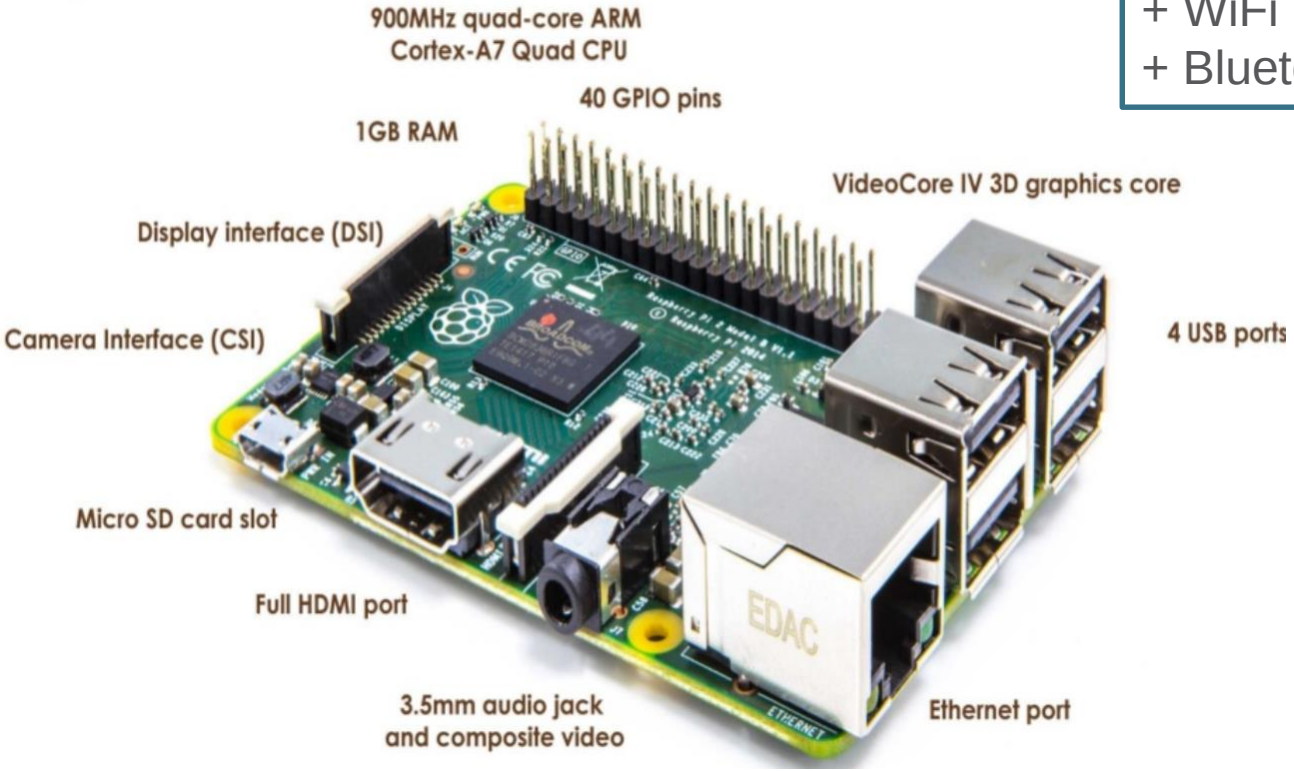
Circuit: soldered
Logic: hardwired

Circuit: changeable
Logic: software-based

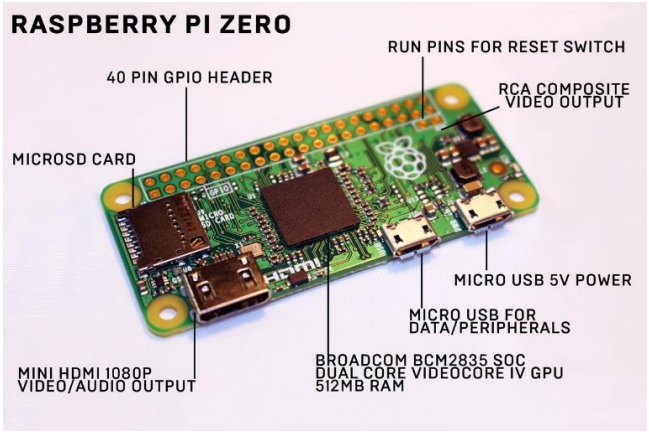
Raspberry Pi

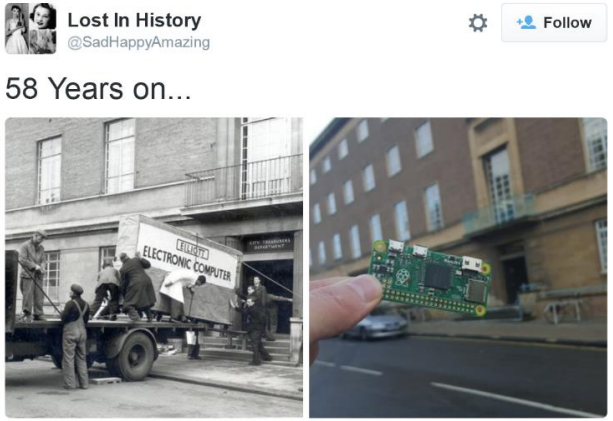
Raspberry Pi 2 Model B

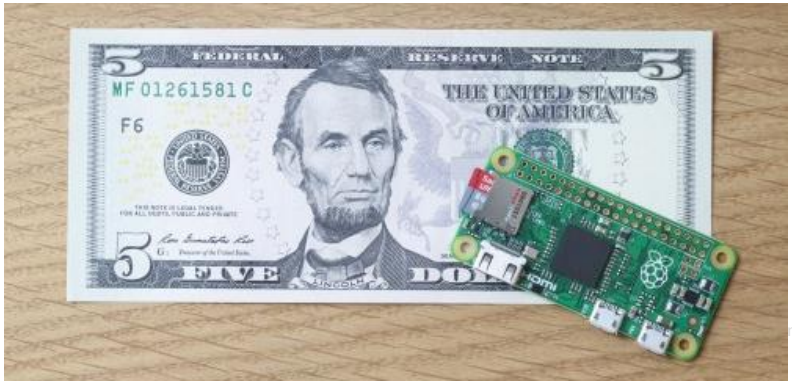
Raspberry Pi 3:
+ faster
+ WiFi
+ Bluetooth



