

# Centre de formació i ocupació

**Enginyers**  
Industrials de Catalunya

**Curs pràctic de programació d'electrònica  
"embedded low-cost" per a la IoT**

# Curs pràctic de programació d'electrònica "embedded low- cost" per a la IoT

<https://formacio.eic.cat/cursos/1123418>

## Grup de Treball IoT & Embedded Systems

**Xavier Pi**

*Vicepresident GT IoT & Embedded Systems*  
*Comissió Indústria 4.0 Enginyers de Catalunya*  
*[www.linkedin.com/in/xavierpi](http://www.linkedin.com/in/xavierpi)*

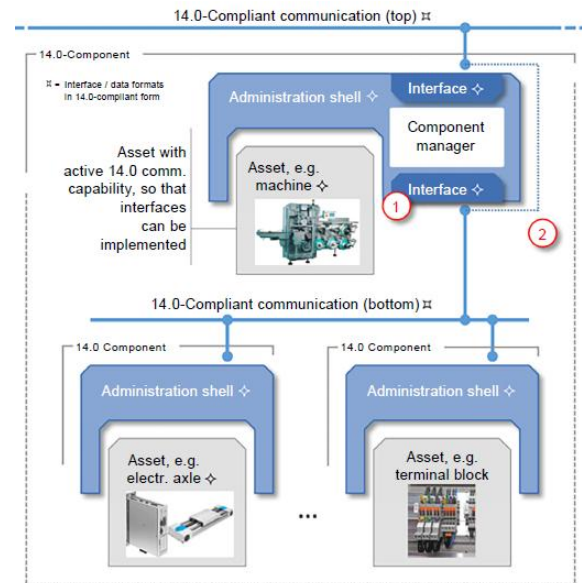
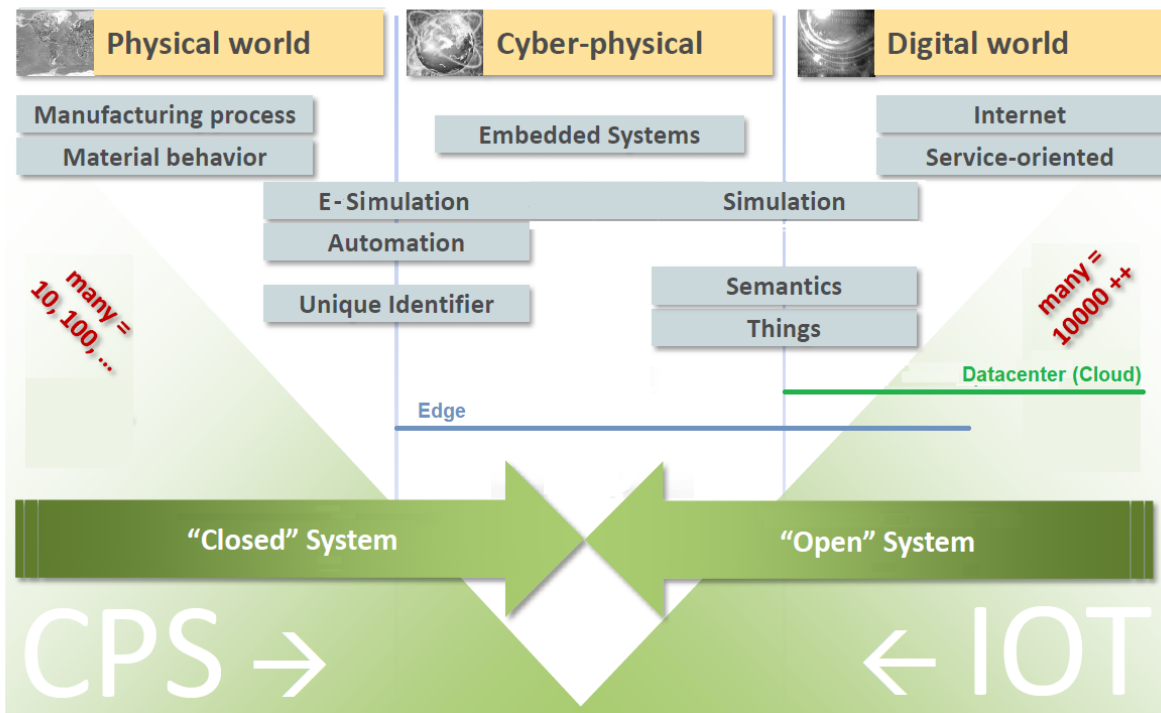
# Programa

- Els embedded systems
- Exemples bàsics d'actuadors
- Exemples bàsics de sensòrica
- Protocols de comunicacions IoT: MQTT i HTTP
- Sensors i actuadors en entorns industrials
- Ciberseguretat
- Exemples avançats

# ELS EMBEDDED SYSTEMS

# Elements centrals de la Indústria 4.0

Two Worlds coming together



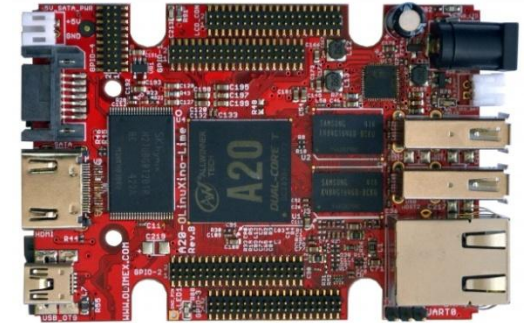
Nestability of I4.0 components

IEC PAS 63088

# Tipus d'embedded systems

- Classificació del prof. Raj Kamal
  - Escala petita (Ex. Arduino)
  - Escala mitjana (Ex. ESP32)
  - Escala sofisticada (Ex. Raspberry Pi, Olimex)

# Tipus d'embedded systems



# Proveïdors

- [www.diotronic.com](http://www.diotronic.com)
- [www.industrialshields.com](http://www.industrialshields.com)
- [www.amazon.com](http://www.amazon.com)
- [www.aliexpress.com](http://www.aliexpress.com)



# Llenguatges low-code de blocs (Snap!)

The screenshot displays the Snap! IDE interface for a Scratch project named "HelloWorld". The script area contains the following blocks:

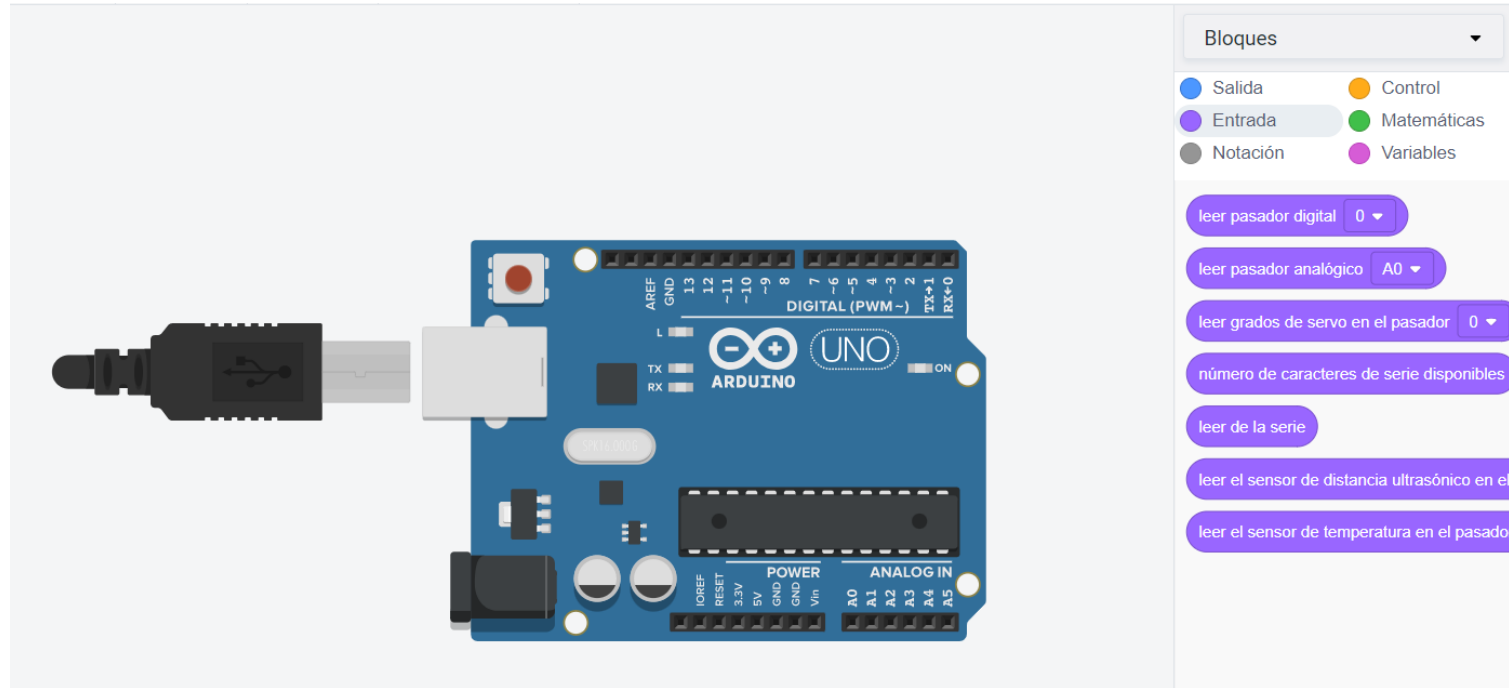
- Sprite (draggable)
- Scripts: PubSub Hello World!
- Control: set broker to broker.emqx.io:8084/mqtt
- Control: MQTT connect broker
- Control: MQTT sub broker topicName
- Control: say payload for 2 secs input names: topic payload
- Control: MQTT pub broker topicName Hello-World!
- Control: Client-server Hello World!
- Control: MQTT response broker topicName
- Control: example MQTT response reporter topic payload input names: topic payload
- Control: set resp to MQTT request broker topicName timer Hello
- Control: say resp for 2 secs

The right-hand preview window shows a variable named "broker" with the value "broker.emqx.io:8084/mqtt". Below the preview is a "Sprite" block and an "Escenari" (Stage) area.

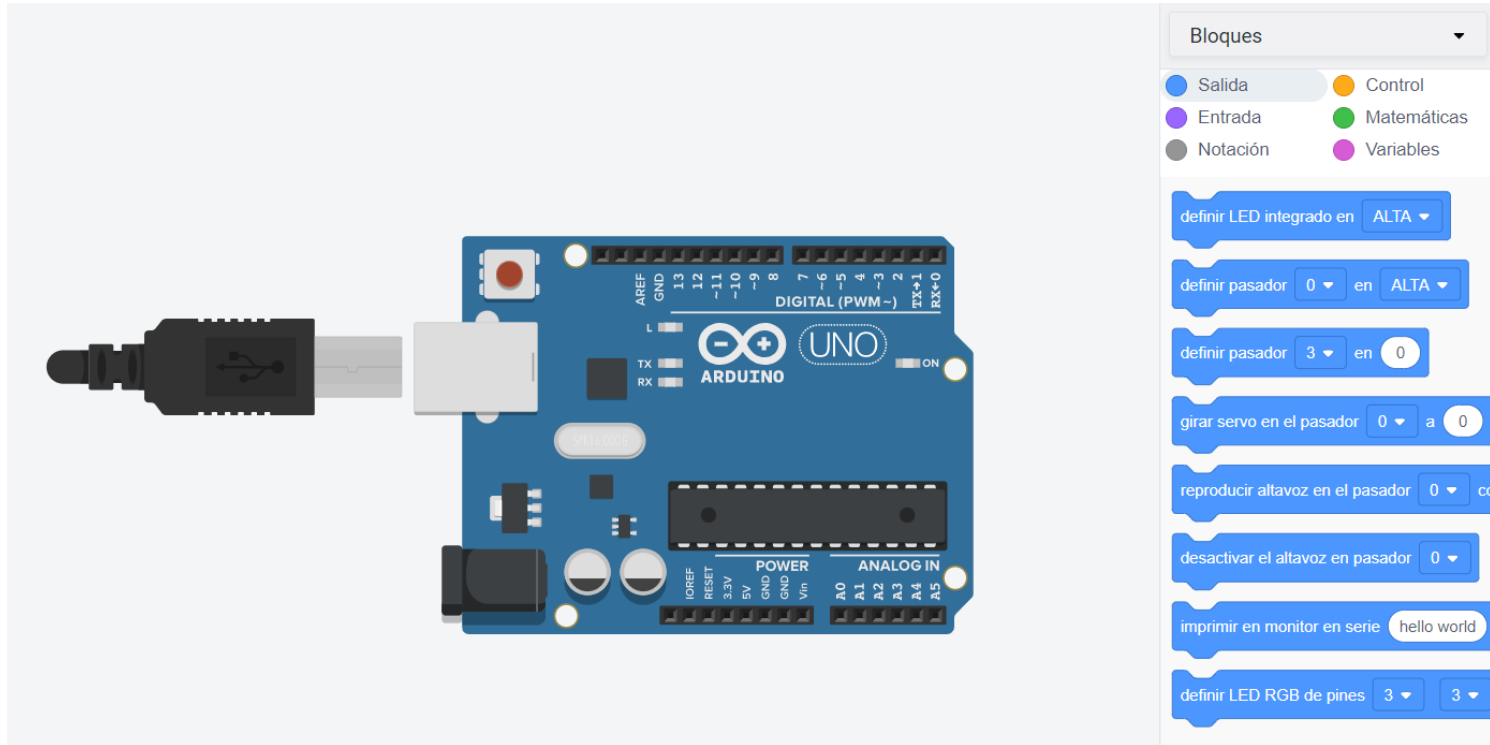
# L'esquema setup() - loop()

```
main() {  
    setup();  
    while (1) {  
        loop();  
    }  
}
```

