Internet of Things (IoT)

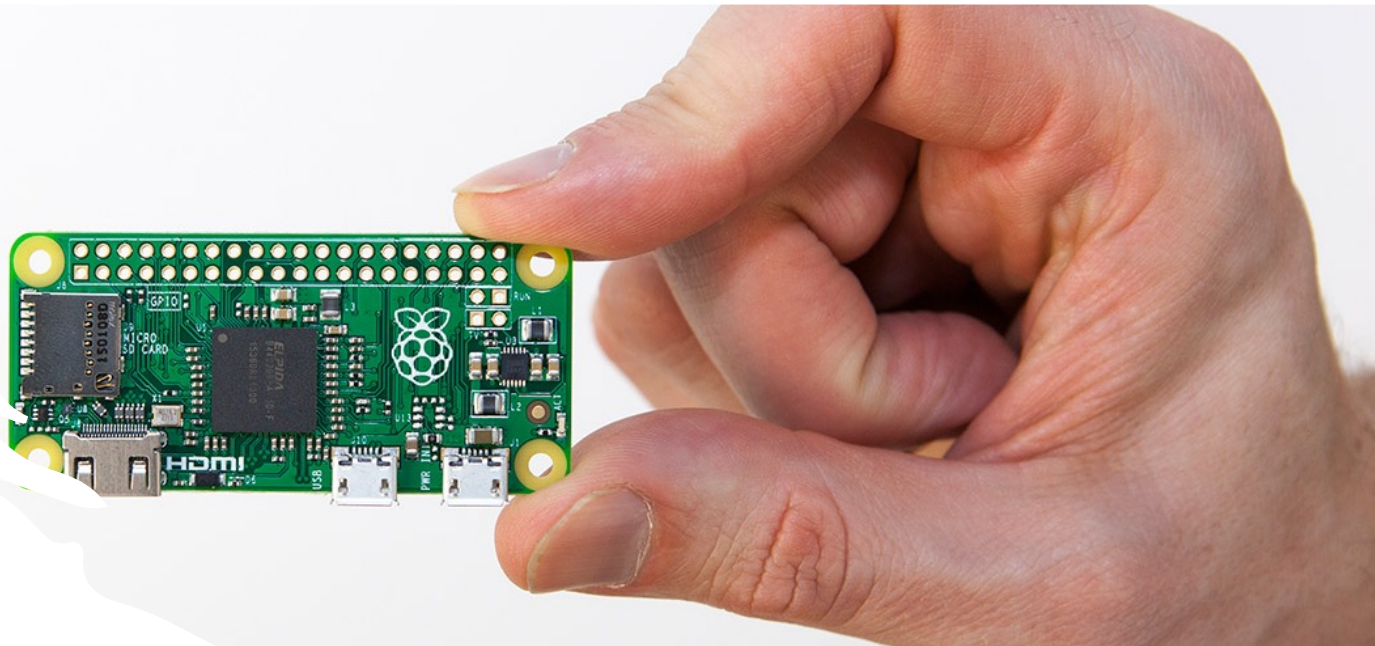
Prof. Michalis Vlachos

UNIL | Université de Lausanne

From Big (cloud)

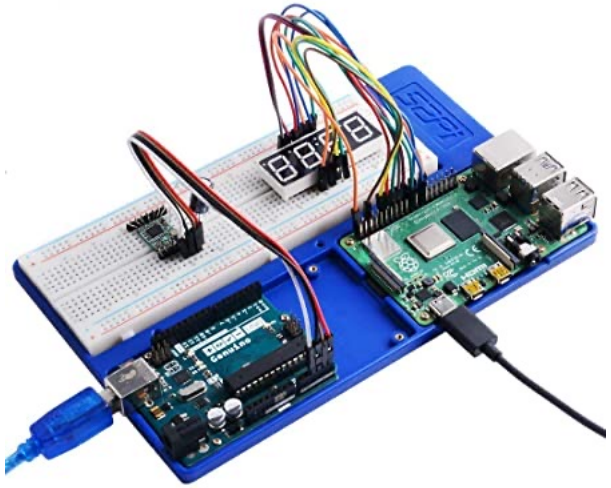
to

small (IoT)



Internet of Things (IoT)

- The **Internet of things** (**IoT**) describes physical objects (or groups of such objects) that are embedded with sensors, processing ability, software, and other technologies that connect and exchange data with other devices and systems over the Internet or other communications networks.
- IoT is an application of existing technologies like – connectivity, security, storage, sensing – in new ways.
- Value of IoT: real commercial cases, utilitarian and engaging way.



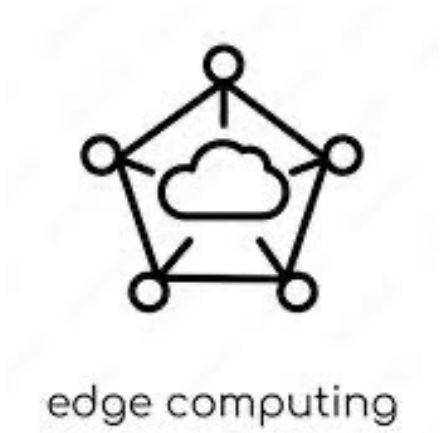
Why do you think we bother with IoT devices?

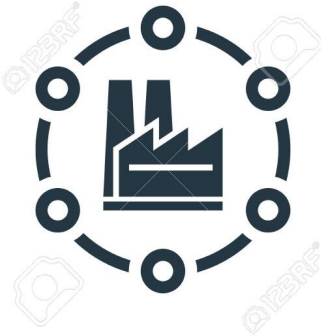
IoT devices = Digital Tools (?)

- Humans, since the beginning of their existence, build *tools*.
- Such devices allow us to build the **tools of the digital age**.

Related terms

- **Edge computing** is “a distributed computing paradigm that brings computation and data storage closer to the sources of data. This is expected to improve response times and save bandwidth.”
- **Industry 4.0** “conceptualizes rapid change to technology, industries, and societal patterns and processes in the 21st century due to increasing interconnectivity and smart automation”



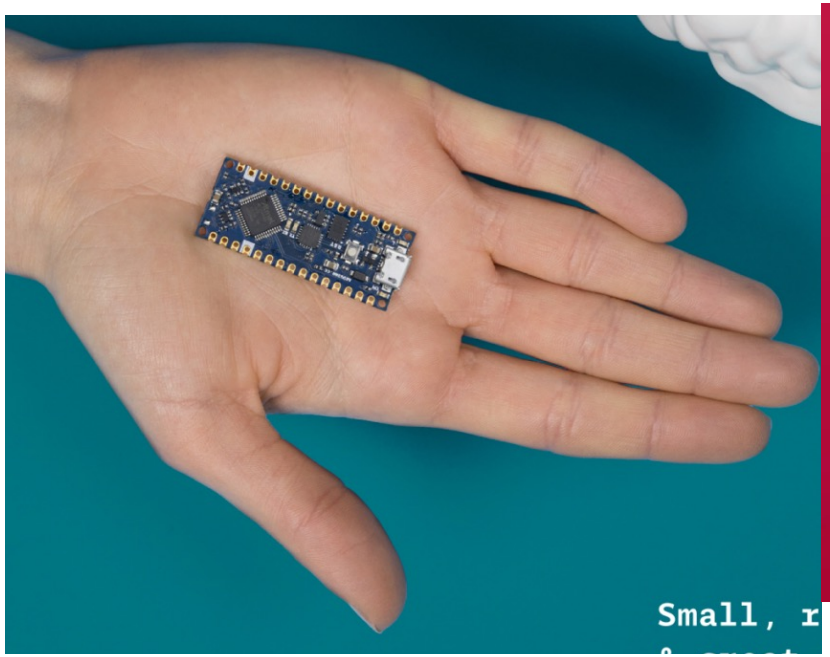


SmartHome

Maker Economy

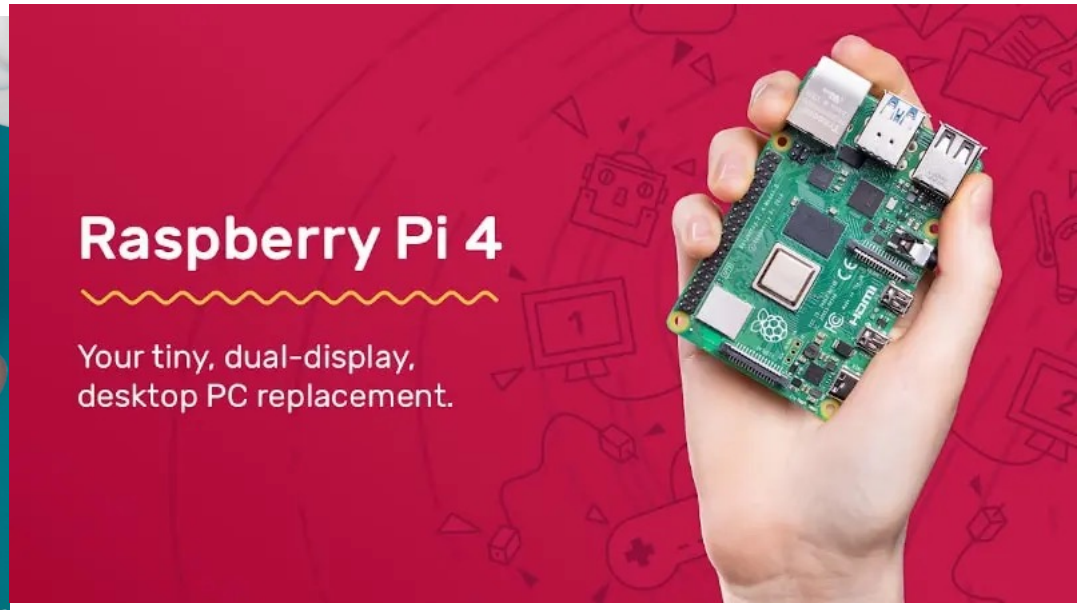
“Puts power in the hands of people to fund, design, prototype, produce, manufacture, distribute, market, and sell their own goods.”

The devices



Arduino
(microcontroller)

Code in C (typically)
or in Python
(Most have no WiFi)



Raspberry pi 4
(just like a computer)

Code in Python
You boot linux

ESP32

Code in Scratch
or Python

Comparison of devices

	ARDUINO	RASPBERRY PI Zero	Raspberry Pi 4	ESP32 eg M5GO or Core2
TYPE	Microcontroller	Microprocessor / Single Board Computer	Microprocessor / Single Board Computer	Microprocessor
CPU	ATmega328P	ARM Cortex	ARM Cortex	Dual-Core 32-bit LX6
RAM	48 – 256 KB	512 MB	1 GB – 8 GB	16 MB
WIFI	Usually not	YES	YES	YES
Programming ?	C / MicroPython	Any Language (Python, C++,…)	Any Language (Python, C++,…)	C, MicroPython, Blockly
Goal	Control devices	Regular Computer	Regular Computer	
Power Consumption	40 mA (0.2W)	120 mA (0.7W)	500 - 1000 mA (5 W)	130 mA
CPU Speed	16 - 48 MHz	1 GHz	1.5 GHz	240 MHz

What we will use

- **Product Features**
- ESP32-based, built-in Bluetooth,WiFi
- 16M Flash, 8M PSRAM
- Built-in speaker, power indicator, vibration motor, RTC, I2S amplifier, capacitive touch screen, power button, reset button
- TF card slot (16G Maximum size)
- Built-in lithium battery, equipped with power management chip
- Independent small board built-in 6-axis IMU, PDM microphone
- M-Bus Socket & Pins
- Program Platform: UIFlow, MicroPython, Arduino

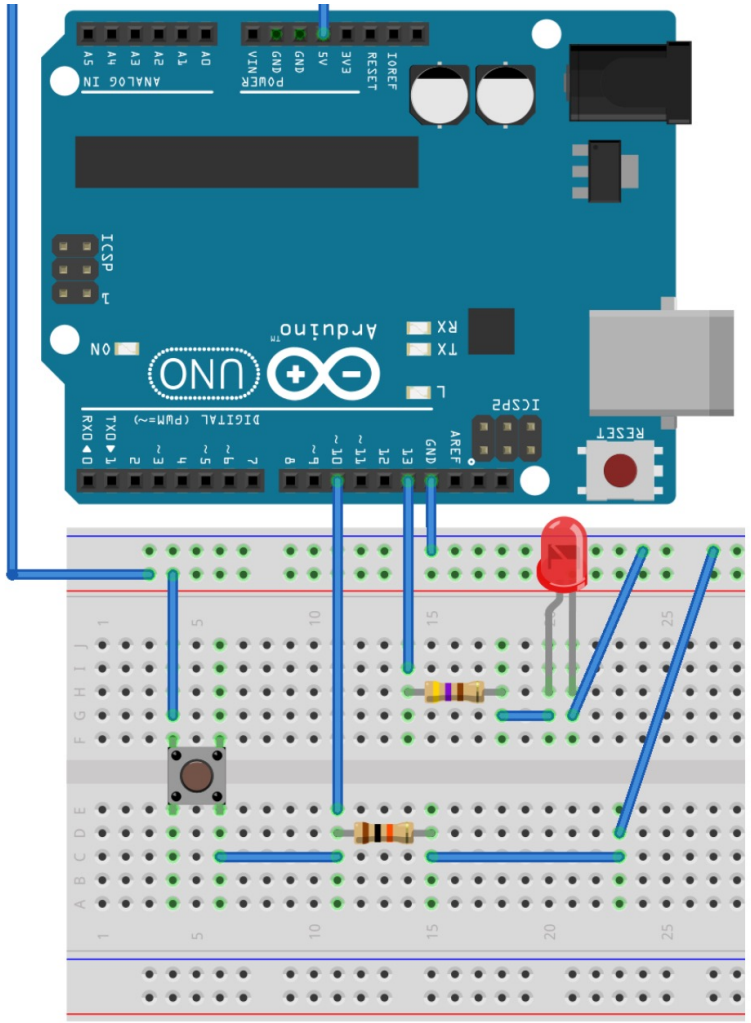
What we will use

ESP32	240MHz dual core, 600 DMIPS, 520KB SRAM, Wi-Fi, dual mode Bluetooth
Flash Memory	16MB
PSRAM	4MB
Power Input	5V @ 500mA
Port	TypeC x 1, GROVE(I2C+I/0+UART) x 1
IPS Screen	2 inch, 320x240 Colorful TFT LCD, ILI9342C
Speaker	1W-0928
Button	Custom button x 3
MEMS	BMM150 + MPU6886
Battery	500 mAh @ 3.7V
Operating Temperature	32°F to 104°F (0°C to 40°C)
Size	54.2 x 54.2 x 30.5mm
Weight	62.3g
Case Material	Plastic (PC)

Development Environments

- Arduino IDE
- Python
- Micropython
- Scratch

Connecting to the outside world!

