

Raduino 32

Ramón Lorenzo EA4GZI

Se trata en este proyecto de utilizar para el control de la radio **Ubitx v6** un microcontrolador ESP32 en lugar de un Arduino Nano. El ESP32 es más rápido y, además, dispone de hasta 4Mb de RAM y de la posibilidad de utilizar archivos mediante SPIFFS.

Adicionalmente, permite utilizar conexión WiFi o BT.

He partido del firmware original de Ashhar Farhan, VU2ESE y partes del firmware CEC de KD8CEC, a los que agradezco su esfuerzo.

Consideraré con mucho gusto cualquier sugerencia o corrección que se me haga, aunque no puedo garantizar su puesta en práctica en un plazo determinado.

FUNCIONES

Se han conservado las funciones básicas del diseño original, por ejemplo

- Modos LSB,USB,CW.
- Doble VFO
- RIT
- Split
- Entrada directa de frecuencia.
- CAT control.

y se han añadido algunas **nuevas**:

- La configuración reside en un fichero llamado ***ubitx.cnf***, no en EEPROM.
- Cambios en la interfaz de usuario.
- Múltiples pasos de sintonía, desde 1 hz hasta 100 Mhz.
- Función SCAN.
- Temperatura transistores de potencia.
- Medidor de ROE (SWR)
- Control y configuración vía Web
 - Calibración
 - Parámetros usuario.
 - Límites de bandas
 - Servicios de red: IFTTT, Dweet.io, MQTT, FTP server.
- RTC vía NTP
- Actualización remota del firmware vía web.
- Segundo puerto serie para depuración u otros usos, además del primero usado para el CAT.
- Idiomas Español e Inglés. Pueden añadirse fácilmente otros idiomas mediante un fichero.

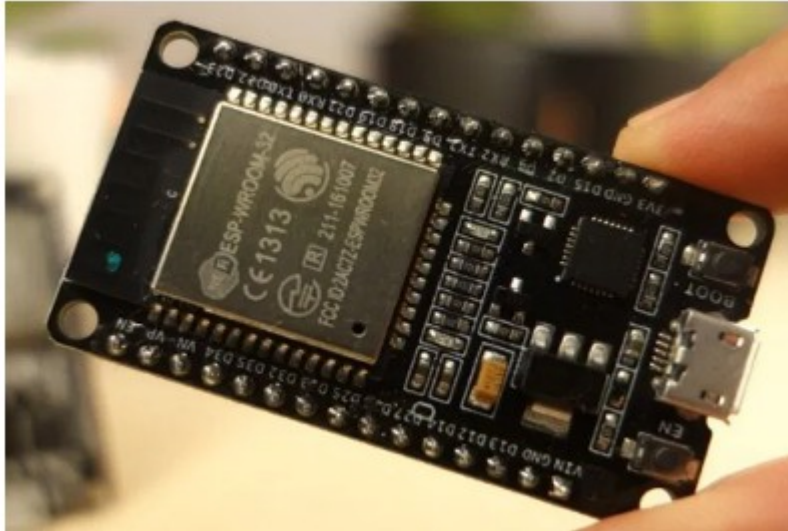
Otras funciones en desarrollo:

- Número ilimitado de canales de memoria.
- Control de temperatura de los disipadores de calor.

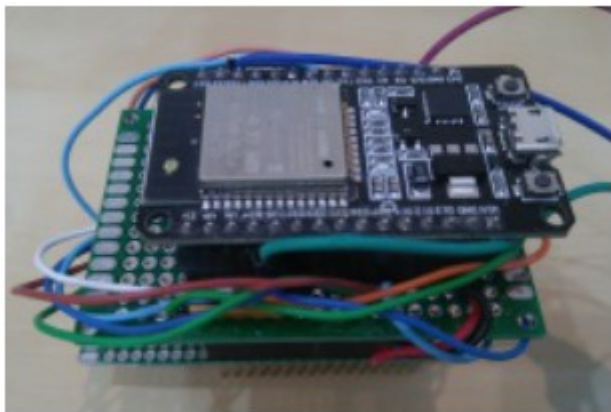
HARDWARE

La única modificación hardware necesaria es la sustitución del Arduino Nano por un módulo ESP32, en particular el módulo DOIT Esp32 DevKit v1 que puede conseguirse online.

No es necesario intervenir ni soldar en el circuito original.



Para el prototipo y desarrollo se ha conectado a la radio mediante un adaptador cableado que se monta directamente en el zócalo del Arduino Nano original.



Ha sido necesario añadir unos prolongadores para evitar el disipador de calor del módulo Arduino. En el diseño final de la PCB adaptadora podrá colocarse el ESP32 más cerca.

Los pines utilizados, obviamente, son diferentes a los utilizados en el el Arduino Nano. En la siguiente tabla pueden verse los pines en cada caso.

SOFTWARE

El software está en desarrollo y puede haber nuevas funciones o cambios menores en las ya existentes.

Utilizo para el desarrollo el editor UECIDE (<https://uecide.org/>), más potente y rápido que el IDE de Arduino. No he probado a compilarlo con Arduino IDE.

