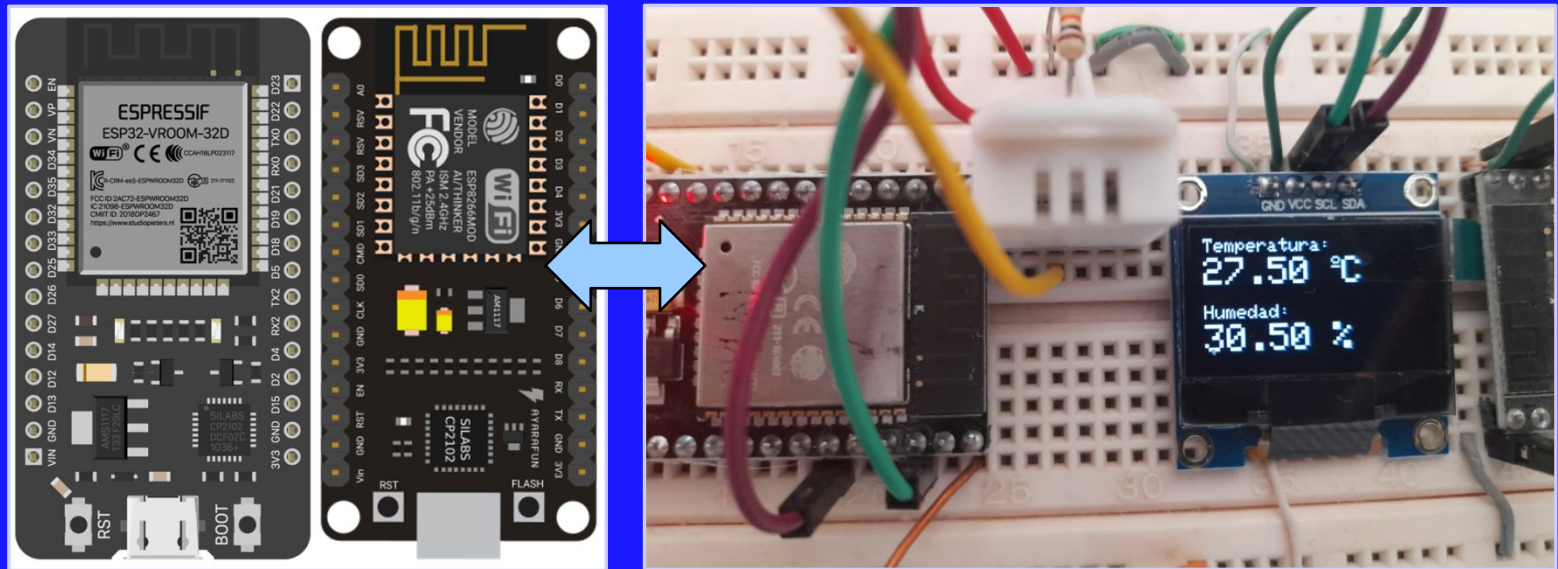


Aplicaciones de sistemas embebidos ESP32 e implementación de un módulo de medición de pH basado en NodeMCU



Parte I (Introducción) - Mg. Ing. Rafael Oliva, Ing. Esp. Nestor Cortez
Instituto de Tecnología Aplicada (ITA-UARG) Area Energías Alternativas

XX JORNADAS DE INFORMÁTICA 2024 – UNPA /UARG
12 de noviembre 2024

CONTENIDOS:

- Interfaz Web en Internet – conceptos básicos y modelo de capas
- Historia de familia ESP
- Uso de ESP8266 y ESP32
- Sistemas embebidos – antecedentes
- Configuraciones posibles de conexión ESP-Sistema Embebido
- Caso demostración – Conexión a SISMED/SJ24 con módulo ESP y con Raspberry Pi4B
- Otros modelos y casos prácticos

Que es un sistema embebido?

“Un sistema embebido (*embedded*) es cualquier aplicación en que una computadora dedicada se construye como parte del sistema”

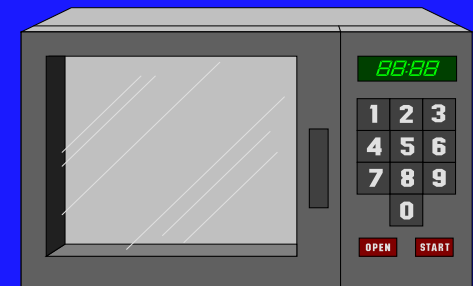
(Jack Ganssle - The Art of Programming Embedded Systems)



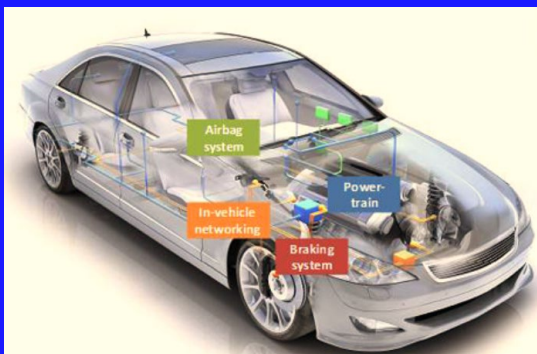
Instrumental



Celular y SmartTV



Microondas

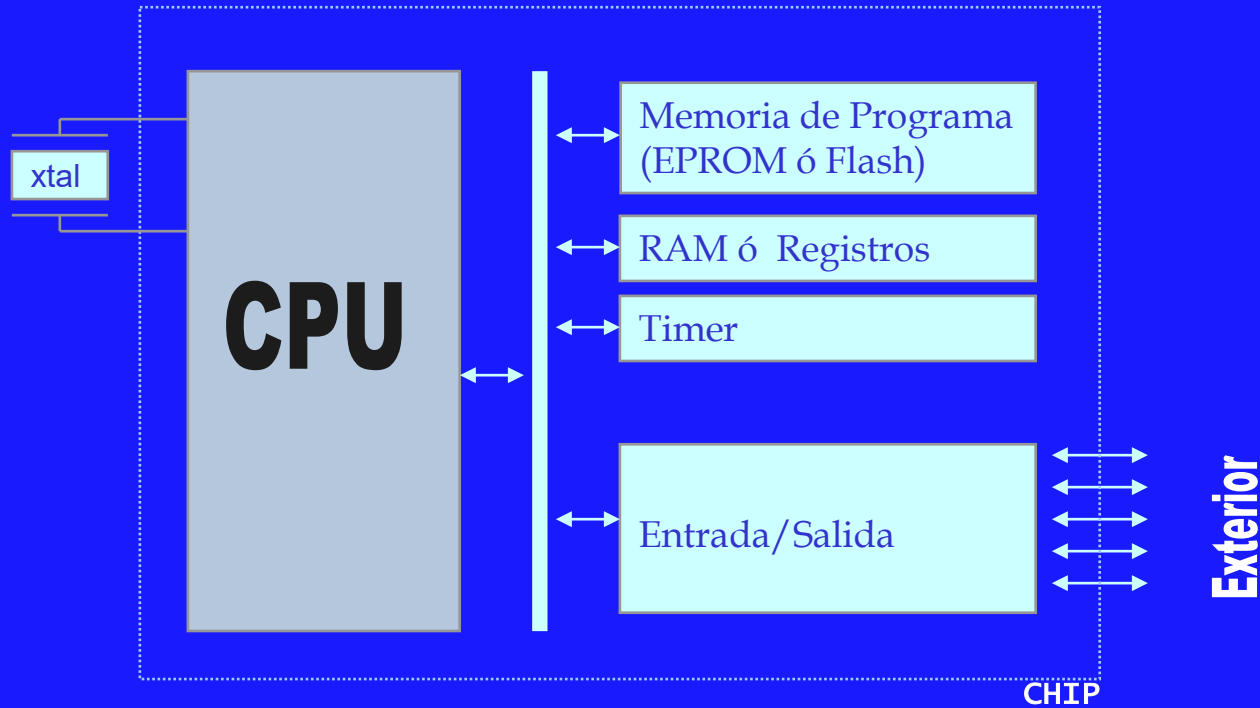


En Vehículos

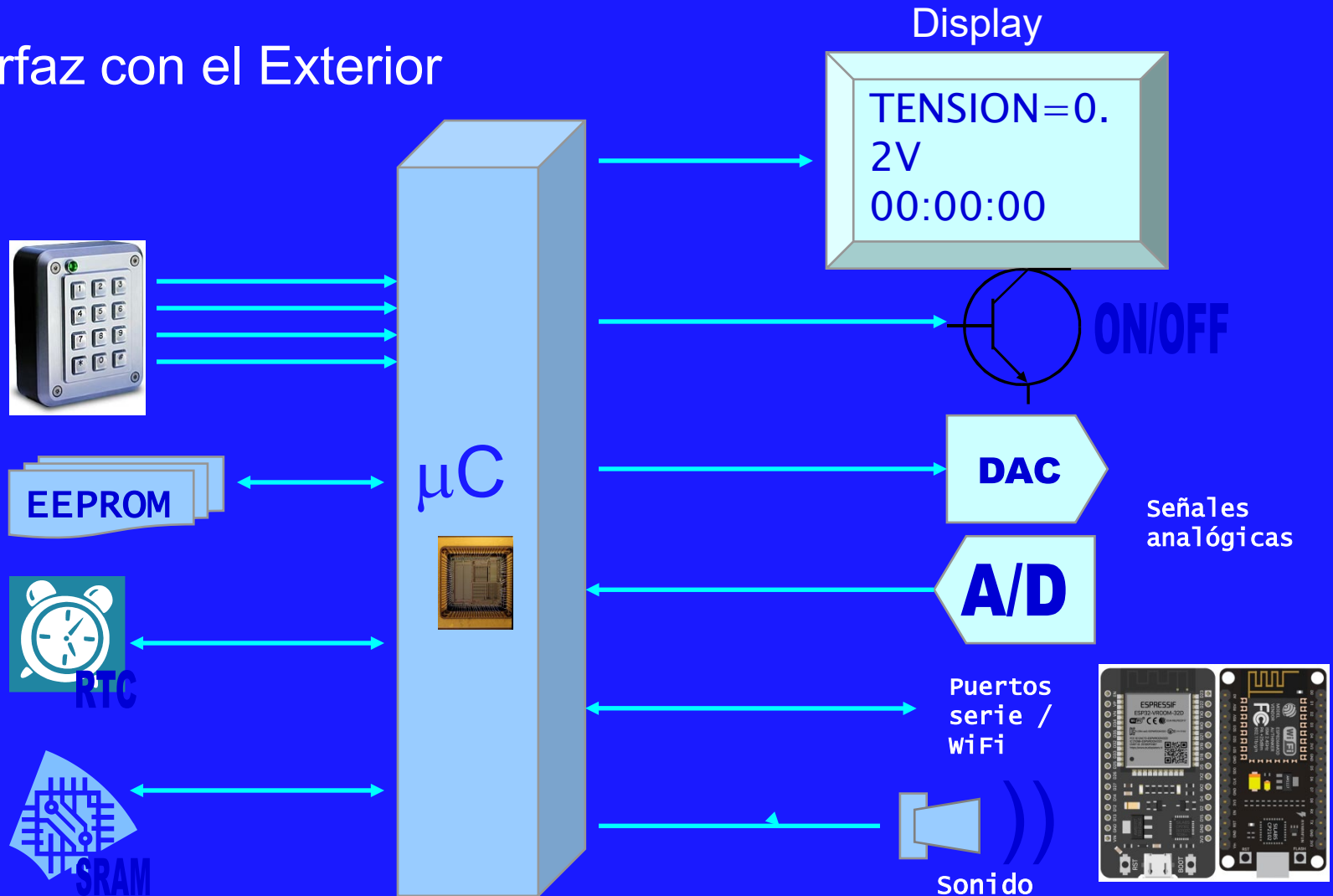


Alarmas

COMPONENTES DE UN MICROCONTROLADOR ELEMENTAL



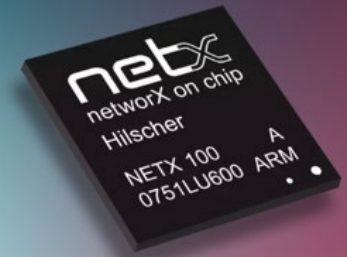
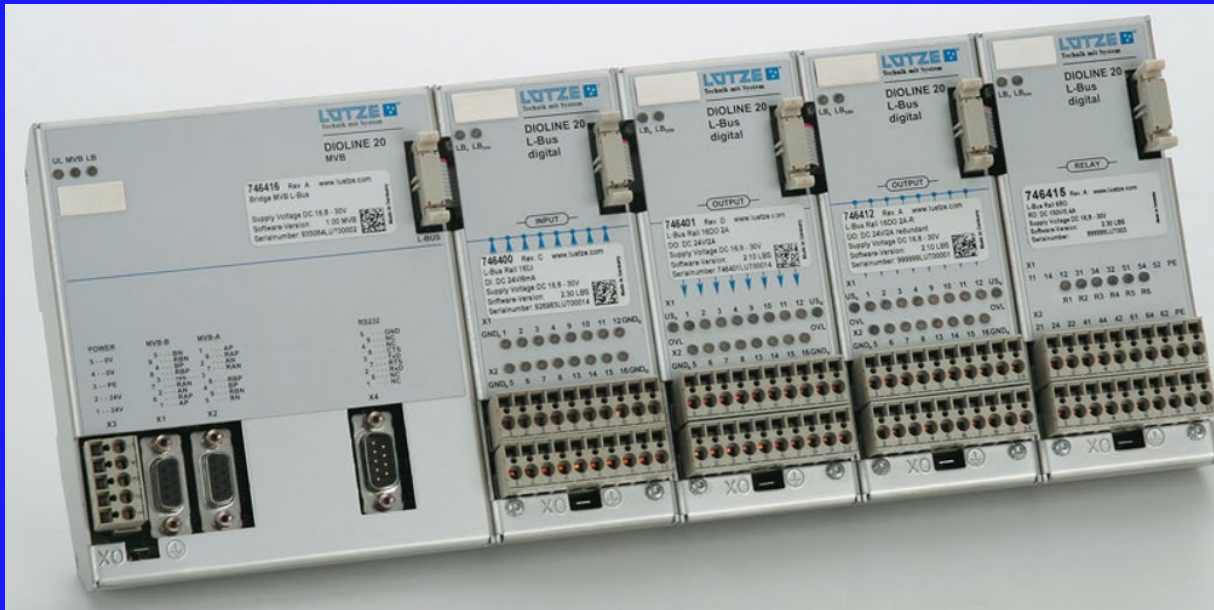
Interfaz con el Exterior



Interfaz con el Exterior – Importancia de las conexiones



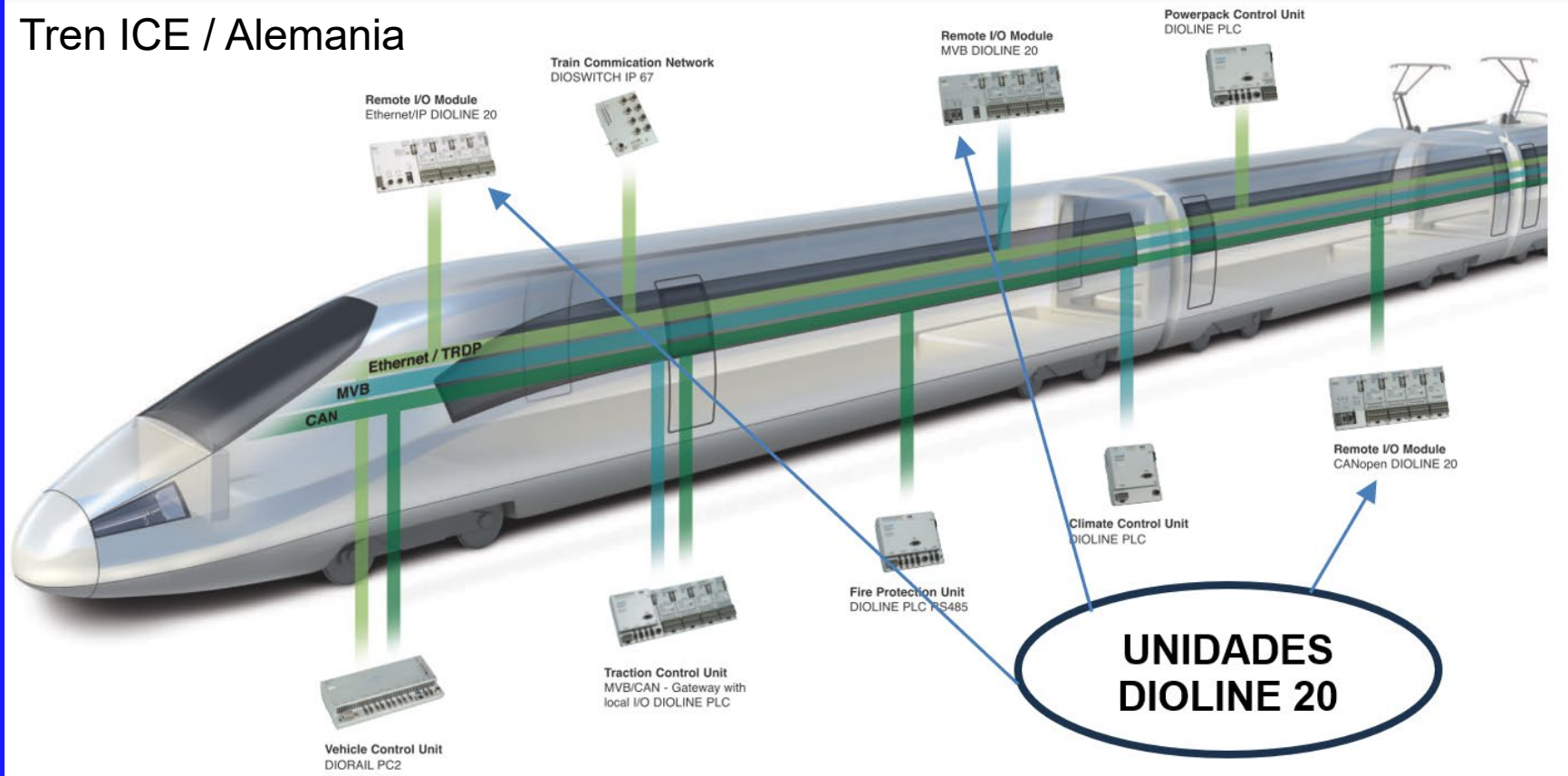
Sistemas embebidos críticos



* DIOLINE20 es un sistema de E/S para uso en vehículos ferroviarios
Controlador DIOLINE 20 de Lütze (<https://www.luetze-transportation.com/products/control-technology/dioline-20>)