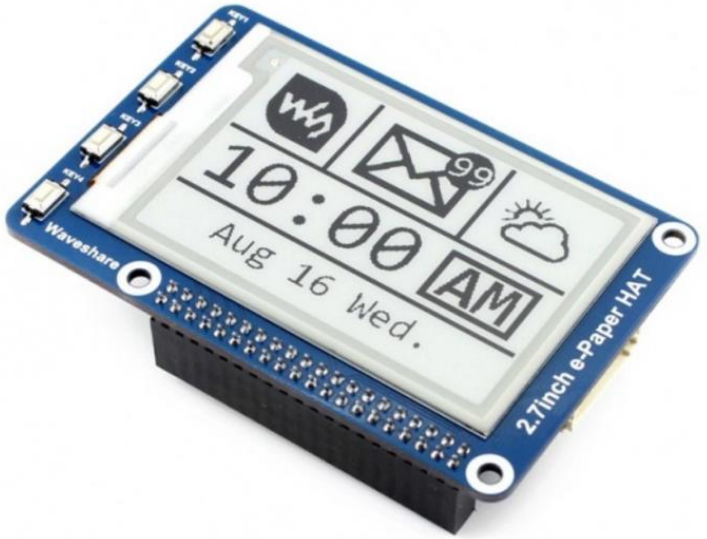


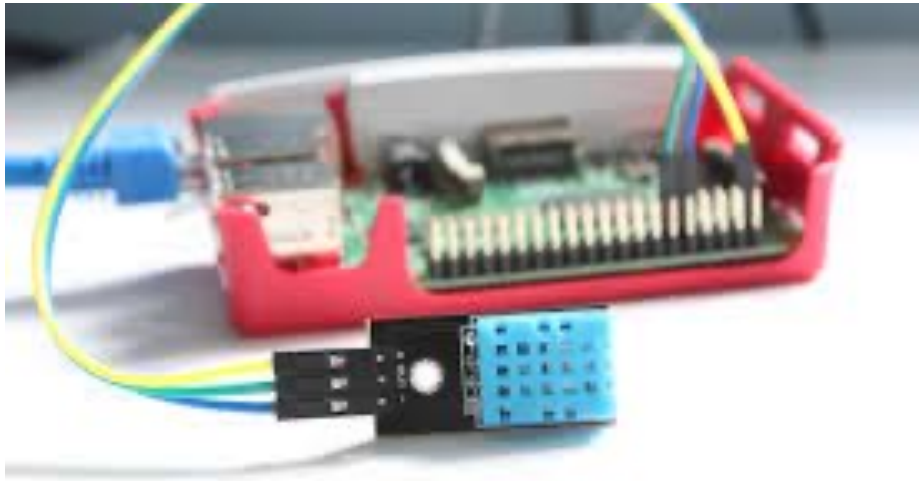
The appeal of those devices is that you can connect it to and interact with the outside world

ESP32 (M5Stack)

Connectivity

Screen/e-ink

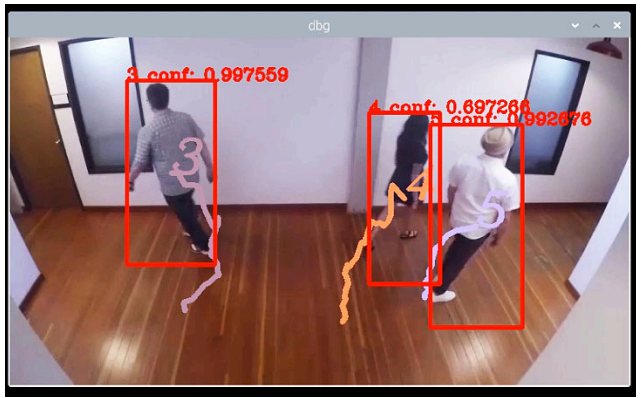




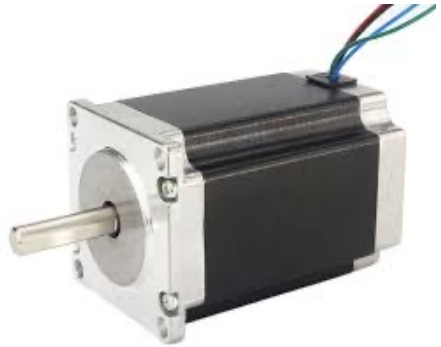
CO2/temperature sensors

GPS

Camera



Relay Switch (control high voltage devices)



Step motors

Talk to your neighbor

- Brainstorm on some project you would like to do with such a device.

Applications of IoT

- Remote measurement
- Smart watering
- Smart Devices
- Automation
- Drones
- Sustainability

Smart watering

At a vast vineyard outside Modesto, Calif., E&J Gallo Winery is testing a new irrigation system developed with IBM to grow grapes using less water. The plan is to eventually apply the lessons learned to Watson so that IBM's data crunching technology can help farmers around the world.

Cutting water use can save huge amounts of money in the agriculture industry. It can also play a big role in water conservation, especially during droughts like the one that has plagued California for several years.

"Water conservation has been a central focus," said Dr. Nick Dokoozlian, vice president, viticulture, chemistry, and enology at E&J Gallo. "Our stewardship of water continues."

Smart Devices (Fridge)

See what's inside your fridge from anywhere

View Inside[4] (+ auto tagging of food expiration date)

LIDAR AND ARCHAEOLOGY

WHAT IS LIDAR?

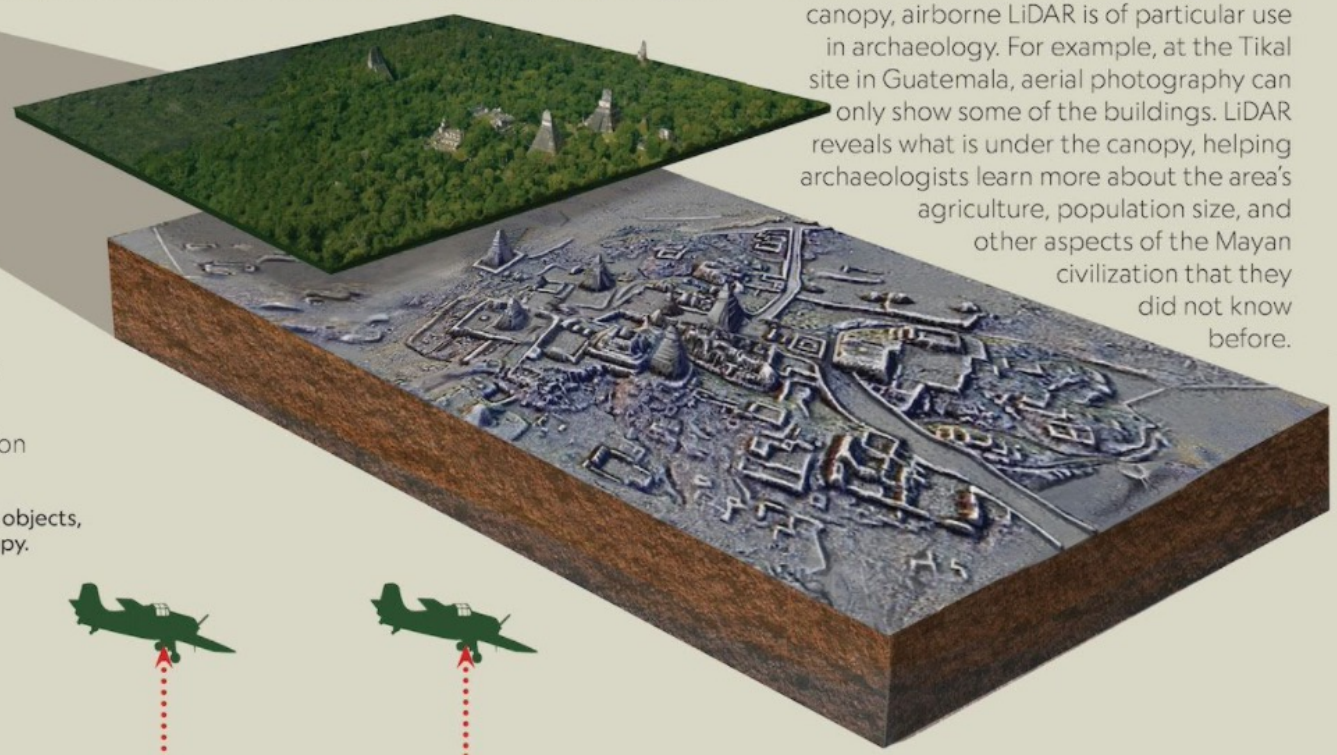
LiDAR stands for Light Detection And Ranging, a technology invented in the 1960s. It a remote sensing technology system that collects data about ranges, or distances, using lasers. It has a number of benefits:

- It can more easily collect data over a large area.
- It can be used to create maps.
- It can map areas that are not visible to the human eye.

Because of its special features, LiDAR is now being used by archaeologists.

UNCOVERING TIKAL

Because of its ability to "see" through a tree canopy, airborne LiDAR is of particular use in archaeology. For example, at the Tikal site in Guatemala, aerial photography can only show some of the buildings. LiDAR reveals what is under the canopy, helping archaeologists learn more about the area's agriculture, population size, and other aspects of the Mayan civilization that they did not know before.



MAIN PARTS OF AN AIRBORNE LIDAR SYSTEM

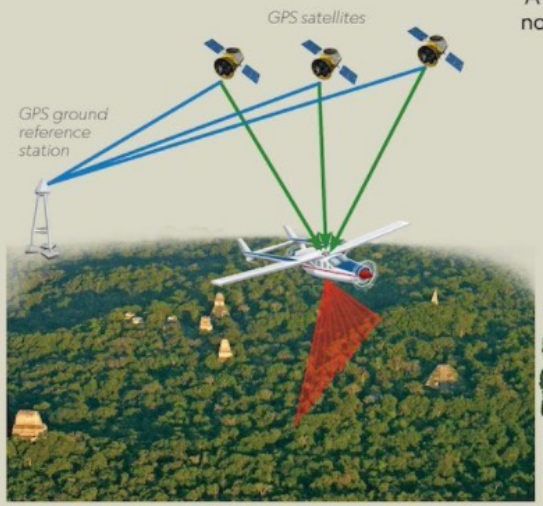
Plane/Helicopter/Drone—carries the moving technology

Laser—sends out an echo pulse

Laser Sensor—detects the reflected pulse to measure the range

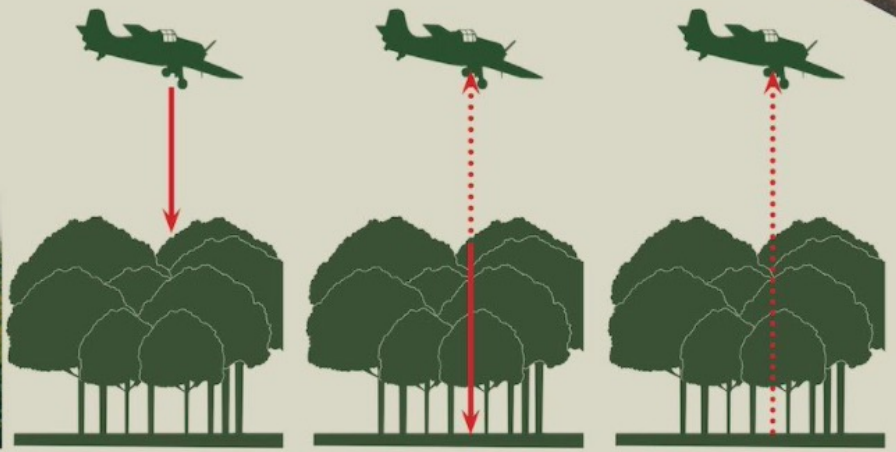
Global Positional System (GPS) ground reference and satellites—detect aircraft position

Inertial Measurement Unit system (IMU)—determines aircraft's orientation



Lasers sweep across the terrain measuring

A camera sees only the nearest objects, not what's underneath the canopy.



LIDAR IN ACTION

Vegetation limits aerial photography. It does not limit LiDAR. Where light can go, LiDAR can go. LiDAR pulses reflect from multiple points, starting at the first item they hit, and continue to the ground.
So LiDAR can provide information about things that people cannot see.

