

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 (en pruebas)

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.
- 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:

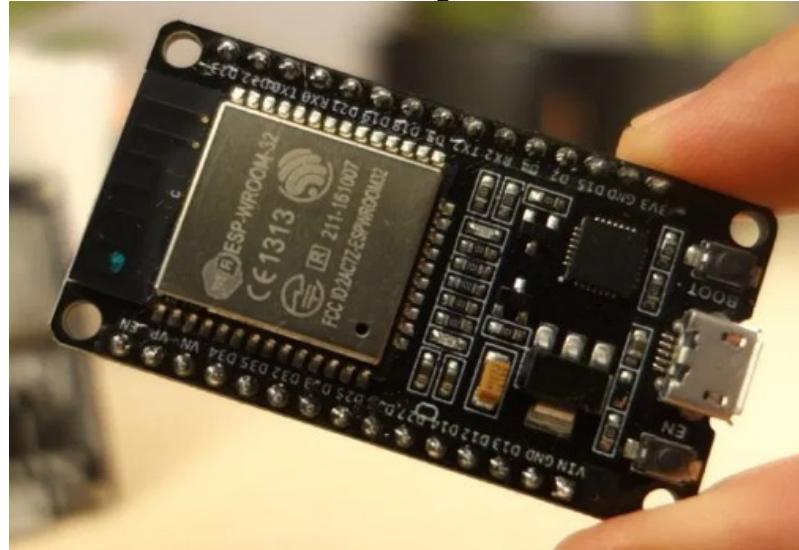
- Número ilimitado de canales de memoria.
- Control de temperatura de los disipadores de calor.

HARDWARE

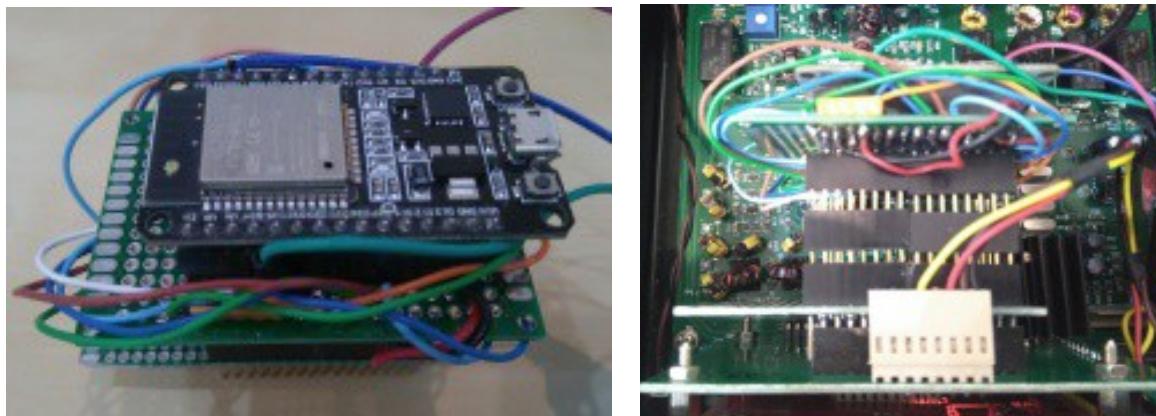
La única modificación hardware necesaria es la sustitución del Arduino Nano por un módulo

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 Raduino. 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.

Señal	Pin Raduino v6 Arduino Nano	Pin Raduino32 ESP32 Dev Mod	
ENC_A	A0	34	
ENC_B	A1	35	
FBUTTON	A2	36	
PTT	A3	39	
SDA	A4	18	
SDC (SCL)	A5	19	
ANALOG_KEYER	A6	4	testing
ANALOG_SPARE	A7	33	testing
TX_RX	7	23	
CW_TONE	6	16	testing
TX_LPF_A	5	25	
TX_LPF_B	4	26	
TX_LPF_C	3	27	
CW_KEY	2	2	testing
TFT_DC	9	21	
TFT_CS	10	15	
TOUCH_CS	8		
TFT_SDI (MISO)	11	12	
TFT_SDO (MOSI)	12	13	
TFT_SCK	13	14	
RXD2 (UART2)	-	22	
TXD2 (UART2)	-	32	

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 mezcolanza 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>