Sistemas embebidos críticos

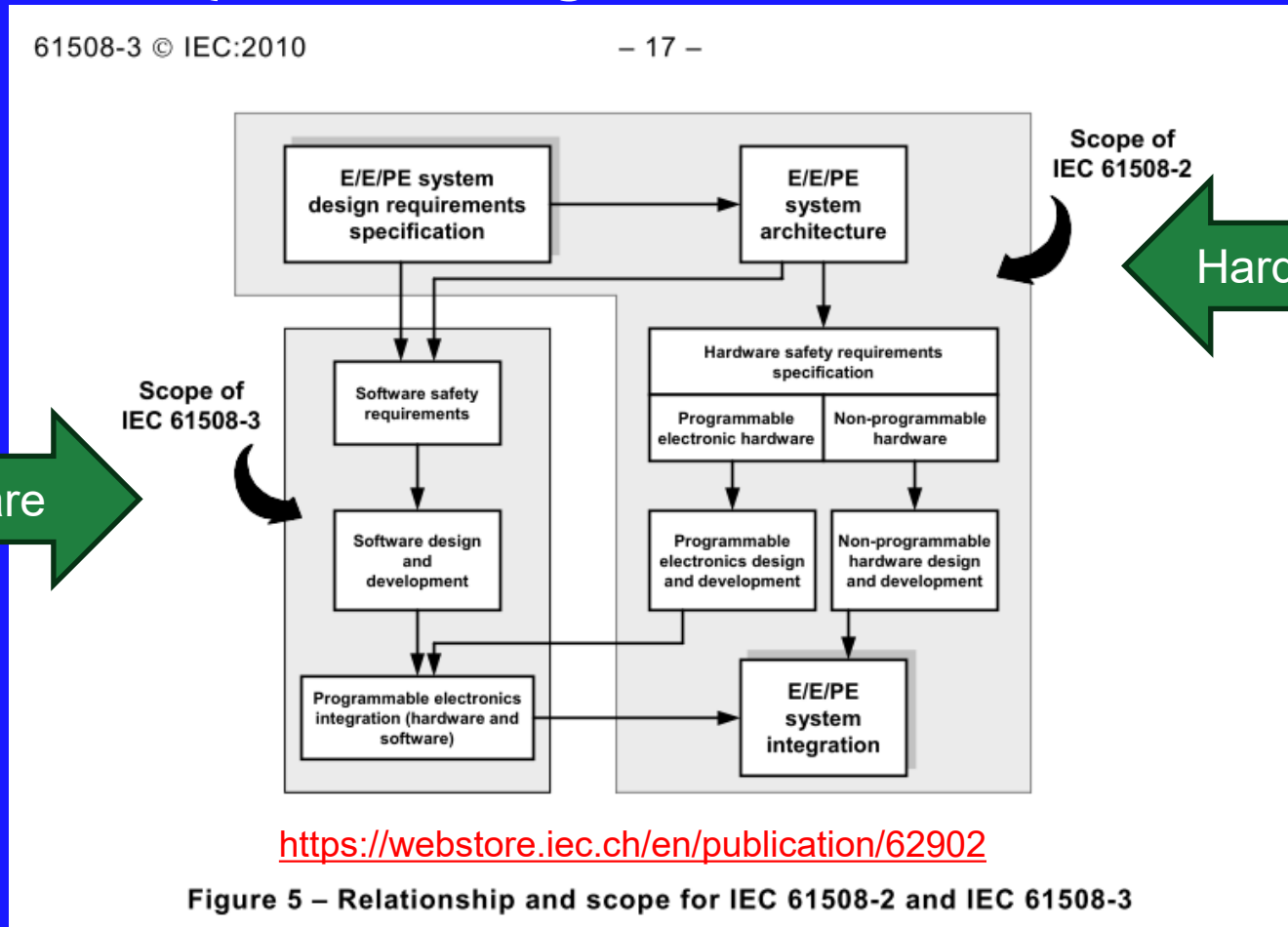
Tren ICE / Alemania



* Aquí 3 DIOLINE20 conectados sobre CANOpen, MVB (*Multifunction Vehicle Bus*), y Ethernet/TRDP (*Train Real-time Data Protocol*).

Sistemas embebidos críticos

❖ IEC 61508 (Norma de Seguridad funcional de sistemas)

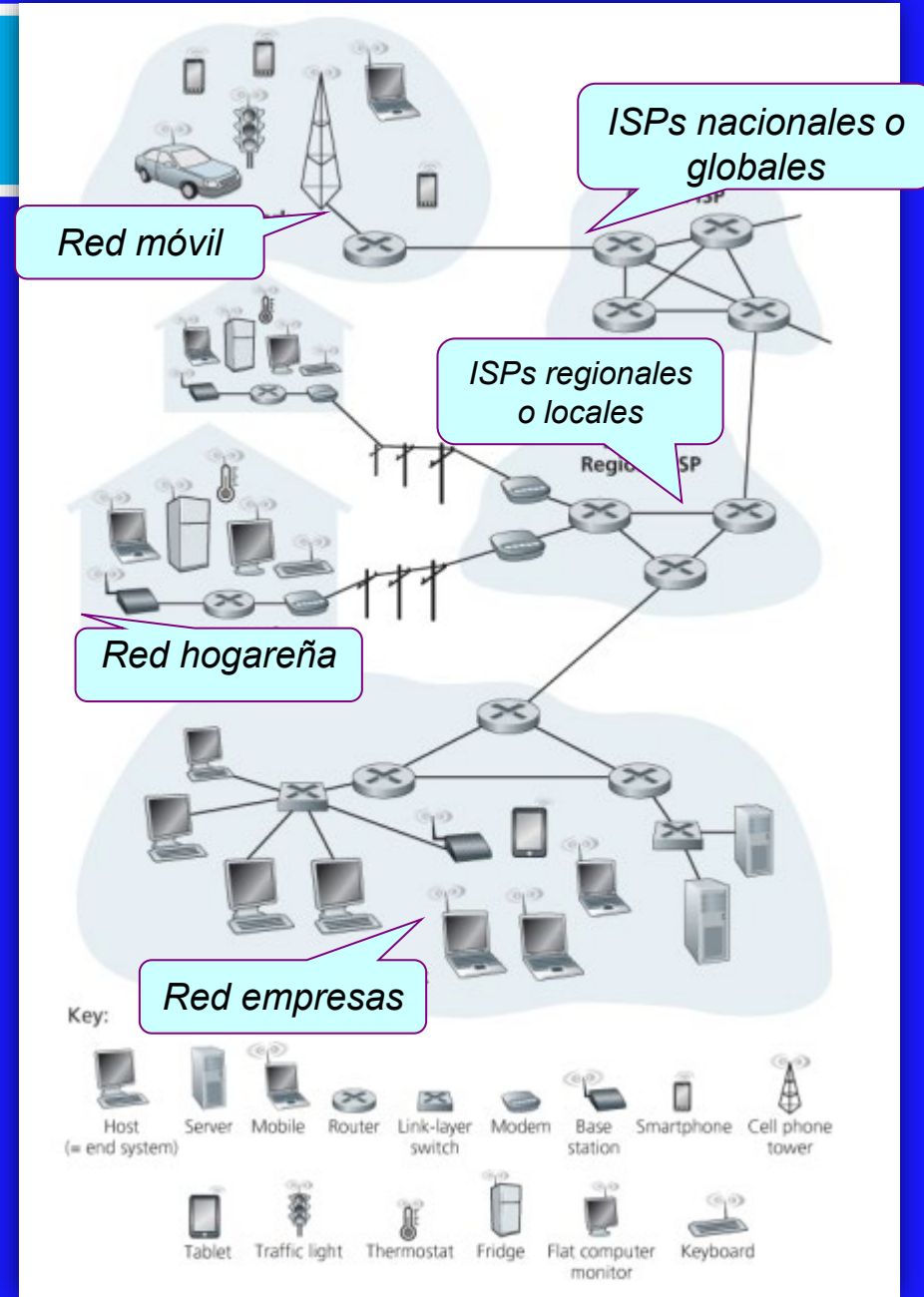


E/E/PE Systems = Sistemas Eléctricos/Electrónicos y Electrónicos Programables

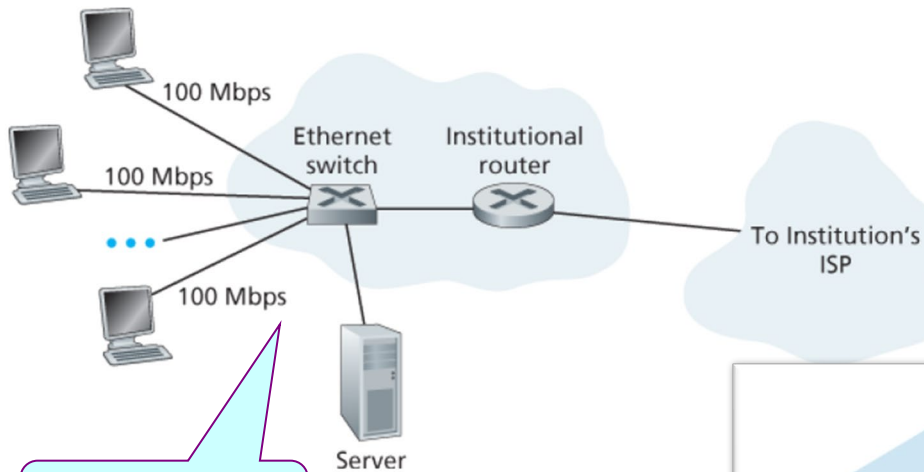
Uso de Internet para la conexión de dispositivos.. y casi todo lo demás

*ISP: Proveedor de Servicios
de Internet*

Gráficos de:
Kurose, Ross "Computer networking:
a top-down approach"
7ma Edición 2017 Ed. Pearson ISBN
0133594149

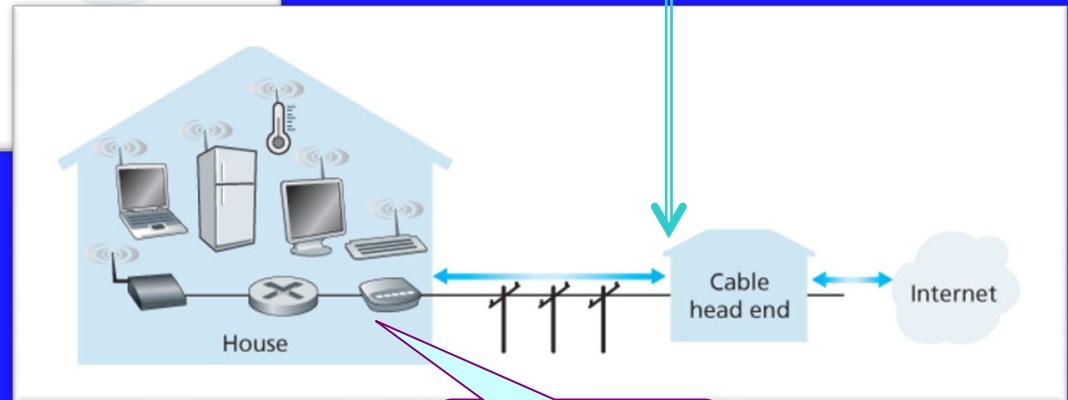


Accesos cableados (Ethernet) y WiFi



Red institucional

ISP: Proveedor de Servicios de Internet



Red hogareña típica

Gráficos de:
Kurose, Ross "Computer networking:
a top-down approach"
7ma Edición 2017 Ed. Pearson ISBN
0133594149

“Stack” o pila de protocolos de Internet

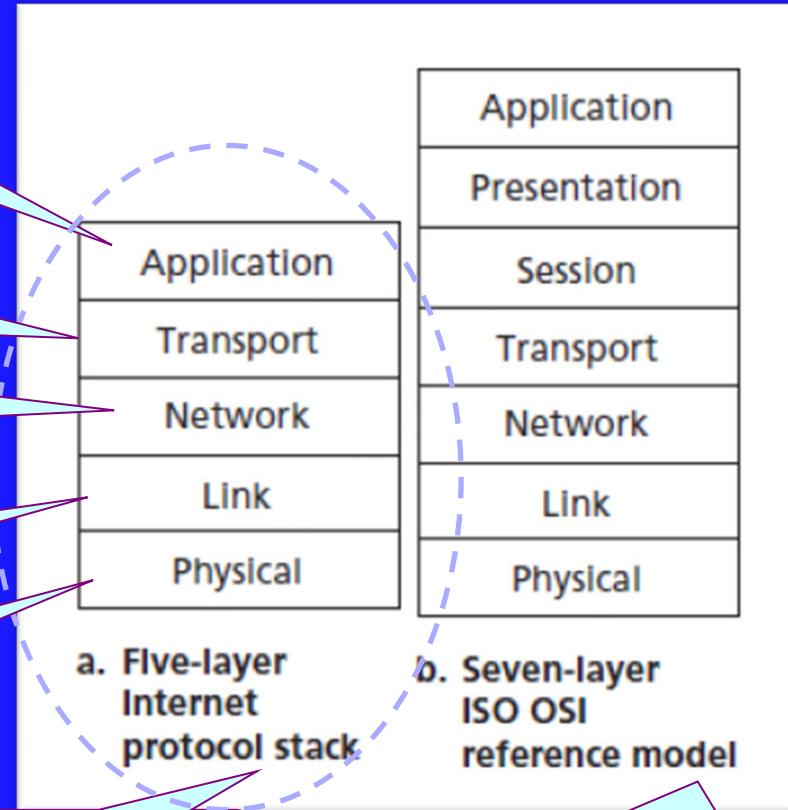
Capa de **aplicación** del usuario y los **protocolos asociados**: HTTP (Web), FTP, SMTP (mail)...

Capa de **transporte**: protocolos asociados: TCP (confiable, mas complejo), UDP (simple)

Capa de **red o network**: protocolo IP (v4 o v6) para enrutamiento de los paquetes de transporte

Capa de **enlace** para movimiento de los datagramas (Ethernet, WiFi, PPP..)

Capa **física** para movimiento de los bits individuales (cable UTP, fibra, RF)



Stack de 5 capas usado en Internet

OSI 7 capas completo, raramente usado

Gráficos de:

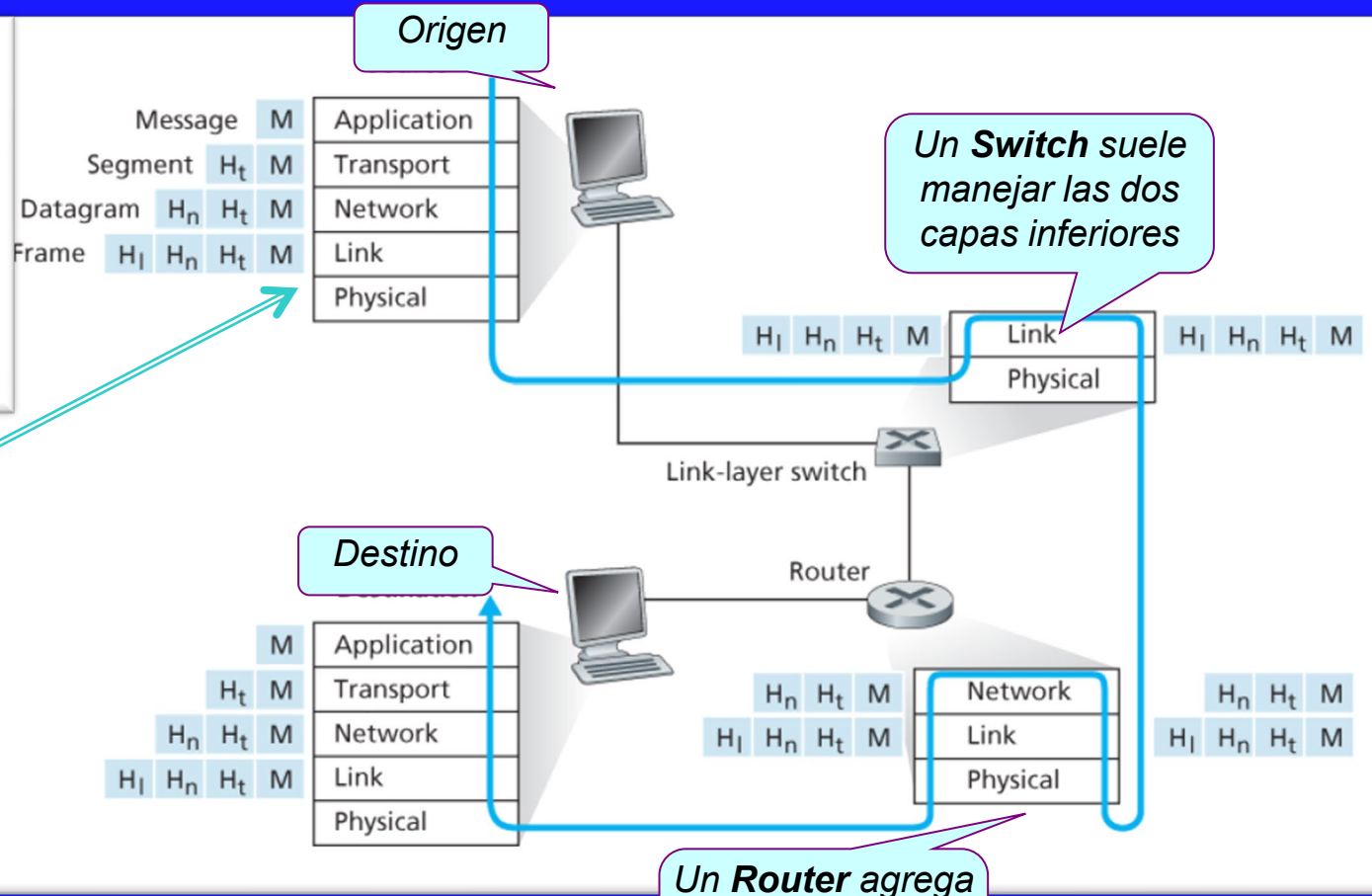
Kurose, Ross “Computer networking: a top-down approach”
7ma Edición 2017 Ed. Pearson ISBN
0133594149