# Entrades analògiques i digitals



# Sortides analògiques i digitals

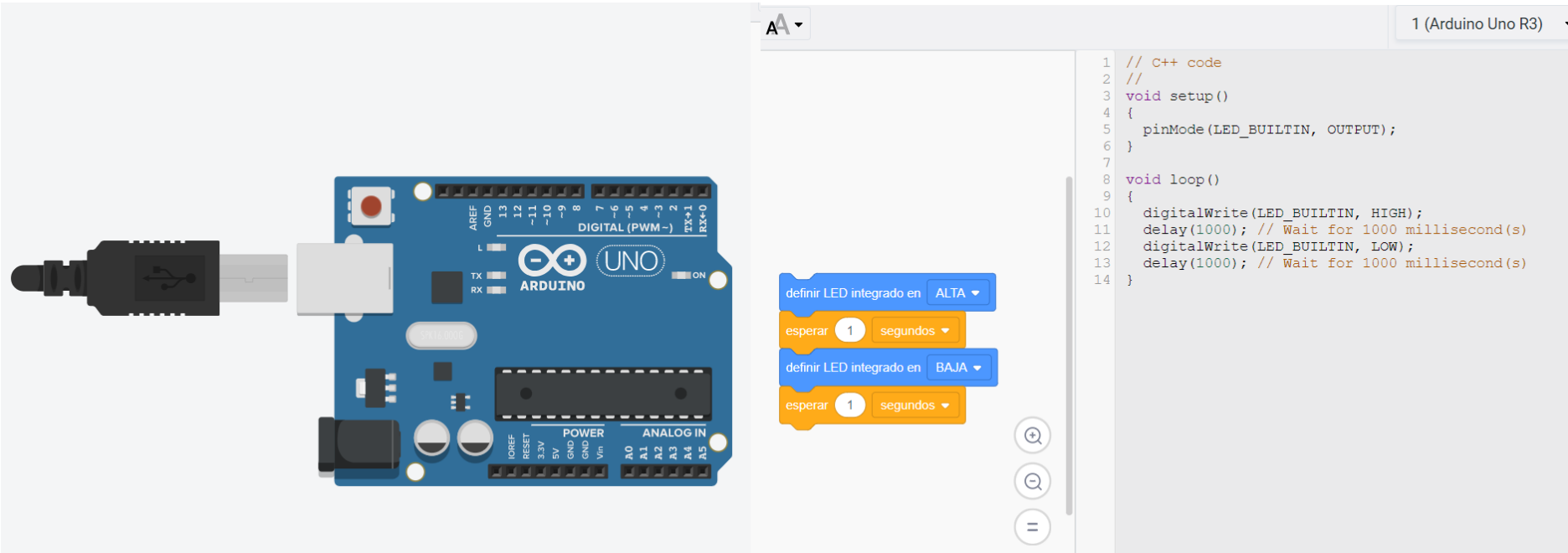


The image shows an Arduino Uno board with a USB cable plugged into it. To the right, a block diagram of an analog output program is displayed. The program consists of the following blocks:

- definir LED integrado en ALTA
- definir pasador 0 en ALTA
- definir pasador 3 en 0
- girar servo en el pasador 0 a 0
- reproducir altavoz en el pasador 0 cc
- desactivar el altavoz en pasador 0
- imprimir en monitor en serie hello world
- definir LED RGB de pines 3 3

# EXEMPLES BÀSICS D'ACTUADORS

# Control d'un led (blink)



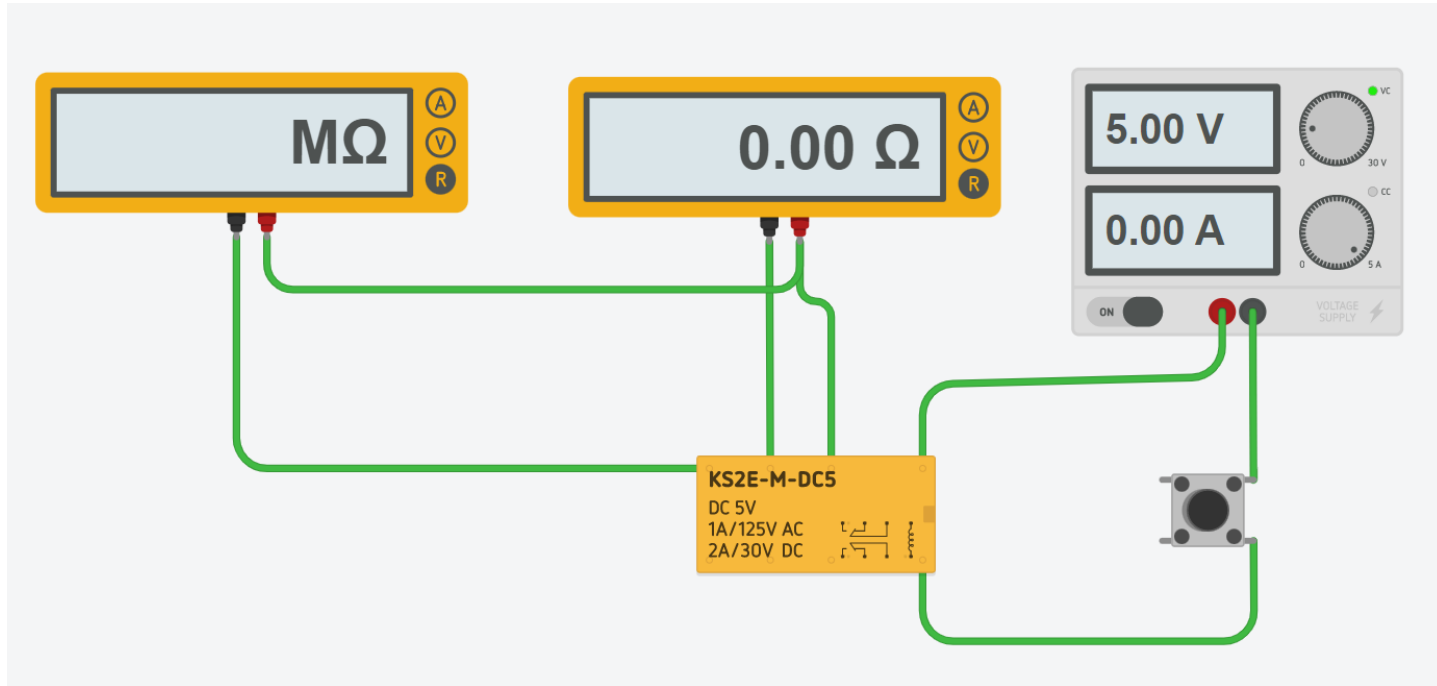
The image displays an Arduino Uno R3 board on the left and its IDE interface on the right. The board is blue with a USB Type-C port, a USB Type-A port, a DC power jack, and various digital and analog pins. The IDE interface shows a code editor with the following C++ code for a blink program:

```
1 // C++ code
2 //
3 void setup()
4 {
5   pinMode(LED_BUILTIN, OUTPUT);
6 }
7
8 void loop()
9 {
10  digitalWrite(LED_BUILTIN, HIGH);
11  delay(1000); // Wait for 1000 millisecond(s)
12  digitalWrite(LED_BUILTIN, LOW);
13  delay(1000); // Wait for 1000 millisecond(s)
14 }
```

Below the code editor, there is a block of visual programming code consisting of four blocks: two blue blocks labeled "definir LED integrado en" with dropdown menus set to "ALTA" and "BAJA", and two orange blocks labeled "esperar 1 segundos".

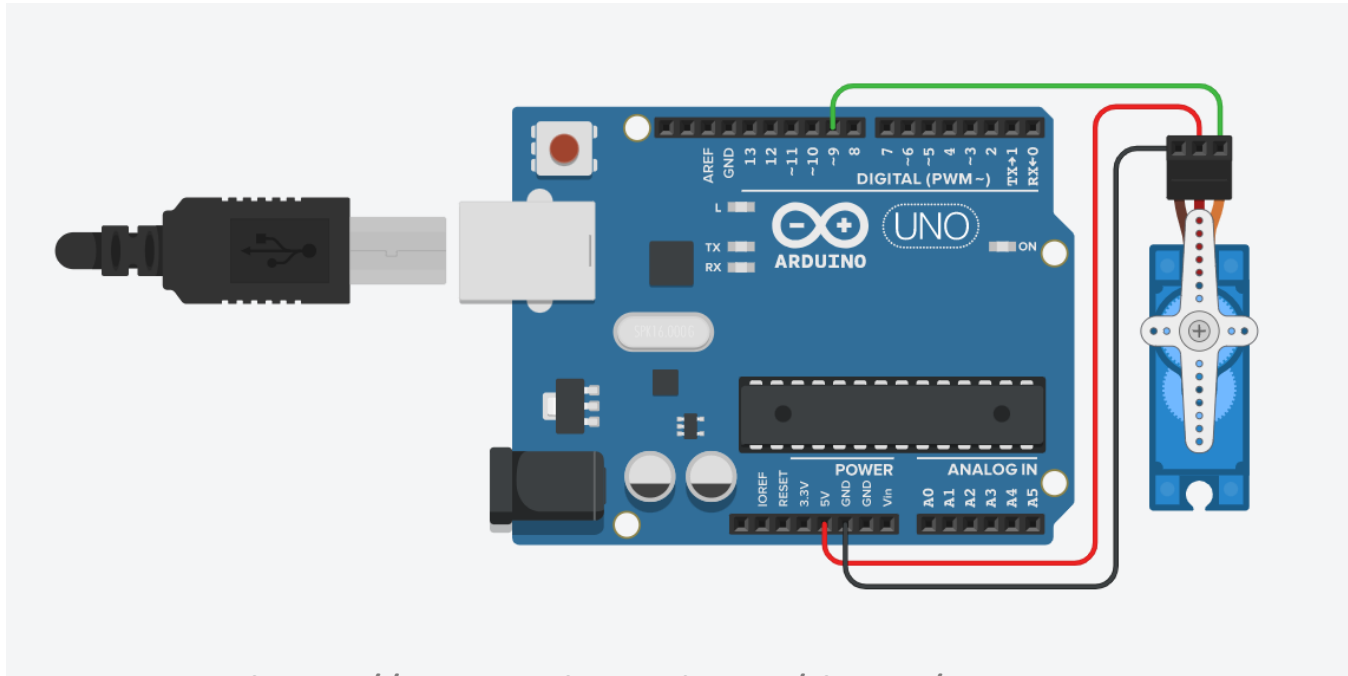
Exercici: Afegir un LED extern, amb la resistència corresponent.