He partido del código de KD8CEC, aunque he modificado muchas partes del mismo con el propósito de simplificarlo, eliminando las partes no usadas. Algunas funciones las he utilizado tal como estaban, otras las he modificado y en otros casos he utilizado las originales de VU2ESE.

El resultado es una mezcla de código propio y de unos y de otros. Si a alguien le resulta interesante me daré por satisfecho.

El código, también el fichero ***ubitx32_20.bin***, puede encontrarse en <https://github.com/ramonlh/ubitx32>

Para la programación del módulo ESP32 directamente con el fichero ***.bin*** puede usarse *Flash Download Tools de Espressif* : <https://www.espressif.com/en/support/download/other-tools>

INSTALLING UECIDE para ubitx32_20

DESCARGA

Descargar Uecide desde <https://uecide.org/download> (última versión 0.11.8)

Si es la primera vez, descargar la versión “full”.

Instalar con las opciones por defecto.

EJECUCIÓN

Si no está instalado Java Runtime, ir a <https://java.com/es/download/> e instalarlo.

Descargar Java para Windows

Recomendado Version 8 Update 261 (Tamaño de archivo: 1.99 MB)

Fecha de versión: 14 de julio de 2020



Actualización importante de la licencia de Oracle Java

La licencia de Oracle Java ha cambiado para las versiones publicadas a partir del 16 de abril de 2019.

El nuevo [acuerdo de licencia de Oracle Technology Network para Oracle Java SE](#) es sustancialmente diferente a las licencias de Oracle Java anteriores. La nueva licencia permite ciertos usos, como el uso personal y de desarrollo, sin coste alguno (aunque podría haber otros usos autorizados en licencias de Oracle Java anteriores que ya no estén disponibles). Revise las condiciones con atención antes de descargar y utilizar este producto. Puede consultar las preguntas frecuentes [aquí](#).

La licencia comercial y el soporte están disponibles con una [suscripción de Java SE](#) de bajo coste.

Oracle también ofrece la última versión de OpenJDK con la [licencia pública general](#) de código abierto en jdk.java.net.

Aceptar e iniciar descarga gratuita

Al descargar Java, confirma que ha leído y acepta las condiciones del [acuerdo de licencia de Oracle Technology Network para Oracle Java SE](#)

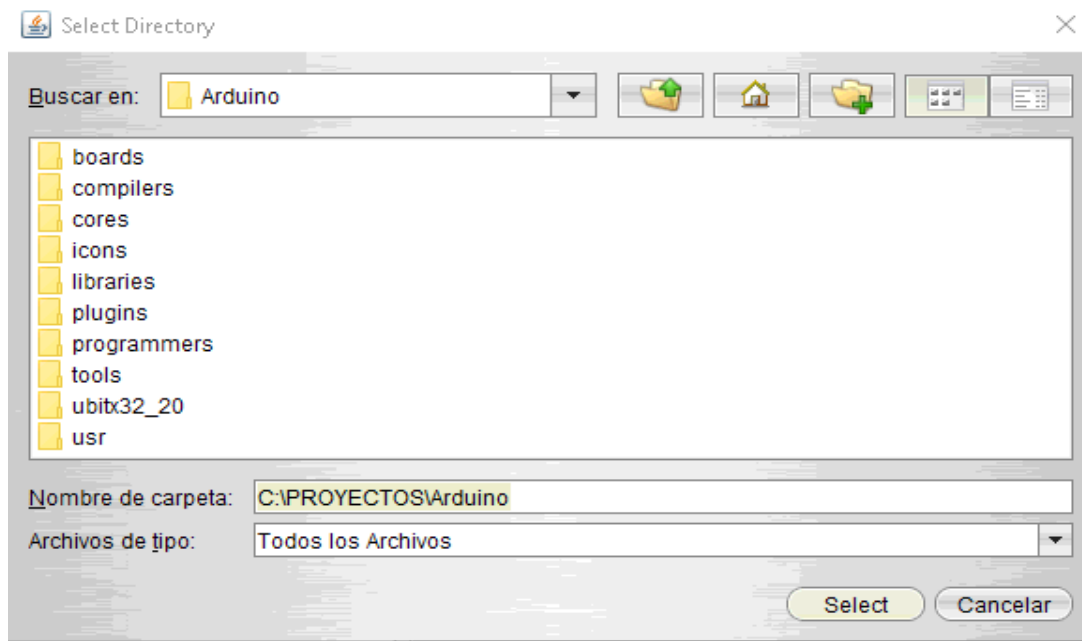
Una vez instalado Java Runtime, ejecutar de nuevo Uecide.