Raspberry Pi Zero – a $5 computer



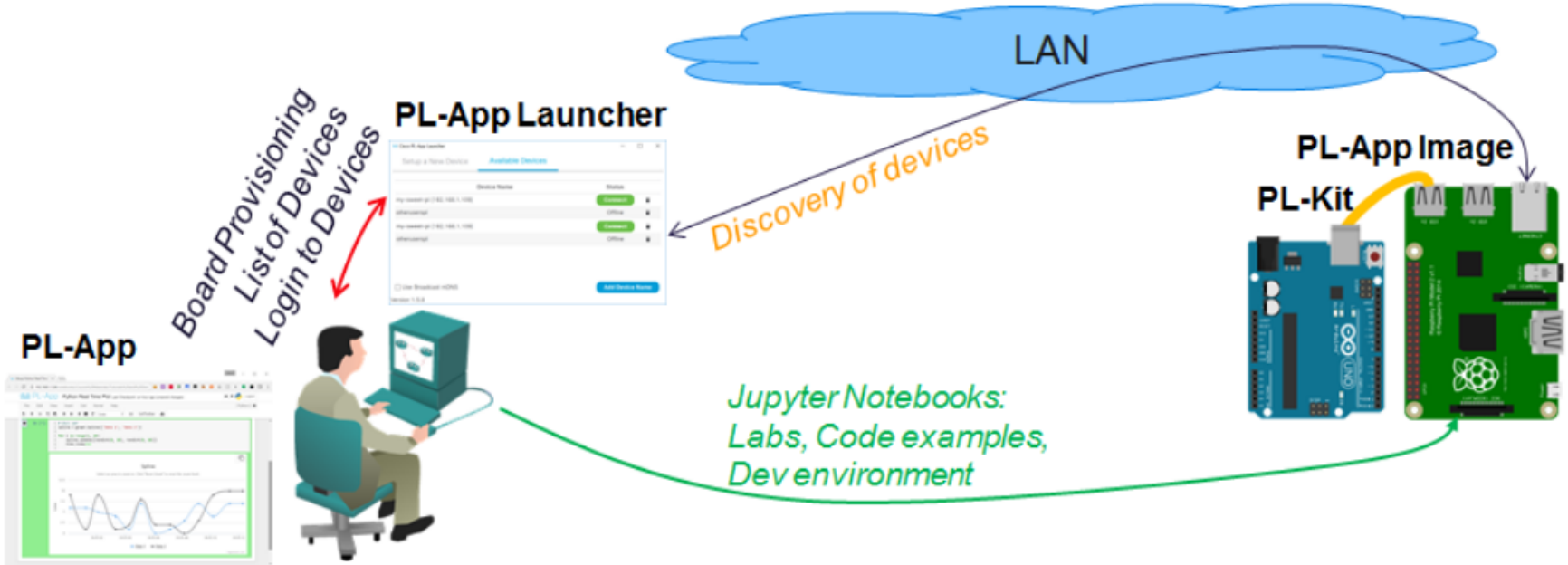




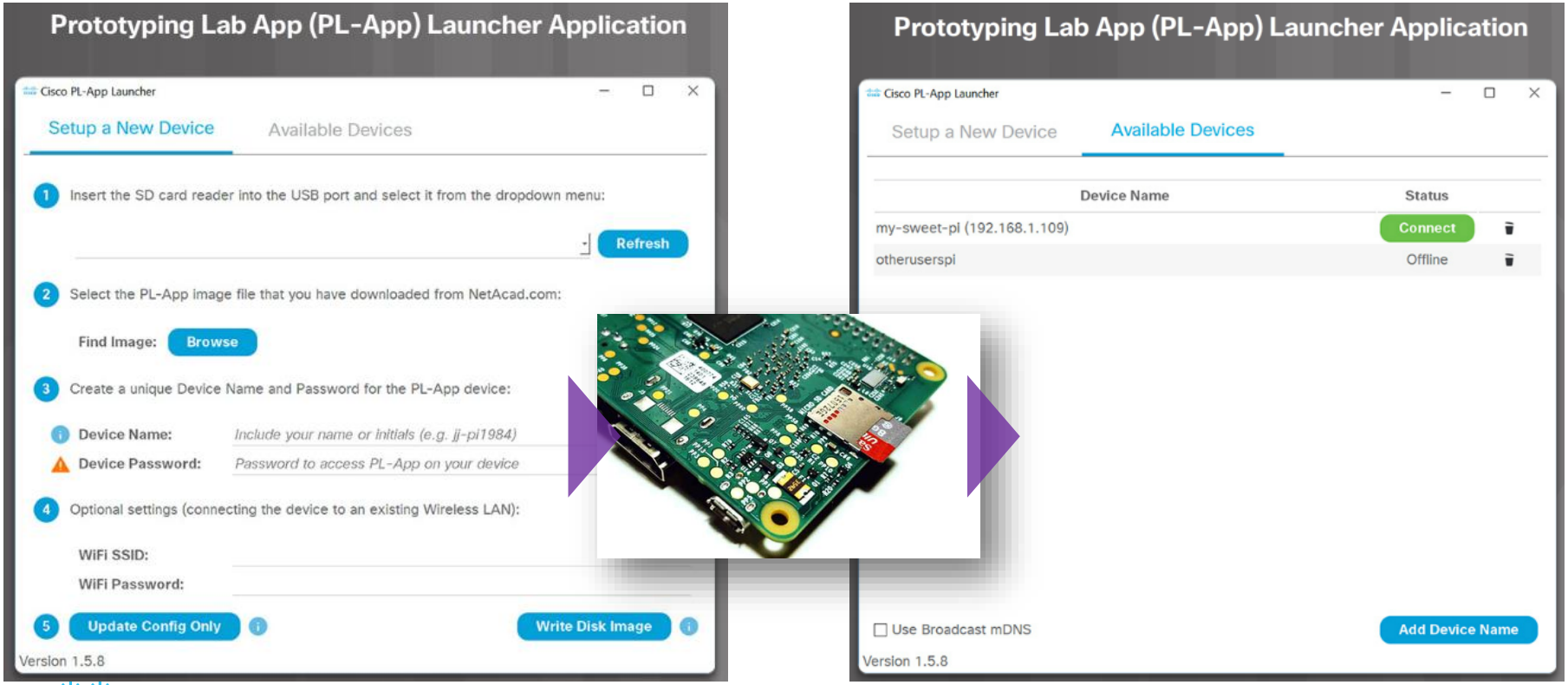
PL-App

Prototyping Lab App

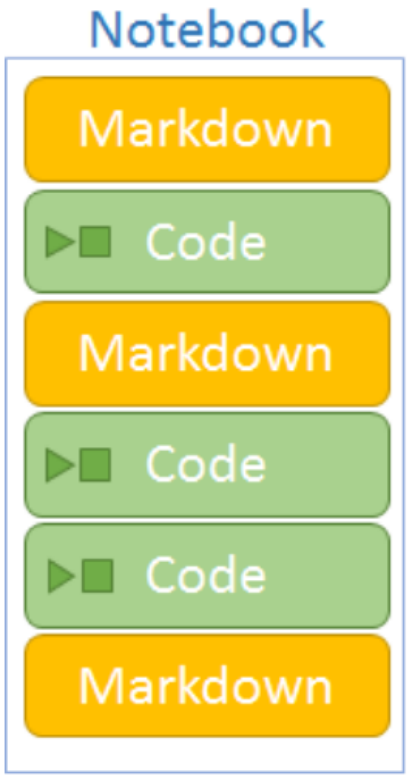
- Standard tool used in IoT Fundamentals courses