DESCRIPCIÓN

La pantalla principal tiene este aspecto



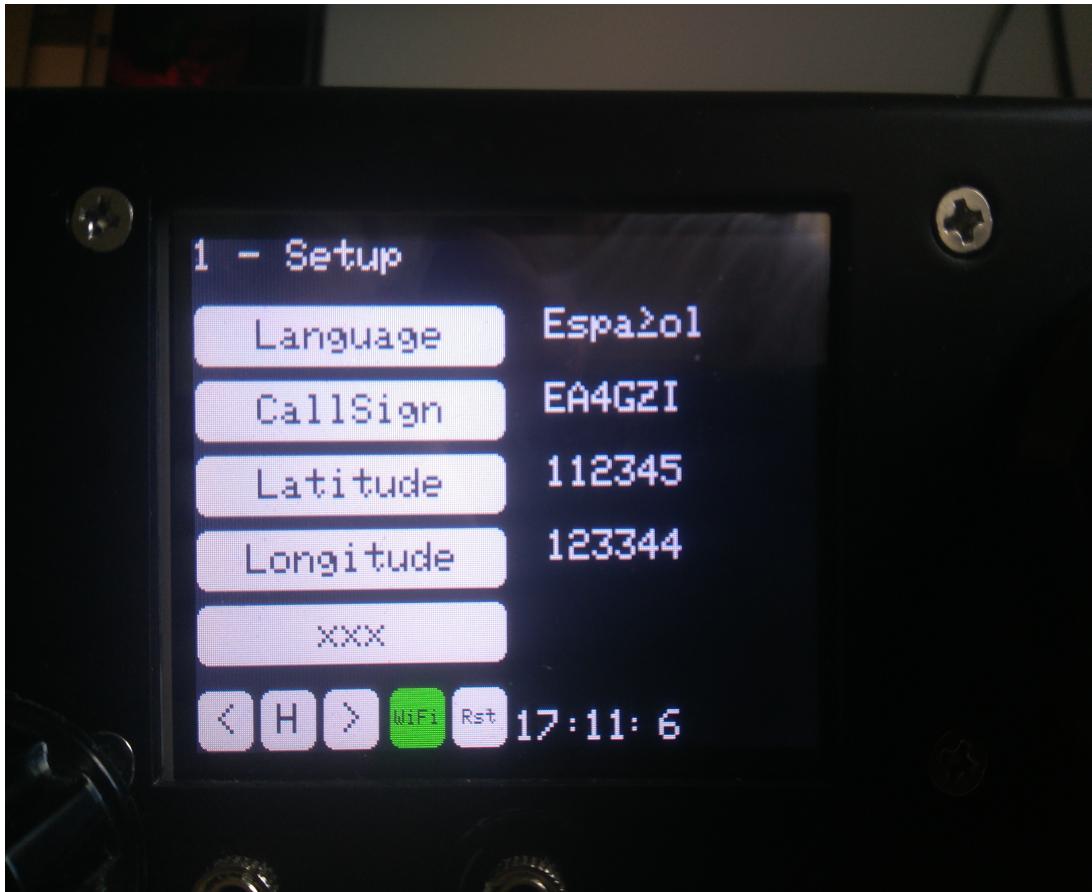
Botones



Configuración

Pulsando el botón  se avanza por las páginas de configuración

Página “1 – Setup”



