

Librería Arduino "Base"

Librería realizada para funcionar con la placa ESP32

Fernando Oviedo

Abril 2022

0.1 Aplicación IoT escalable

Existe un notable crecimiento en IoT que no hará más que aumentar, es indispensable contar con herramientas para hacerlo, pese a no ser responsable del diseño del microcontrolador al que se dirige ésta librería, ésta en sí misma es creada con el objetivo de representar una herramienta para el desarrollo de dispositivos IoT, desde prototipos hasta diseños finales.

Prácticamente todo dispositivo de Internet Of Things cumple con al menos dos características. Sin considerar la alimentación eléctrica que deben tener estos dispositivos, tenemos que incorporar.

- Capacidad de conectarnos con la red local
- Posibilidad de recibir actualizaciones OTA

0.2 La librería

0.2.1 Otras librerías incluidas

Cada librería listada aquí, cuenta con su propia documentación por tanto no serán incluidas características extensas de cada una, para funcional la librería base requiere:

- WiFi.h
- EEPROM.h
- WebServer.h
- ESPmDNS.h
- Update.h

Listadas con finalidad de evitar redundancia en el compilador al usar la base.

0.2.2 Sobre el uso

La forma de utilizar la librería base se ha hecho tan simple cómo me fue posible, partido en dos secciones para sus principales funciones.

Conexión con WiFi local

Cómo en todo dispositivo IoT, es de suma importancia añadir la posibilidad de registrar una red WiFi propia del usuario, esto se realiza con la función:

```
Void base(){
    while (!wifistate()){
        createap();
    }
}
```

Ésta función deberá ser incluida en el void loop().

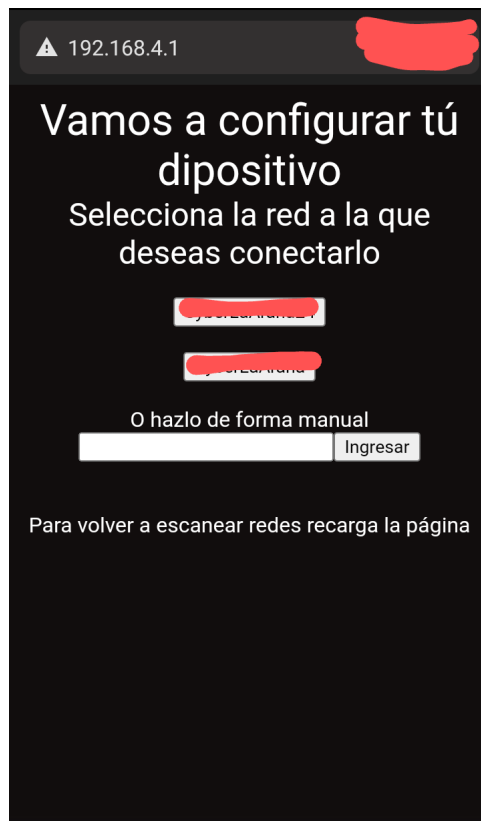
```
void loop(){
    base();
    //Aquí tu código.
}
```

Se notará que al contener un while este pedazo de código se ejecutará en bucle hasta conectar con una red WiFi. El código ha sido diseñado para operar con la función millis() en lugar de la función delay().

De cara al Usuario la configuración es realizada al conectarse a la red WiFi generada por el dispositivo, por la forma en que se realizó el código fuente esta red se generará incluso teniendo una red previamente guardada si no se logra conectar, permitiendo reconfigurar en caso de cambios de contraseña en la red.

Para realizar la configuración, se deberá, conectarse a la red WiFi creada por el dispositivo "Naceli", desde cualquier dispositivo que disponga de un navegador, utilizando la contraseña por defecto "Naceli-Home#".

Para configurar una vez conectado será necesario abrir la página web <http://192.168.4.1> que desplegará la página de selección de SSID.



Se dará la posibilidad de seleccionar con alguno de los botones desplegados la red a la que se desea conectar, o, en caso de ser necesario, escribirlo de forma manual, una vez hecho esto se solicitará la contraseña de la red.



Una vez introducida la contraseña el dispositivo intentará conectarse, en caso de ser correcta y establecer conexión guardará el nombre de red y contraseña en la memoria EEPROM en las direcciones 0 y 50 respectivamente

Actualizaciones OTA

OTA, siglas de Over The Air, una forma de actualización extremadamente común y cómoda de cara al usuario, incorporada para producción y prototipado, al facilitar incluir código nuevo sin necesidad de retirar el dispositivo del lugar en que se encuentre, mientras esté alimentado, claro está.

Para utilizarlo se debe hacer uso de la función:

```
void updateota(bool usar = false){  
  if (usar and wifistate){  
    //Código de inicio de actualización  
  }  
}
```

```
}
```

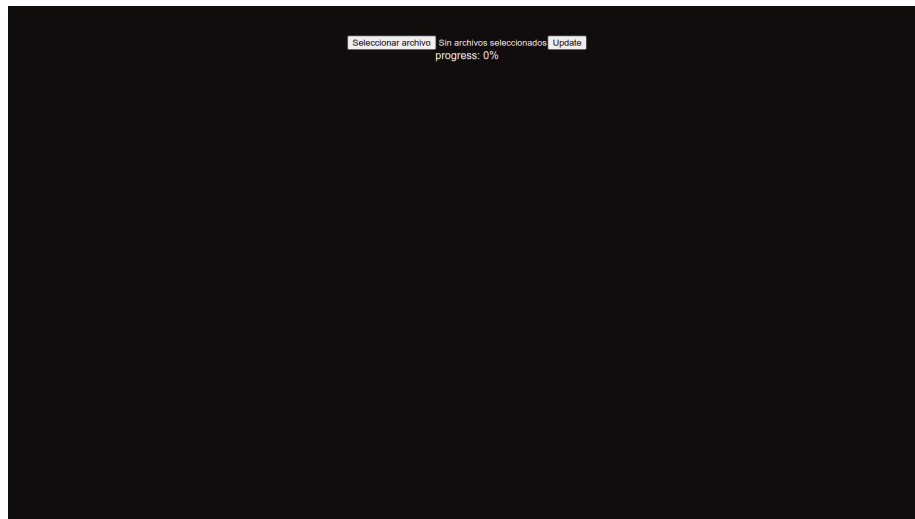
Cómo se notará, para iniciar el proceso de actualización basta invocar la función en la parte del código que mejor nos parezca

```
...  
if (quiero_actualizar){  
    updateota(true);  
}  
...
```

Para detenerlo, cómo se podrá ver, basta con enviar un false al valor "hacer" o simplemente esperar a que el proceso de actualización termine, esto reiniciará el dispositivo con el nuevo programa cargado.

El proceso de actualización es extremadamente similar al de configuración de la red WiF, ya que hace uso de una página HTML en el dispositivo, sin embargo, por necesidad de hacer esto únicamente si se encuentra conectado a una red WiFi, será complicado dar con la dirección IP asignada al dispositivo, para conectarnos y ver la página lanzada, es por ello que se incorpora un mDNS, encargado de asignar un valor fijo para acceder a esta.

Para iniciar el proceso de actualización, de cara al Usuario, es necesario estar conectado a la misma red que el dispositivo, e ingresar a la página `naceliupdate.local`



Será tan simple como cargar un archivo binario (extensión .bin) y dar al botón Update

0.3 Mensaje final

Ésta es la primer versión de la librería base, existe mucho margen de mejora, y existen muchas posibilidades de uso para el estado actual de la misma, es una de tantas herramientas disponibles y probablemente no sea la más completa, pero está hecha con pasión y ganas de crear algo útil.