# Control d'un relé



<https://www.tinkercad.com/things/1O4C6ZoLpdB>

# Control d'un servomotor

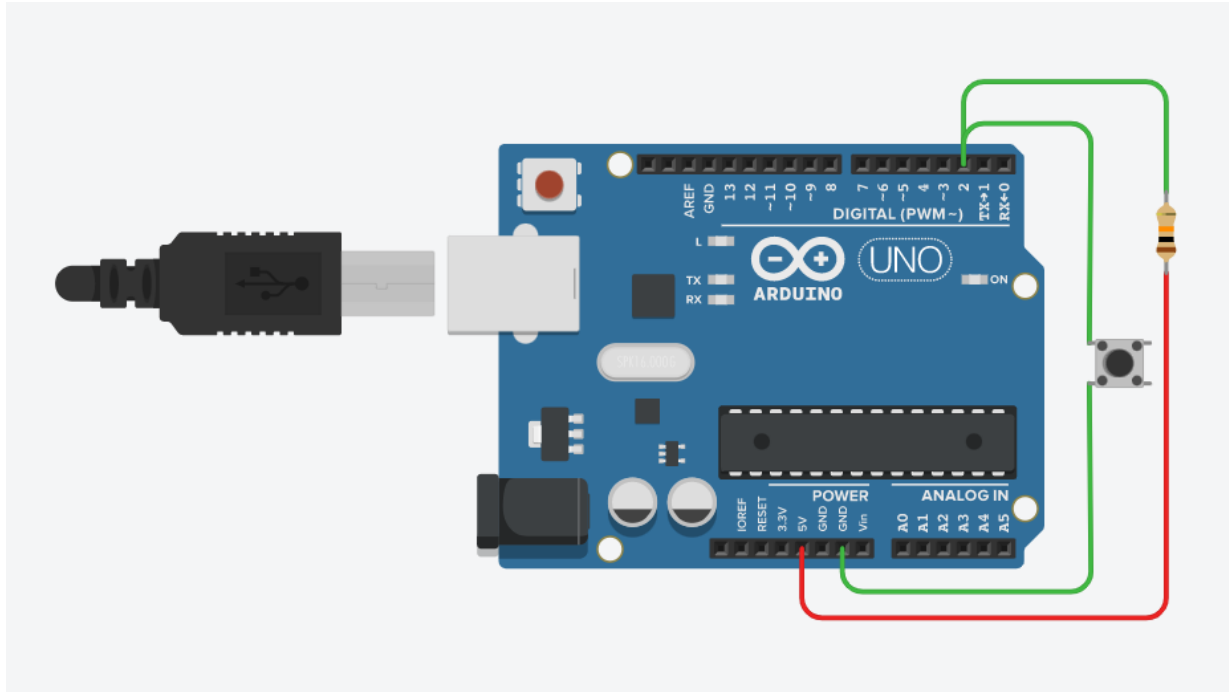


<https://www.tinkercad.com/things/3NPlinNEm8D>



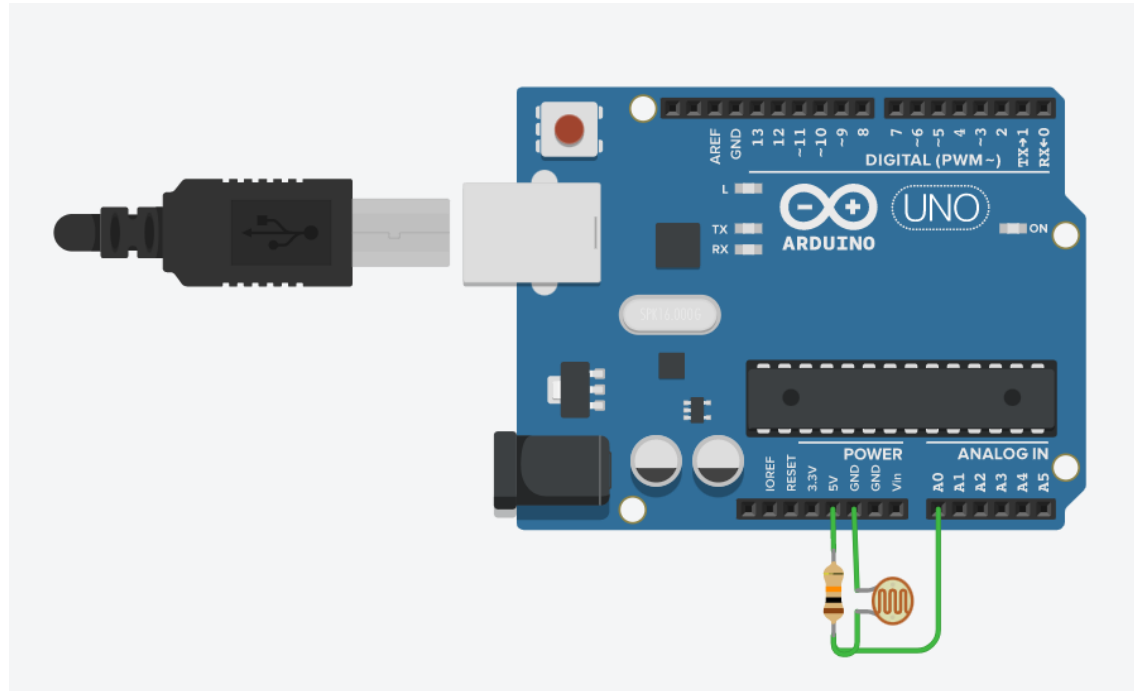
# EXEMPLES BÀSICS DE SENSÒRICA

# Detecció d'un final de carrera



<https://www.tinkercad.com/things/jcljb4UQUdP>

# Mesura d'un nivell de lluminositat



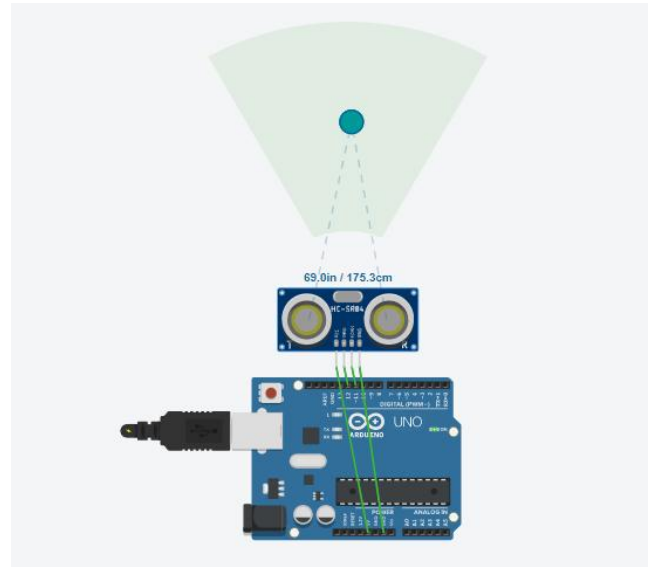
<https://www.tinkercad.com/things/eZBCioTaA5V>