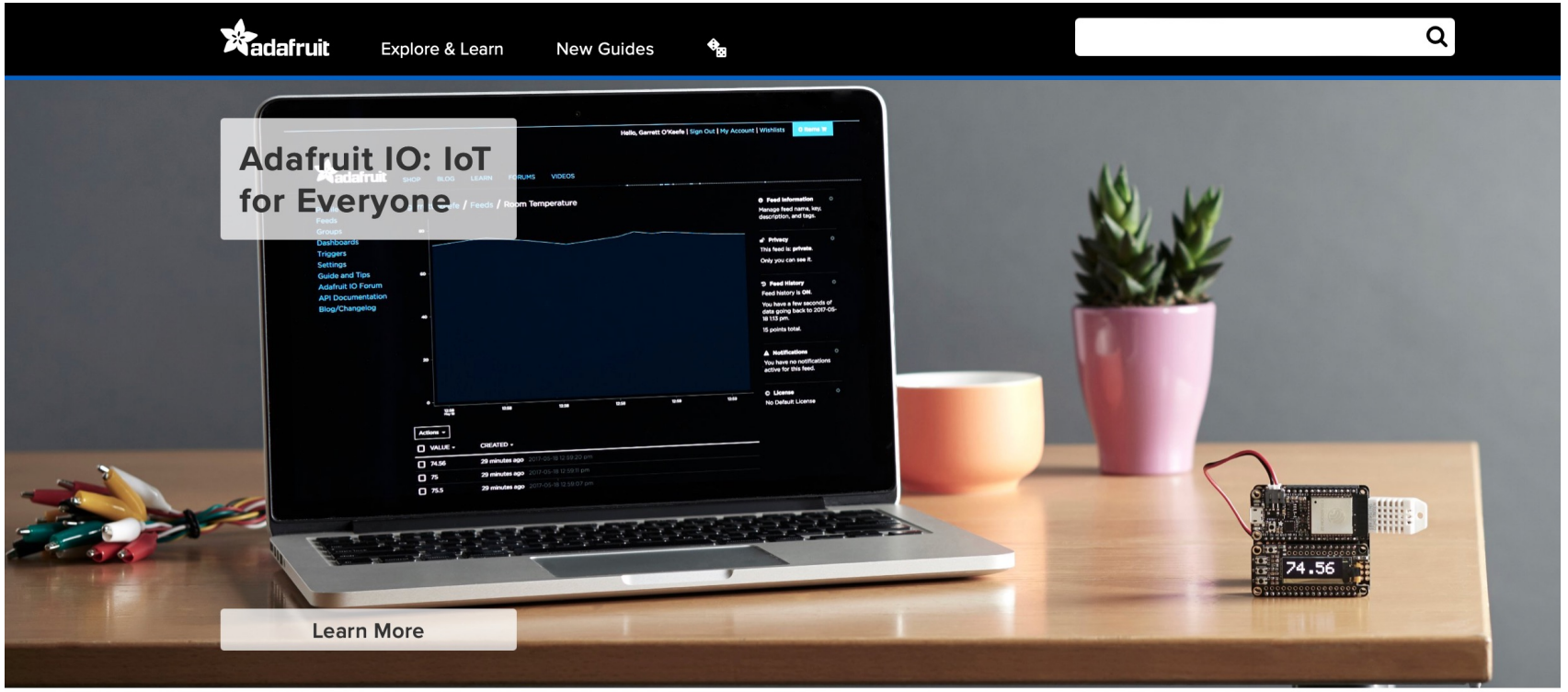
NATIONAL GEOGRAPHIC

Aspects to consider when developing an IoT project…

- Communications (Internet needed or not?)
- Processor Power (how much power do I need?)
- Local Storage (RAM, EEPROM, Flash)
- Power Consumption (autonomy for how many hours/days)
- Functionality (GPIO interface, sensors, touch screen, ...)
- Cost (how many devices will I need?)

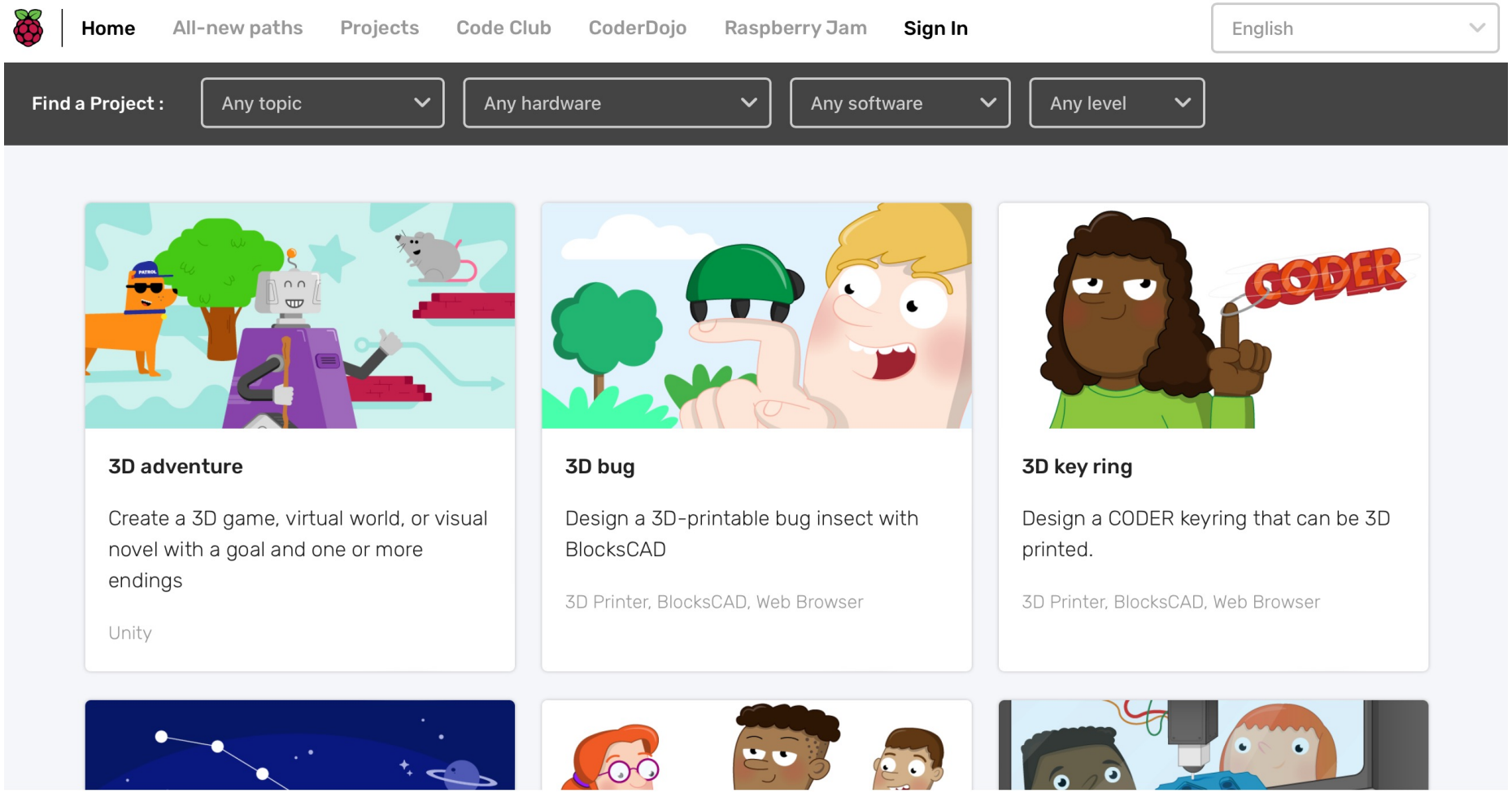
Learning Resources

http://learn.adafruit.com



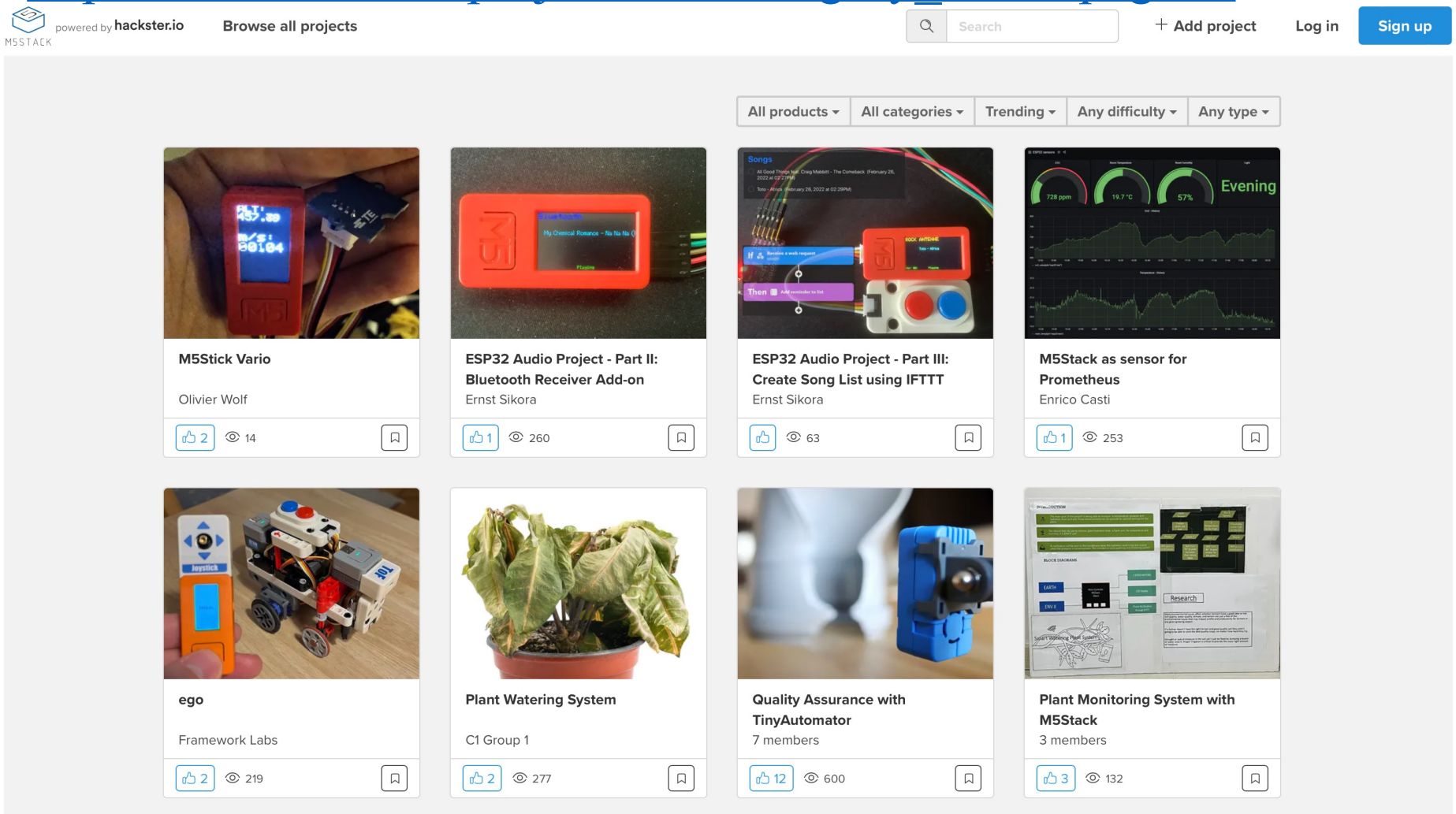
Learning Resources

https://projects.raspberrypi.org/en/projects



Learning Resources

- https://m5stack.hackster.io/projects
- https://m5stack.com/project-hub?category_id=1&page=1



Other cool projects you can build

- Remote monitoring and automation for House/RV
- Smart supermarket
- Air quality monitor
- Portable maps for your bike (or any other vehicle)
- A speedometer for your vehicle

- …get more ideas for projects here

Hands-on Part

M5Stack

- Extendible devices
- Easy connect with sensors, etc
- Can be coded over the Internet!

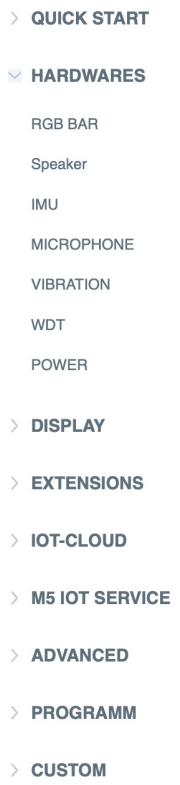
M5Stick

M5Go

Core 2

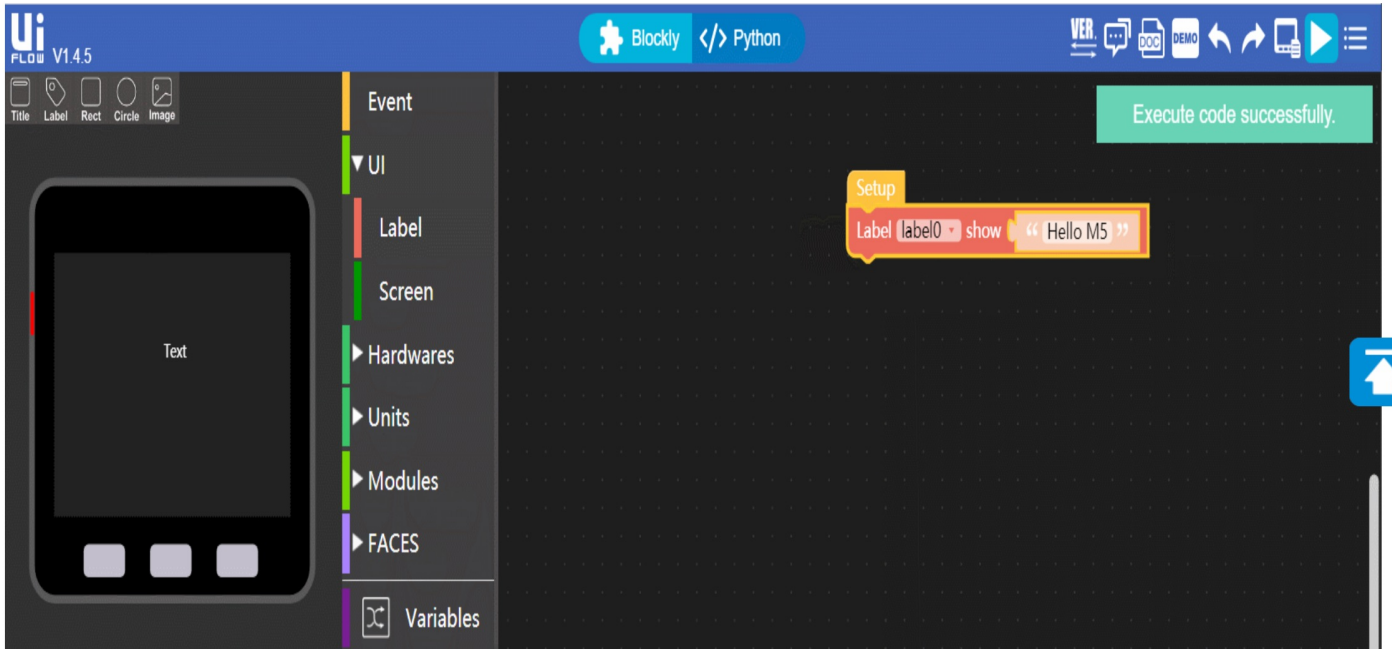
Documentation

- https://docs.m5stack.com/en/quick_start/m5core/uiflow
- https://m5stack.oss-cn-shenzhen.aliyuncs.com/resource/docs/M5GO_Guide_English.pdf