“Stack” o pila de protocolos

Al mensaje **M** a enviar, se le agregan “Headers” o encabezados de cada capa:

H_t = header transporte
 H_n = header de red
 H_l = header de enlace

- Aplicación
- Transporte
- Red
- Enlace
- Físico

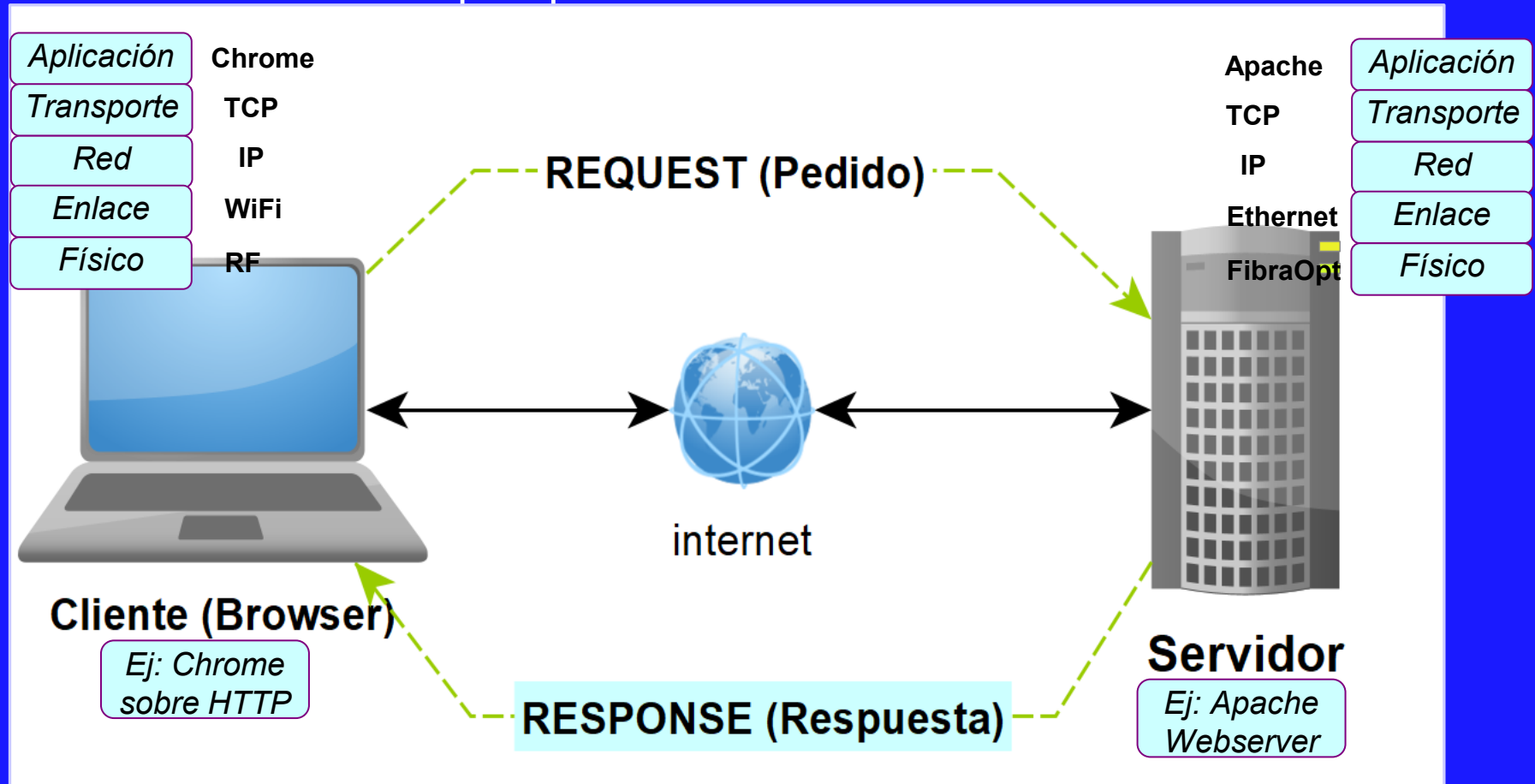


Gráficos de:

Kurose, Ross “Computer networking: a top-down approach”
 7ma Edición 2017 Ed. Pearson ISBN 0133594149

Interfaz Web elemental – a Servidor convencional

Protocolo HTTP en capa Aplicación + TCP/IP

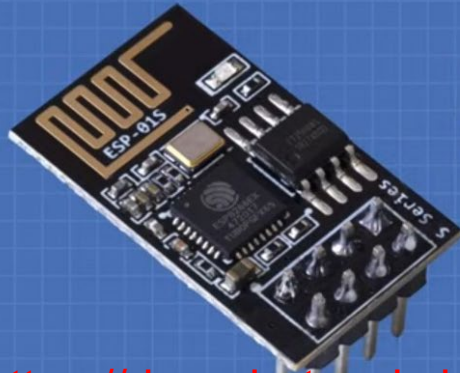


Espressif – ESP / Familia de módulos para Conexión a Internet



- Espressif Systems founded in 2008
- Wireless chips for Bluetooth & Wi-Fi
- Introduced ESP8089 Wi-Fi SoC in 2013
- Introduced ESP8266 Wi-Fi SoC in 2014

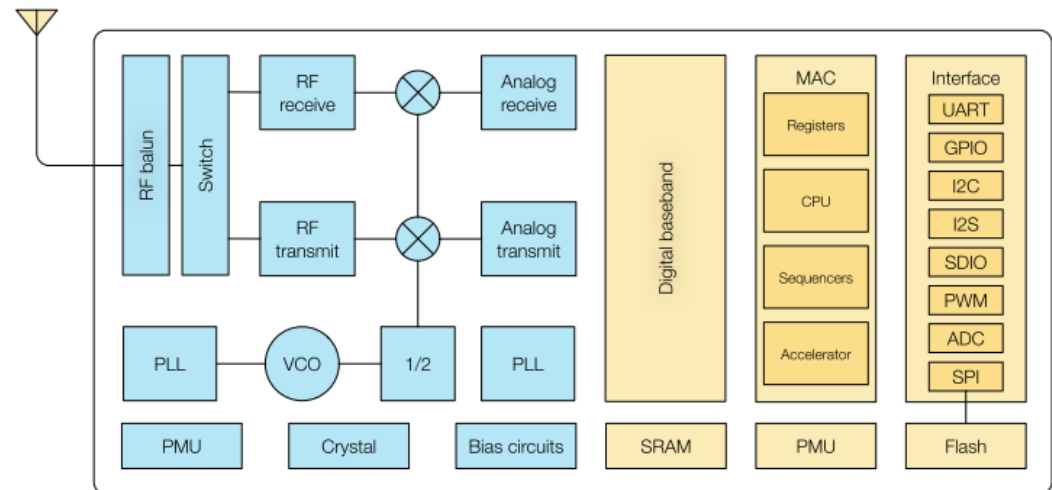
<https://www.espressif.com/>



- AI-Thinker **ESP-01S** Wi-Fi Module
- Based on the **ESP8266EX**

<https://dronebotworkshop.com/>

The functional diagram of ESP8266EX is shown as in Figure 3-1.



Espressif – ESP / Familia de módulos para Conexión a Internet



- ESP32 introduced September 2016
- Improvement over ESP8266
- Wi-Fi and Bluetooth
- Extensive documentation and DevKits



ESP-WROOM ESP DevKit V1

<https://www.espressif.com/>

<https://dronebotworkshop.com/>



Espressif – ESP / Familia de módulos para Conexión a Internet

Espressif ESP32



2016

160 - 240 MHz

Xtensa LX6

FPU

- Original ESP32
- Camera Bus
- Hall Sensor
- SD Interface

32

32-bit MCU Single & Dual Core & 2.4 GHz Wi-Fi & Bluetooth/Bluetooth LE

Espressif ESP32



2020

240 MHz

Xtensa LX7

- S2 has no Bluetooth
- S2 has 13-Bit ADC

S2

32-bit Single Core MCU & 2.4 GHz Wi-Fi

S3

32-bit Dual Core MCU & 2.4 GHz Wi-Fi & Bluetooth 5 (LE)

Espressif – ESP / Familia de módulos para Conexión a Internet

Espressif ESP32



2020

THREAD

ZIGBEE

RISC-V

120 - 160 MHz

- All Single Core
- C2 replaces ESP8266
- C3 has RTC
- C6 for Thread/Zigbee

C2

32-bit RISC-V MCU & 2.4 GHz Wi-Fi & Bluetooth 5 (LE)

C3

32-bit RISC-V MCU & 2.4 GHz Wi-Fi & Bluetooth 5 (LE)

C6

32-bit RISC-V MCU & 2.4 GHz Wi-Fi 6 & Bluetooth 5 (LE) & IEEE 802.15.4

ESP32



2023

THREAD

ZIGBEE

RISC-V

96 MHz

- IoT Applications
- External Flash
- Thread End Device

H2

32-bit RISC-V MCU & Bluetooth 5 (LE) & IEEE 802.15.4

Espressif – Módulos ESP32

<https://www.espressif.com/>

Familia ESP

Wi-Fi SoC (System on Chip)
para IoT

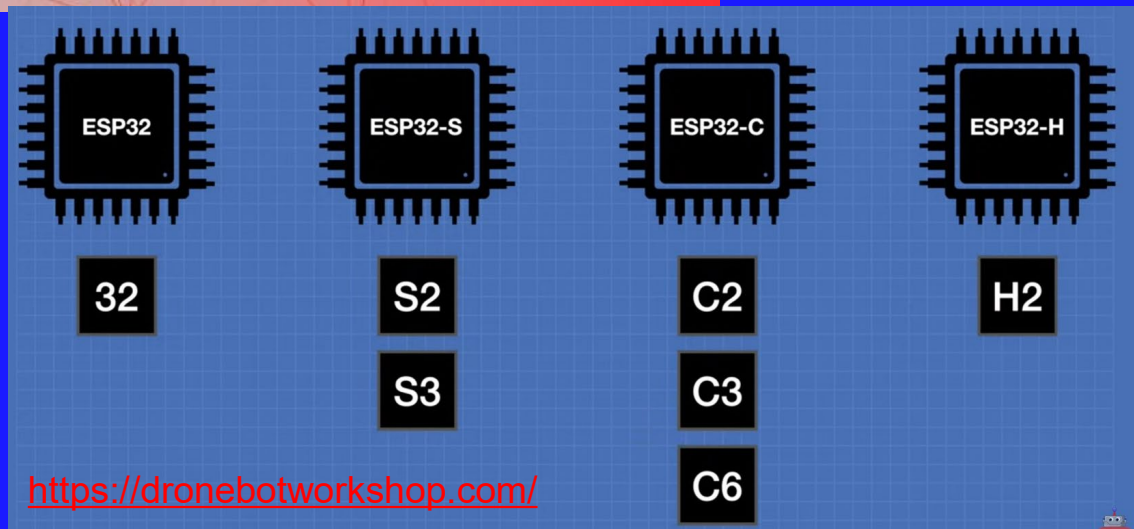


ESP32-DevKitC

ESP32-DevKitC is a low-footprint and entry-level development board that is part of the ESP32 series. This board has a rich peripheral set. The built-in ESP32 pinout is optimized for hassle-free prototyping!

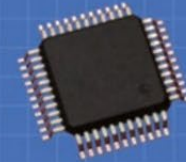
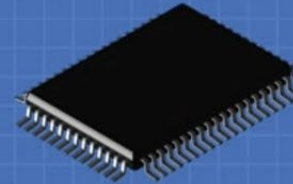
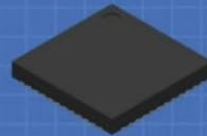


Espressif – ESP / Familia de módulos para Conexión a Internet



Espressif ESP32

SoC



Module

WROOM

WROVER

SOLO

PICO

MINI

DevKit



<https://dronebotworkshop.com/>

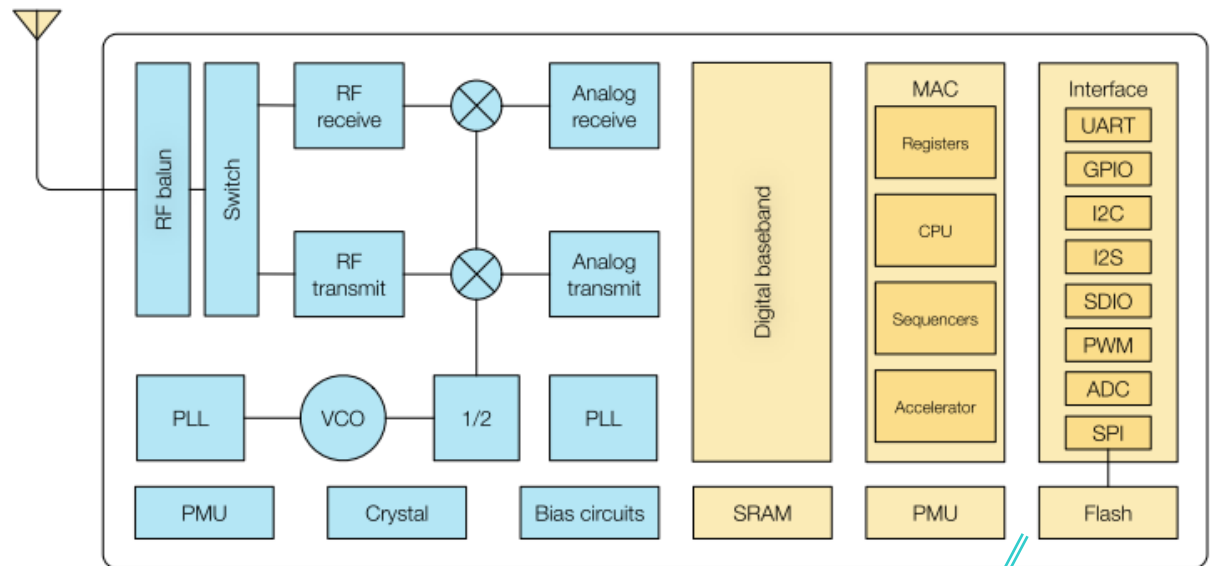
Espressif – Módulos ESP8266 y ESP32

Familia ESP

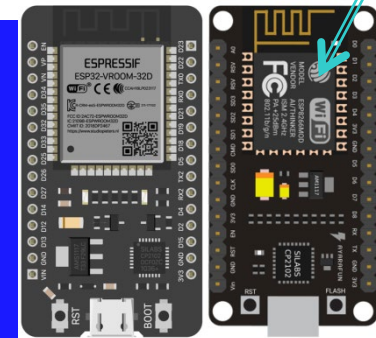
Wi-Fi SoC (System on Chip) para IoT

- Capacidades de WiFi Integradas, para funcionar sólo o como esclavo en comunicación con otros sistemas embebidos
- Contienen RAM y Memoria Flash interna, y se los puede programar con **Arduino IDE (similar a C++)** o con **Visual Studio Code / PlatformIO** y otras herramientas (C++)
- En el arranque bootea desde su flash interna. Puede funcionar como adaptador WiFi via UART o SPI
- ESPxxx integra antena, switches, RF balun, amplificador, receptor, filtros.

The functional diagram of ESP8266EX is shown as in Figure 3-1.

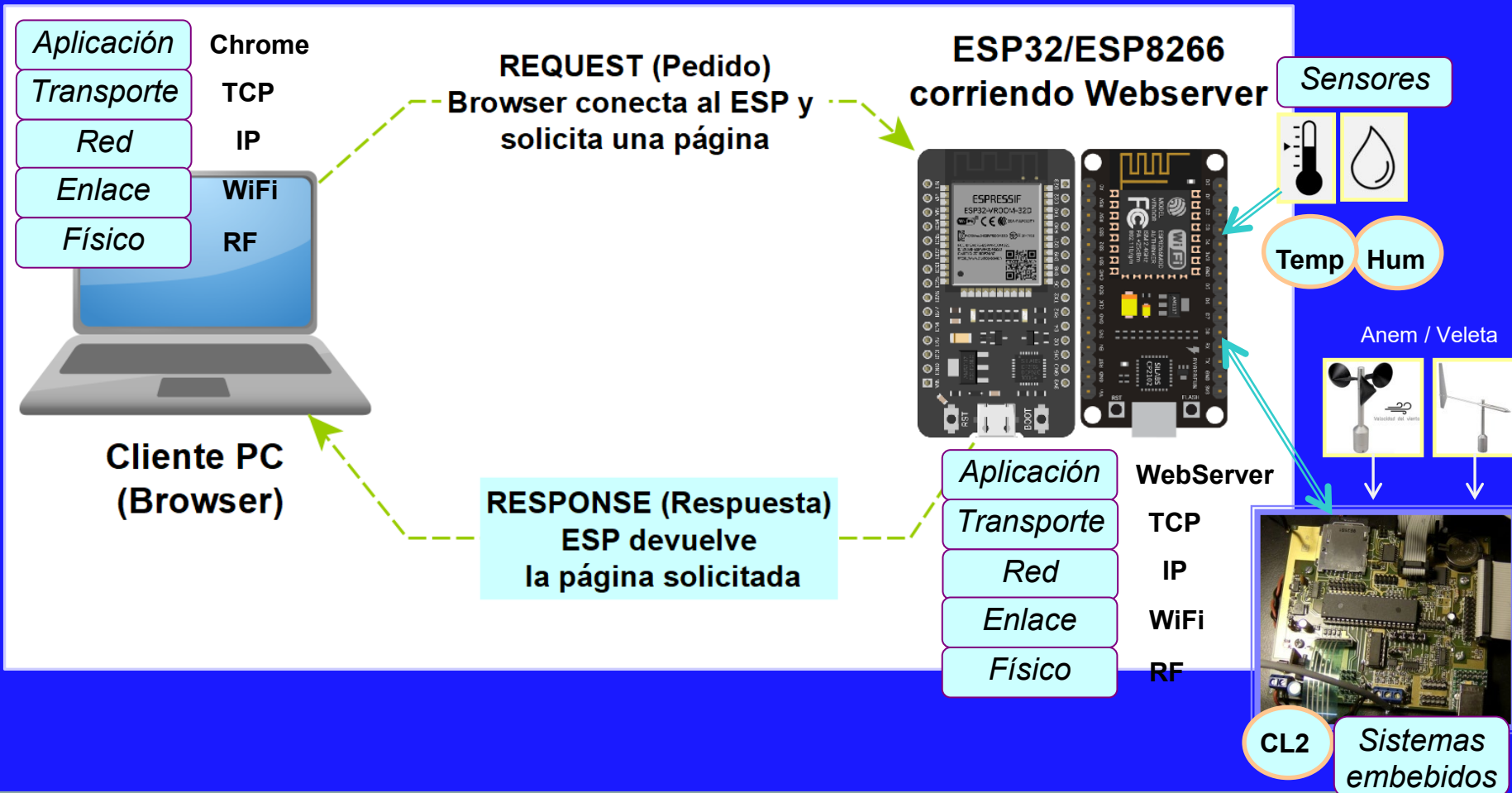


<https://www.espressif.com/>



Interfaz Web – reemplazo por Servidores ESP

Conexión directa a sensores u a otros sistemas embebidos

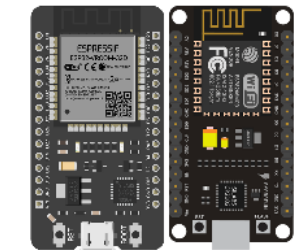


Configuración (i) – ESPs como Estación WiFi

Aplicación	WebServer
Transporte	TCP
Red	IP
Enlace	WiFi
Físico	RF

Sensores

Temp Hum



ESP32/ESP8266 corriendo Webserver (modo Station)