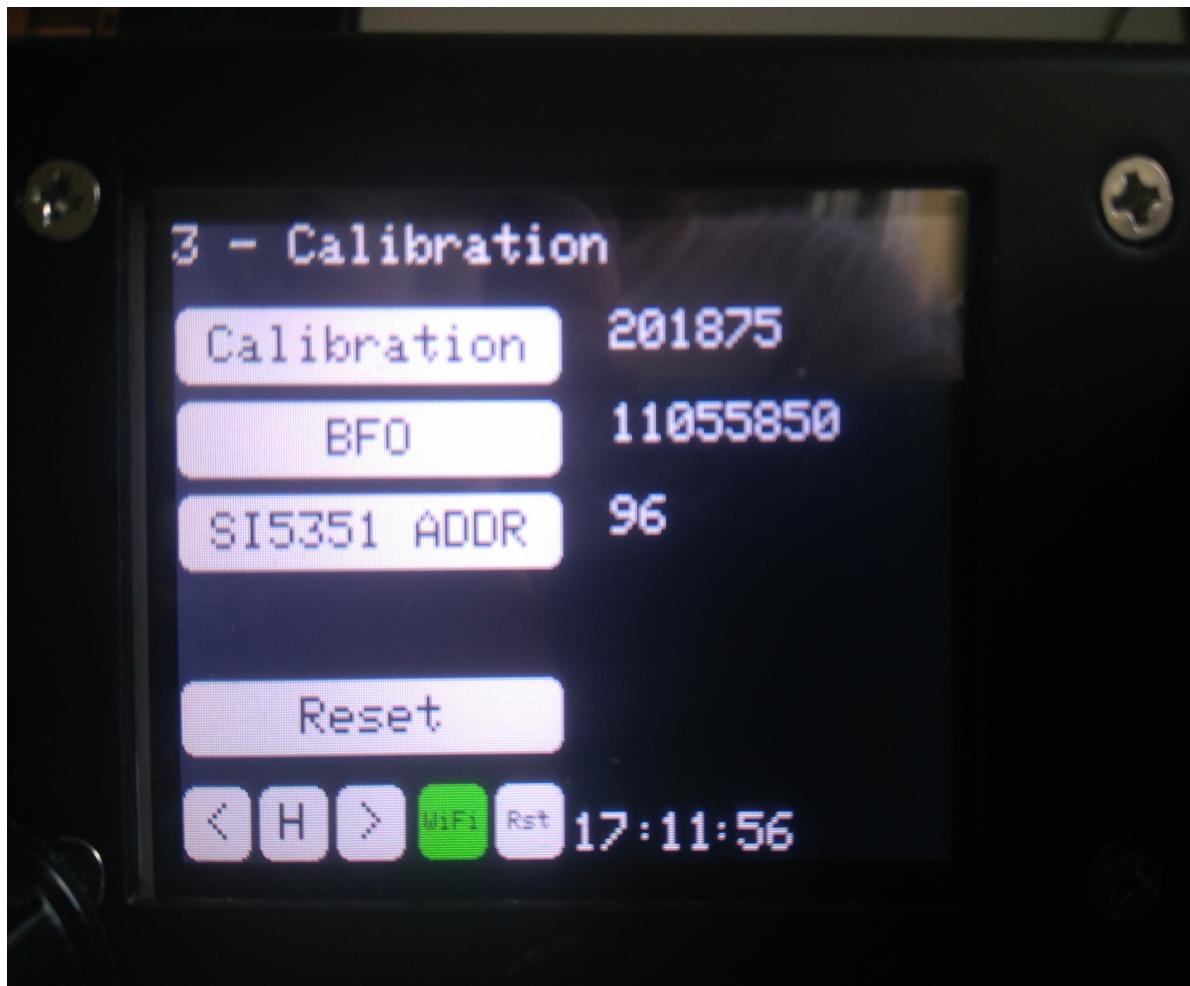
En esta página se pueden editar Idioma, Distintivo, Latitud y Longitud pulsando en el botón correspondiente.



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”.