INSTALAR PLACA

Ir a **Tools->Plugin manager->Boards**

Buscar “ESP32”, instalar la placa **esp32doit-devkit-v1**

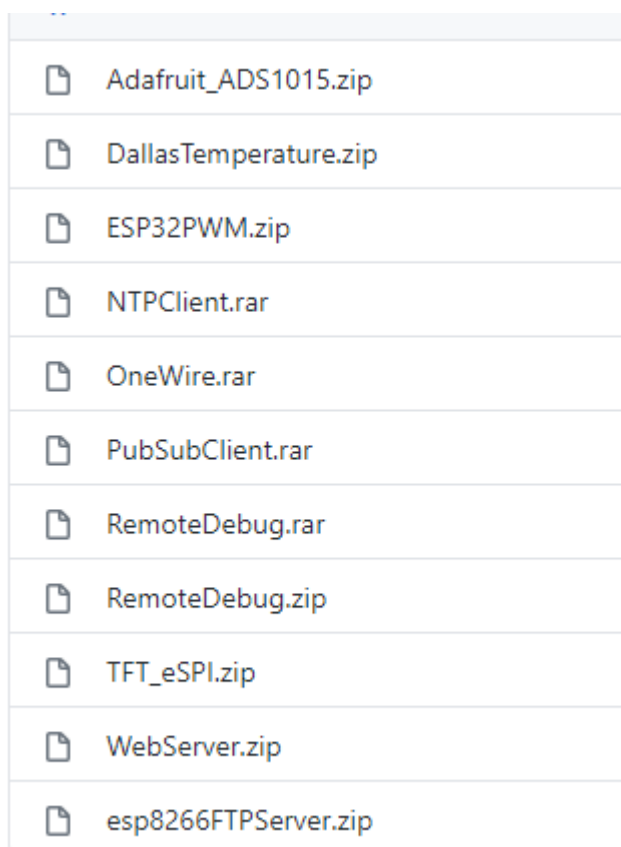
Ir a **File->Preferences->Locations**: poner “SketchBook location” a la carpeta donde están los proyectos Arduino (por ejemplo



INSTALAR LIBRERÍAS

Descargar las librerías desde <https://github.com/ramonlh/ubitx32/tree/master/libraries> en una carpeta a nuestra elección.

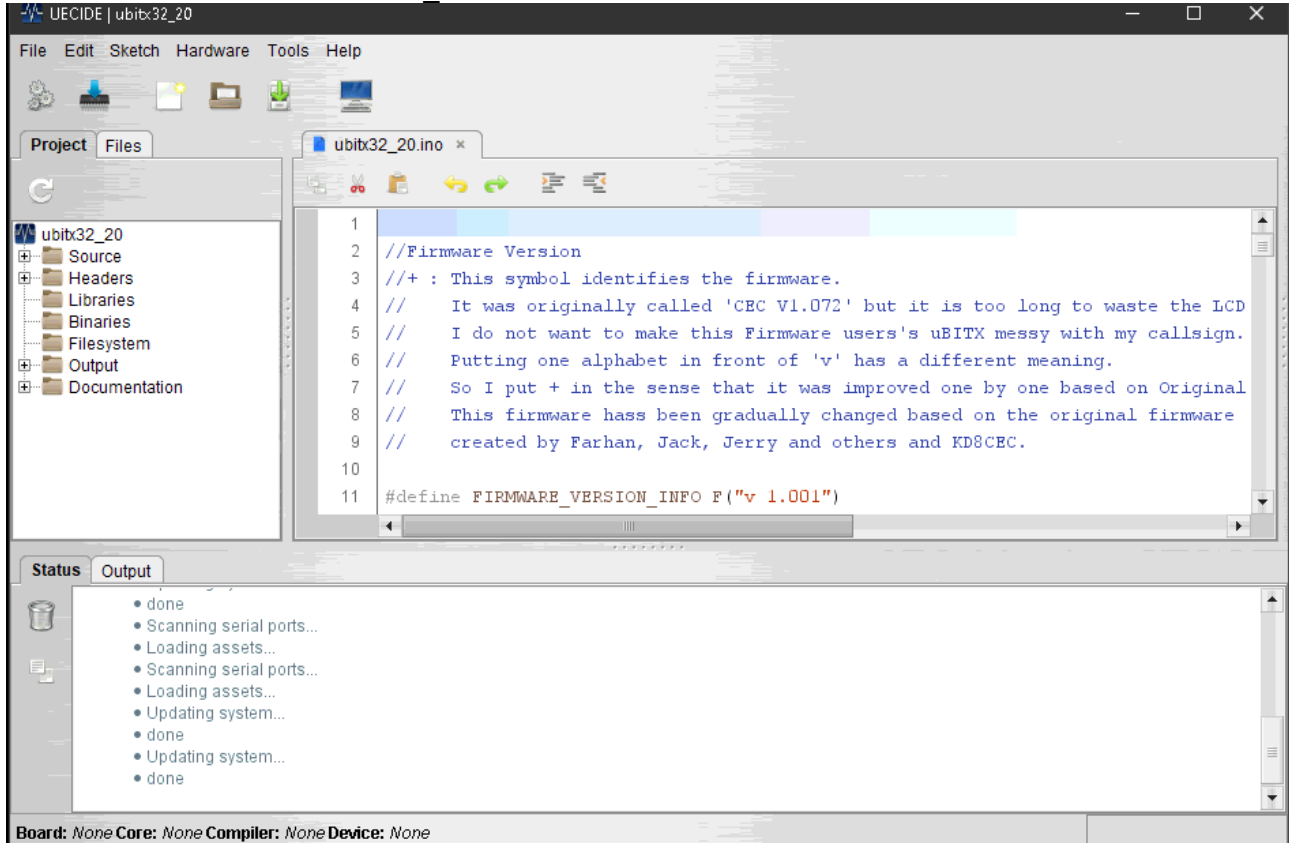
Ir a **File->Preferences->Locations**, click “**Add Location**”, seleccionar la carpeta en la que hemos descargado las librerías.