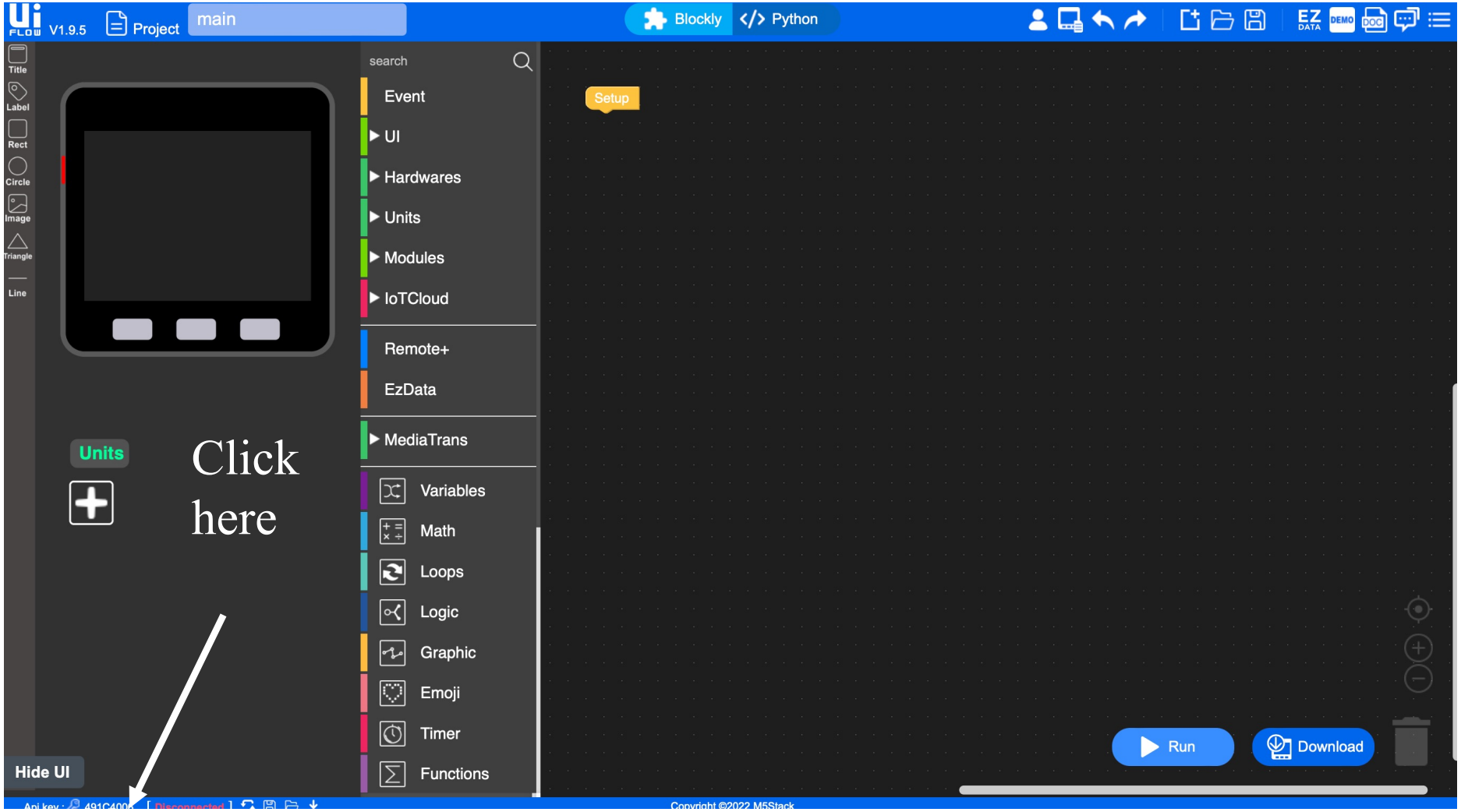


Hello M5

After completing the above steps, you can start programming with UIFlow. The following will demonstrate a simple program for you to drive the screen to display "Hello M5". (1. Place the label 2. Add the label block. 3 Click the run button in the upper right corner)