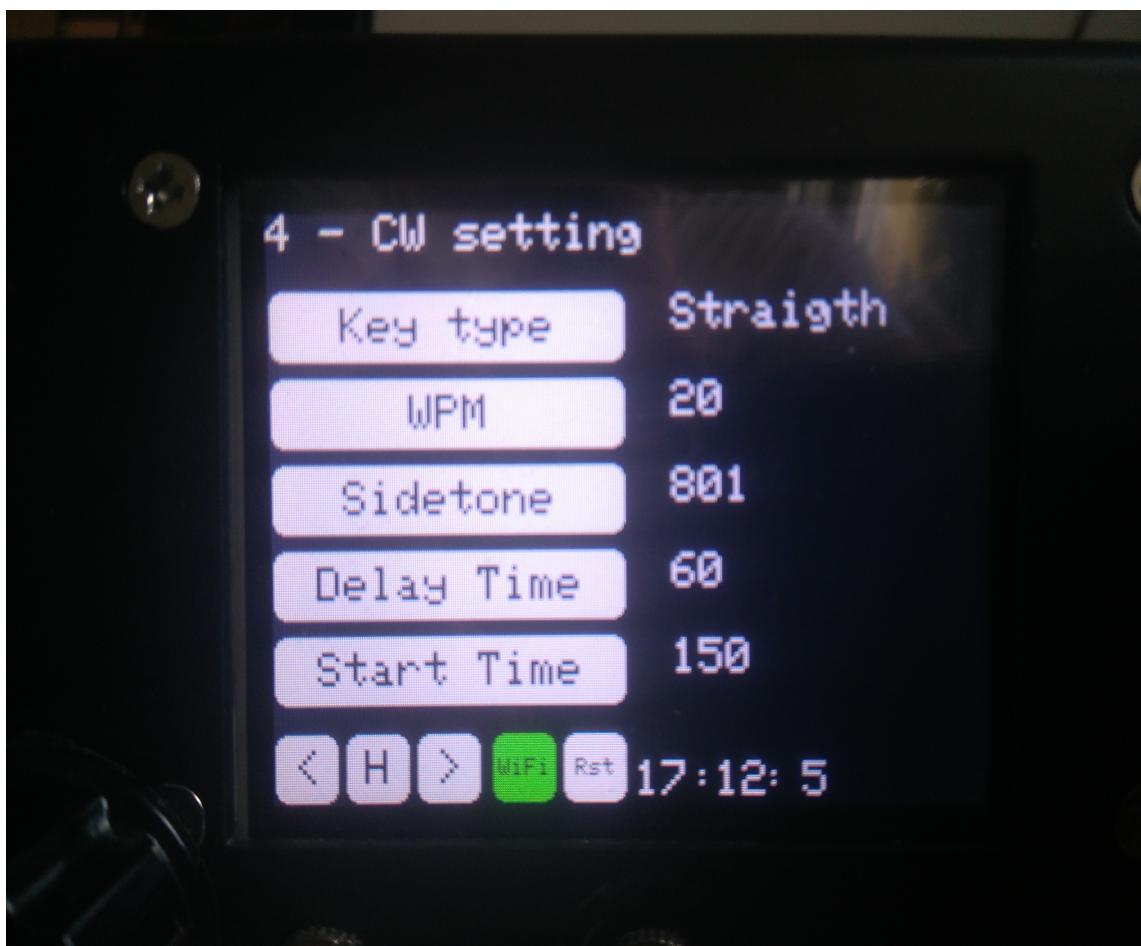
Este procedimiento es igual en todos aquellos casos en que hay que seleccionar un valor.

También puede encontrarse mucha ayuda en:

Grupo bitx20: groups.io/g/bitx20

Página <https://ubitx.net/>

Página “4 – CW setting”



Acceso al equipo mediante navegador

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

El panel principal

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	EA4GZI			
Calibration/usbCarrier	201875		11055850	
Latitud/ Longitud	112345.000000		123344.000000	
Idioma	Español ▾			
CW Speed	60			
cwDelayTime	60			
cwKeyType	iambica ▾			
SI5351BX_ADDR	96			
Reset periodico (horas)	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
Banda	desde	hasta khz			
160 m	1810	2000			
80 m	3500	3800			
60 m	5351	5367			
40 m	7000	7300			
30 m	10100	10150			
20 m	14000	14350			
17 m	18068	18168			
15 m	21000	21450			
12 m	24890	24999			
10 m	28000	29700			
<input type="button" value="Guardar"/>					

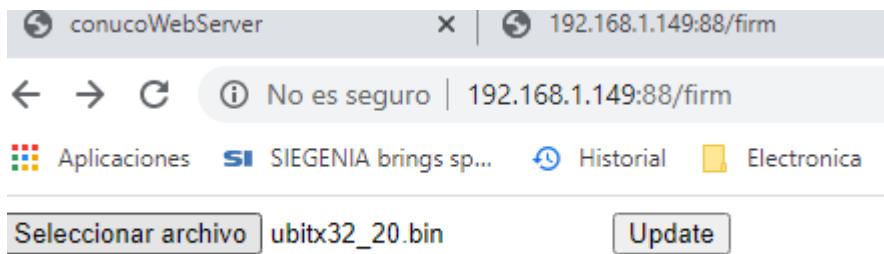
Pantalla “Red”

Acceso a los parámetros completos de la conexión WiFi

Panel		Configuración		Sistema	
Dispositivo		Bandas		Red	Serv. Red
Auto WiFi		<input checked="" type="checkbox"/>			
Modo		AP+STA ▾			
STA mode SSID		conuco4		Scan	
STA mode Password		██████████			
AP mode SSID		UBITX			
AP mode Password		12341234			
Canal		3 ▾			
MAC					
IP estática		<input checked="" type="checkbox"/>			
Dir. IP		192.168.1.149			
IP		192	168	1	149
Mask		255	255	255	0
Gateway		192	168	1	1
DNS		8	8	8	8
IP pública		185.178.80.2			
<input type="button" value="Guardar"/>					