INTERNET



Chrome
TCP
IP
WiFi
RF



Cliente Wi-Fi (Browser)



Router (ACCESS POINT)

Red	IP
Enlace	WiFi
Físico	RF



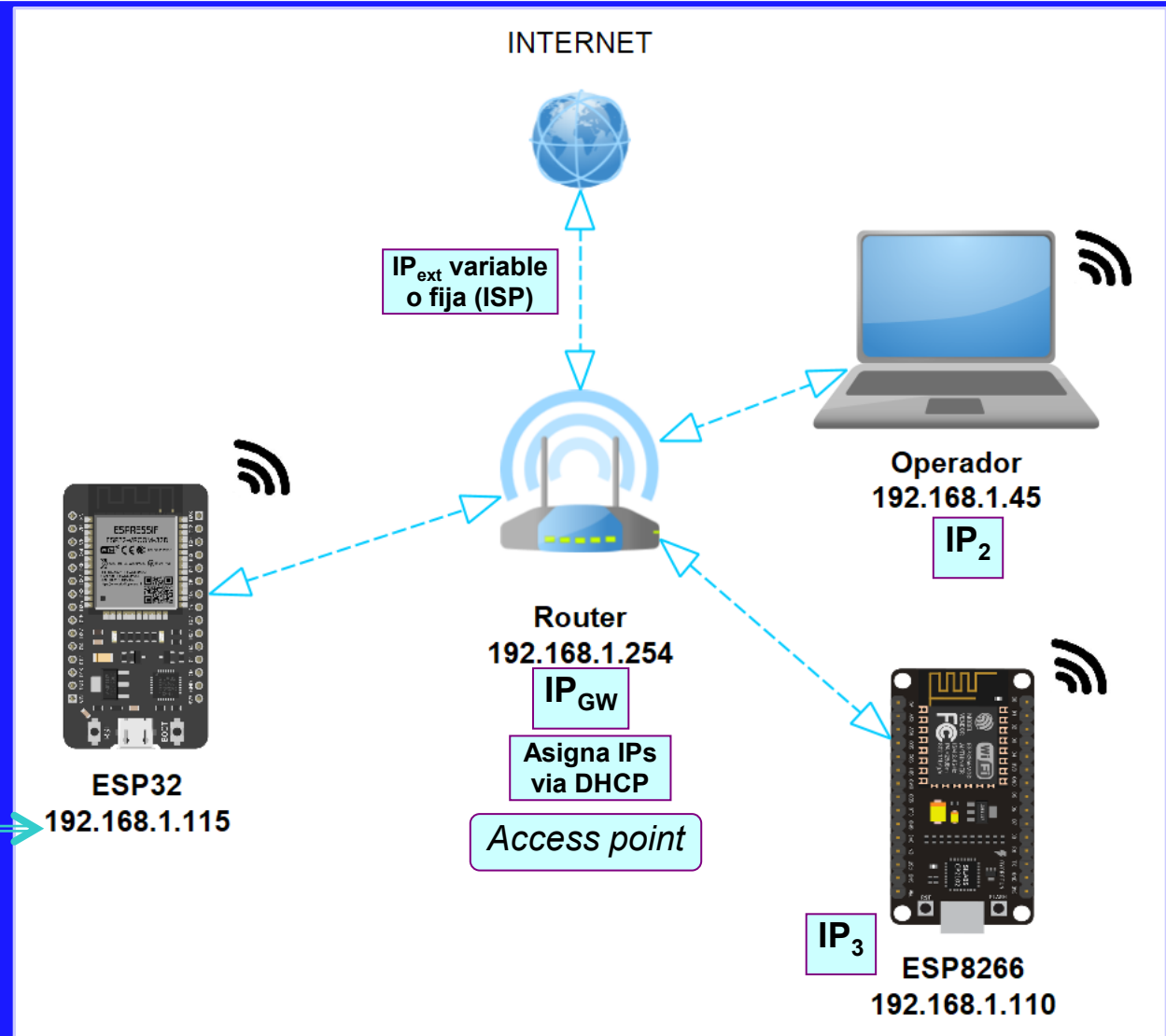
App
TCP
IP
WiFi
RF

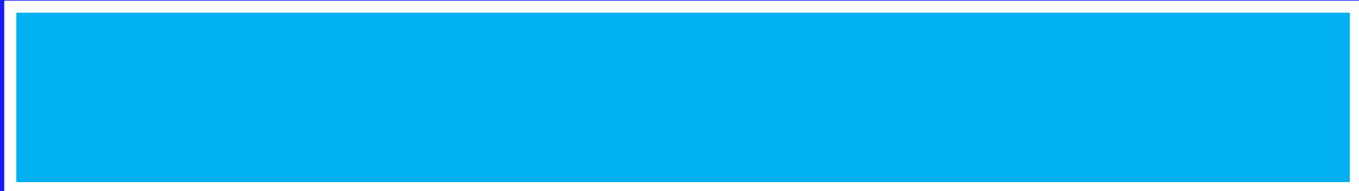
Cliente Wi-Fi

Configuración (ib) – ESPs como Estación WiFi

Direcciones típicas en una red interna

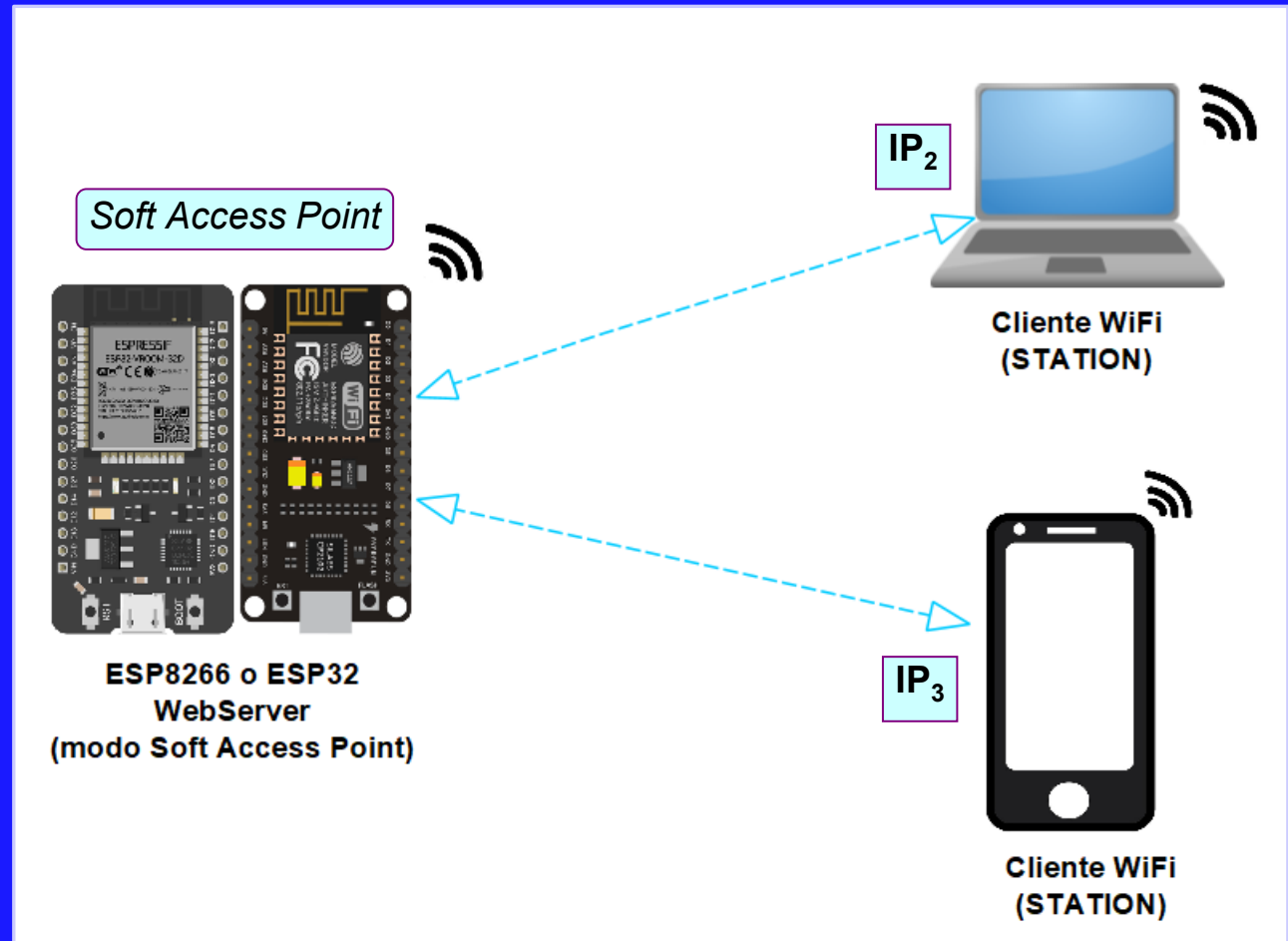
Aplicación	
Transporte	TCP
Red	IP ₁
Enlace	WiFi
Físico	RF





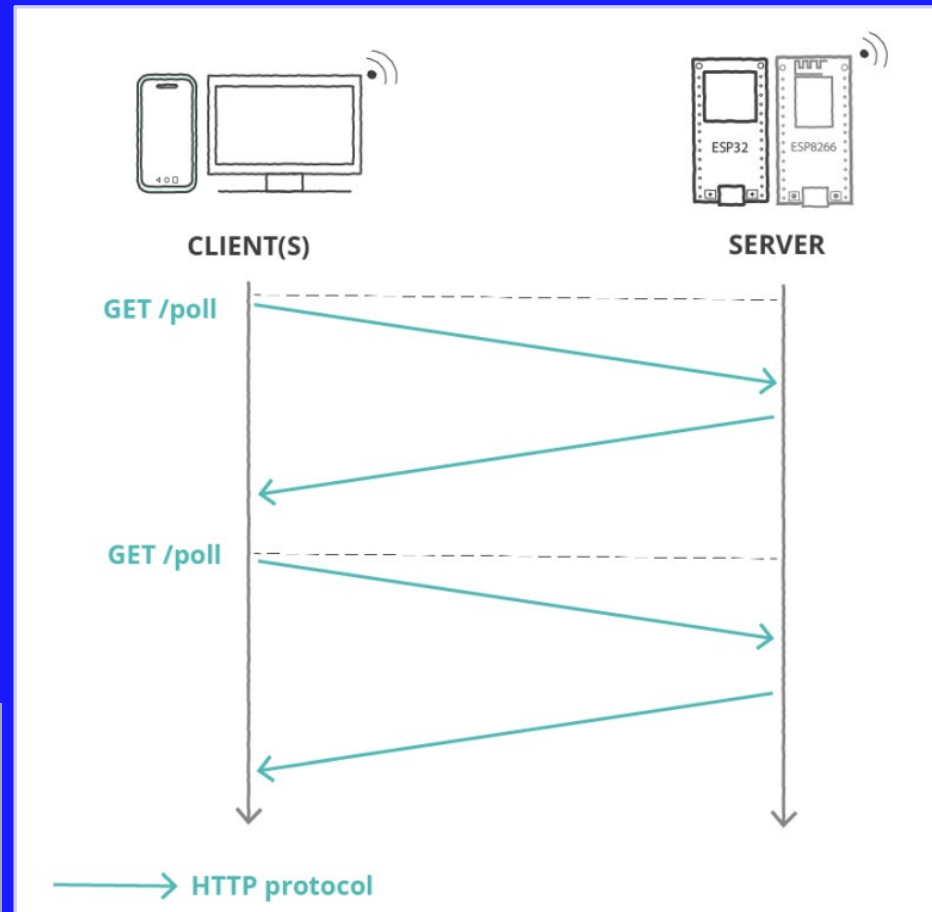
Configuración (ii) – ESPs como “Soft Access Point”

(sin vinculación
a Internet)



Comunicación Cliente-Servidor (i) – HTTP Polling

El cliente “consulta” o “encuesta” (*polls*) al servidor periódicamente utilizando comandos GET/poll. El Servidor ESP solamente responde ante consultas del cliente (usuario)



Gráficos de:

R.Santos, S.Santos “Building Web Servers with ESP8266/ESP32”
2ndEd 2021 ebook

<https://randomnerdtutorials.com/courses>

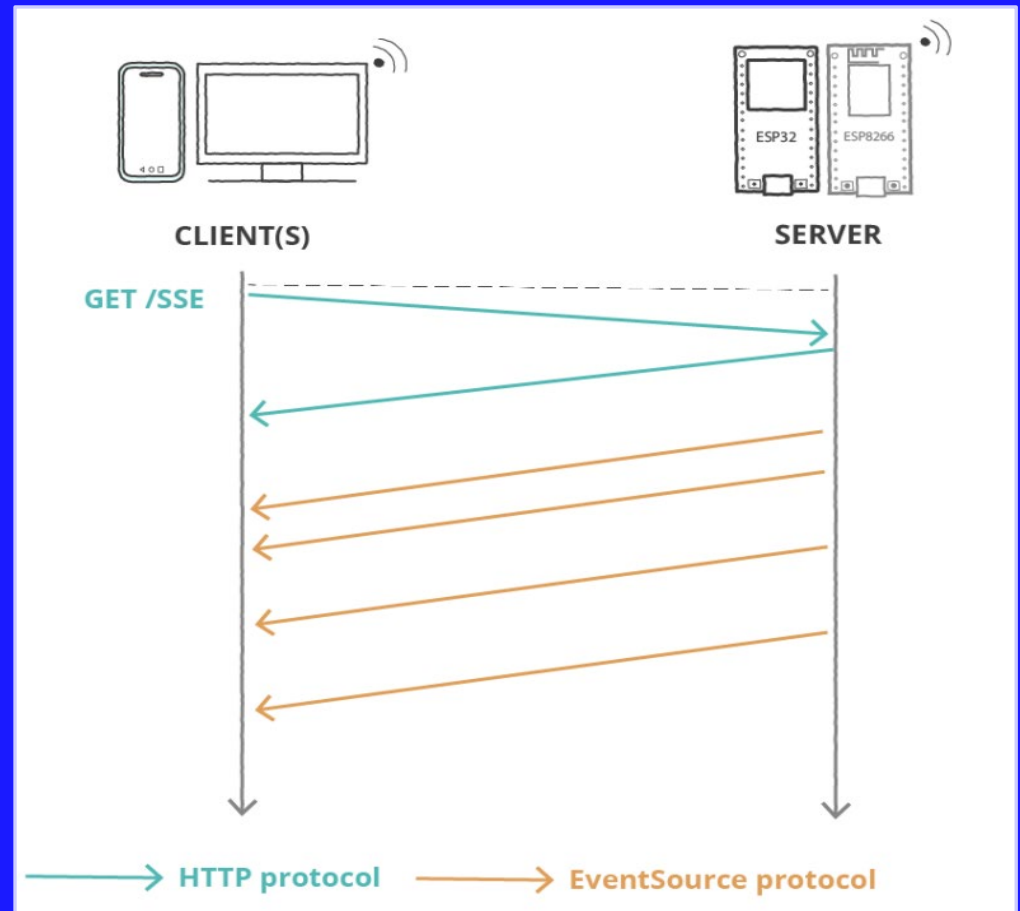
Comunicación Cliente-Servidor (ii) – Eventos del Lado Servidor (SSE)

El cliente inicia el modo SSE, después de lo cual el Servidor ESP envía automáticamente respuestas a eventos (por ejemplo lecturas de sensor, pulsaciones de botones, etc.) El cliente no puede enviar datos hacia el servidor después del protocolo inicial GET/SSE, salvo que se lo termine.

Gráficos de:

R.Santos, S.Santos "Building Web Servers with ESP8266/ESP32"
2ndEd 2021 ebook

<https://randomnerdtutorials.com/courses>



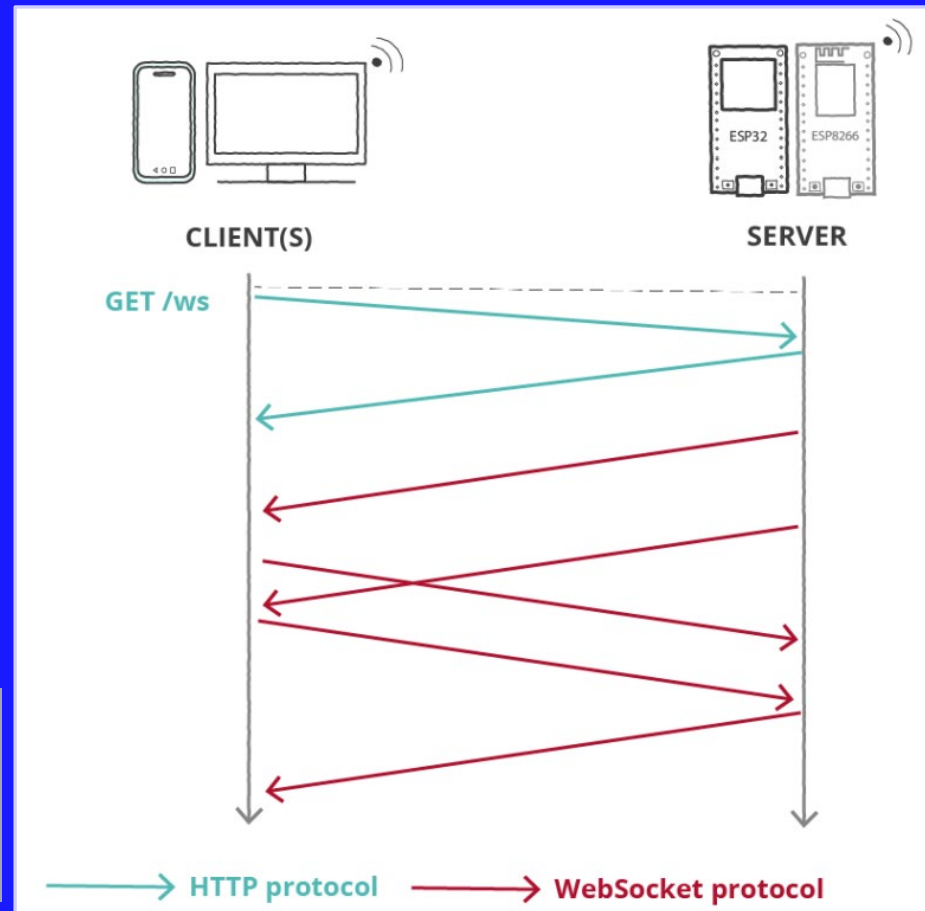
Comunicación Cliente-Servidor (iii) – WebSocket

Un WebSocket es una conexión persistente bidireccional, que inicia el cliente con el Servidor ESP, después de lo cual ambas partes pueden enviar y recibir datos usando una conexión TCP.