# Detecció d'un llindar de lluminositat

- Exercici:

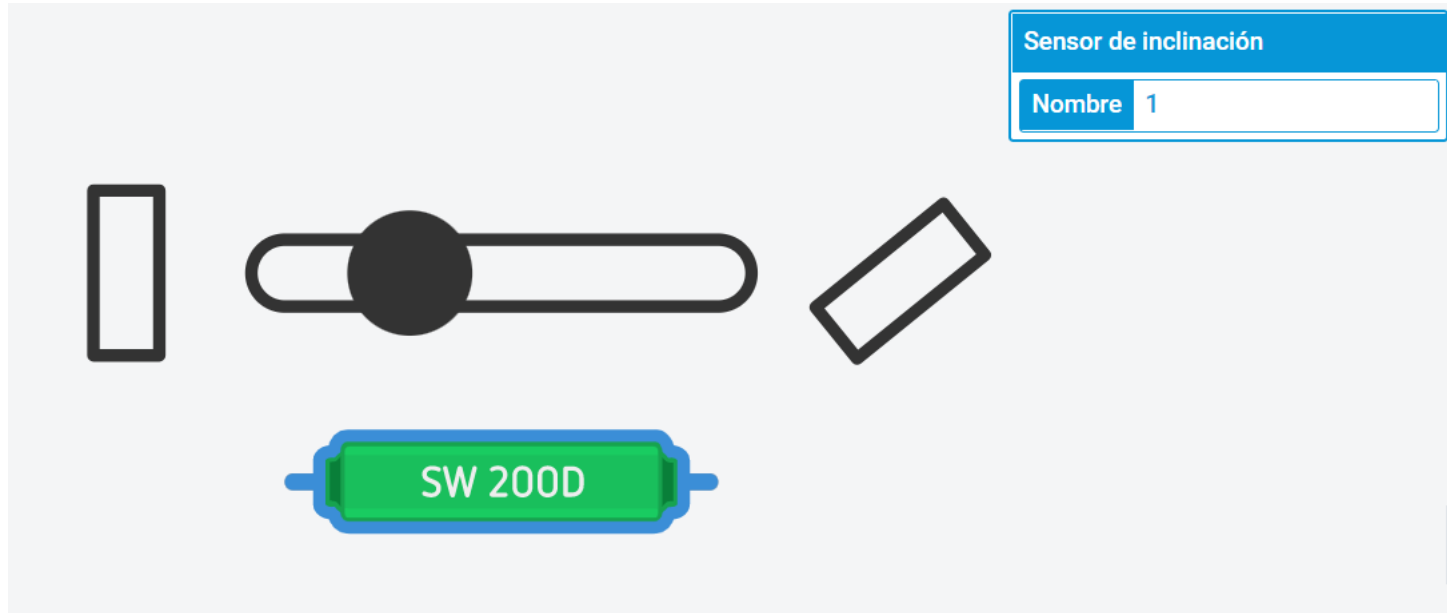
Encendre/apagar un LED si traspasa un valor llindar de lluminositat.

# Mesura d'una distància



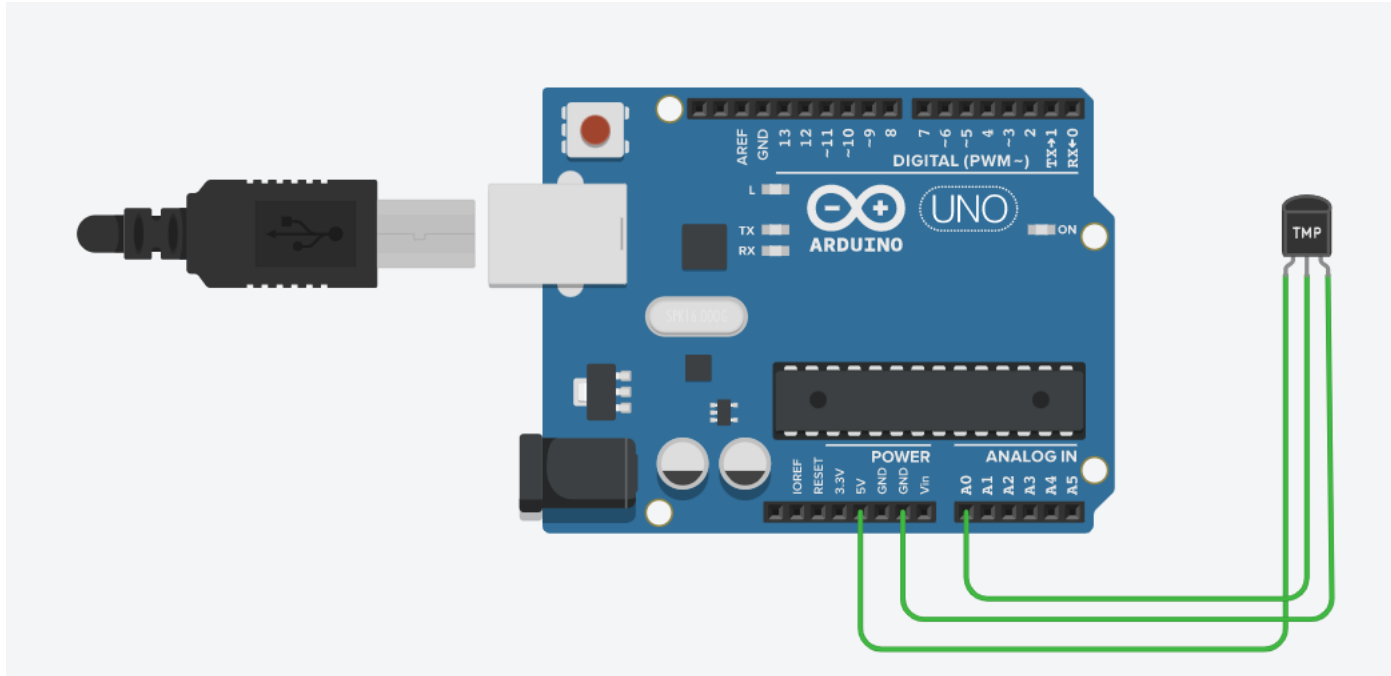
<https://www.tinkercad.com/things/9I2rZw0cjLS>

# Mesura d'inclinacions (acceleracions)



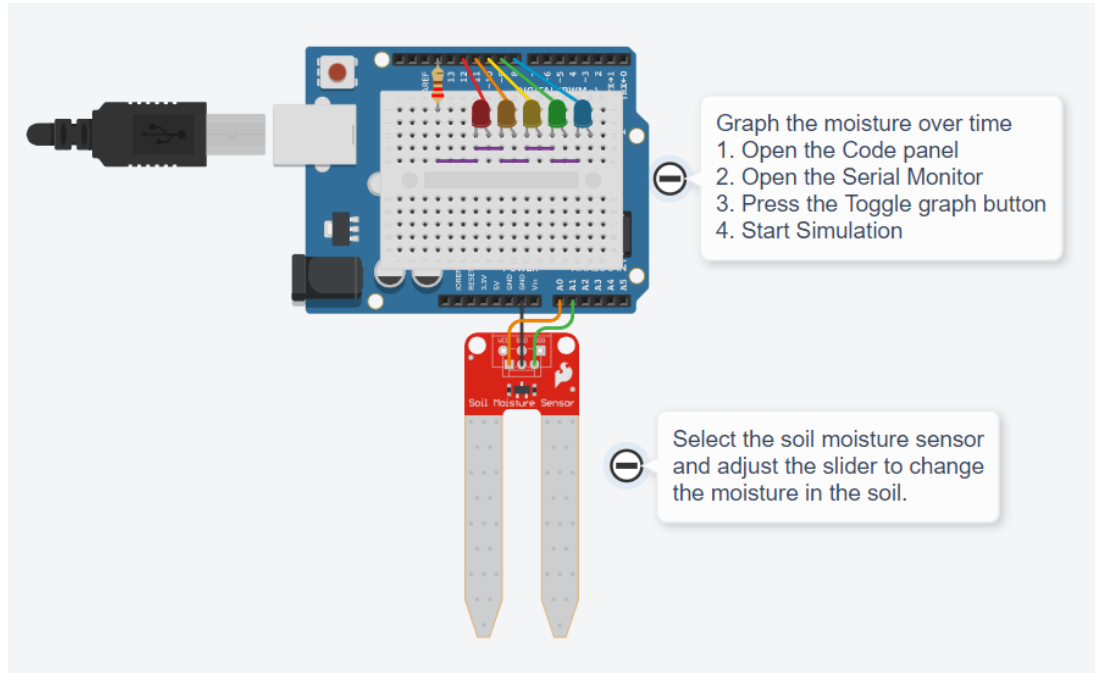
Exercici: Analitzar el funcionament amb el multímetre