Descomprimir todas las librerías, cada una en su carpeta.

Después de añadir las librerías es necesario reiniciar Uecide.

ABRIR PROYECTO “ubitx32_20”.



Seleccionar la placa: **Hardware->Board->ESP32->Doit.....**

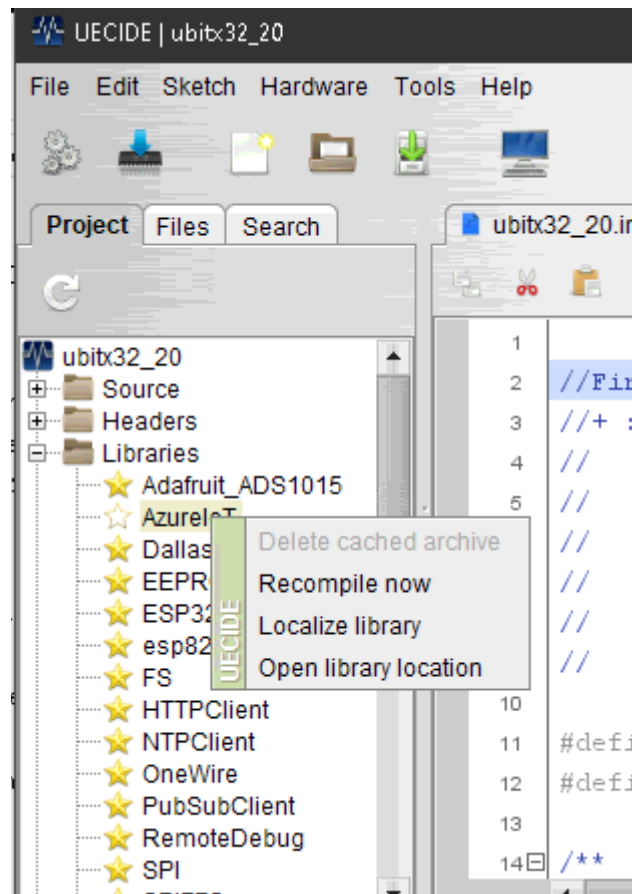
Seleccionar puerto serie: **Hardware->Devices->COMXX**

Compile ubitx32_20

Abrir proyecto: **File->Open...ubitx32_20**

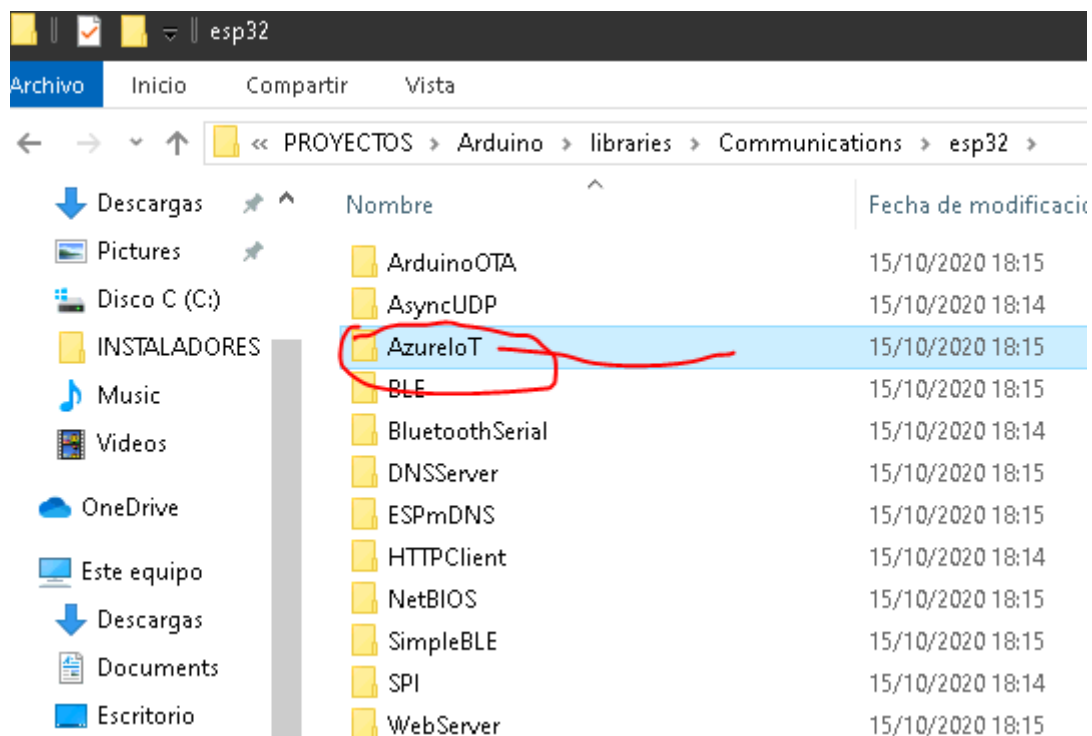
Por alguna razón, se añade al proyecto la librería **AzureIoT** y se produce un error. Es necesario eliminarla del proyecto:

Click sobre el botón derecho y “Open library location”.