Gráficos de:

R.Santos, S.Santos "Building Web Servers with ESP8266/ESP32"
2ndEd 2021 ebook

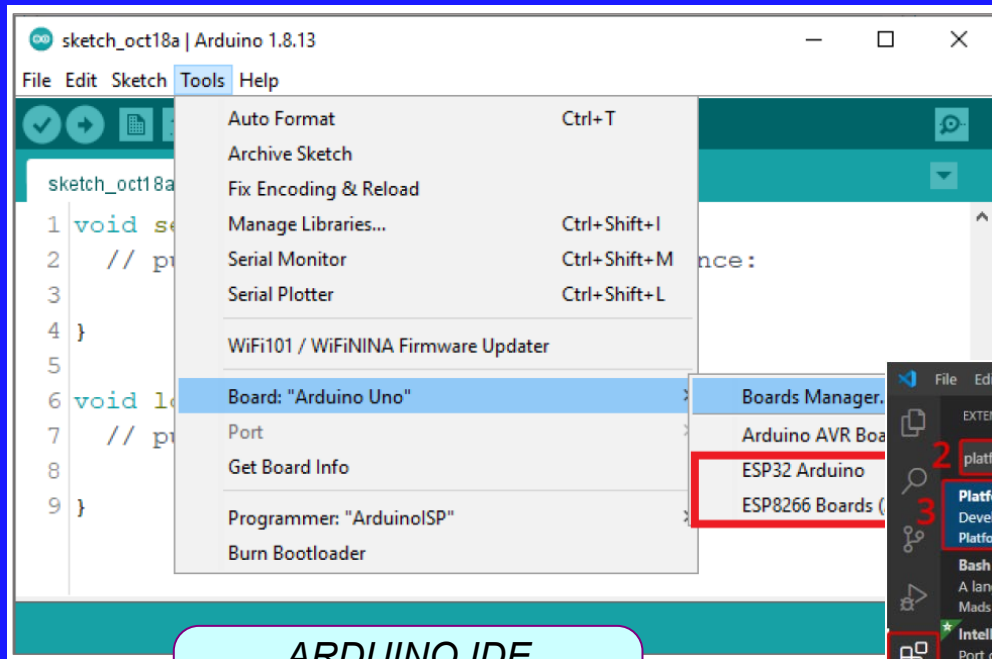
<https://randomnerdtutorials.com/courses>



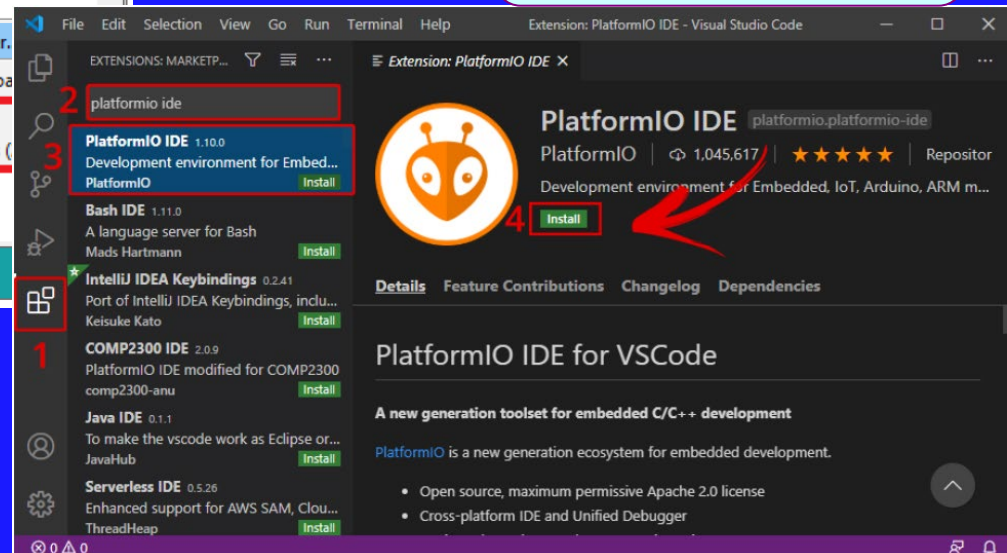
Primer Proyecto con ESP8266 / ESP32 “Hola Mundo” programable desde Arduino IDE o VSC / PIO

*Ambos GRATUITOS y
disponibles para Windows,
Ubuntu o MacOS*

VISUAL STUDIO CODE
<https://code.visualstudio.com/download>



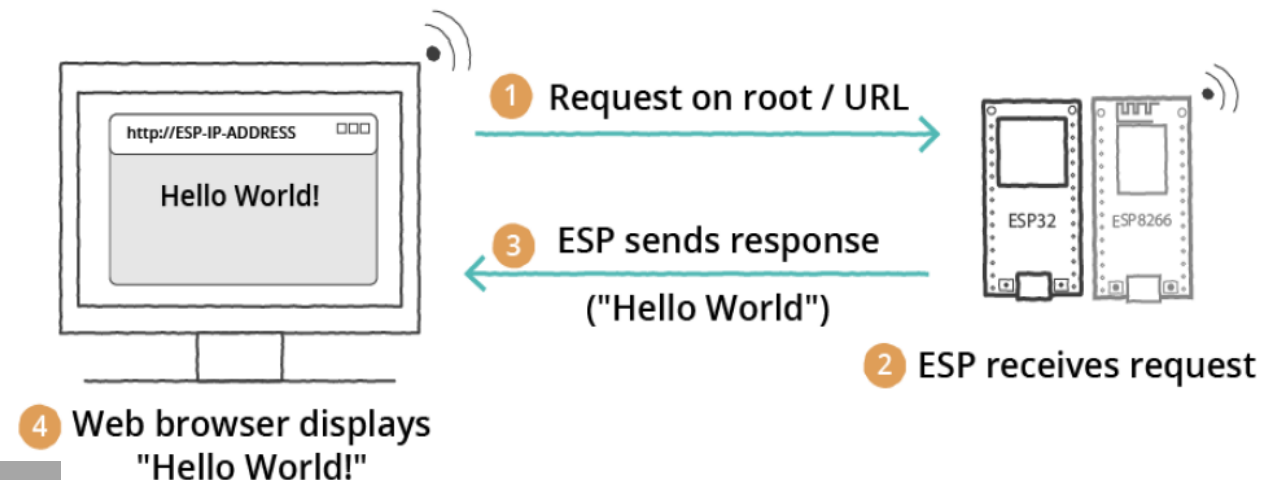
ARDUINO IDE
<https://www.arduino.cc/en/software>



Primer Proyecto con ESP8266 / ESP32 “Hola Mundo”

(i) HTML en código .cpp

Project Overview

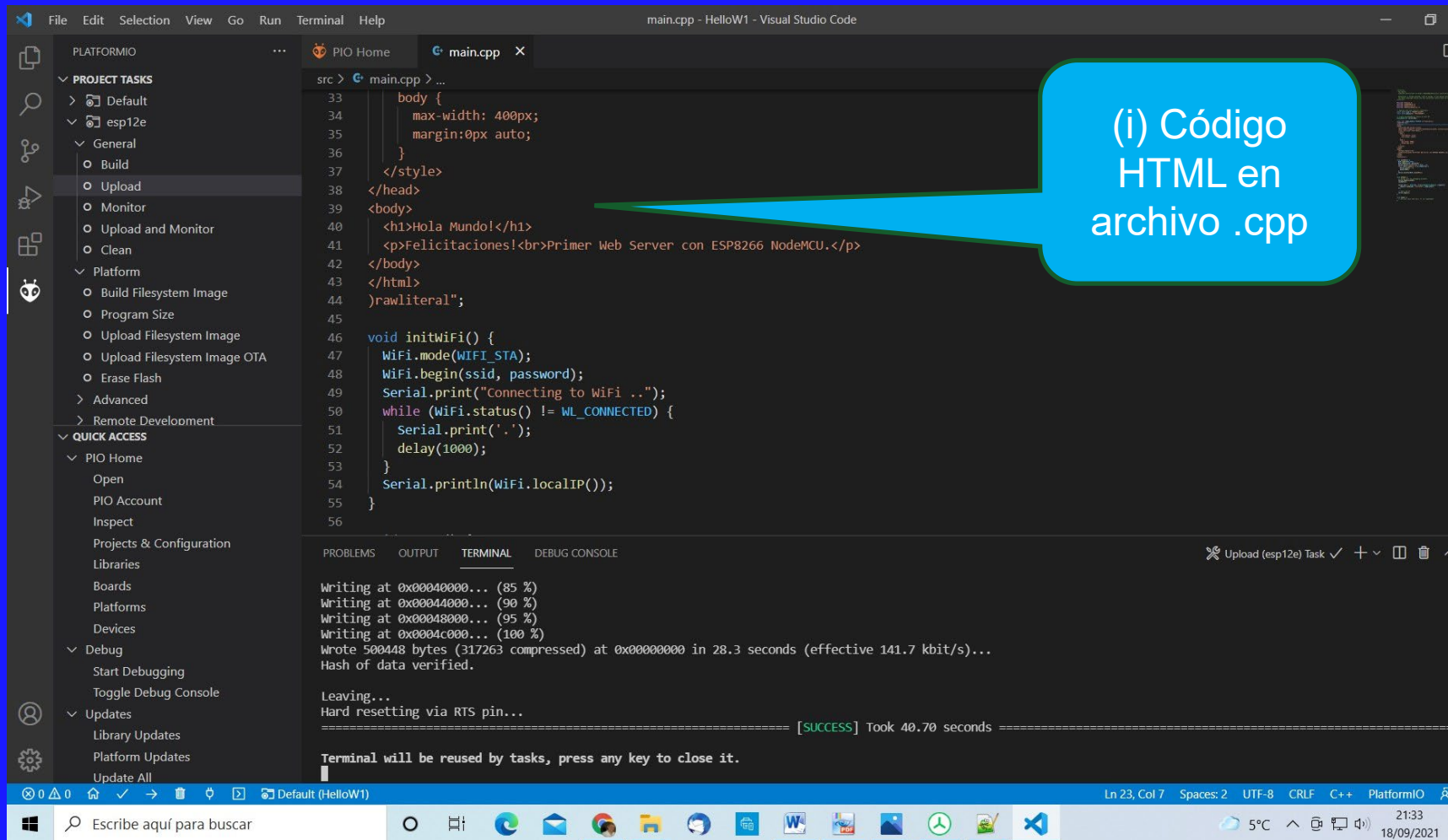


Gráficos de:

R.Santos, S.Santos “Building Web Servers with ESP8266/ESP32”
2ndEd 2021 ebook

<https://randomnerdtutorials.com/courses>

Primer Proyecto con ESP8266 / ESP32 “Hola Mundo” (alt-i)



The screenshot shows the Visual Studio Code interface with the following elements:

- Editor:** Displays the `main.cpp` file with the following code:


```

src > main.cpp > ...
33     body {
34         max-width: 400px;
35         margin:0px auto;
36     }
37     </style>
38 </head>
39 <body>
40     <h1>Hola Mundo!</h1>
41     <p>Felicitaciones!<br>Primer Web Server con ESP8266 NodeMCU.</p>
42 </body>
43 </html>
44 )rawliteral";
45
46 void initWiFi() {
47     WiFi.mode(WIFI_STA);
48     WiFi.begin(ssid, password);
49     Serial.print("connecting to WiFi ..");
50     while (WiFi.status() != WL_CONNECTED) {
51         Serial.print('.');
52         delay(1000);
53     }
54     Serial.println(WiFi.localIP());
55 }
56
      
```
- Terminal:** Shows the upload progress and a successful completion message:


```

Writing at 0x00040000... (85 %)
Writing at 0x00044000... (90 %)
Writing at 0x00048000... (95 %)
Writing at 0x0004c000... (100 %)
Wrote 500448 bytes (317263 compressed) at 0x00000000 in 28.3 seconds (effective 141.7 kbit/s)...
Hash of data verified.

Leaving...
Hard resetting via RTS pin...

===== [SUCCESS] Took 40.70 seconds =====

Terminal will be reused by tasks, press any key to close it.
      
```
- Callout:** A blue speech bubble points to the HTML code in the editor, containing the text: (i) Código HTML en archivo .cpp

Primer Proyecto con ESP8266 / ESP32 “Hola Mundo” (alt-i)

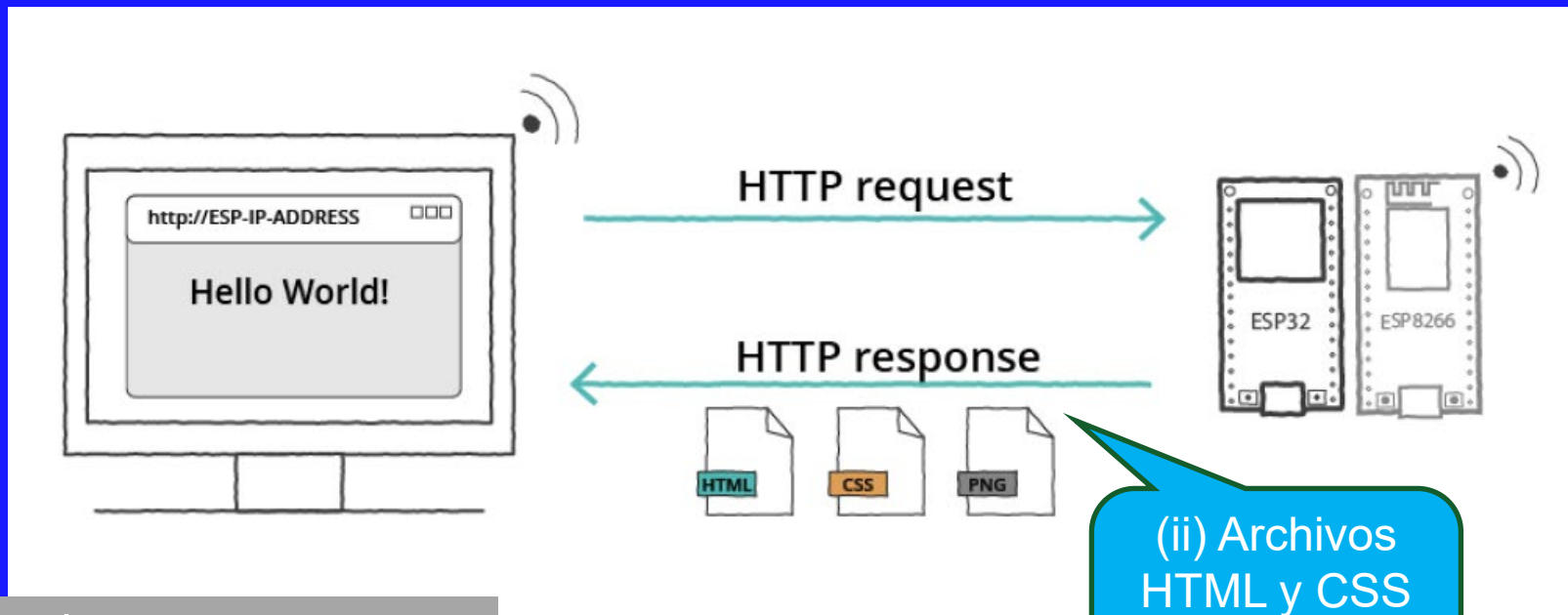
```
52   delay(1000);
53   }
54   Serial.println(WiFi.localIP());
55   }
56
57   void setup() {
58     // Serial port for debugging purposes
59     Serial.begin(115200);
60     initWiFi();
61     (const char [2])"/"
62     server.on("/", HTTP_GET, [](AsyncWebServerRequest *request){
63       request->send(200, "text/html", index_html);
64     });
65
66     // Start server
```

(i) Código
HTML en
archivo .cpp



Primer Proyecto con ESP8266 / ESP32 “Hola Mundo”

(ii) HTML almacenado en Flash File System (SPIFFS)