# Mesura temperatura



<https://www.tinkercad.com/things/6M4rpEIIvRE>

# Mesura d'humitat

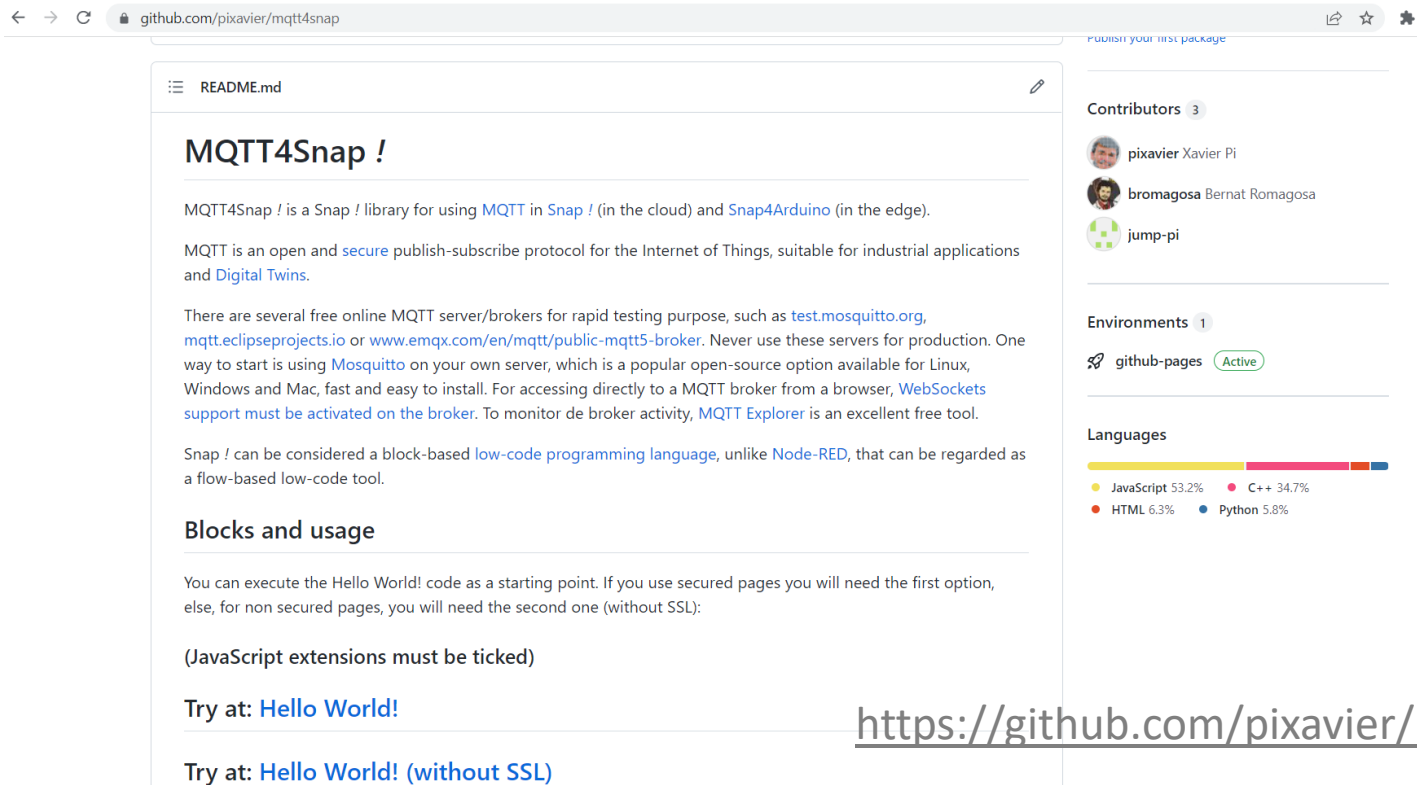


Moisture Arduino starter. Exercici: Analitzar i entendre l'exemple.



# PROTOCOLS DE COMUNICACIONS IOT: MQTT I HTTP

# Protocol MQTT. PubSub vs client-server



← → ↻ [github.com/pixavier/mqtt4snap](https://github.com/pixavier/mqtt4snap) Publish your first package

☰ README.md ✎

## MQTT4Snap !

MQTT4Snap ! is a Snap ! library for using MQTT in Snap ! (in the cloud) and Snap4Arduino (in the edge).

MQTT is an open and [secure](#) publish-subscribe protocol for the Internet of Things, suitable for industrial applications and [Digital Twins](#).

There are several free online MQTT server/brokers for rapid testing purpose, such as [test.mosquitto.org](#), [mqtt.eclipseprojects.io](#) or [www.emqx.com/en/mqtt/public-mqtt5-broker](#). Never use these servers for production. One way to start is using [Mosquitto](#) on your own server, which is a popular open-source option available for Linux, Windows and Mac, fast and easy to install. For accessing directly to a MQTT broker from a browser, [WebSockets support must be activated on the broker](#). To monitor de broker activity, [MQTT Explorer](#) is an excellent free tool.

Snap ! can be considered a block-based [low-code programming language](#), unlike [Node-RED](#), that can be regarded as a flow-based low-code tool.

## Blocks and usage




You can execute the Hello World! code as a starting point. If you use secured pages you will need the first option, else, for non secured pages, you will need the second one (without SSL):

(JavaScript extensions must be ticked)


Try at: [Hello World!](#)

Try at: [Hello World! \(without SSL\)](#)