Se abrirá una ventana del explorador de archivos:

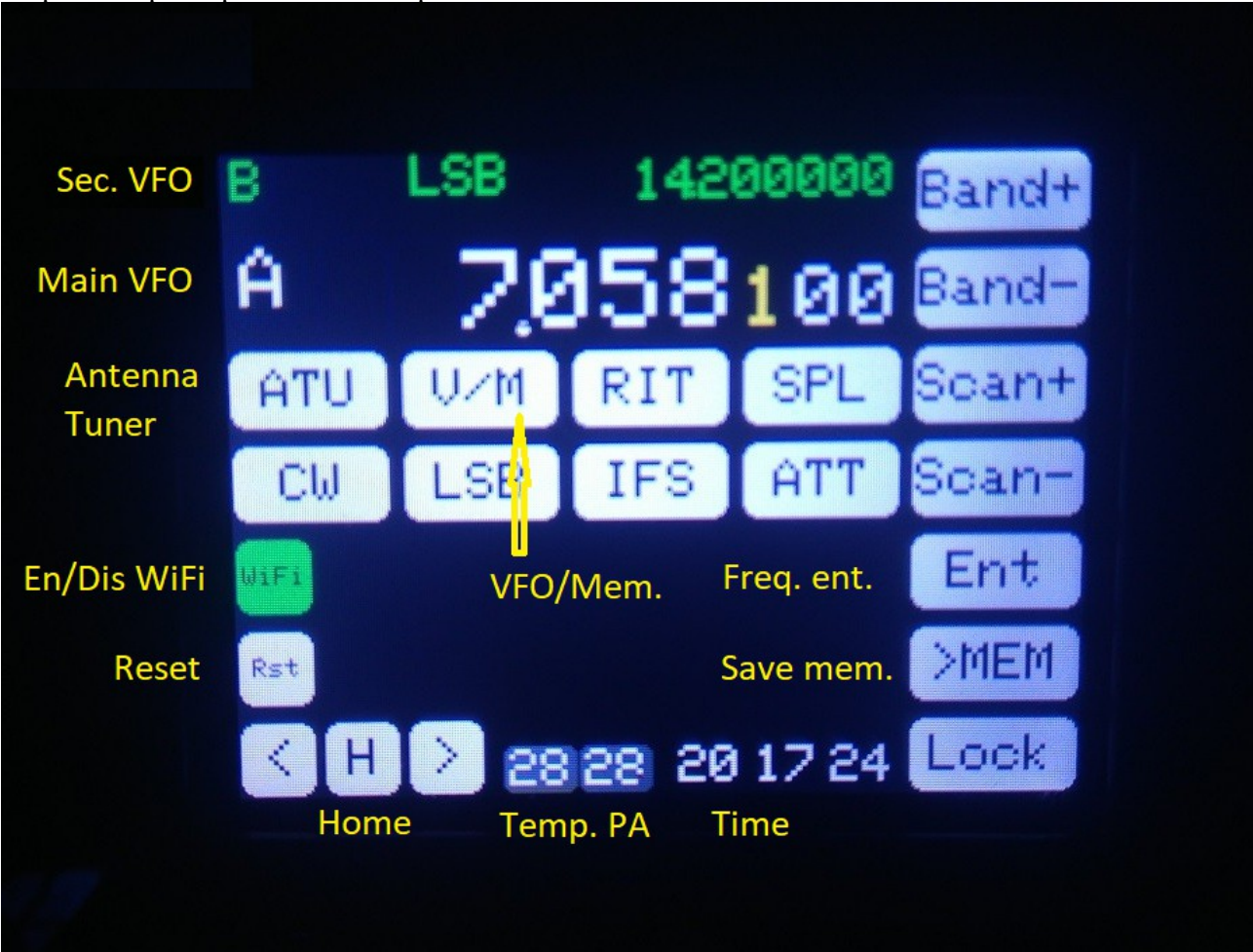
Borrar la carpeta **AzureIoT**.



Cerrar y volver a abrir Uecide. La librería ya no estará y compilará correctamente.

DESCRIPCIÓN


La pantalla principal tiene este aspecto



Botones



Configuración

Pulsando el botón  se avanza por las páginas de configuración

Página “1 – Setup”