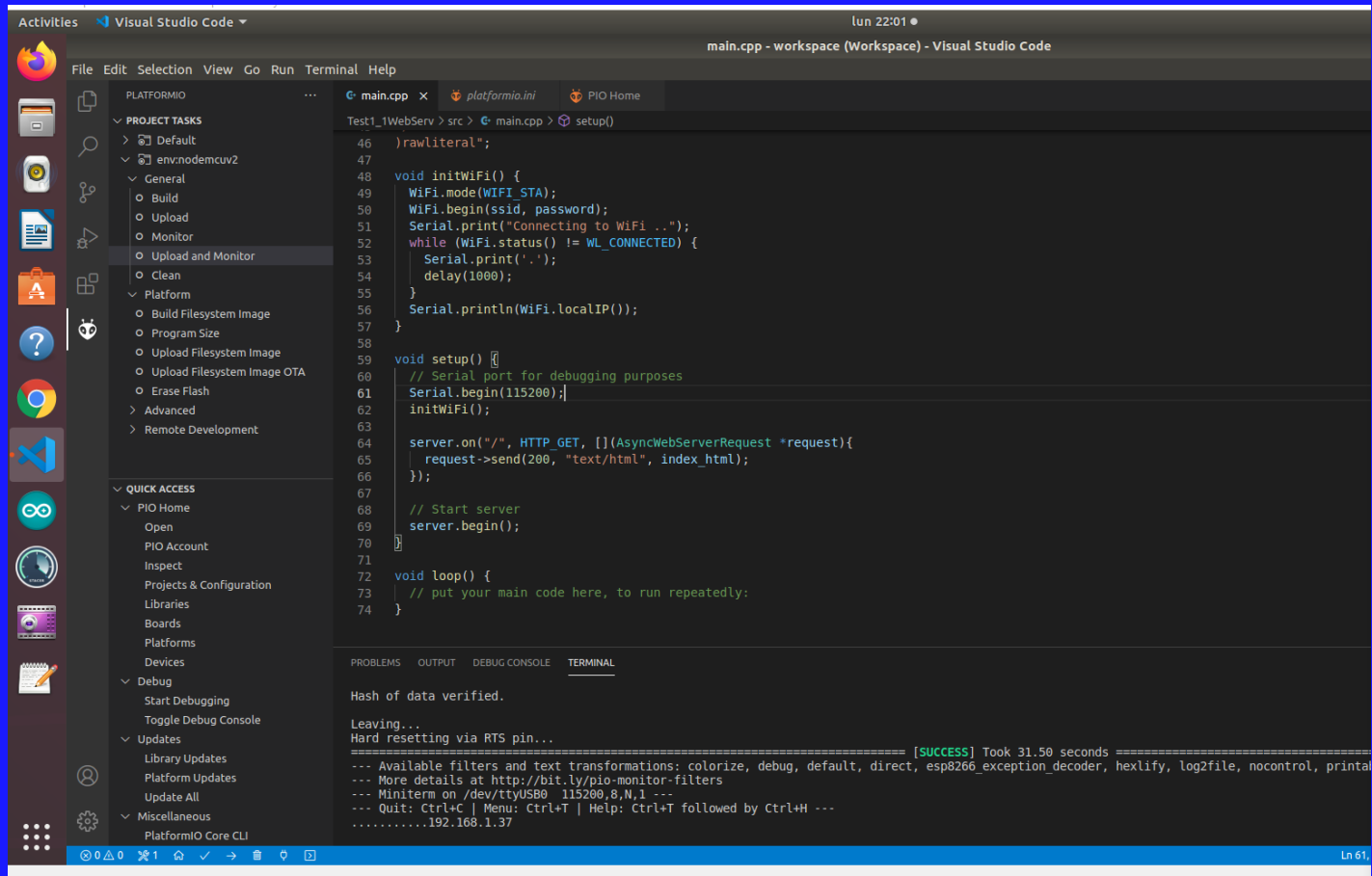
Gráficos de:

R.Santos, S.Santos “Building Web Servers with ESP8266/ESP32”
2ndEd 2021 ebook

<https://randomnerdtutorials.com/courses>

(ii) Archivos
HTML y CSS
residentes en
Flash del ESP

Interfaz Web a Sistemas Embebidos Existentes (VSC / PIO bajo Ubuntu 18.04 LTS)



Visual Studio Code interface showing a workspace for a NodeMCU project. The code editor displays the following C++ code:

```

46 )rawliteral";
47
48 void initWiFi() {
49     WiFi.mode(WIFI_STA);
50     WiFi.begin(ssid, password);
51     Serial.print("Connecting to WIFI ..");
52     while (WiFi.status() != WL_CONNECTED) {
53         Serial.print('.');
54         delay(1000);
55     }
56     Serial.println(WiFi.localIP());
57 }
58
59 void setup() {
60     // Serial port for debugging purposes
61     Serial.begin(115200);
62     initWiFi();
63
64     server.on("/", HTTP_GET, [](AsyncWebServerRequest *request){
65         request->send(200, "text/html", index_html);
66     });
67
68     // Start server
69     server.begin();
70
71 }
72
73 void loop() {
74     // put your main code here, to run repeatedly:
75 }

```

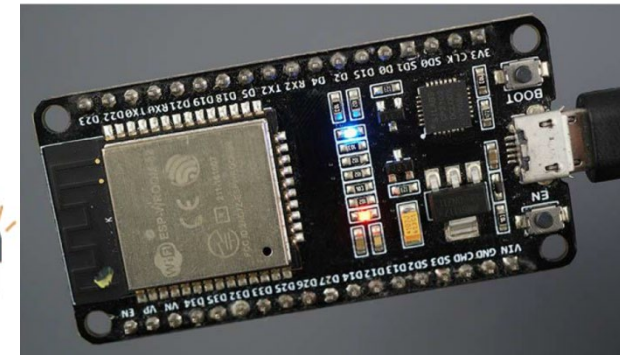
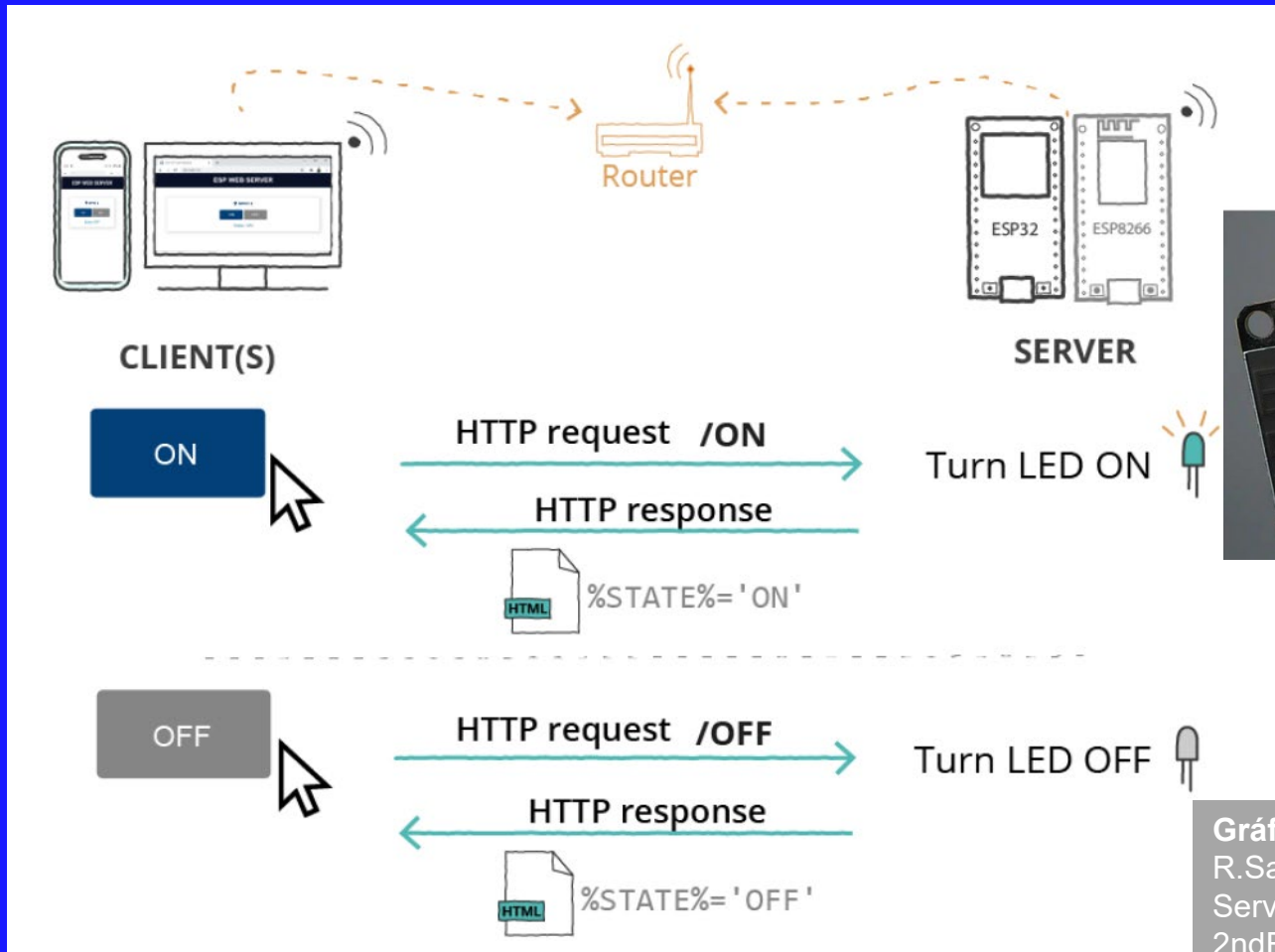
The terminal output shows the upload process:

```

Hash of data verified.
Leaving...
Hard resetting via RTS pin...
===== [SUCCESS] Took 31.50 seconds =====
--- Available filters and text transformations: colorize, debug, default, direct, esp8266_exception_decoder, hexlify, log2file, nocontrol, print
--- More details at http://bit.ly/pio-monitor-filters
--- Miniterm on /dev/ttyUSB0 115200,8,N,1 ---
--- Quit: Ctrl+C | Menu: Ctrl+T | Help: Ctrl+T followed by Ctrl+H ---
.....192.168.1.37

```

Control de Entrada / Salida vía Webserver ESP

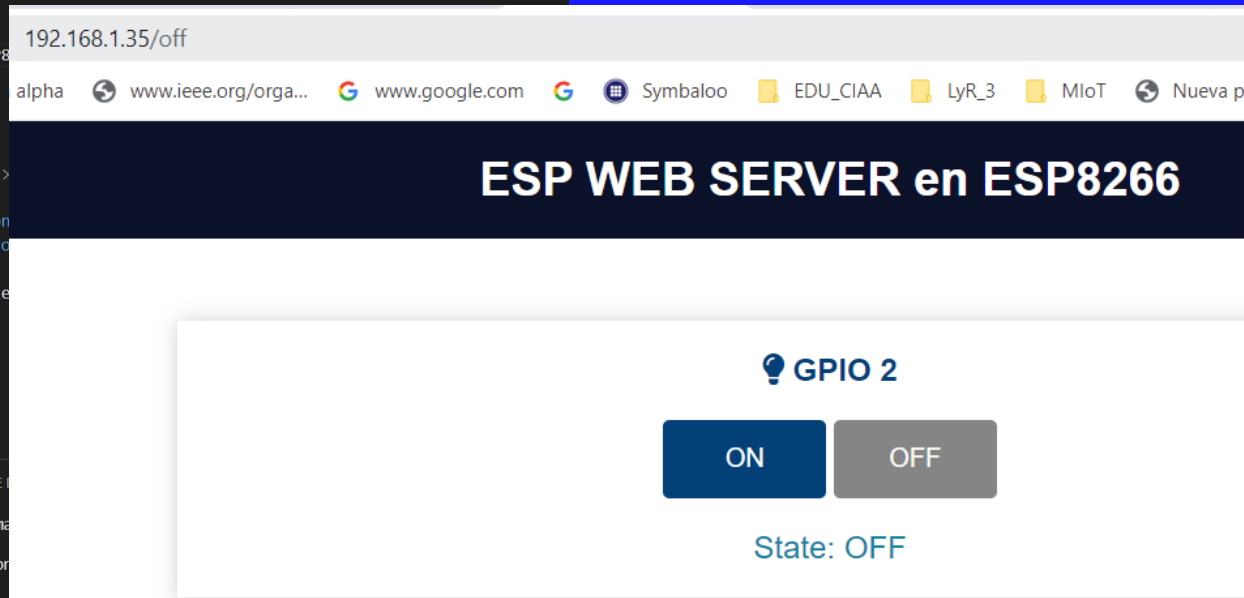


Gráficos de:
R.Santos, S.Santos "Building Web Servers with ESP8266/ESP32"
2ndEd 2021 ebook
<https://randomnerdtutorials.com/courses>

Control de Entrada / Salida vía Webserver ESP

(ii) Archivos HTML y CSS en /data

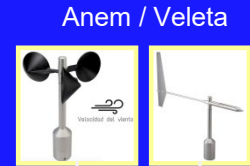
```
index.html - LEDonoff1 - Visual Studio Code
EXPLORADOR
main.cpp PIO Home index.html X
LEDON...
  .pio
  .vscode
  data
    favicon.png
    index.html
    style.css
  include
  lib
  src
    main.cpp
  test
  .gitignore
  .travis.yml
  platformio.ini
data > index.html > html > body > div.topnav > h1
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <title>ESP IOT DASHBOARD / +3+3
5 <meta charset="utf-8" />
6 <link rel="stylesheet" type="text/css" href="style.css">
7 <link rel="icon" type="image/png" href="favicon.png">
8 <link rel="stylesheet" href="https://use.fontawesome.com/releases/v5.7.2/css/all.css">
9 </head>
10 <body>
11 <div class="topnav">
12 <h1>ESP WEB SERVER en ESP8266</h1>
13 </div>
14 <div class="content">
15 <div class="card-grid">
16 <div class="card">
17 <p class="card-title">GPIO 2</p>
18 <p>
19 <a href="on"><button class="btn btn-primary">ON</button>
20 <a href="off"><button class="btn btn-secondary">OFF</button>
21 </p>
22 <p class="state">State: OFF</p>
23 </div>
24 </div>
25 </div>
26 </body>
27 </html>
```



Interfaz Web a sistema embebido tradicional



Aplicación	WebServer
Transporte	TCP
Red	IP
Enlace	WiFi
Físico	RF



Serial/UART

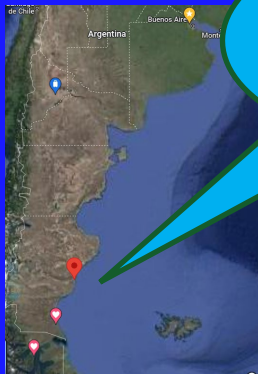


CL2 Sistemas embebidos tradicionales

Caso demostración – Conexión a SISMED/SJ24 con módulo ESP Sistema de Medición SJ24 UA Puerto San Julian (Instalado 2013)