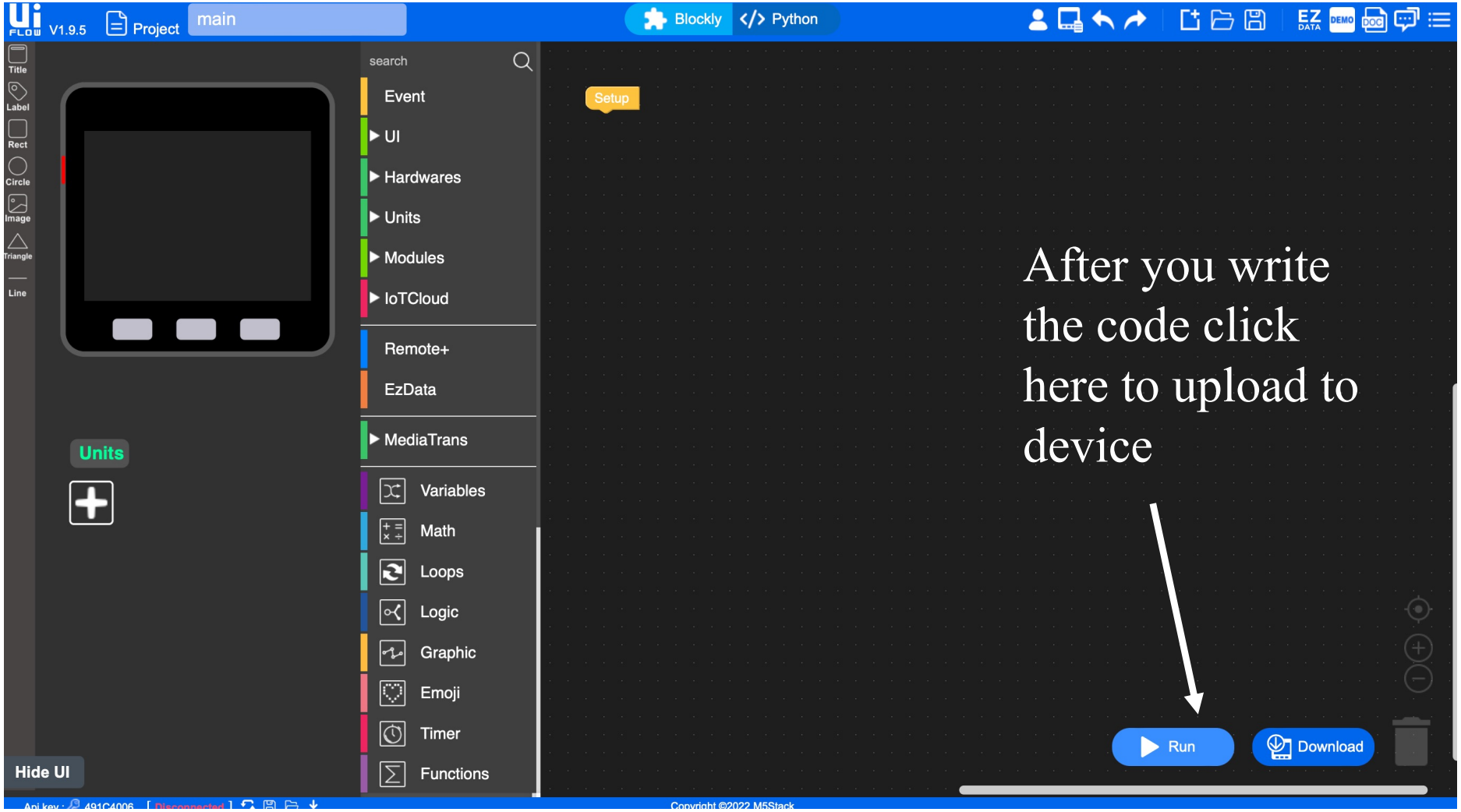


htttp://flow.m5stack.com

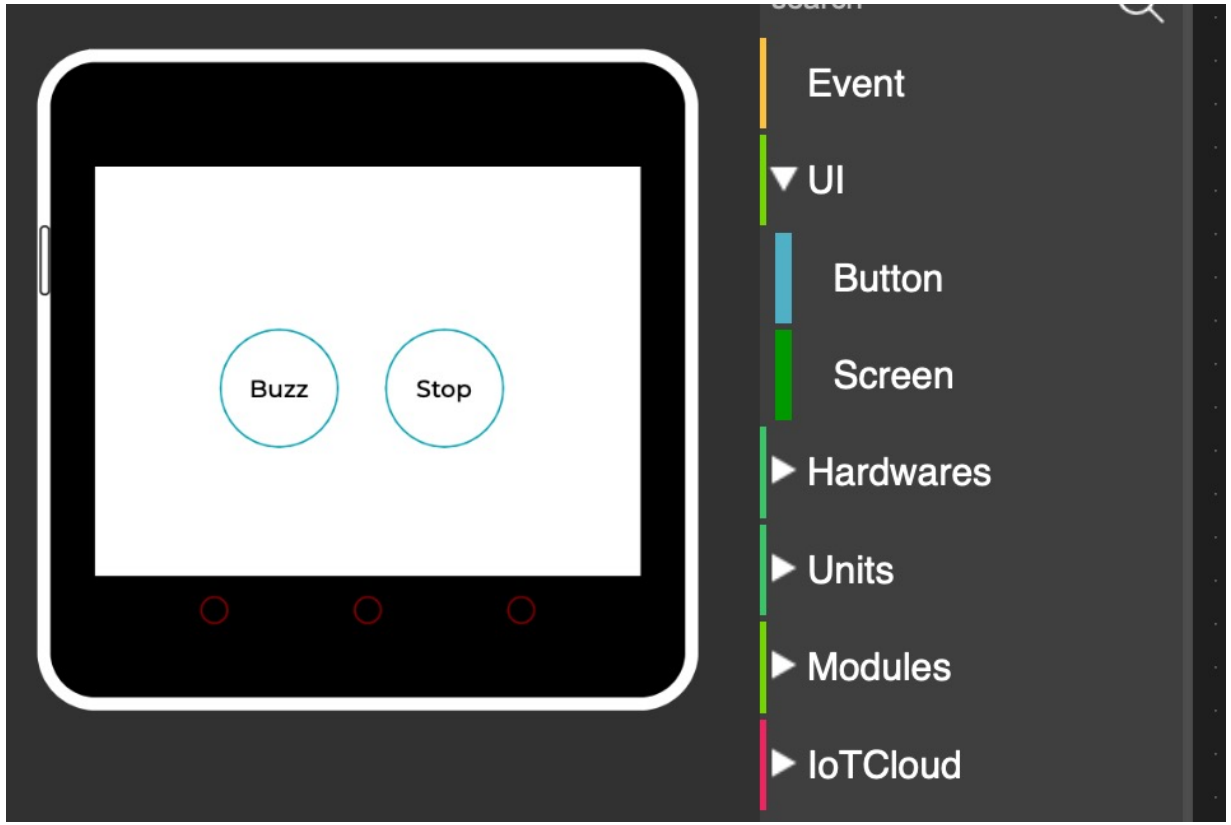


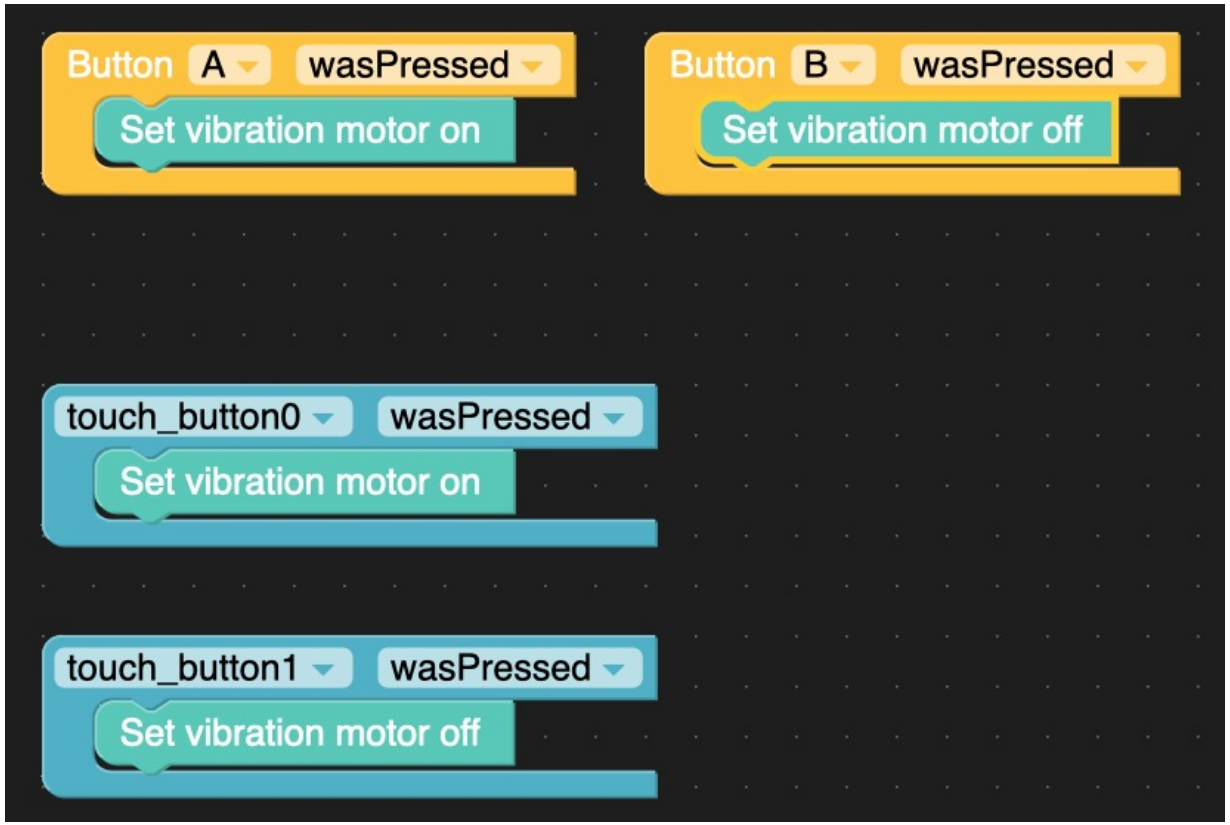
Flow.m5stack.com

Flow.m5stack.com

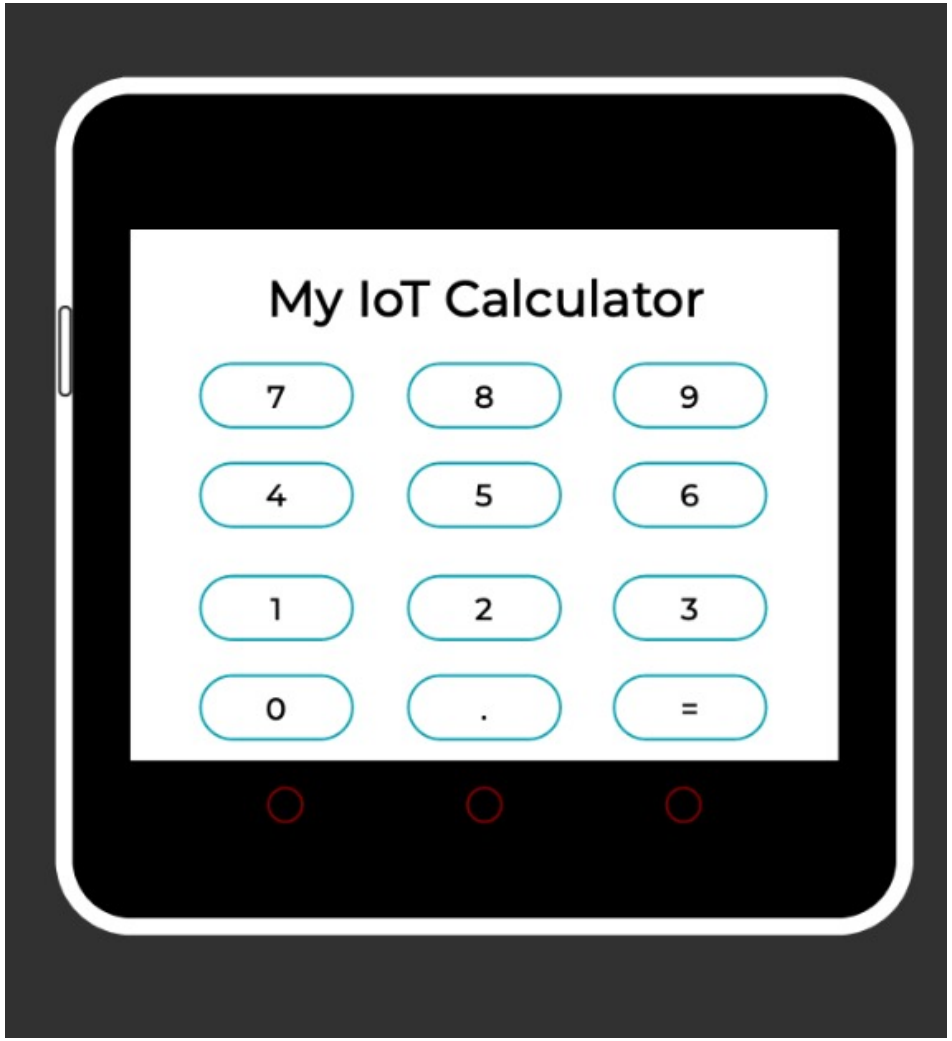


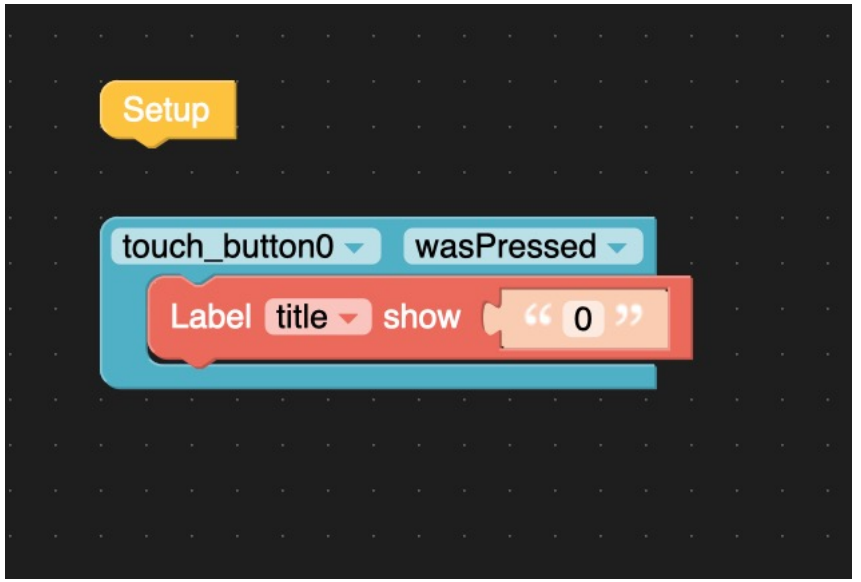
Warm up: The “Hello World” of IoT





Warm up: Use UIFlow to create and upload a simple interface, eg a calculator on the device

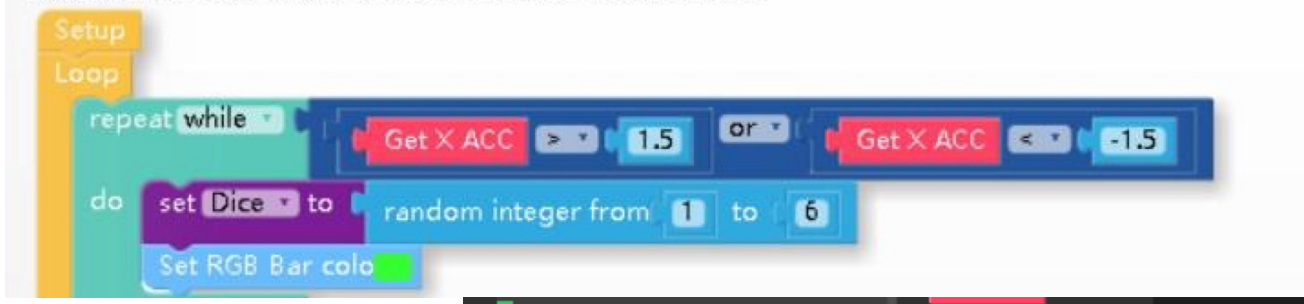




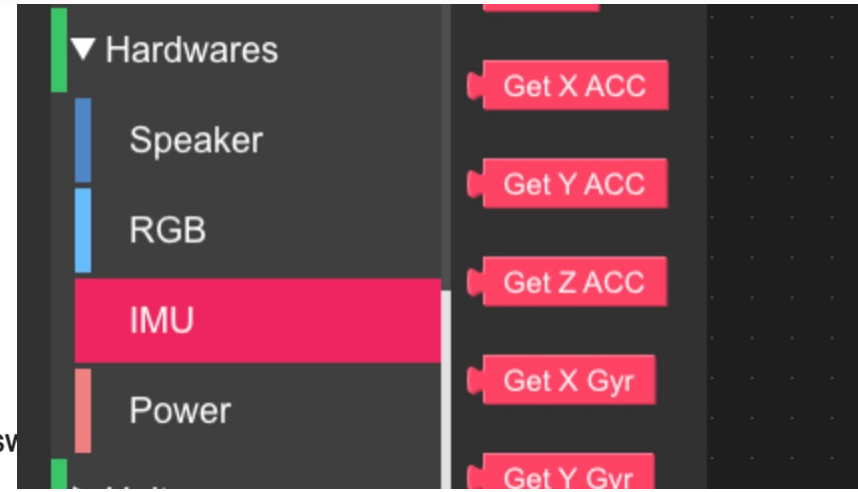

```
touch_button9 = M5Btn(text='9', x=218, y=60, w=70, h=30, bg_c=0xFFFFFF, text_c=0x000000,
touch_button = M5Btn(text='0', x=31, y=200, w=70, h=30, bg_c=0xFFFFFF, text_c=0x000000,
touch_buttonComma = M5Btn(text='.', x=125, y=200, w=70, h=30, bg_c=0xFFFFFF, text_c=0x00
touch_buttonEQ = M5Btn(text='=', x=218, y=200, w=70, h=30, bg_c=0xFFFFFF, text_c=0x00000

def touch_button0_pressed():
  # global params
  title.set_text('0')
  pass
touch_button0.pressed(touch_button0_pressed)
```

Warm up: Magic 8 Ball



Use the accelerometer of the device to create a magic 8 ball



A standard Magic 8 Ball has 20 possible answers, including 10 affirmative answers (●), 5 non-committal answ

● It is certain.	● As I see it, yes.	● Reply hazy, try again.	● Don't count on it.
● It is decidedly so.	● Most likely.	● Ask again later.	● My reply is no.
● Without a doubt.	● Outlook good.	● Better not tell you now.	● My sources say no.
● Yes definitely.	● Yes.	● Cannot predict now.	● Outlook not so good.
● You may rely on it.	● Signs point to yes.	● Concentrate and ask again.	● Very doubtful.

Magic 8-Ball

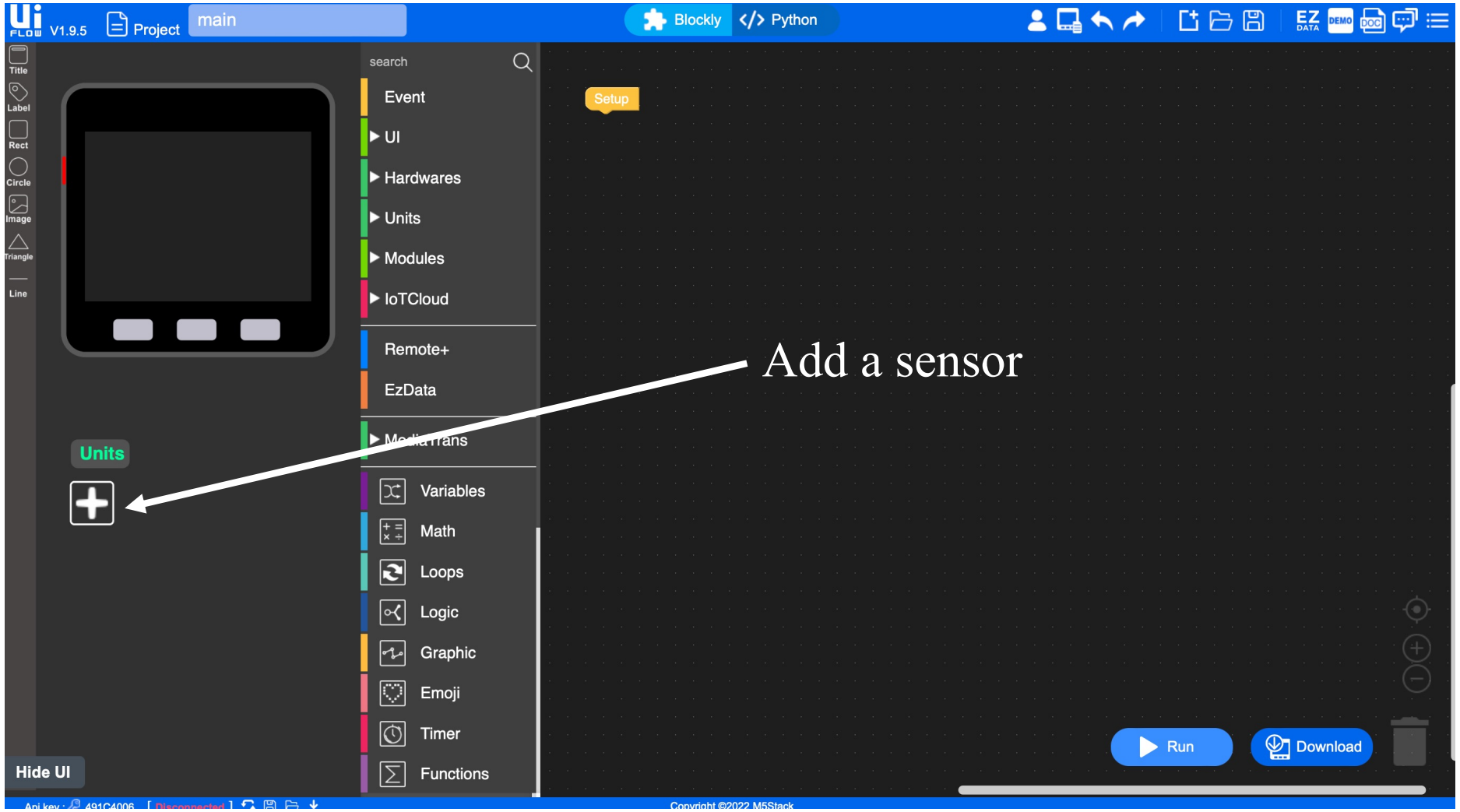
The Magic 8-Ball.