PL-App Launcher

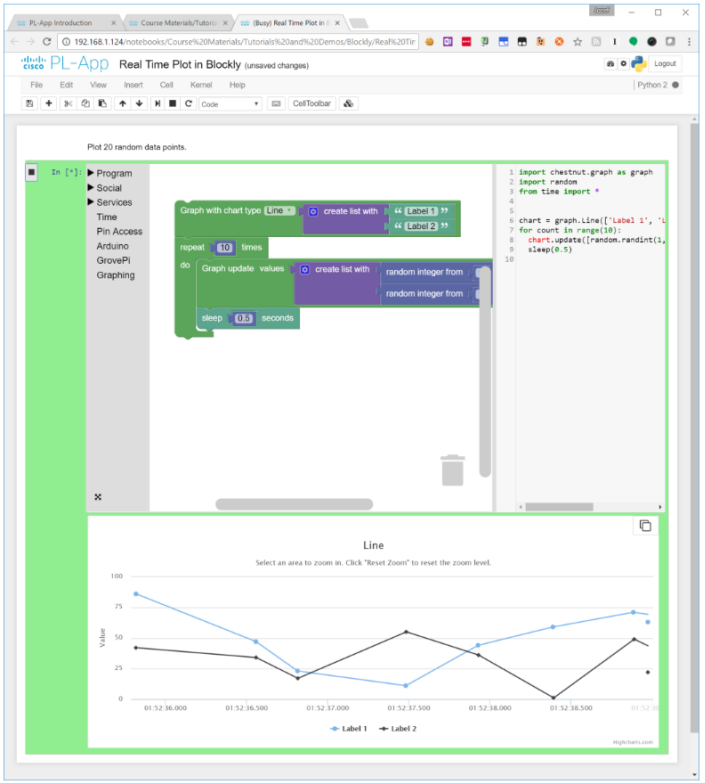


PL-App Notebooks



Combines:

- Explanatory text, graphics, media
- Programming code in Python, Bash or Blockly
- Data visualization