Puerto San
Julián,
Argentina



Predio
Chacra
UASJ-UNPA

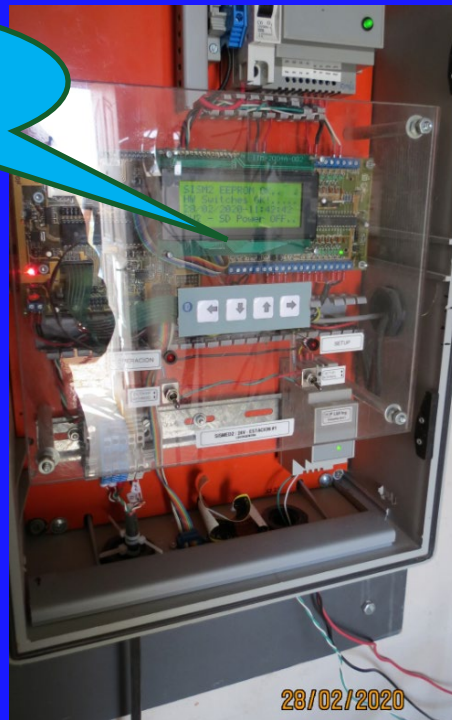


Sistema de Medición SISMED/SJ24



VISTA DEL SISTEMA

Sistema de
Registro



Rectificador,
reguladores, cargador



Control
Baterías

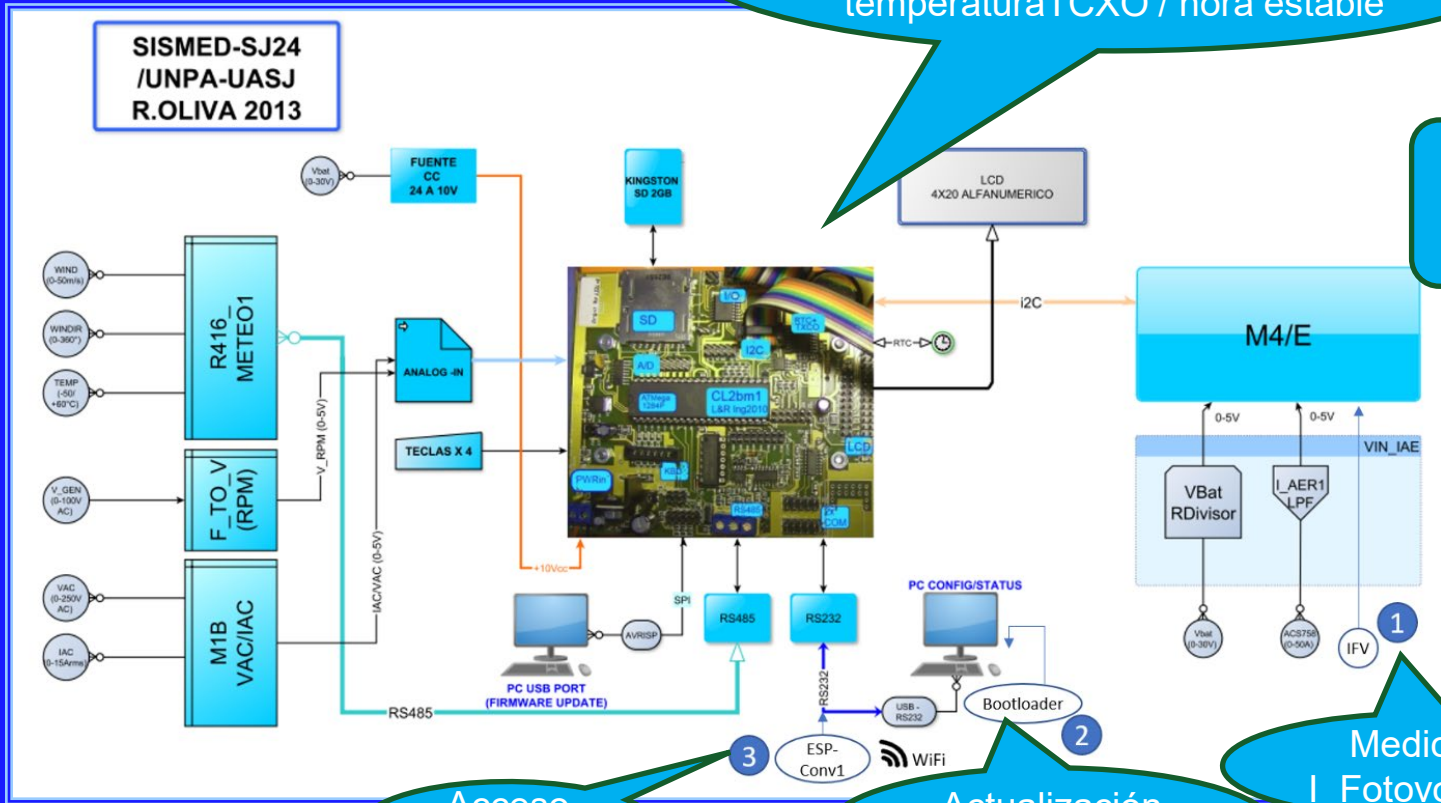


Sistema de Medición SISMED/SJ24

ESTRUCTURA DE

Reloj interno RTC con oscilador compensado por temperatura TCXO / hora estable

(1), (2), (3) agregados posteriores



Acceso WiFi

Actualización Firmware Serial

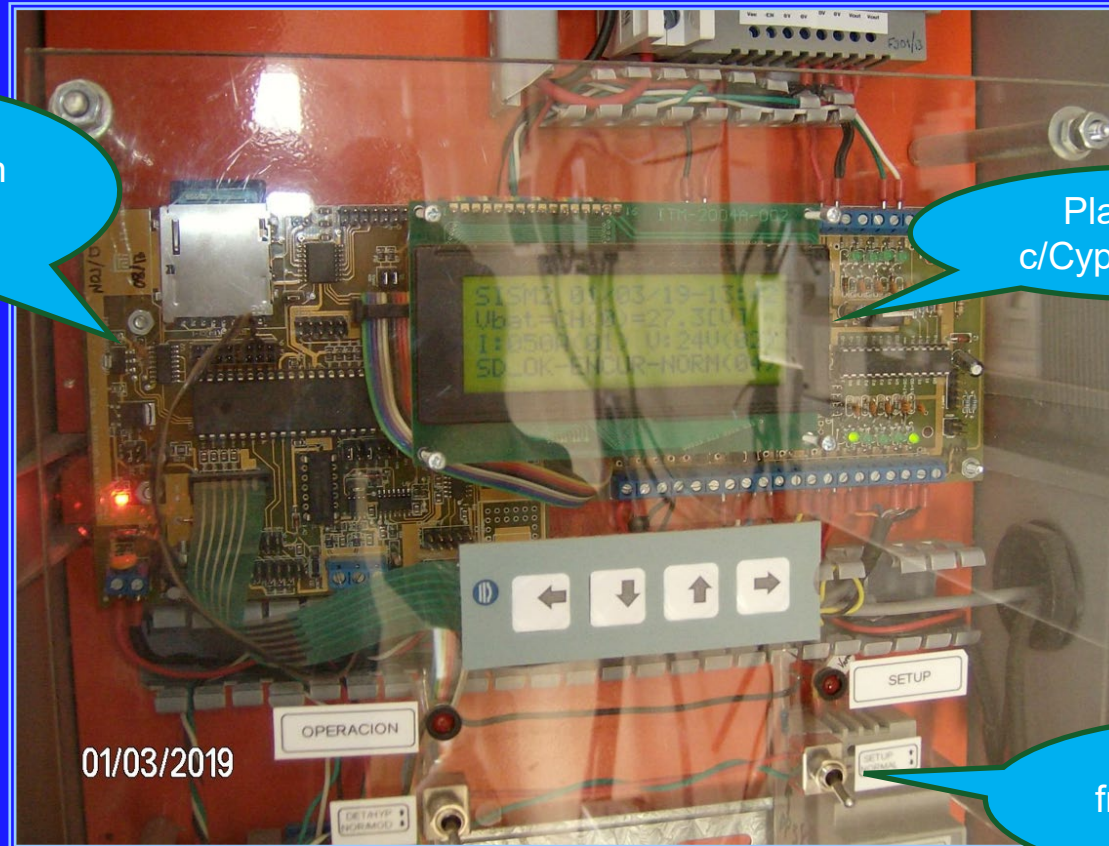
Medición I_Fotovoltaica

Sistema de Medición SISMED/SJ24

GABINETE PRINCIPAL / MÓDULOS DE HARDWARE

Placa CL2b con
ATMega1284P

Placa M4/E
c/Cypress 29466



Switches
frontales: Modo
Operación

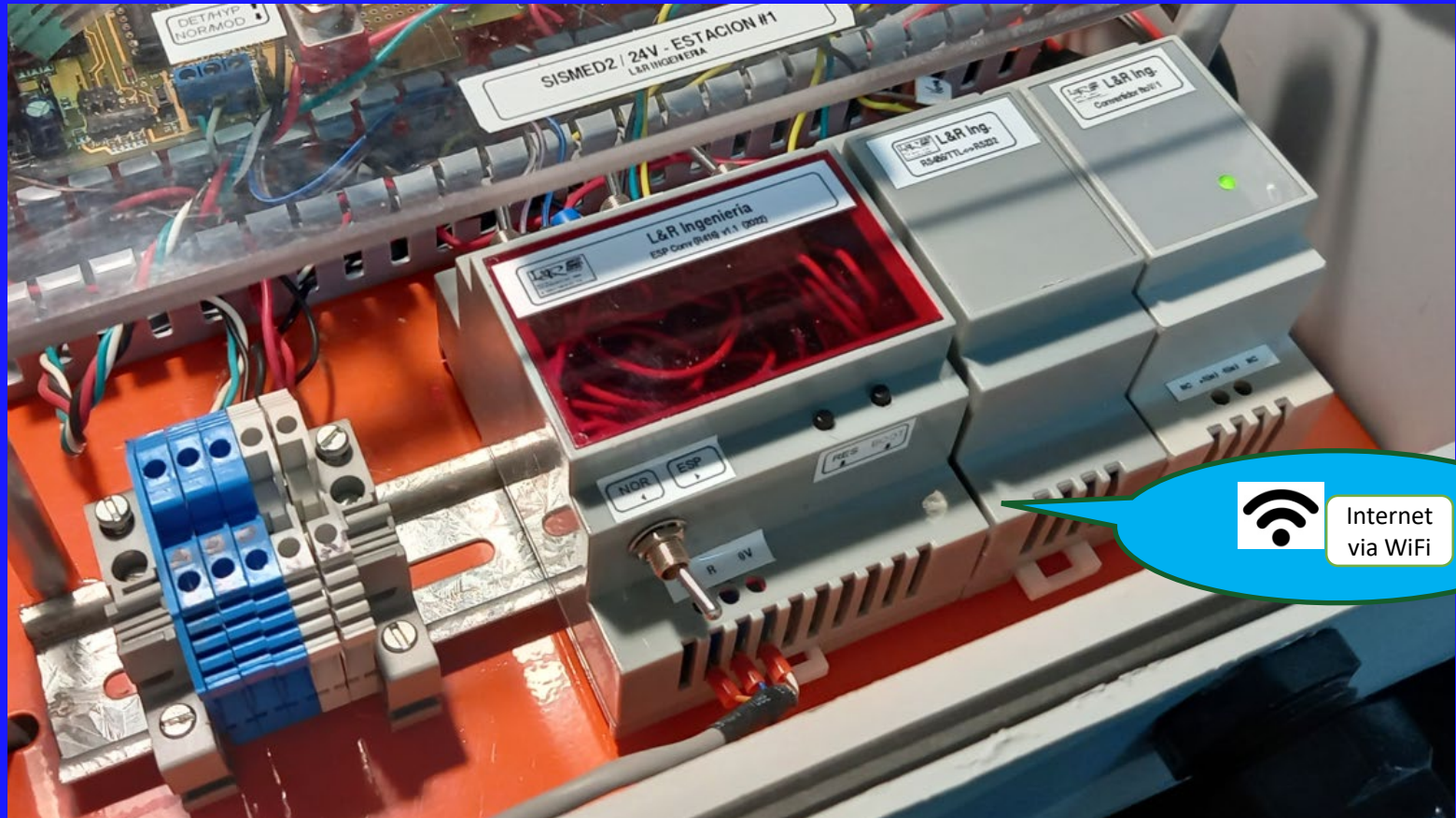


SISTEMAS
EMBEBIDOS
2024



Sistema de Medición SISMED/SJ24

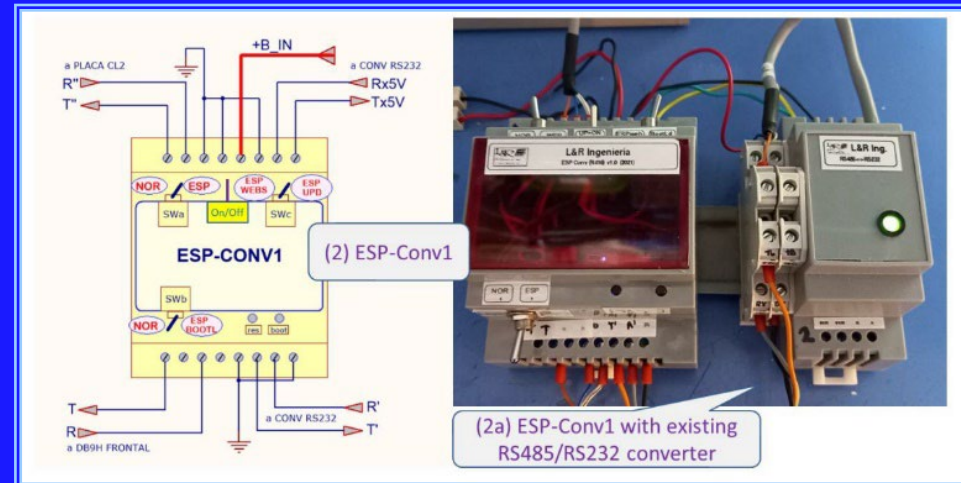
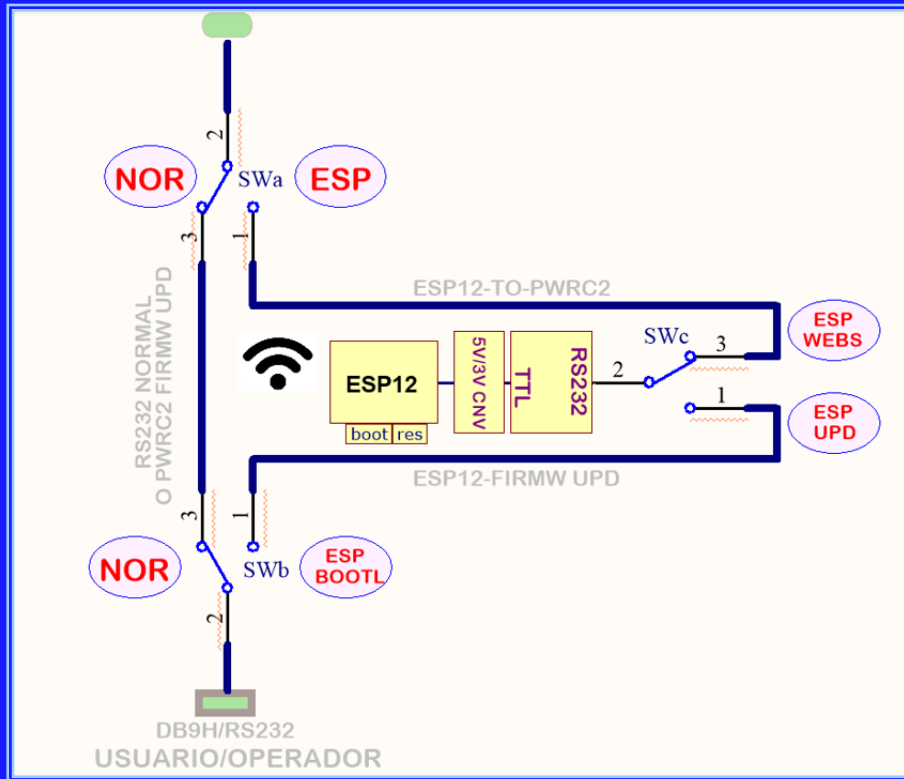
AGREGADO DE MODULOS WI-FI /2022



Sistema de Medición SISMED/SJ24

AGREGADO DE MODULOS WI-FI /2022

ESP-Conv1 -módulo que permite trabajar en forma convencional (con RS232 cableada), o vía un módulo ESP-12 (Espressif) basado en el controlador ESP8266, que integra una interfaz wi-fi y una serie de funcionalidades a un costo muy reducido.



Referencias:

- [a] Espressif (2022) Módulos WiFi ESP:
<https://espressif.com/en/products/modules/esp8266>
- [b] RS485 Modules (2022) online:
<https://www.lyringeneria.com.ar/productos/rs485-brd-placa-y-o-modulo-rs-485-rs-232-ttl/>



Sistema de Medición SISMED/SJ24

Acceso WiFi / Caso PWRC2

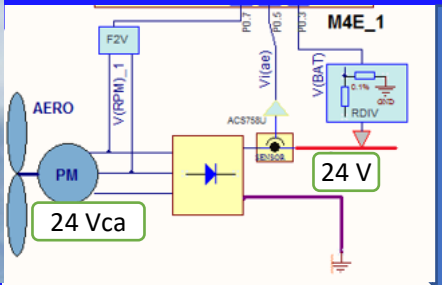
Modo 3 Menu de Configuración
PWRC2 vía WiFi ESP-Conv1

The screenshot shows a web browser window with the URL `espwrc2.local/config.html`. The page title is "ESPWRC2 Configuration". At the top, there are navigation buttons: "Main Page", "Software Upload", and "SPIFFS Editor". Below these are buttons for "Save Configuration", "Load Configuration", "Restart", and "Clean". The main section is titled "Configuration" and includes an "Edit JSON" button. It contains several input fields with labels and instructions:

- Host Name:** Input field containing "ESPWRC2". Below it, a note says: *"" for a generated unique hostname.*
- Your access point SSID (your router):** Input field containing "SPEEDY-ACC1E9". Below it, a note says: *Blank to disable connecting to an access point. * means connect to the last one used.*
- Password of your access point:** Input field with masked characters "*****".
- SSID when acting as an access point:** Empty input field. Below it, a note says: *Blank will create a suitable ssid.*
- Password when acting as an access point:** Empty input field.

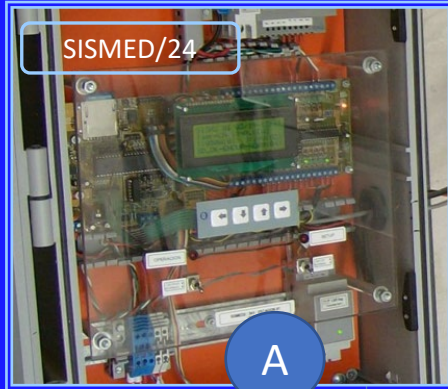
Sistema de Medición SISMED/SJ24

COMUNICACIÓN VÍA RBPi4 – WiFi (2023)

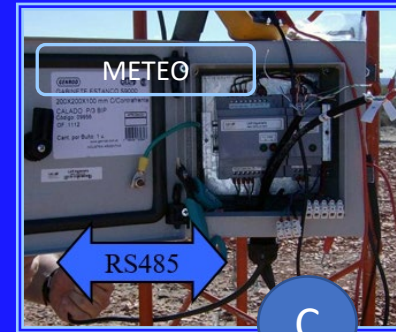


PV reg

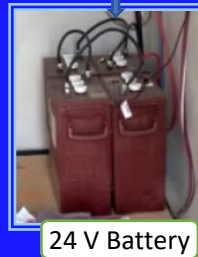
Acceso Web basado en RBPi4 con Interfaz gráfica (05-2023)



I_{PV}



Sensores de viento en UASJ



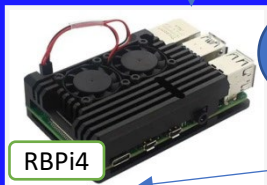
Inversor

24 V 220V

CA Invernadero

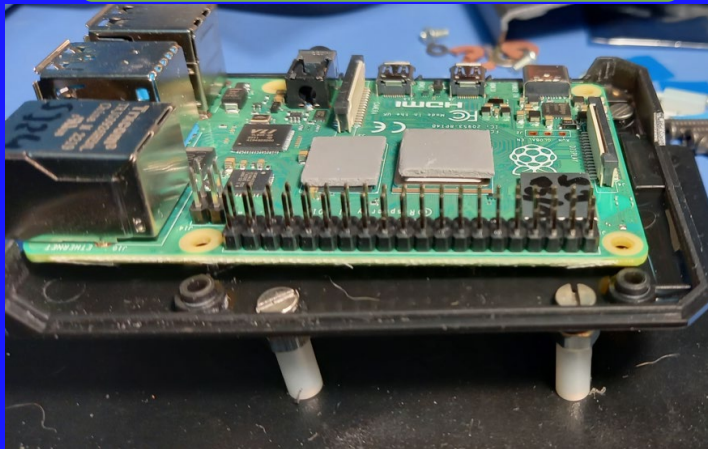
RS485 @38400 baud

RS232<-> USB

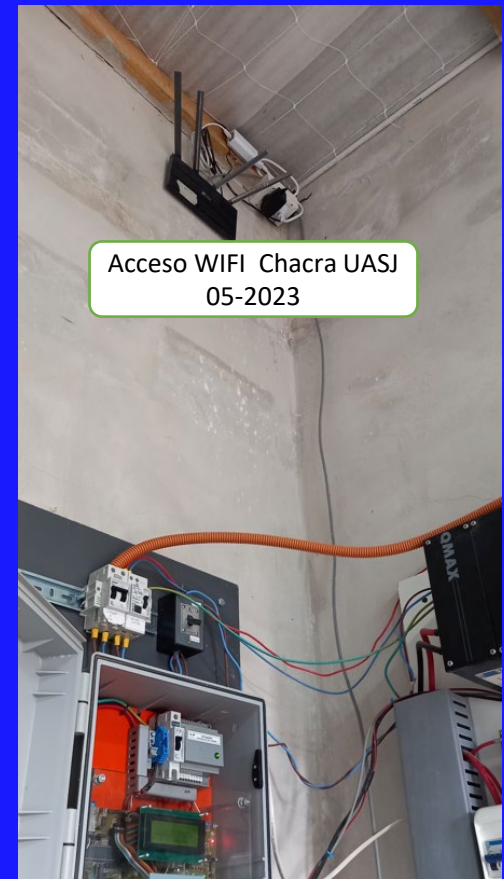


Internet via WiFi

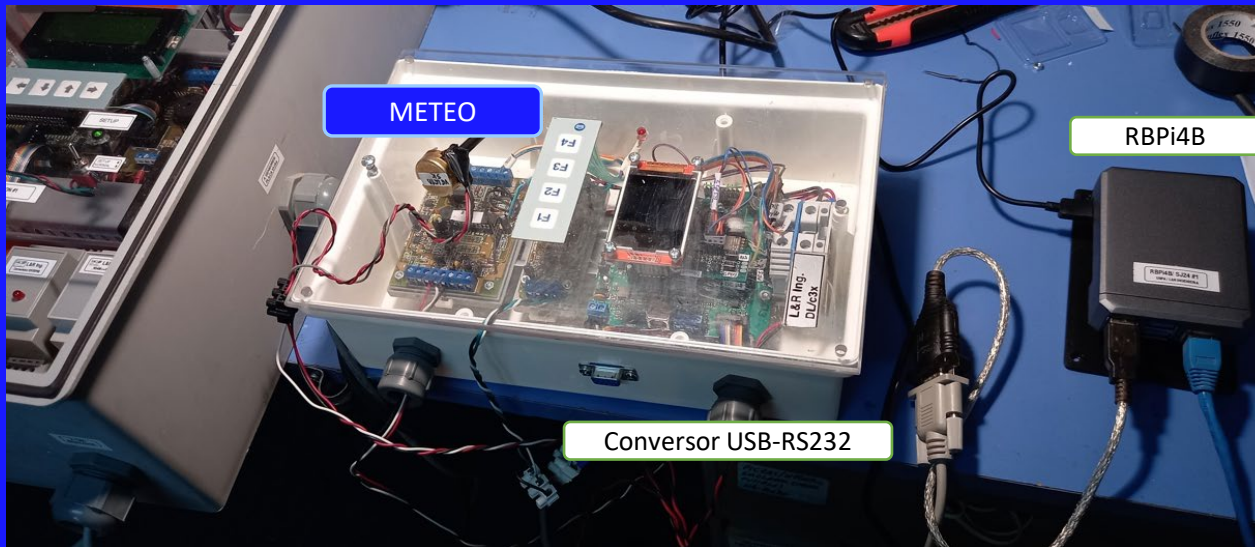
Ensayos y armado RBPi4 – bajo Raspbian Bullseye
(64-bit) version 2023