Project 1: Read from Internet

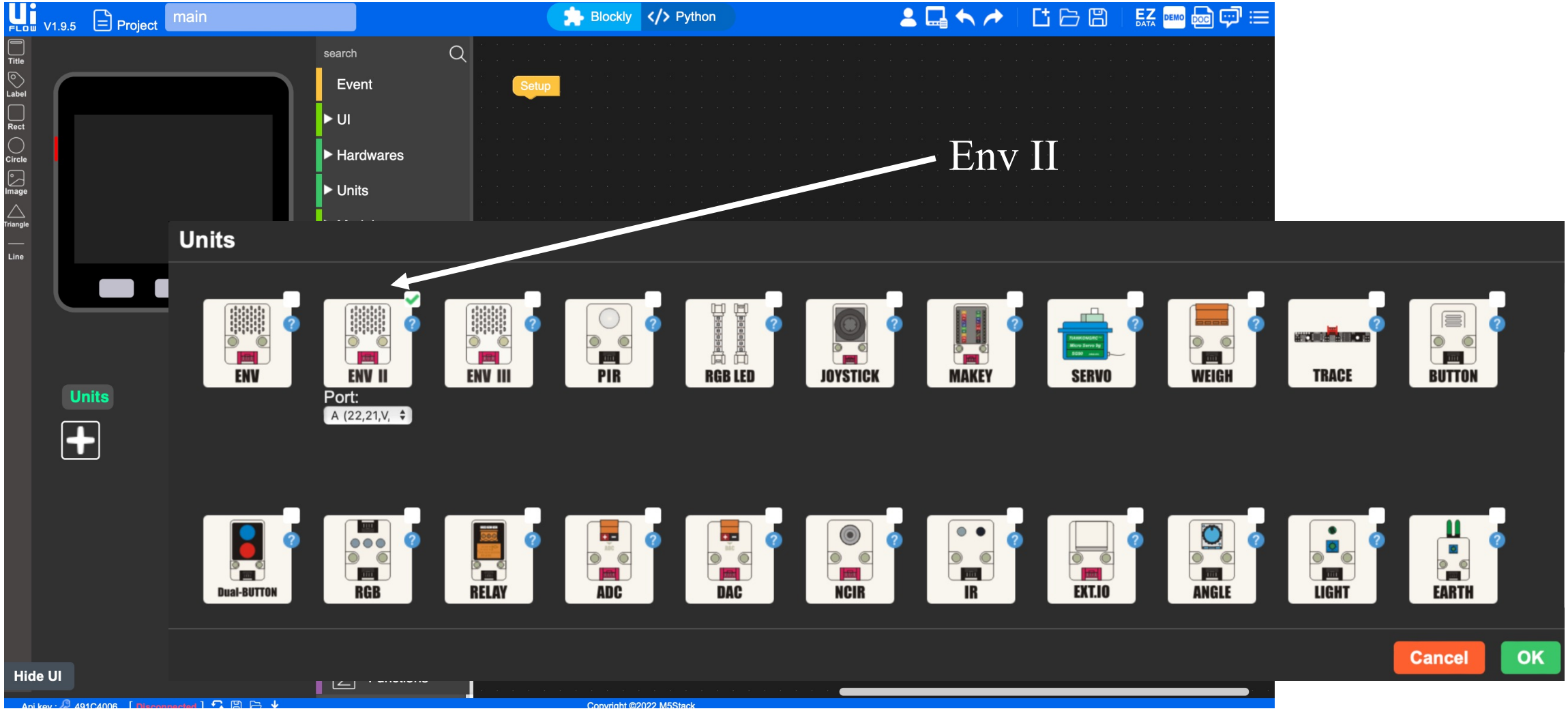
Read the weather in “Lausanne, CH” from an internet service and show it on the screen of the device

- Create a free account at openweathermap. Get the key.

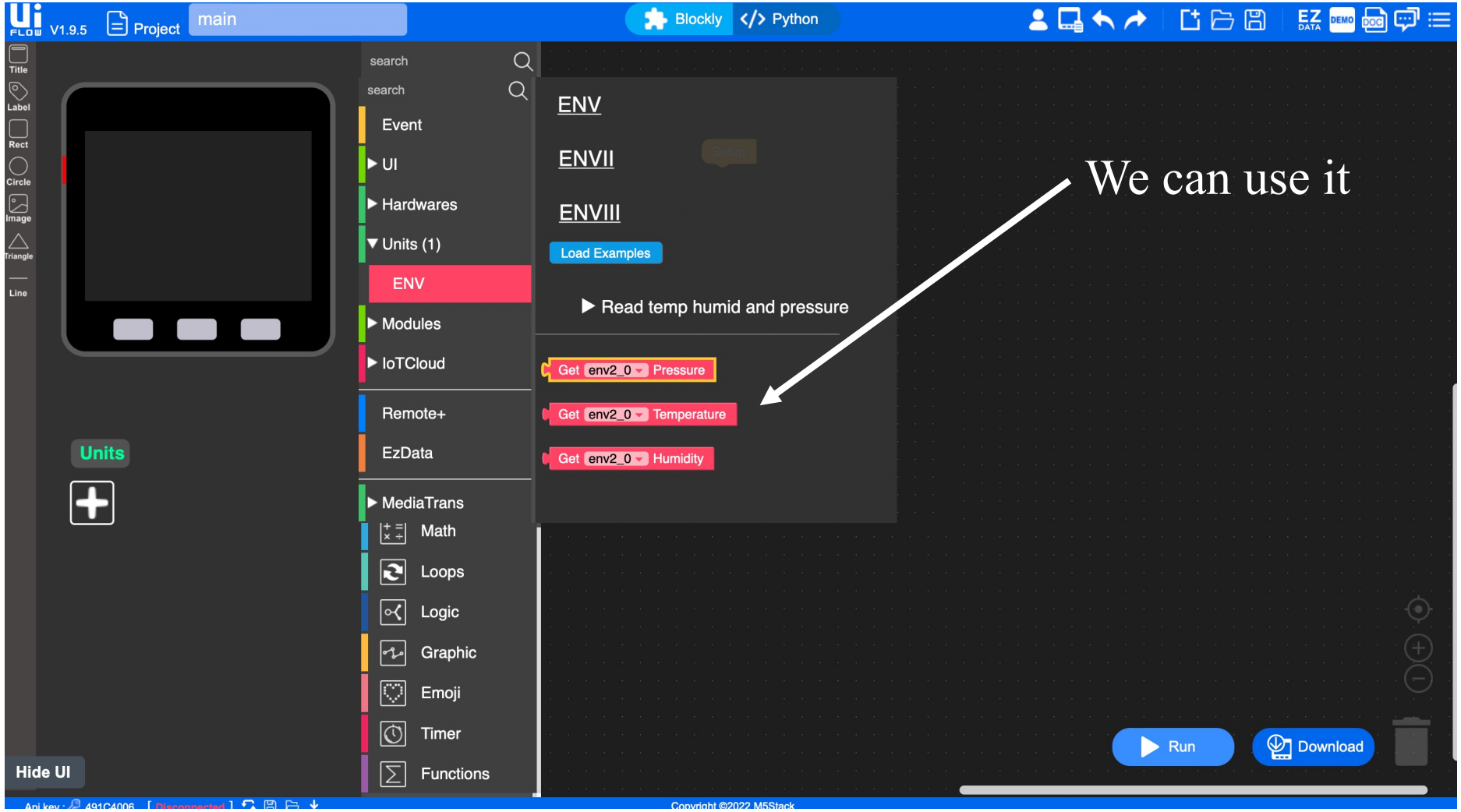
Project 2: Read from Sensors



Project 2: Read from Sensors



Project 2: Read from Sensors



Project 3: Write data from the device to the Internet

There are many ways to push data to the internet:

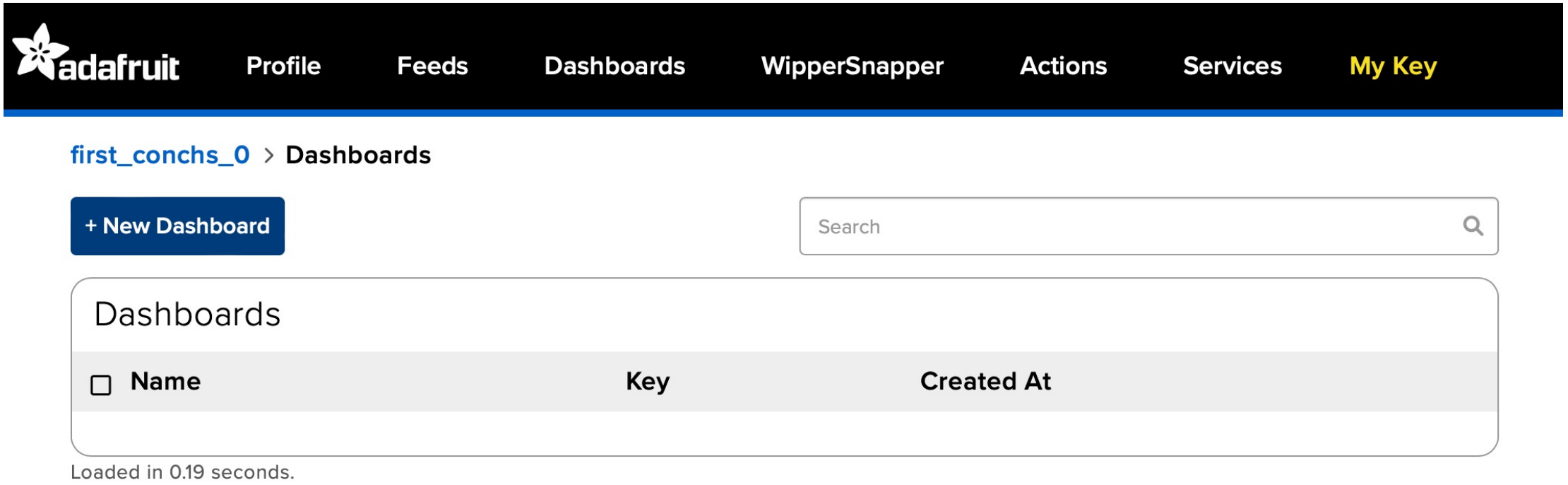
- Push it to Google Cloud (eg BigQuery)
- Wolfram Data Drop (universal data accumulator)
- Blynk
- Adafruit.io