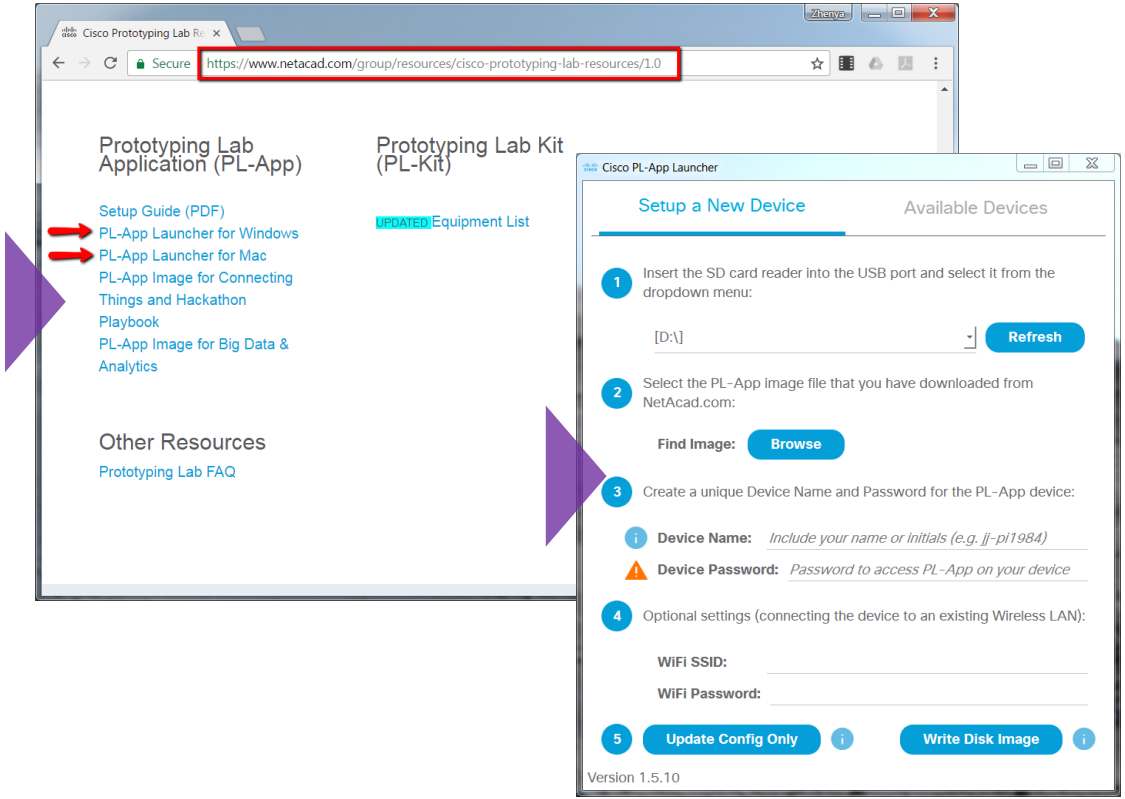


Step 1:
Preparing the Platform

Download and install PL-App Launcher

- Connect your laptop to the WiFi network
  - SSID: netacad
  - Password: workshop
- Go to NetAcad.com → Resources → Course Resources → IoT Fundamentals: Connecting Things →

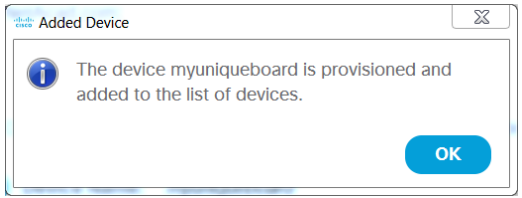
  PL-App Launcher and Image →
- Download, install and launch PL-App Launcher

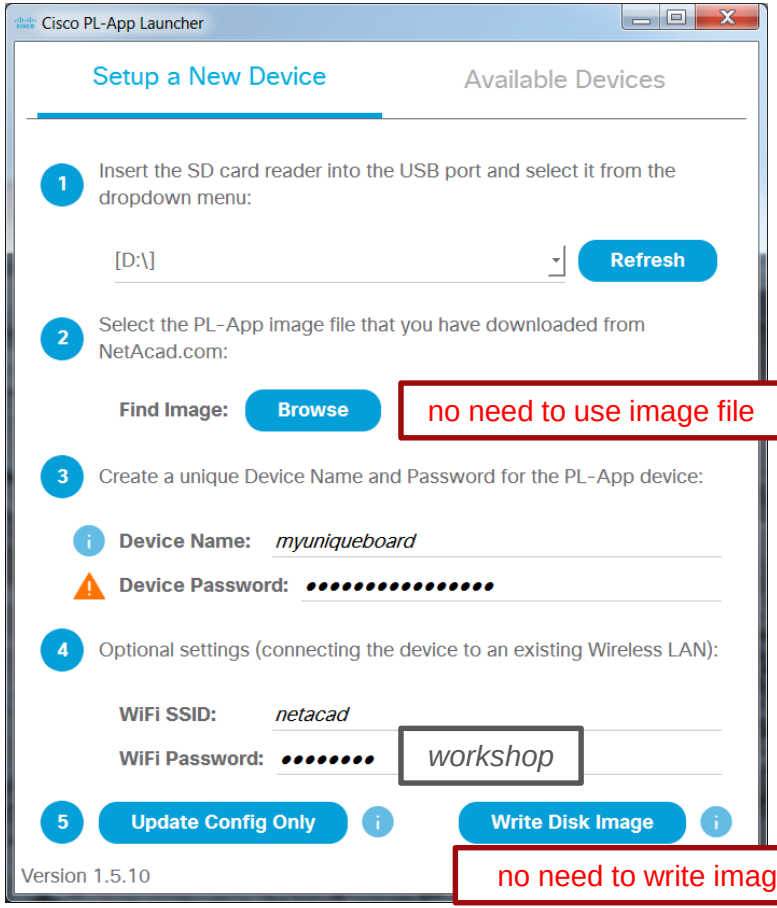


Configure PL-App

Your SD card is already flashed with PL-App image. You only need to configure it.