Pantalla “Actualizar”

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



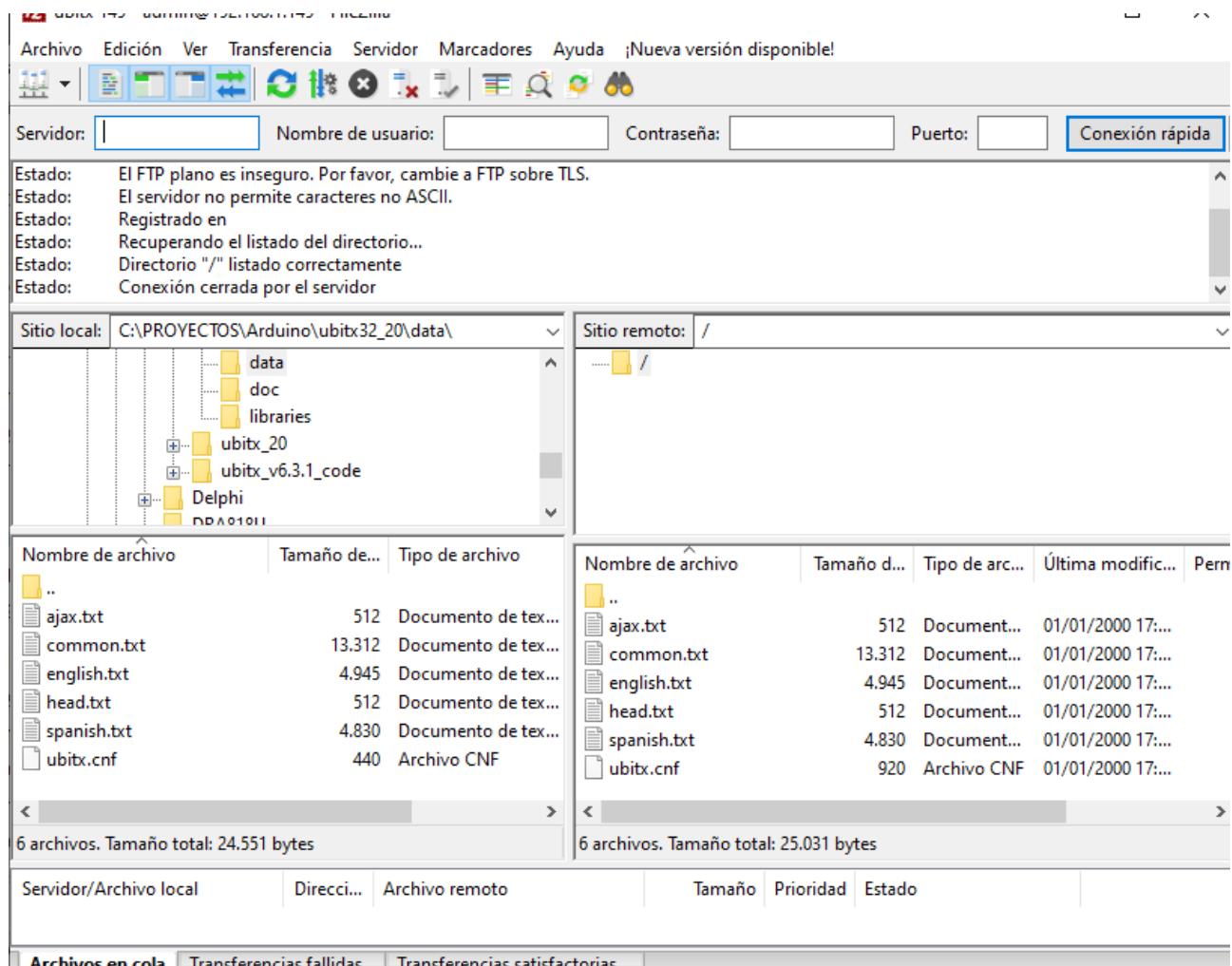
Si todo funciona como debe, se actualizará el firmware con el nuevo fichero y el equipo se reiniciará.

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