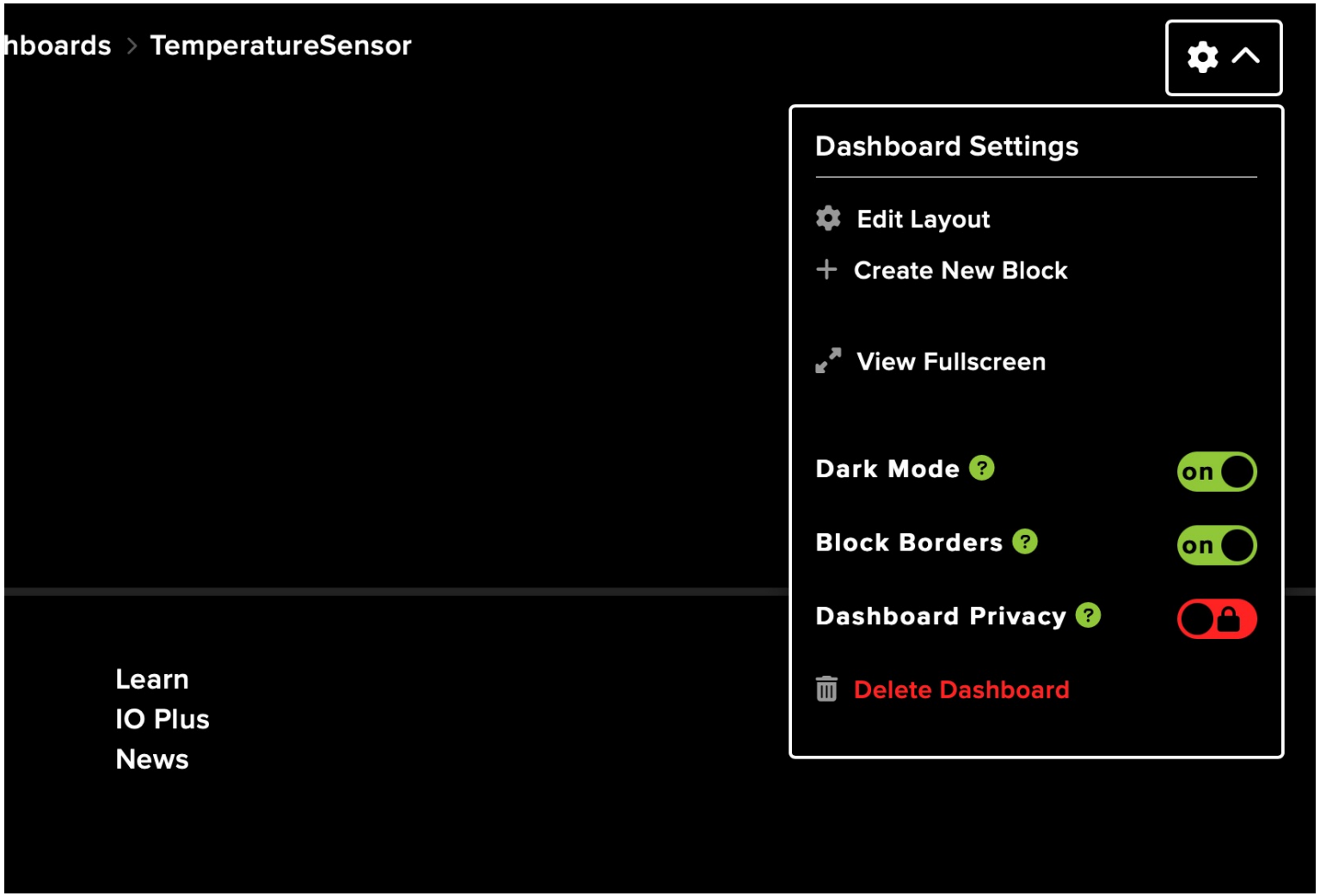
Exercise

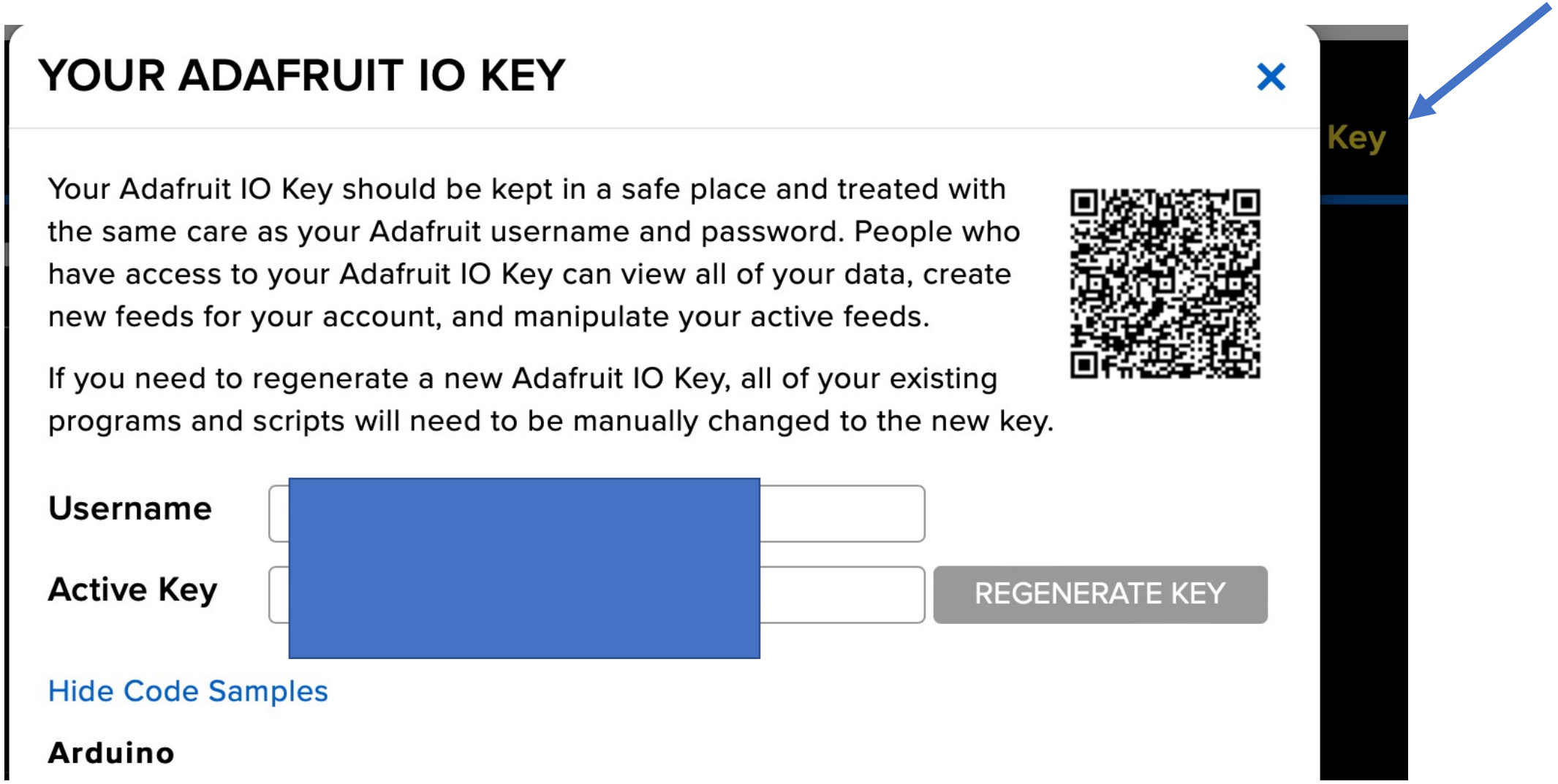
Now we will save those data also on the Internet.

For this we will setup an MQTT pub/sub platform.

- Create an account at Adafruit.io
- Create a dashboard and a feed.
- **Publish** your data on a **feed** of that platform.


adafruit
Profile
Feeds
Dashboards
WipperSnapper
Actions
Services
My Key
first_conchs_0 > Dashboards
+ New Dashboard
Search
Dashboards
Name
Key
Created At
Loaded in 0.19 seconds.


hboards
TemperatureSensor
Dashboard Settings
Edit Layout
Create New Block
View Fullscreen
Dark Mode
on
Block Borders
on
Dashboard Privacy
Delete Dashboard
Learn
IO Plus
News


YOUR ADAFRUIT IO KEY
Key
Your Adafruit IO Key should be kept in a safe place and treated with the same care as your Adafruit username and password. People who have access to your Adafruit IO Key can view all of your data, create new feeds for your account, and manipulate your active feeds.
If you need to regenerate a new Adafruit IO Key, all of your existing programs and scripts will need to be manually changed to the new key.
Username
Active Key
REGENERATE KEY
Hide Code Samples
Arduino

Project 4: Control the device from the Internet

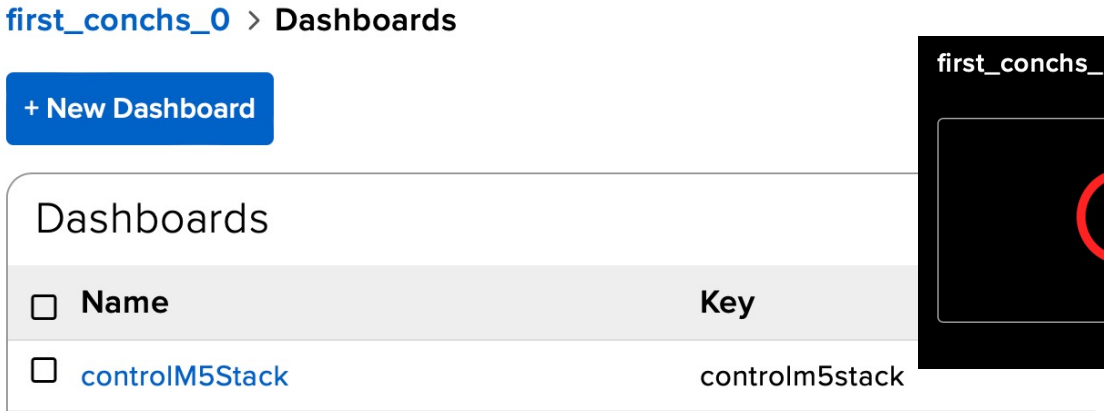
Now we do the reverse of what we did before.

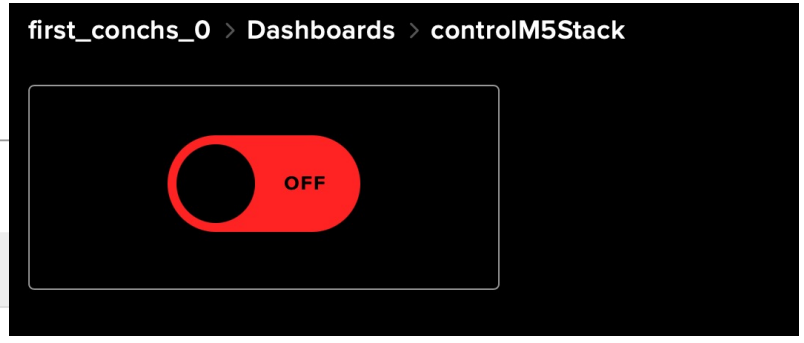
- Our device **subscribes** to a **feed.** The feed get the input from a dashboard at adafruit.io which contains a togglebutton.
- On the device the togglebutton controls something, eg buzzing, light, prints a text, etc.