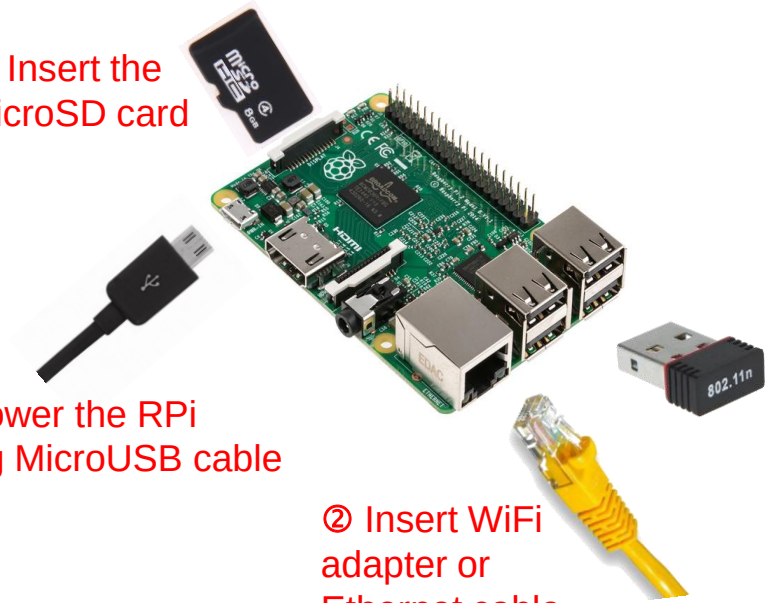
- Insert microSD card into your laptop using an appropriate adapter
- Select it from the dropdown menu 1
- Give your device a unique name and password 3
- Input WiFi SSID and password, if wireless network is used 4
- Click **Update Config Only** to update configuration without reimaging the SD card
- Configuration will be updated:

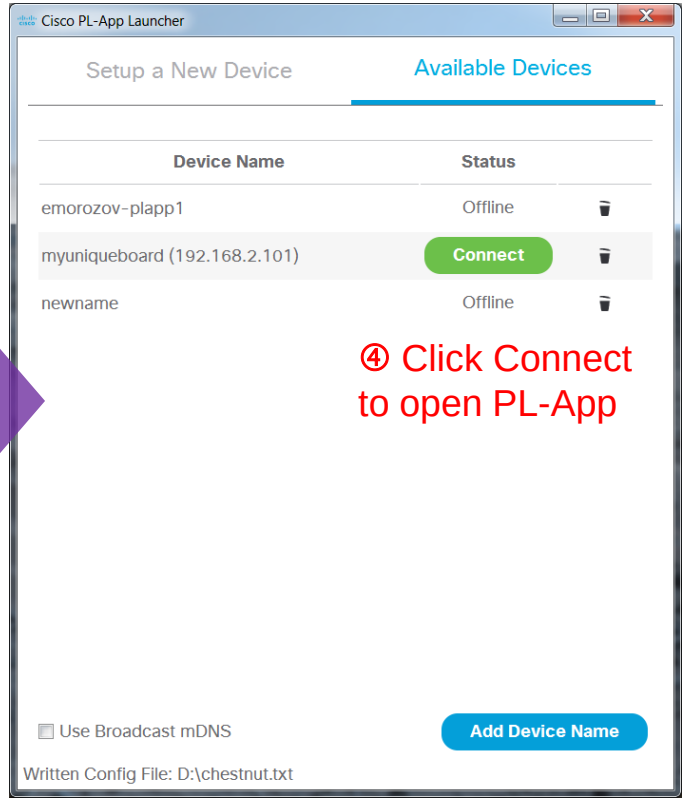




Connect and start your Raspberry Pi

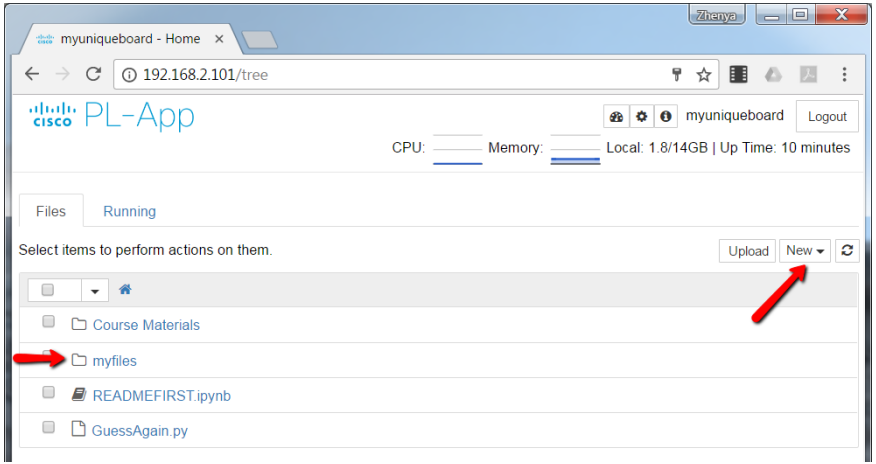
- Safely remove SD card from your laptop