**Contributors** 3

-  **pixavier** Xavier Pi
-  **bromagosa** Bernat Romagosa
-  **jump-pi**

**Environments** 1

-  **github-pages** Active

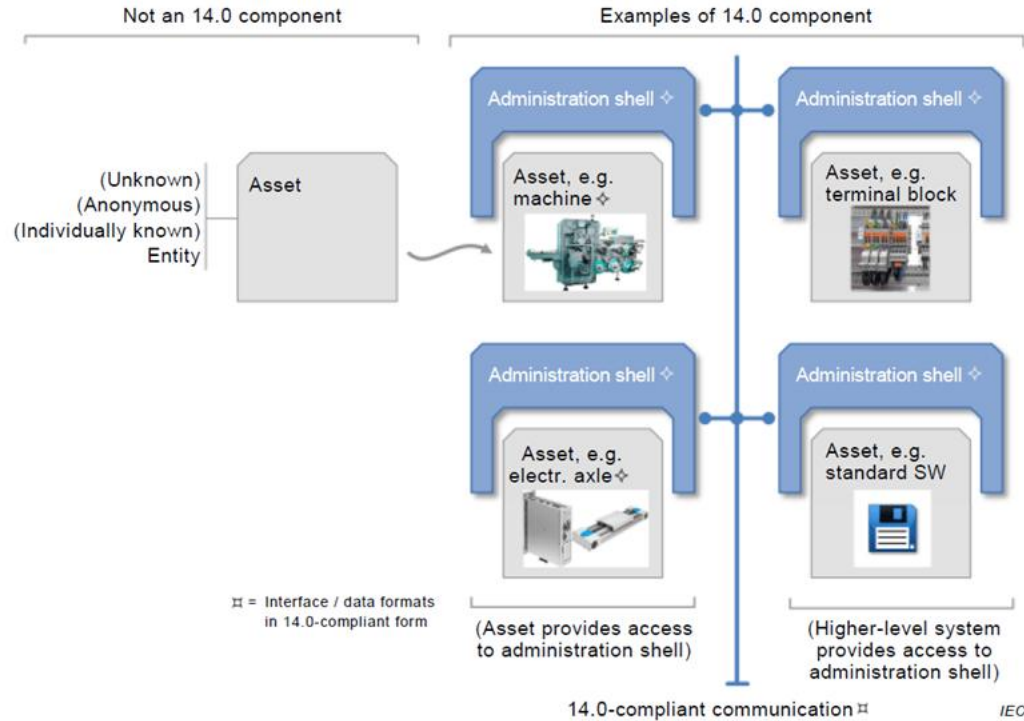
**Languages**



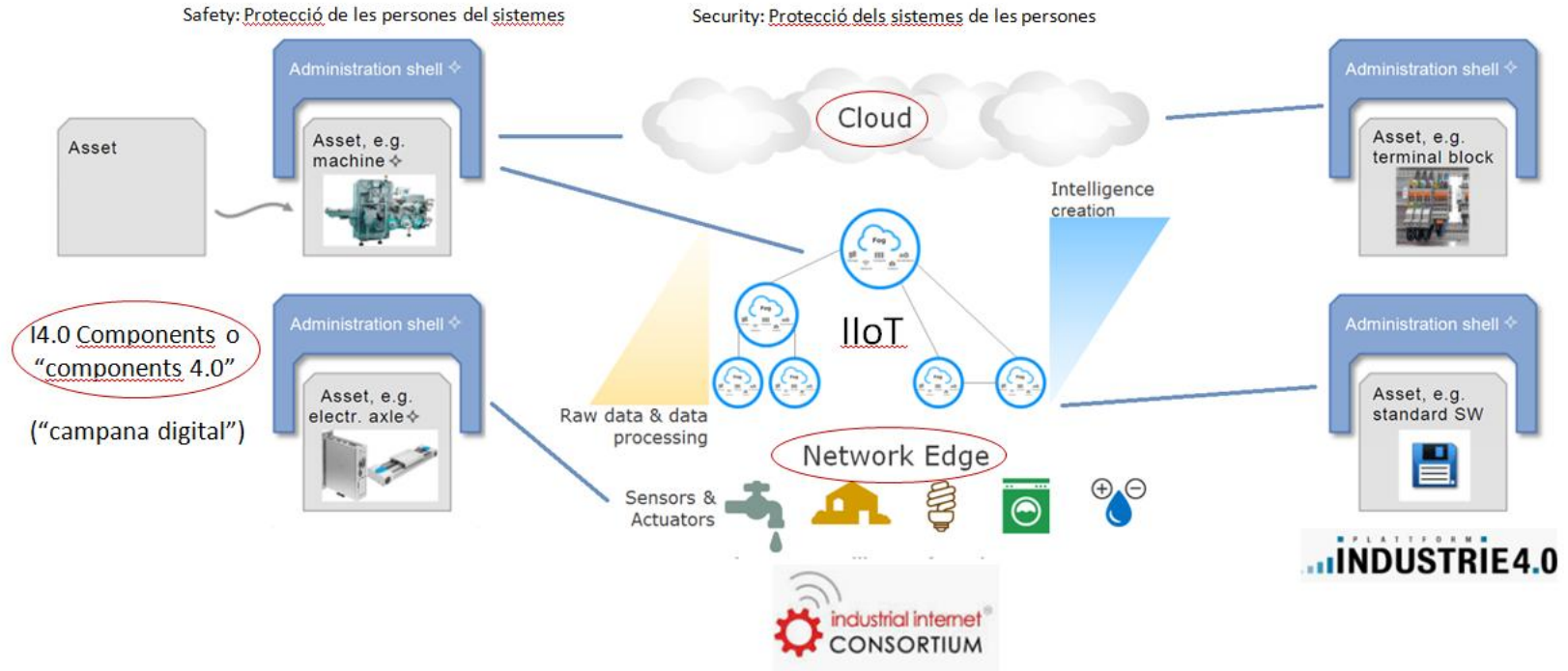
Language	Percentage
JavaScript	53.2%
C++	34.7%
HTML	6.3%
Python	5.8%

<https://github.com/pixavier/mqtt4snap>

# Noció de núvol. Cloud i Edge computing



# Noció de núvol. Cloud i Edge computing



# Llenguatges low-code de fluxos (Node-RED)

The screenshot shows the Node-RED web interface in a browser. The address bar displays the URL: `vps656540.ovh.net:51880/#flow/d775676113e223e0`. The interface includes a left sidebar with a node palette, a central workspace for building flows, and a right sidebar with a debug console. The flow, titled "Flow 1", consists of two nodes: a "timestamp" node (blue) connected to a "msg.payload" node (green). The debug console on the right shows the following output:

```
node: 417da6efa7f5b383  
msg.payload: number  
1651022933026
```

<http://vps656540.ovh.net:5x880>

# Connexió dels exemples al núvol

## Noció de Bessó Digital (Digital Twin)

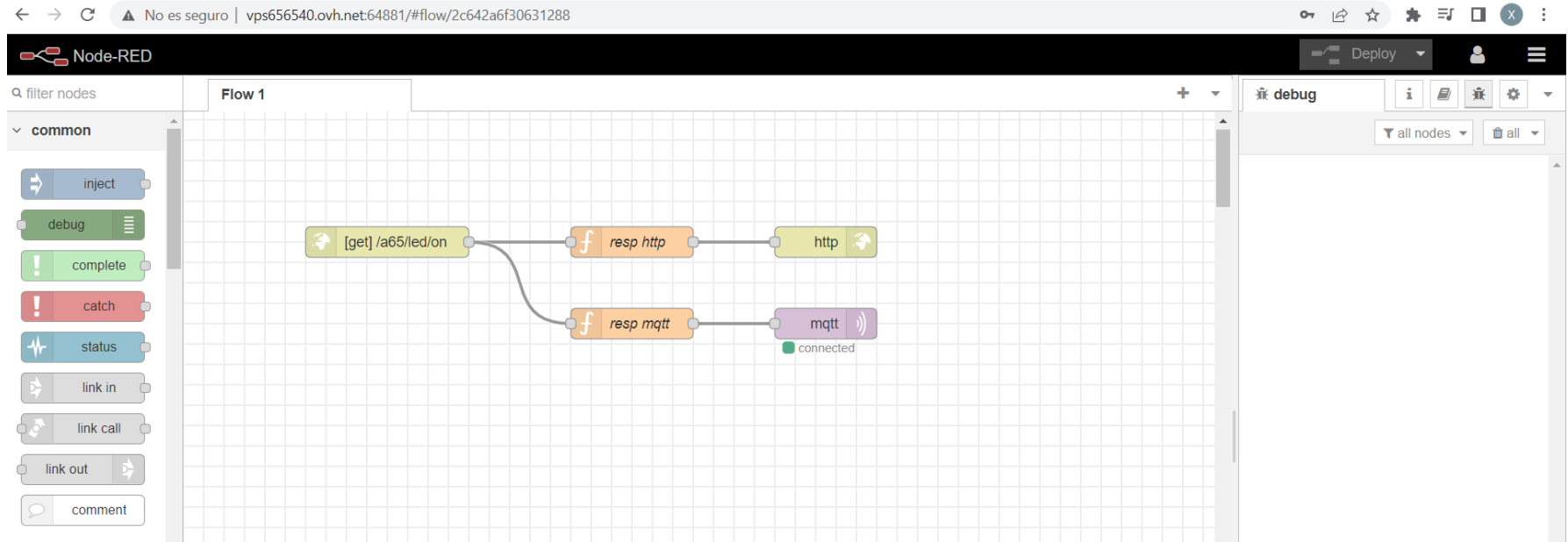
The screenshot displays the Snap! IDE interface for a Scratch project named 'esp32\_DT'. The left sidebar shows the 'Motion' category selected. The main workspace contains a Scratch script with the following blocks:

- set broker to broker.emqx.io:8083/mqtt
- MQTT connect broker
- switch to costume esp32off
- MQTT sub broker topicName
- If payload = on: switch to costume esp32on
- If payload = off: switch to costume esp32off
- MQTT pub broker topicName on
- MQTT pub broker topicName off

The visual area on the right shows a digital representation of an ESP32 board with a red LED and a resistor connected to it. The bottom right corner of the IDE shows a 'Sprite' area with a 'Sprite' icon and an 'Escenari' (Stage) area with an 'Escenari' icon.

[http://vps656540.ovh.net/snap/snap.html#open:esp32\\_DT.xml](http://vps656540.ovh.net/snap/snap.html#open:esp32_DT.xml)

# Protocol HTTP. Noció d'URL. Bridge HTTP-MQTT

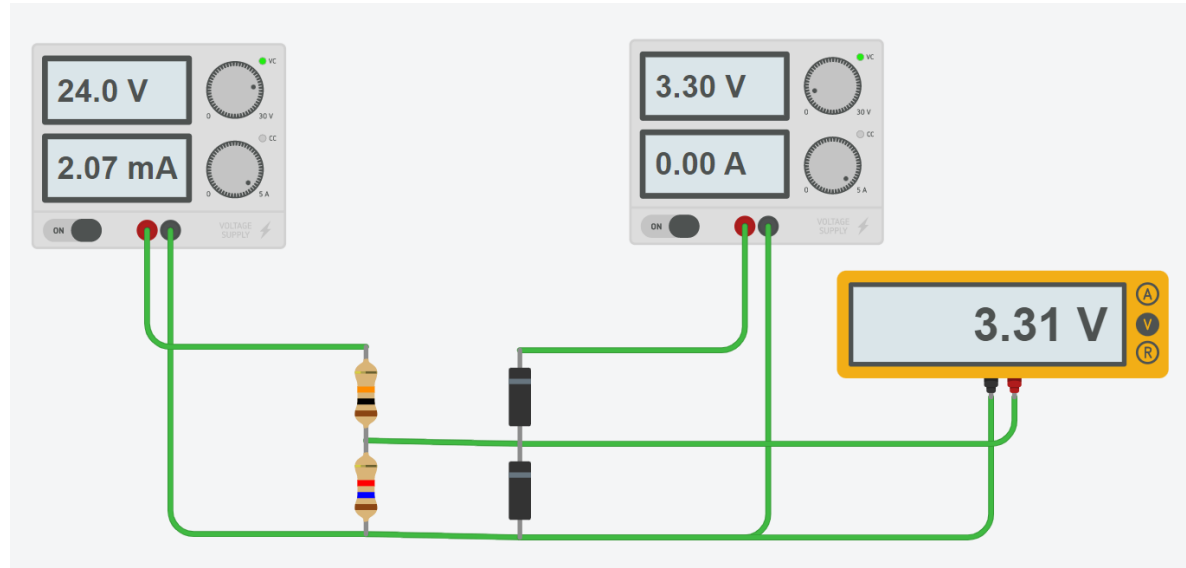


<http://vps656540.ovh.net:64881/a65/led/on>

# **SENSORS I ACTUADORS EN ENTORNS INDUSTRIALS**

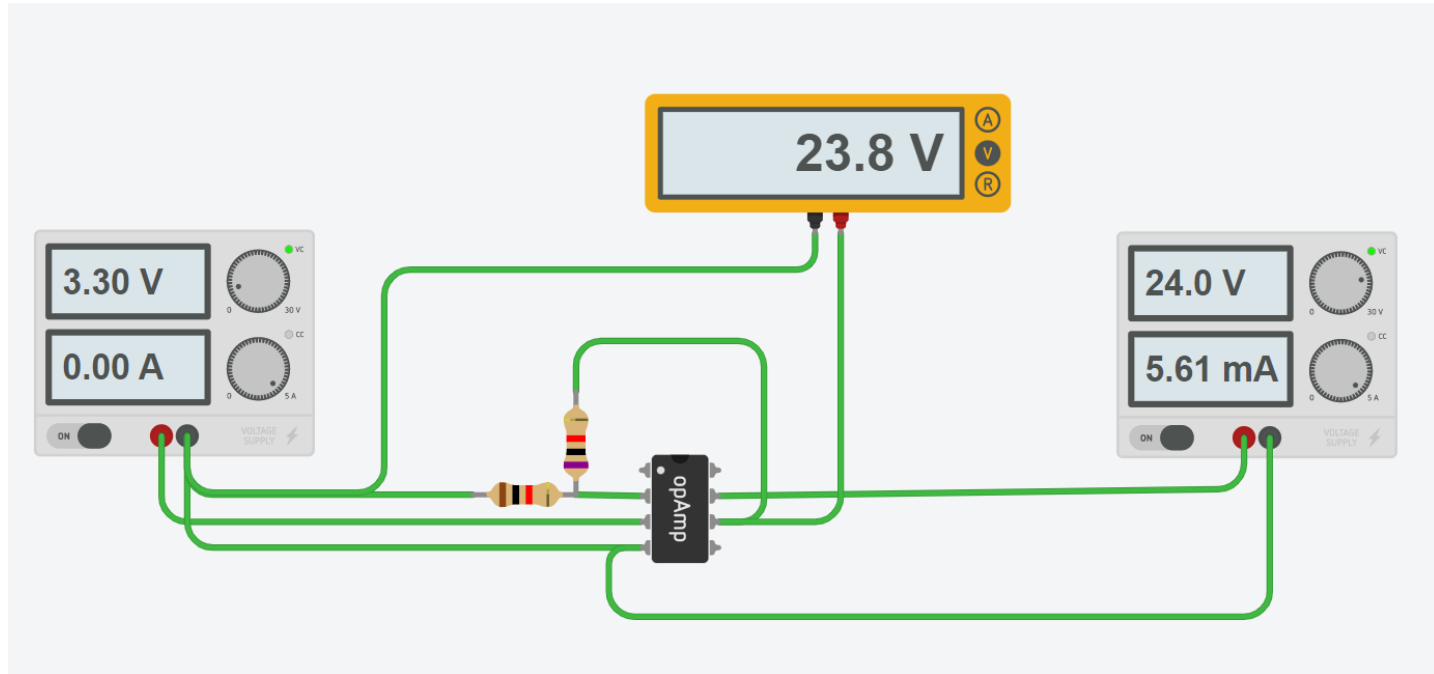


# Entrades analògiques i digitals. 12V-24V



<https://www.tinkercad.com/things/428kNkFBRgR>

# Control de dispositius 0-10V, 24V



<https://www.tinkercad.com/things/2QMMBjVqQFp>

# Gràcies !

Xavier Pi

[xpi@enginyers.net](mailto:xpi@enginyers.net)

<https://www.eic.cat/content/gt-embedded-systems-iot>

[www.comissioindustria40.cat](http://www.comissioindustria40.cat)

# Centre de formació i ocupació

**Enginyers**  
Industrials de Catalunya