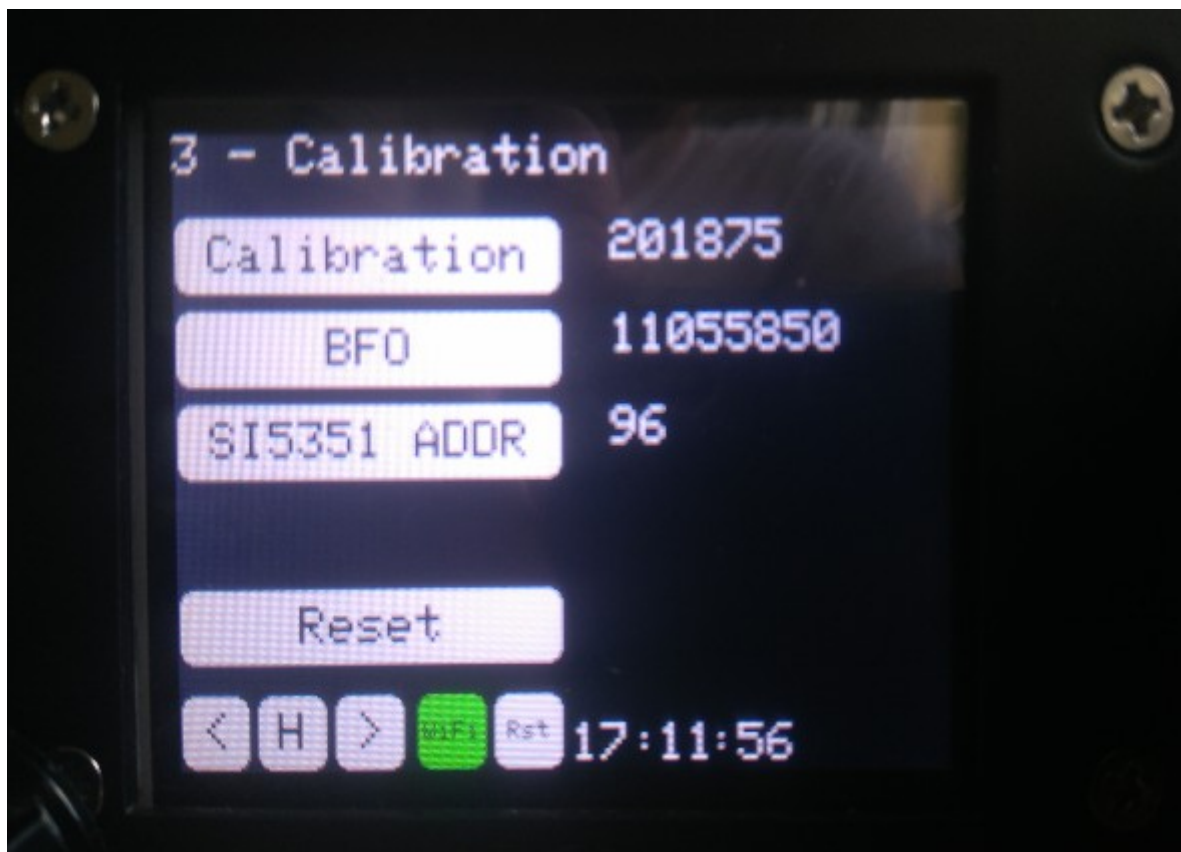


En esta página se pueden editar Idioma, Distintivo, Latitud y Longitud pulsando en el botón correspondiente.

Opciones:

- “**Auto Connect**”: si está activado, el equipo intenta conectarse a la red definida cuando arranca.
- “**Scan SSID**”: explora las redes WiFi para seleccionar a la que nos queremos conectar.
- “**Password**”: Contraseña de la red WiFi a la que nos queremos conectar.
- “**WiFi Mode**”: el ESP32 puede funcionar de tres modos. Modo STA, como estación para conectarse a un AP. Modo AP, como punto de acceso para conectarnos desde otro equipo y modo AP+STA como ambos a la vez.
El modo AP podremos conectarnos con el PC u otro equipo y el ESP32 nos proporcionará una dirección del tipo 192.168.4.xx. El SSID será UBITX y la dirección IP de la radio será 192.168.4.1.
- “**Static IP**”: si está activado la dirección IP del equipo en modo STA será fija (no DHCP) y será 192.168.1.149. (Pendiente poder cambiar).

Página “3 - Calibration”



En esta página se tiene acceso a los procedimientos de calibración inicial de la radio. Dirigirse al manual del equipo original ya que el procedimiento es el mismo.

Después de pulsar el botón “Calibration” aparece esta pantalla



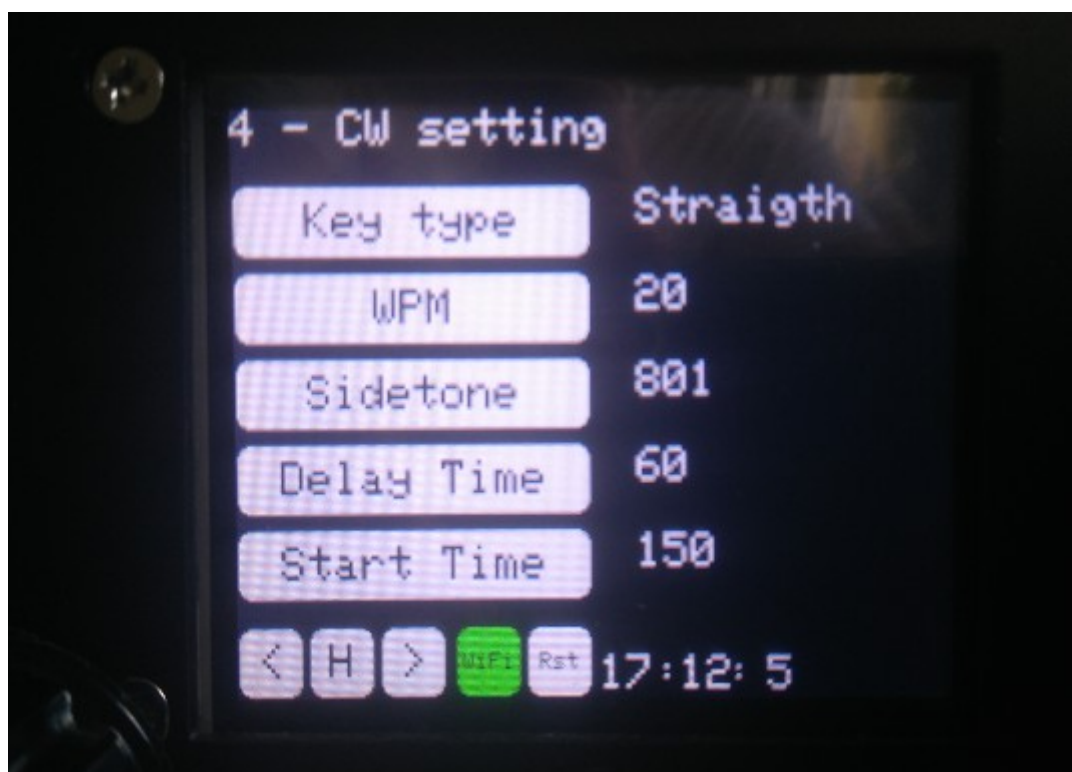
Girando el mando del dial buscaremos el valor adecuado y lo aceptaremos pulsando el mismo mando o pulsando el botón “OK”.