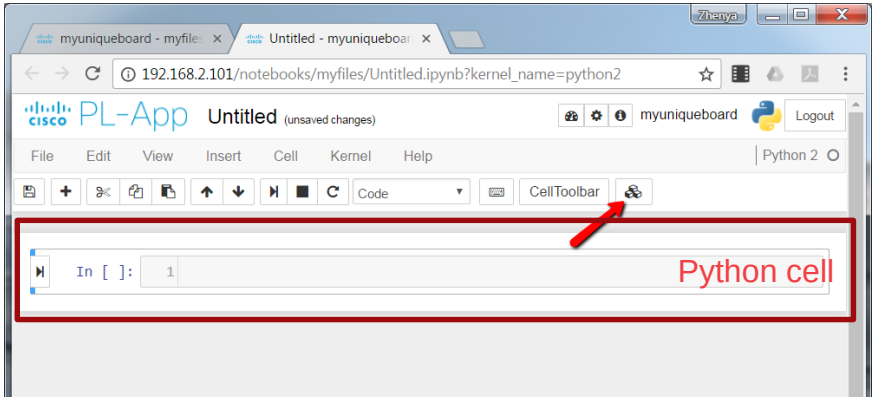




Create a project file

- Google Chrome is the best supported browser for PL-App at the moment
- Log into your board with the password you created
- Go to ‘myfiles’ folder
- Create a new notebook with Python 2
- A new notebook will open with a Python cell in it
- Click the small Blockly  icon to convert the cell into Blockly visual code
- We will use this cell to flash Firmata firmware into Arduino





Prepare your Arduino

In order to communicate with Raspberry Pi, Arduino should be flashed with a special firmware called Firmata

- Build the code to flash Arduino in the cell you have created
- Drag and snap Blockly instructions
- flash Firmata to Ardu instruction located under “Arduino” category
- The port should be **/dev/ttyACM0** or **/dev/ttyUSB0** depending on your hardware
- Connect Arduino to Raspberry Pi with USB cable
- Run the cell with ⏭ button

④ Run the cell

② Build the code to flash Arduino

① Grab instructions from here

③ Note a Python code is generated on the go

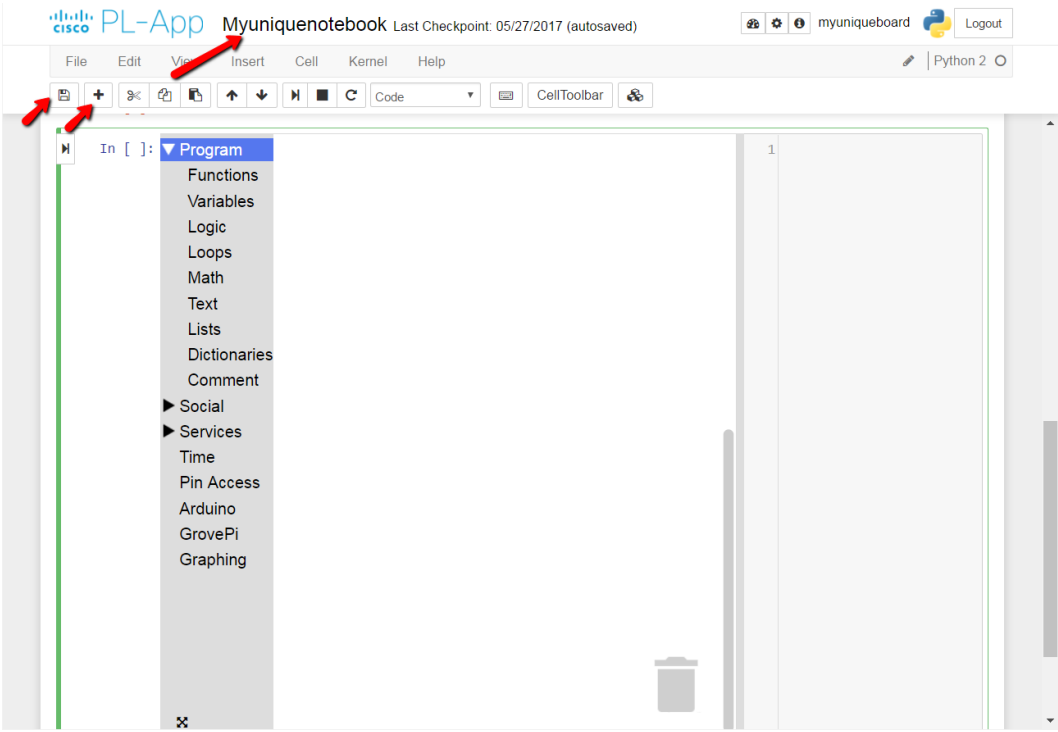
```
Welcome to PL-App
will flash Firmata now...
Flashing Standard Firmata to a USB-connected Arduino. Please wait...
Done flashing.
Out[2]: 0
```

Arduino is flashed

Get ready to build your project

Arduino has to be flashed only once. There is no need to flash it each time you run your project. We will use another cell to build your project.