- See video

On Adafruit.io

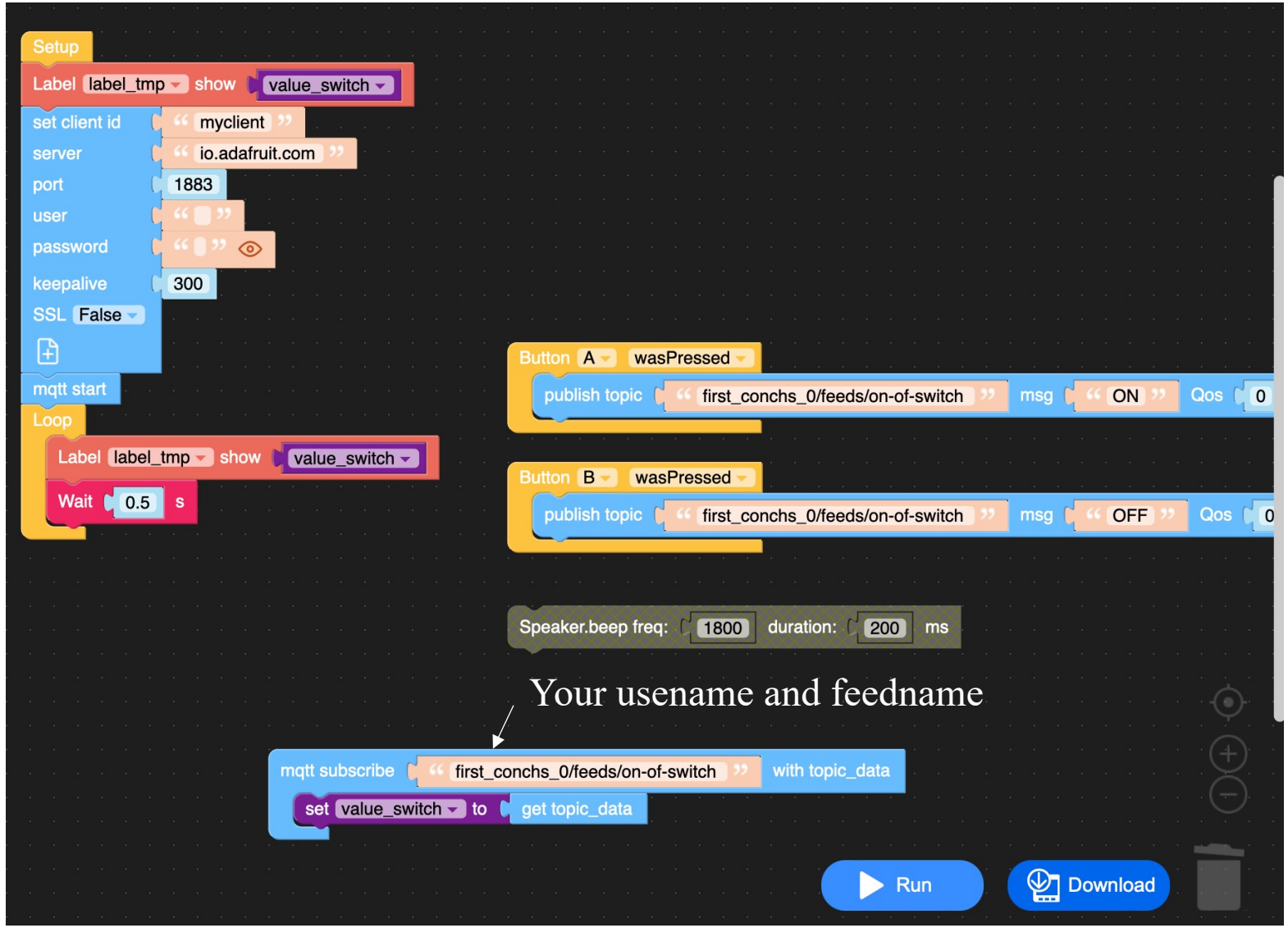
The Dashboard



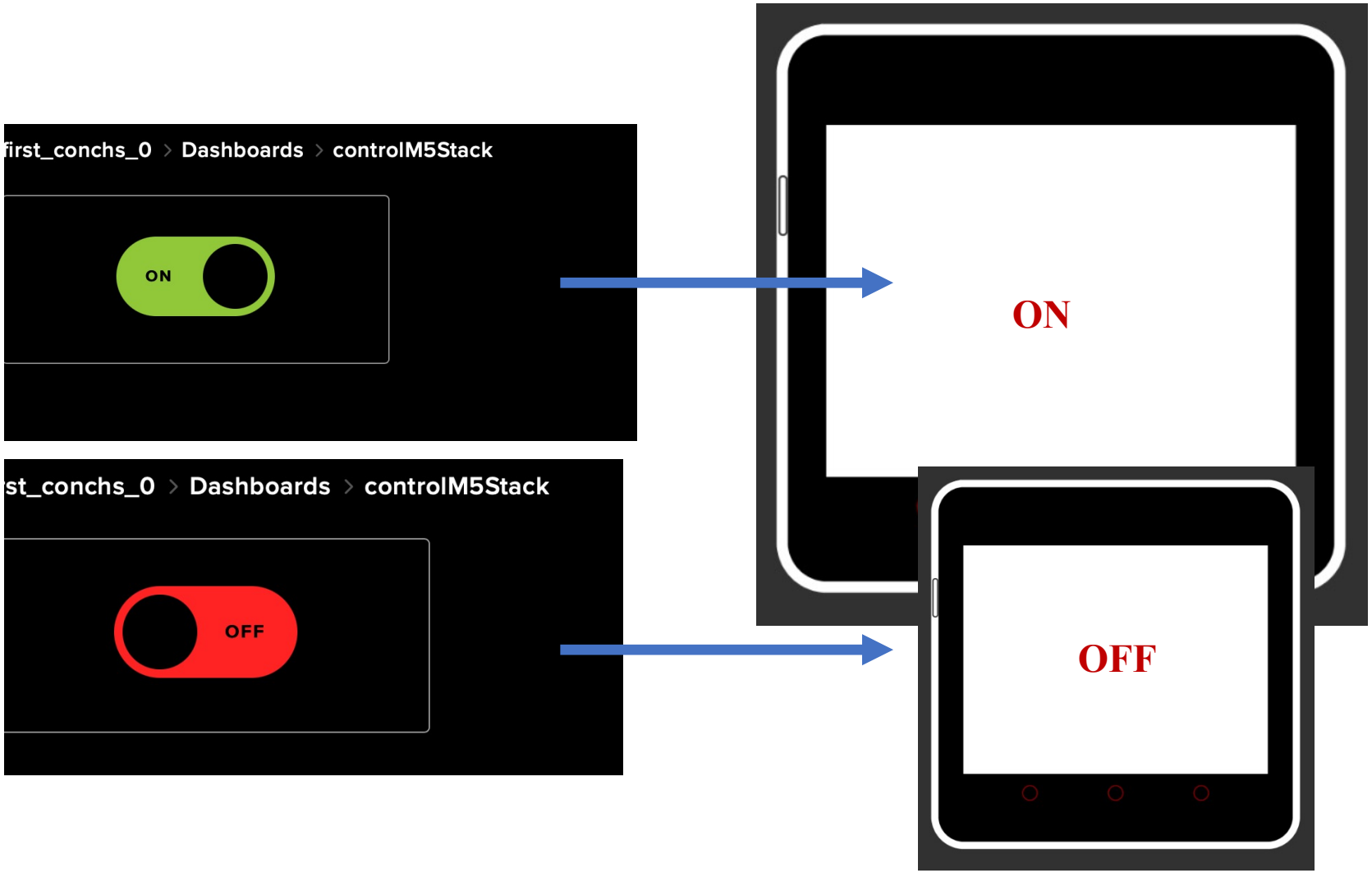


The Feed

On your device



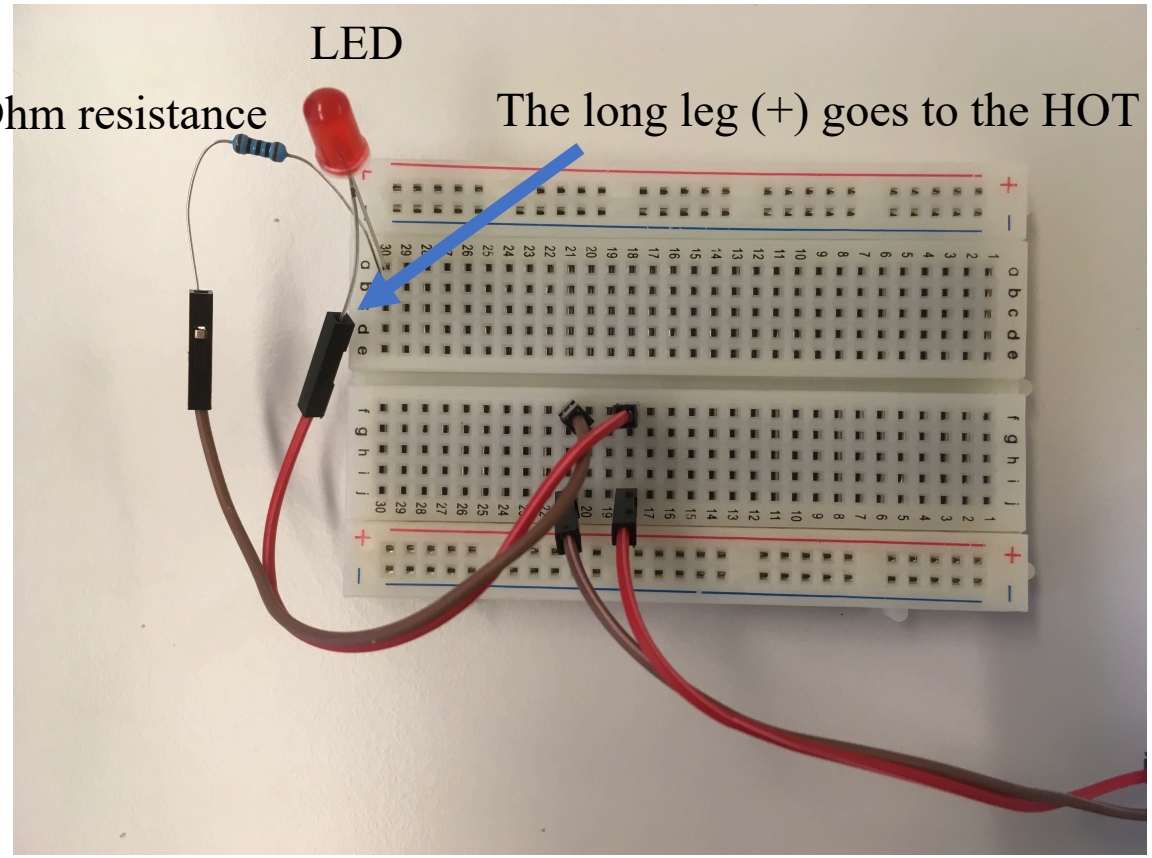
If everything worked ok…

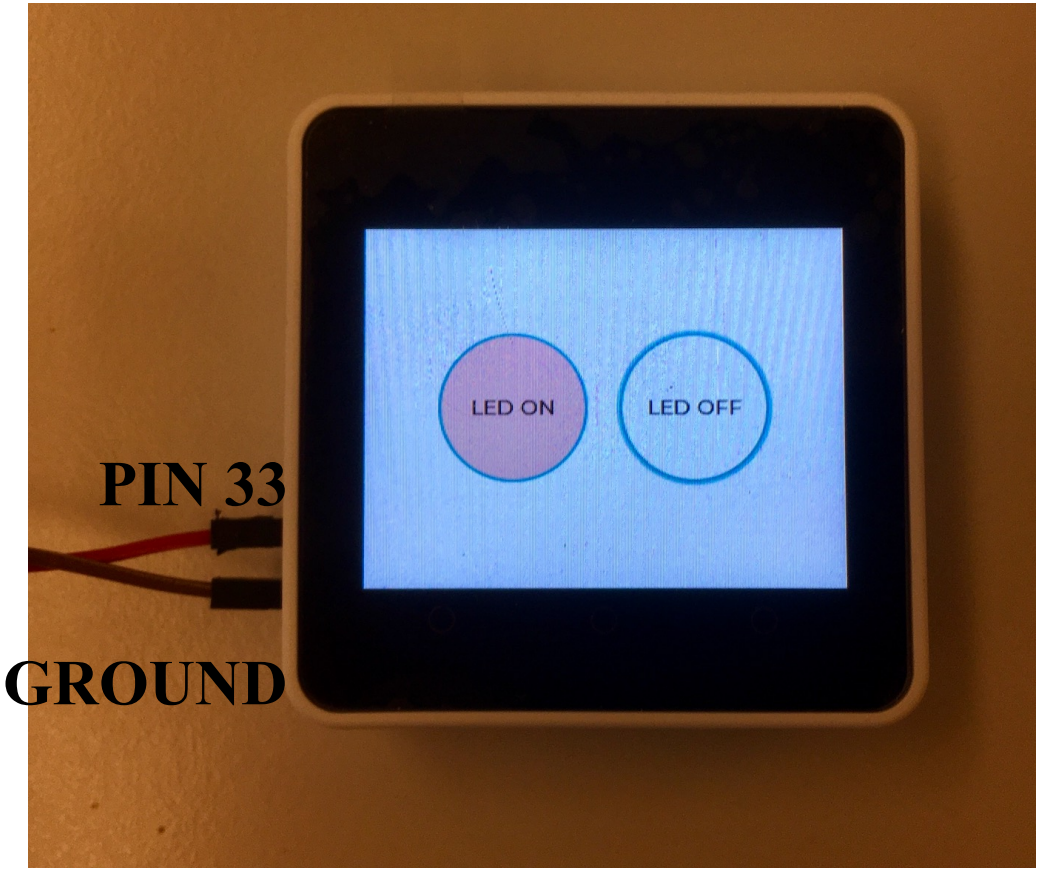


For those that like to tinker things…let's control a LED

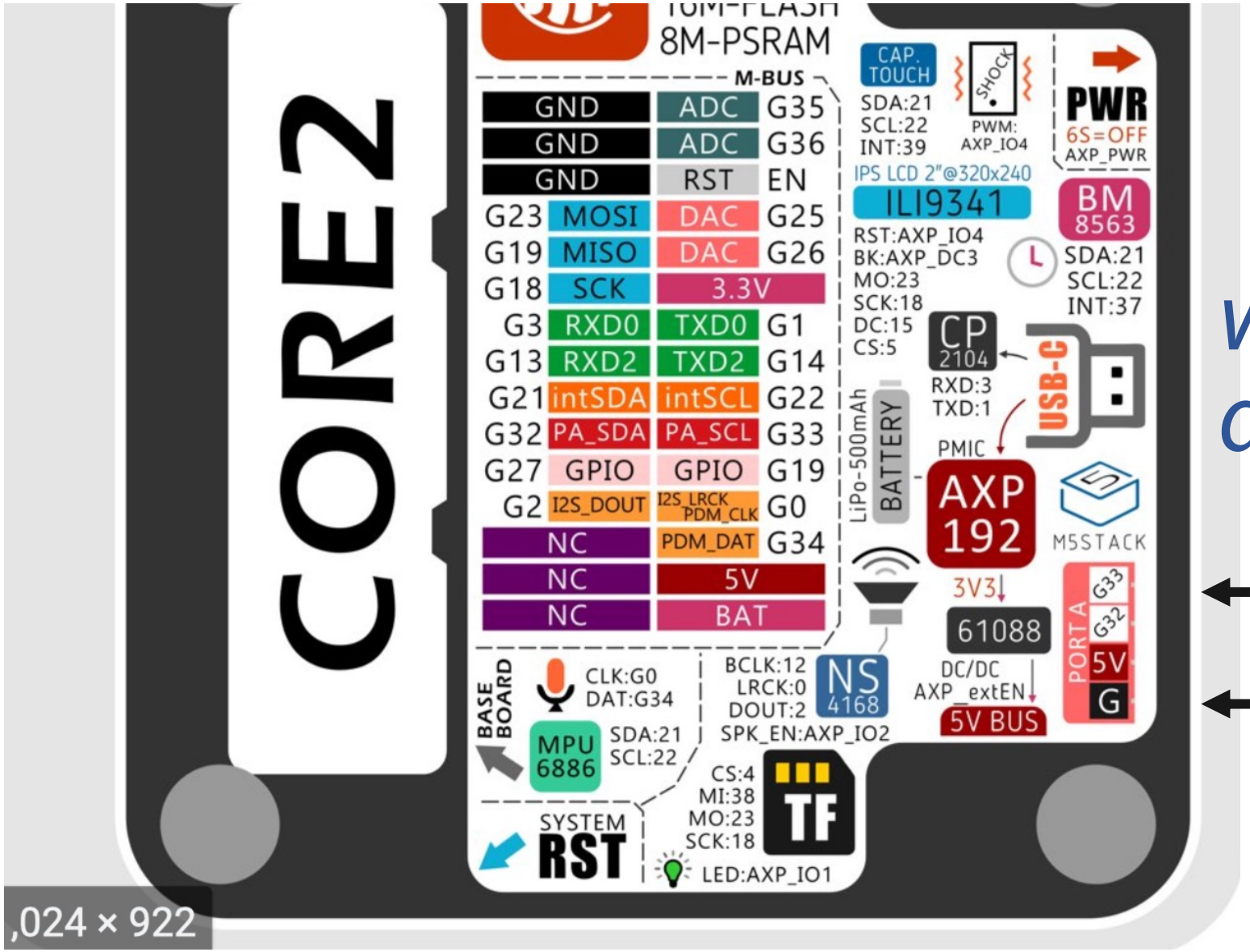
GPIO – Controlling the physical world

- We will control an LED. Turn it ON and OFF.





(just for reference)



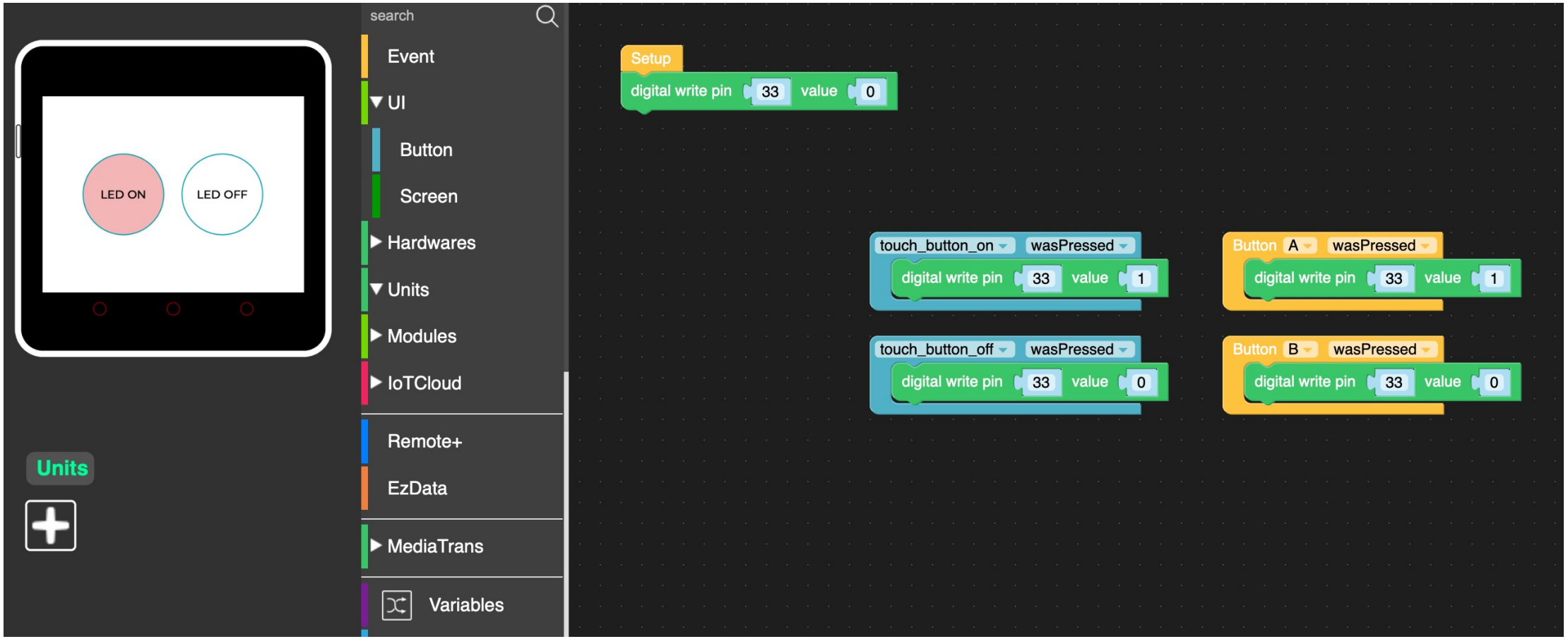
what we just did…

PIN 33

GROUND

GPIO – Controlling the physical world

- The program is super-simple. When button is pressed turn the pin 33 to ON/OFF respectively (i.e., send 5V or not)



Tutorials on YouTube

Selected Lessons on YouTube

- Lesson 6: A plant monitor (soil moisture unit)
- Lesson 7: Using the temperature/humidity sensor
- Lesson 8: A digital Watch
- Lesson 9: Using the Accelerometer
- Lesson 10: Using the Potentiometer
- Weather Station with urequests and json
- Lesson 12: Remote Control a Logo RC device
- Lesson 14: Intruder Alarm with Uiflow and IFTTT
- Lesson 16: A music player
- Lesson 19: Building a game

Terminology

Terminology

- IoT = Internet of Things
- ESP32
- Arduino
- Raspberry Pi
- MQTT (Message Queuing Telemetry Transport)
- pub-sub