Este procedimiento es igual en todos aquellos casos en que hay que seleccionar un valor.

También puede encontrarse mucha ayuda en:

Grupo bitx20: groups.io/g/bitx20

Página <https://ubitx.net/>



Acceso al equipo mediante navegador

Si está activada la IP estática, la dirección será <http://192.168.1.149:88/>

El panel **principal**

Panel	Configuración	Sistema
uBitx		
VFO	A	
Mode	USB	
CW	OFF	
RIT	OFF	
Split	OFF	
Freq. A	14266100	
Freq. B	7131000	
Freq. actual	14266100	
Band 160 m	1989600	1999900
Band 80 m	3769900	3799000
Band 60 m	5366200	5366900

Pantalla “**Configuración**”

Panel	Configuración	Sistema		
Dispositivo	Bandas	Red	Serv. Red	Local I/O
CALLSIGN	<input type="text" value="EA4GZI"/>			
Calibration/usbCarrier	<input type="text" value="201875"/>		<input type="text" value="11055850"/>	
Latitud/ Longitud	<input type="text" value="112345.000000"/>		<input type="text" value="123344.000000"/>	
Idioma	<input type="text" value="Español"/> ▼			
CW Speed	<input type="text" value="60"/>			
cwDelayTime	<input type="text" value="60"/>			
cwKeyType	<input type="text" value="iambica"/> ▼			
SI5351BX_ADDR	<input type="text" value="96"/>			
Reset periodico (horas)	<input type="text" value="1"/> ▼			
<input type="button" value="Guardar"/>				

Pantalla “**Bandas**”

Des esta pantalla se pueden redefinir los límites de cada banda si es necesario.

Panel	Configuración	Sistema
Dispositivo	Bandas	Red Serv. Red Local I/O

Banda	desde	hasta khz
160 m	<input type="text" value="1810"/>	<input type="text" value="2000"/>
80 m	<input type="text" value="3500"/>	<input type="text" value="3800"/>
60 m	<input type="text" value="5351"/>	<input type="text" value="5367"/>
40 m	<input type="text" value="7000"/>	<input type="text" value="7300"/>
30 m	<input type="text" value="10100"/>	<input type="text" value="10150"/>
20 m	<input type="text" value="14000"/>	<input type="text" value="14350"/>
17 m	<input type="text" value="18068"/>	<input type="text" value="18168"/>
15 m	<input type="text" value="21000"/>	<input type="text" value="21450"/>
12 m	<input type="text" value="24890"/>	<input type="text" value="24999"/>
10 m	<input type="text" value="28000"/>	<input type="text" value="29700"/>

Pantalla “**Red**”

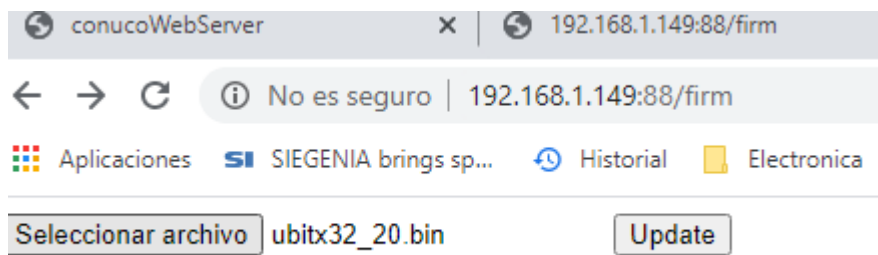
Acceso a los parámetros completos de la conexión WiFi