- Click the + button to add new cell and make it Blockly cell
- Rename your notebook to give it a unique name
- Save your notebook



Step 2:
Creating a project

Project – Door monitoring

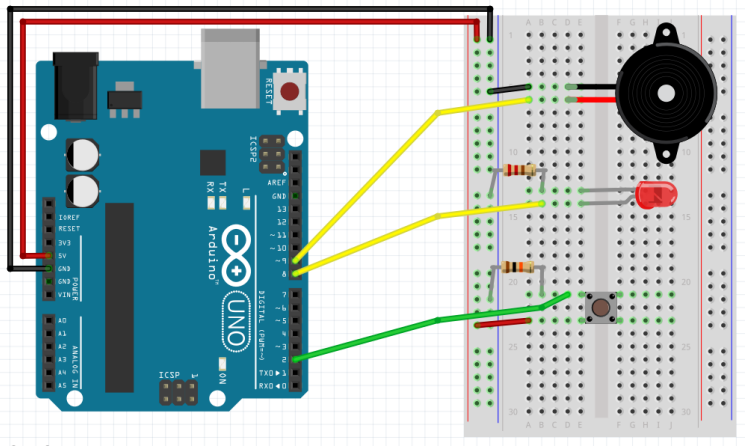
Monitor door using a simulation push-button

While door is closed:

- nothing happens

When door is opened:

- Turn alarm light on
- Sound annoying buzzer alarm
- Send twitter notification



Project – Code

Arduino has to be initialized

```
print "Welcome to PL-App"
connect Arduino on port "/dev/ttyACM0"
Twitter setup
Consumer key "oPKkox9HAiTl6Sdlu0bx2WG0q"
Consumer secret "Uko2Btes3Vpe7PWOeQfW12kTcTP3eU1eiWacsOko72E3PJhLgO"
Access token "734847317069733888-4Qv3j2M0Iyn6iTWfgya3VLiPvP8bQxP"
Access token secret "OpiffppinmDGu0E2h4BEfrouhawcDREX8XbWfg94Xd1Rr"
set Arduino pin 2 to INPUT mode
set Arduino pin 8 to OUTPUT mode
set Arduino pin 9 to PWM mode
set alert to false
print "Door secured"
repeat while alert = false
do
  sleep 1 seconds
  if digital read on pin 2
  do
    set alert to true
    digital write to pin 8 value 1
    analog write to pin 9 value 20
    print "ALARM! Door opened!"
    twitterSend
```

```
analog read on pin 0
```

Weird command (beta issue)

Connect to twitter account. You can follow it with @ioeprototyping

Initialize Arduino pins

A variable used

```
to twitterSend
Post on Twitter
Message  create text with "Team # Secure door has been opened at:"
                          seconds since start
                          "seconds"
print "Alarm notification sent"
```

Put your team number instead of #, add your city in the tweet

Try to build your project without twitter notifications and verify it is working. You can add twitter feature then.

Connect laptop:
SSID: netacad
Password: workshop

Let’s go!

A New NetAcad Hands-On Experience

IoT Fundamentals | Lab Experiences

Analyze the Problem with User Focus

Hands-on Design and Maker Mindset

Rapid Prototyping, Iterating, Presenting

The Networking Academy Learning Portfolio

Current & Planned

Certification-aligned

* Available within 12 months

Collaborate for Impact

Packet Tracer Know How | Packet Tracer | Hackathons | Regional IT Competitions | NetRiders | Internships

	Exploratory	Foundational	Career-Ready
Networking		Networking Essentials Mobility Fundamentals	CCNA R&S: Intro to Networks, Scaling Networks; R&S Essentials, Connecting Networks CCNP R&S: Switch, Route, TShoot
Security	Introduction to Cybersecurity	Cybersecurity Essentials	CCNA Security CCNA Cyber Ops*
IoT	Introduction to IoT	IoT Fundamentals Connecting Things* Big Data and Analytics* Hackathon Playbook*	
OS & IT	NDG Linux Unhatched	NDG Linux Essentials IT Essentials	NDG Linux I NDG Linux II
Programming		Programming Essentials in C Programming Essentials in C++ Python*	Programming in C* Programming in C++*
Business	Be Your Own Boss	Entrepreneurship	
Digital Literacy	Get Connected		

February 2017

IoT Fundamentals Course Summary

	Course Overview	Benefits	
Connecting Things	Students learn how to securely interconnect sensors, actuators, microcontrollers, single-board computers, and cloud services over IP networks to create an end-to-end IoT system.	Students will develop multi-disciplinary skillsets required to prototype an IoT solution for a specific business case with a strong focus on the security considerations for emerging technologies.	Course Delivery: Instructor-led Estimated Time to Complete: 40-50 hours
Big Data & Analytics	Students will learn how to use Python data libraries to create a pipeline to acquire, transform and visualize data collected from IoT sensors and machines.	The transformative element of any IoT system is the data that can be collected from it. Thus the ability to extract data and using data analytics techniques to gain insights increases employability.	Course Delivery: Instructor-led Estimated Time to Complete: 40-50 hours
Hackathon Playbook	The Hackathon Playbook is a comprehensive framework of tools and templates to prepare and run a Hackathon as a result of best practices and lessons-learned collected from the global execution of IoT Hackathons within Networking Academy and by other organizers.	Student reinforce and deepen their multidisciplinary IoT and data skills by defining, designing, prototyping and presenting an IoT solution to a panel of industry experts and peers.	Course Delivery: Instructor-led Estimated Time to Complete: 20-30 hours

IoT Fundamentals: Connecting Things

Course Overview

Students learn how to securely interconnect sensors, actuators, microcontrollers, single-board computers, and cloud services over IP networks to create an end-to-end IoT system.