Acceso Web basado en RBPi4
(05-2023)

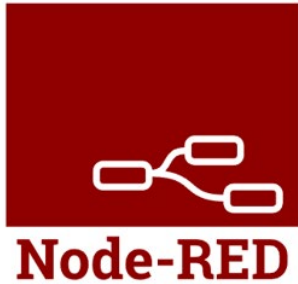


Acceso WIFI Chacra UASJ
05-2023



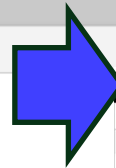
Uso de Node-RED v3.0 sobre RBPi4 – 64 bit

Acceso Web basado en RBPi4 (05-2023)



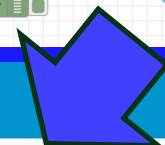
Username:

Password:



Programación acceso a SISMED-SJ24 con Modbus

MODBUS RTU SISMED-SJ24



Lectura Valores SISMEDSJ24

Tension VBat



Tension Vdc
121,38550,2,51



SISTEMAS
EMBEBIDOS
2024

Sistema de Medición SISMED/SJ24

COMUNICACIÓN VÍA RBPi4 – WiFi (2023)

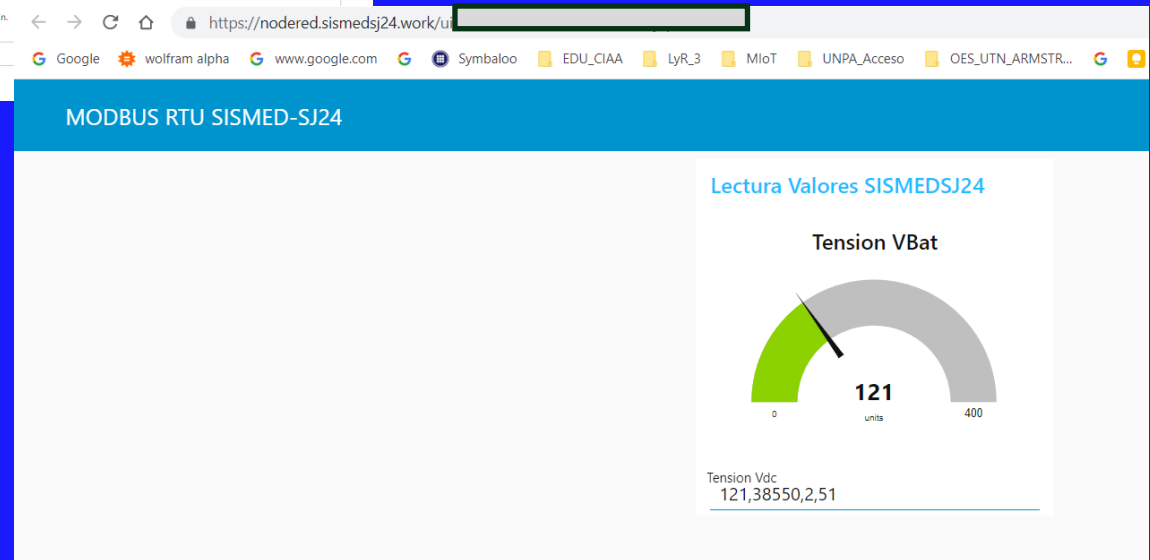


Definición de un dominio gratuito e implementación de un Tunel CloudFlare, versión gratuita

Acceso Web basado en RBPi4 (05-2023)

Acceso remoto a SISMED-SJ24 (trabajo en curso)

The screenshot shows the Cloudflare Tunnel configuration interface. On the left is a navigation menu with options like Overview, Analytics & Logs, DNS, Email, SSL/TLS, Security, Access, Speed, Caching, Workers Routes, Rules, Network, and Traffic. The main content area is titled 'Cloudflare Tunnel' and includes a warning about legacy tunnels being unsupported. Below the warning, there is a 'Launch Zero Trust Dashboard' button. At the bottom, there is an error message: 'Error loading Cloudflare Tunnel data. Refresh to try again.'



Solución de acceso remoto via RBPi4
- instalado 14-05-2023

Acceso WIFI Chacra UASJ
05-2023



RBPi4B

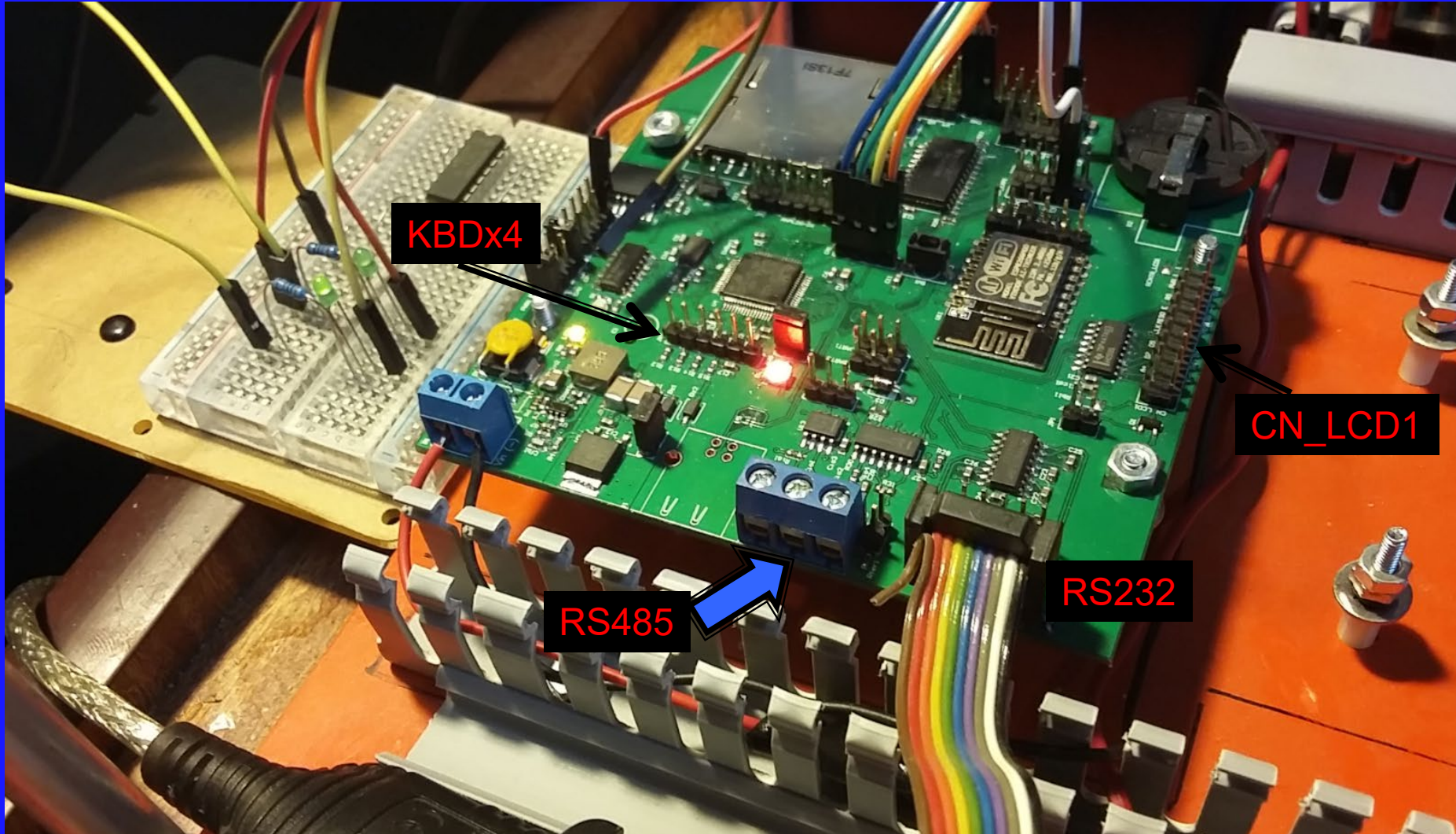
SISMED SJ/24

Acceso Web basado en RBPi4
(05-2023)



Convertor USB-RS232

Desarrollo de nueva placa CL3 (2018)



DETALLES DE PLACAS Y FIRMWARE CL2 Y CL3

Referencias:

Oliva, R. (2023) "Hardware and firmware design and implementation of twin 8-bit and 32-bit microcontroller boards for research and educational applications," in IEEE Embedded Systems Letters, 2022, doi: 10.1109/LES.2022.3189877 - <https://ieeexplore.ieee.org/document/9826356>

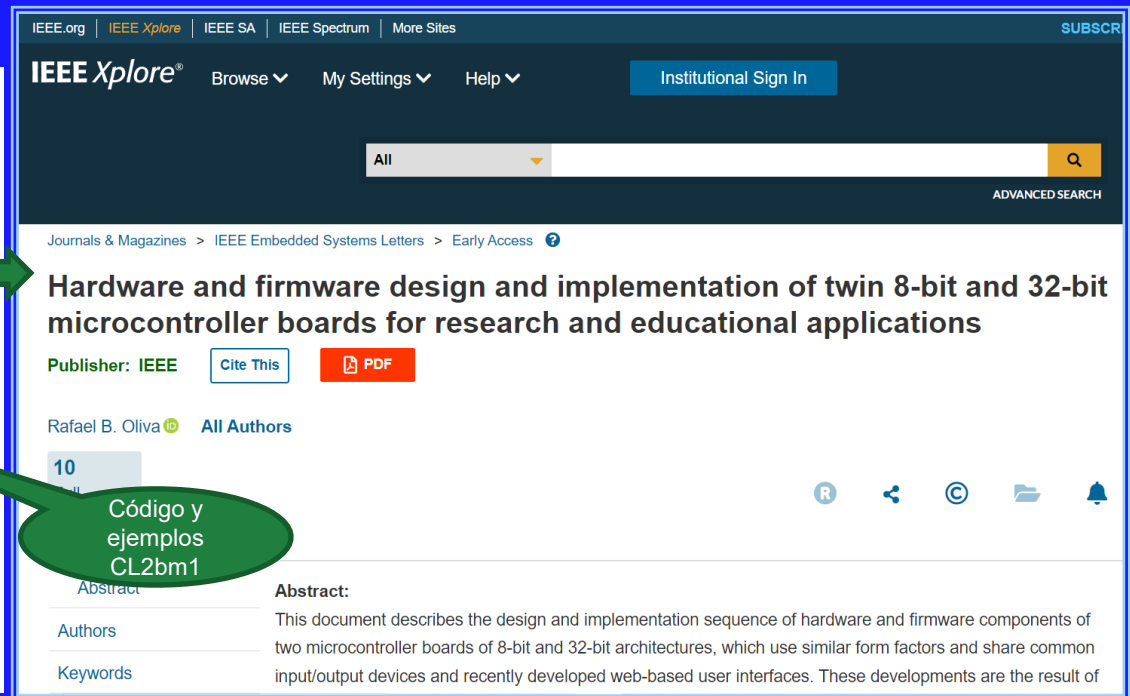
CL2bm1 CPU Board (2022) online:

<https://github.com/LyRIng/CL2bm1>

IHCR (2022) In-House C-Coding Rules / L&R

Ing.:

<https://github.com/LyRIng/InHouseCCodingRules>



IEEE.org | IEEE Xplore | IEEE SA | IEEE Spectrum | More Sites

IEEE Xplore® Browse My Settings Help Institutional Sign In

All

ADVANCED SEARCH

Journals & Magazines > IEEE Embedded Systems Letters > Early Access

Hardware and firmware design and implementation of twin 8-bit and 32-bit microcontroller boards for research and educational applications

Publisher: IEEE Cite This PDF

Rafael B. Oliva All Authors

10

Código y ejemplos CL2bm1

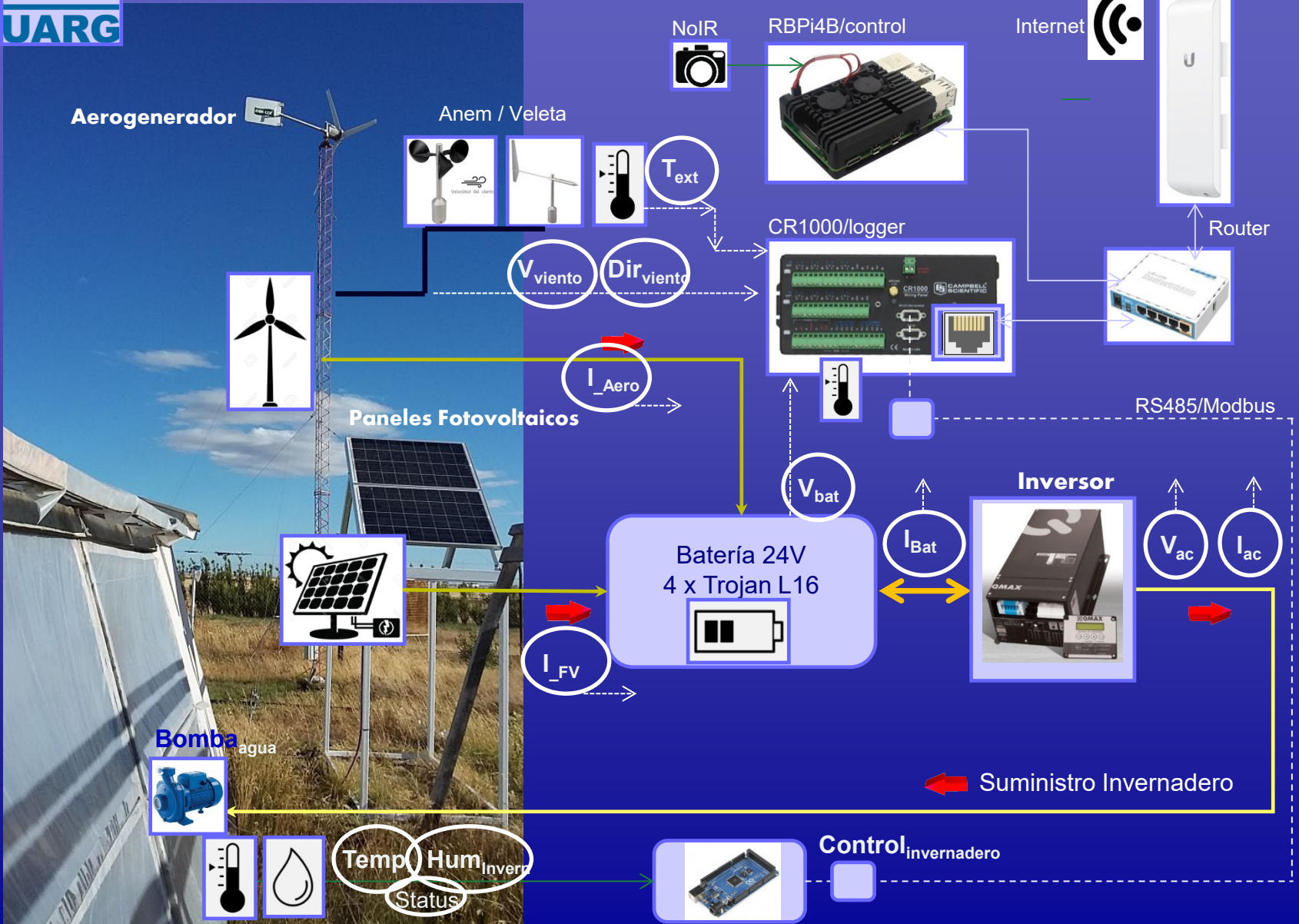
Abstract: This document describes the design and implementation sequence of hardware and firmware components of two microcontroller boards of 8-bit and 32-bit architectures, which use similar form factors and share common input/output devices and recently developed web-based user interfaces. These developments are the result of

Guía IHCR de buenas prácticas de codificación en C



Sistema Eolico/Solar AEA/UNPA-UARG

Suministro a Invernadero





SISTEMAS
EMBEBIDOS
2024



Computadora Industrial
Abierta Argentina
Desarrollo colectivo



Gracias por su atención! →
Continua Ing. Nestor Cortez

<https://www.energiasalternativas-unpa.net/>

<https://www.uarg.unpa.edu.ar/ita/index.php/investigacion-2/grupos-investigacion?showall=&start=1>