Panel	Configuración	Sistema
Dispositivo	Bandas	Red Serv. Red Local I/O

Auto WiFi	<input checked="" type="checkbox"/>
Modo	AP+STA ▼
STA mode SSID	<input type="text" value="conuco4"/> <input type="button" value="Scan"/>
STA mode Password	<input type="password" value="123456789"/>
AP mode SSID	UBITX
AP mode Password	<input type="password" value="12341234"/>
Canal	3 ▼
MAC	
IP estática	<input checked="" type="checkbox"/>
Dir. IP	192.168.1.149
IP	<input type="text" value="192"/> <input type="text" value="168"/> <input type="text" value="1"/> <input type="text" value="149"/>
Mask	<input type="text" value="255"/> <input type="text" value="255"/> <input type="text" value="255"/> <input type="text" value="0"/>
Gateway	<input type="text" value="192"/> <input type="text" value="168"/> <input type="text" value="1"/> <input type="text" value="1"/>
DNS	<input type="text" value="8"/> <input type="text" value="8"/> <input type="text" value="8"/> <input type="text" value="8"/>
IP pública	185.178.80.2

Pantalla “Actualizar”

Desde aquí puede actualizarse el firmware si se dispone del fichero “UBIT32_20.BIN” una vez compilado.



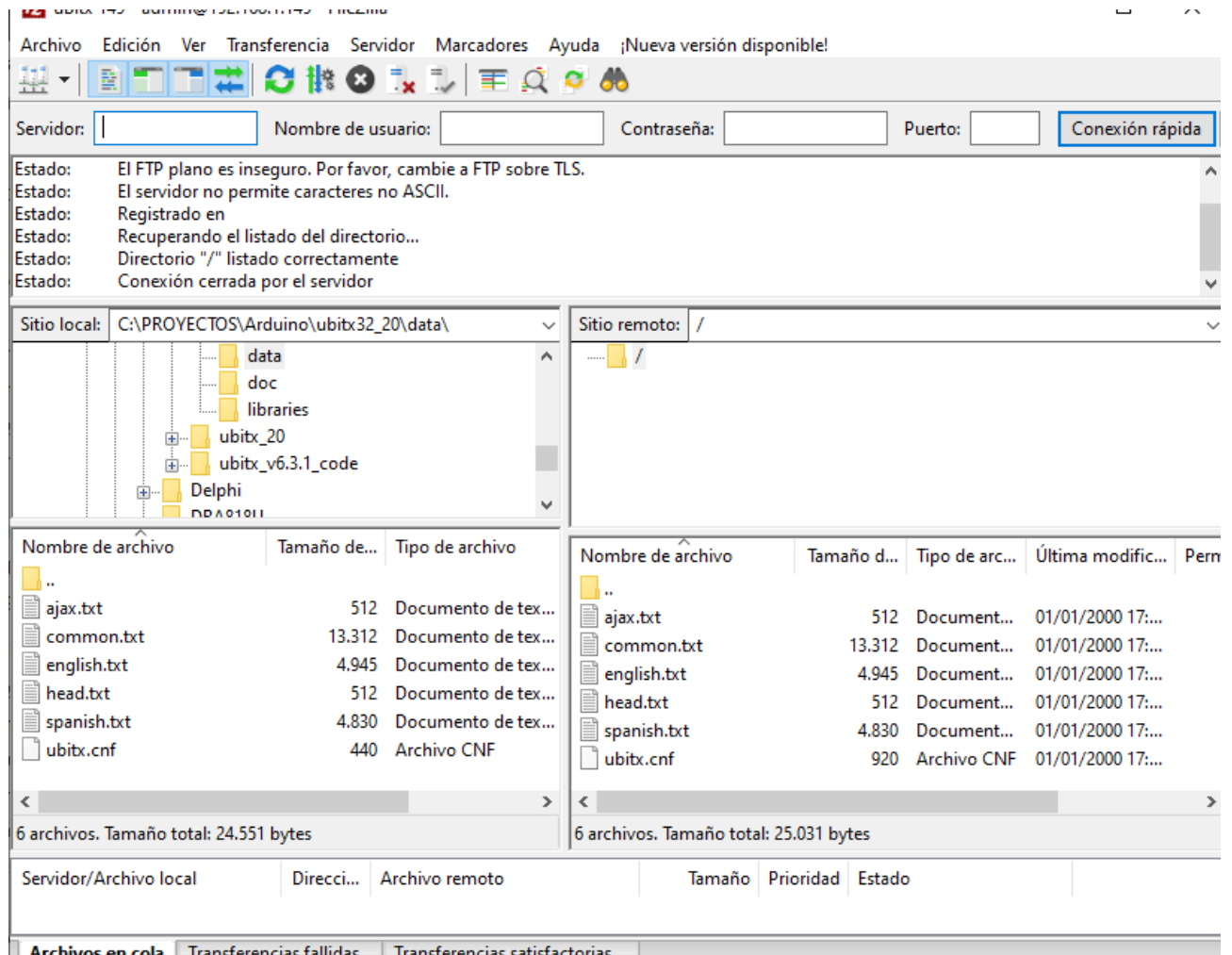
Si todo funciona como debe, se actualizará el firmware con el nuevo fichero y el equipo se reiniciará.

Hay otras pantallas que no se describen porque están aún en desarrollo o no están probadas.

Acceso FTP

El acceso mediante FTP permite gestionar los ficheros del sistema, hacer copia de seguridad o añadir nuevos ficheros de idioma.

Por el momento, la contraseña de acceso FTP es “admin/admin”. Puede desactivarse el servidor FTP des la página web.



Ramón EA4GZI