Benefits

Students will develop multi-disciplinary skillsets required to prototype an IoT solution for a specific business case with a strong focus on the security considerations for emerging technologies.

Learning Components

- Understand and explain the concepts, opportunities and challenges of digital transformation using IoT.
- Interconnect sensors/actuators, microcontrollers (Arduino), Single Board Computers (Raspberry Pi) and cloud services (Cisco Spark restful API) to create an end-to-end IoT system.
- Understand the relevant aspects of cybersecurity and privacy for an IoT solution.
- Understand how digitalization is changing vertical markets such as manufacturing, energy, and smart cars.
- Use simulation tools (Packet Tracer) to create end-to-end IoT system.

Features

Target Audience: Secondary, Vocational, 2-year and 4-year College, 4-Year University students

Prerequisites: Basic programming, networking and electronics

Languages: English

Course Delivery: Instructor-led

Estimated Time to Complete: 40-50 hours

Recommended Next Course: IoT Fundamentals: Big Data & Analytics or Hackathon Playbook

Instructor Training: Required

IoT Fundamentals: Big Data & Analytics

Course Overview

Students will learn how to use Python data libraries to create a pipeline to acquire, transform and visualize data collected from IoT sensors and machines.

Benefits

The transformative element of any IoT system is the data that can be collected from it. Thus the ability to extract data and using data analytics techniques to gain insights increases employability.

Learning Components

- Use Python to read data from sensors and store data in a SQL data base.
- Use Python Data Analysis library to clean, manipulate, integrate data sets.
- Use Python Visualization Libraries to visualize real-time data end explore acquired data sets.
- Explain the fundamental principles of a modern scalable Big Data platforms like Hadoop.
- Use storytelling to present the insights gained from extracted data.

Features

Target Audience: Secondary, Vocational, 2-year and 4-year College, 4-Year University students

Prerequisites: IoT Fundamentals: Connecting Things

Languages: English

Course Delivery: Instructor-led

Estimated Time to Complete: 40-50 hours

Recommended Next Course: IoT Fundamentals: Hackathon Playbook

Instructor Training: Required

IoT Fundamentals: Hackathon Playbook

Course Overview

The Hackathon Playbook is a comprehensive framework of tools and templates to prepare and run a Hackathon as a result of best practices and lessons-learned collected from the global execution of IoT Hackathons within Networking Academy and by other organizers.

Benefits

Student reinforce and deepen their multidisciplinary IoT and data skills by defining, designing, prototyping and presenting an IoT solution to a panel of industry experts and peers.

Learning Components

- Inspiration: understand, select and present the problem to be solved to recruit fellow partners.
- Ideation: invent a concept that doesn't already exist to solve a social issue. Learn how to present the solution to experts who will mentor students.
- Prototyping: create a prototyping action plan, including objects and visuals to illustrate their plan and will help an expert understand the concept and prototyping needs.
- Testing: present the concept and validate the prototype with a second expert, including user experience and enhancements.
- Presentation: present the solution and demo the prototypes to an expert panel.

Features

Target Audience: Secondary, Vocational, 2-year and 4-year College, 4-Year University students

Prerequisites: IoT Fundamentals: Connecting Things and/or Big Data and Analytics

Languages: English

Course Delivery: Instructor-led

Estimated Time to Complete: 20-30 hours

Recommended Next Course: any Career-Ready offering from Cisco or an industry IoT training program

Instructor Training: Required

Cisco Prototyping Lab

Tool Overview

The Cisco Prototyping Lab is a comprehensive learning environment created by Cisco for Networking Academy students to learn and practice key aspects of the foundational IoT technologies. Using an engaging, hands-on approach, it supports both the learning and creative phases of the Networking Fundamentals curriculum.

Career Prep

Provides an easy to use, comprehensive learning environment using real devices, code, coding tools and data that students use to create the physical interconnection of an end-to-end IoT and the logical data pipeline to acquire, analyze and present data.

Learning Components

- Prototyping Lab App
- Prototyping Lab Kit
  - Raspberry Pi 3 CanaKit Ultimate Starter Kit (or equivalent)
  - SparkFun Inventor's Kit for Arduino v3.2 (or equivalent)
  - Cables, sensors & actuators

Features

As an integral part of the Networking Academy learning experience, Cisco Prototyping Lab provides

- Interactive labs using Jupyter Notebook
- Visual programming with Blockly
- Device programming with Python
- Data visualization & analytics
- Connected applications via APIs
- Rapid Prototyping

Packet Tracer

Tool Overview

Packet Tracer is an innovative simulation and visualization tool used for lectures, labs, games, homework, assessments, and competitions. It is embedded in these courses:

- CCNA Routing and Switching
- CCNA Security
- IT Essentials
- Intro to the Internet of Things
- Mobility Fundamentals

Career Prep

The Packet Tracer simulation-based learning environment promotes the development of essential career skills ranging from teamwork and critical thinking to creative problem solving.

Learning Components

- Cisco Packet Tracer (PT)
- PT Mobile Android
- PT Mobile iOS
- PT Games

Features

As an integral part of the Networking Academy learning experience, Packet Tracer provides

- Simulation
- Visualization
- Authoring
- Assessment
- Collaboration capabilities and facilitates the teaching and learning of complex technology concepts.


CISCO