FT-857D CAT
by a web solution
Technical aspects

https://f6czv.fr

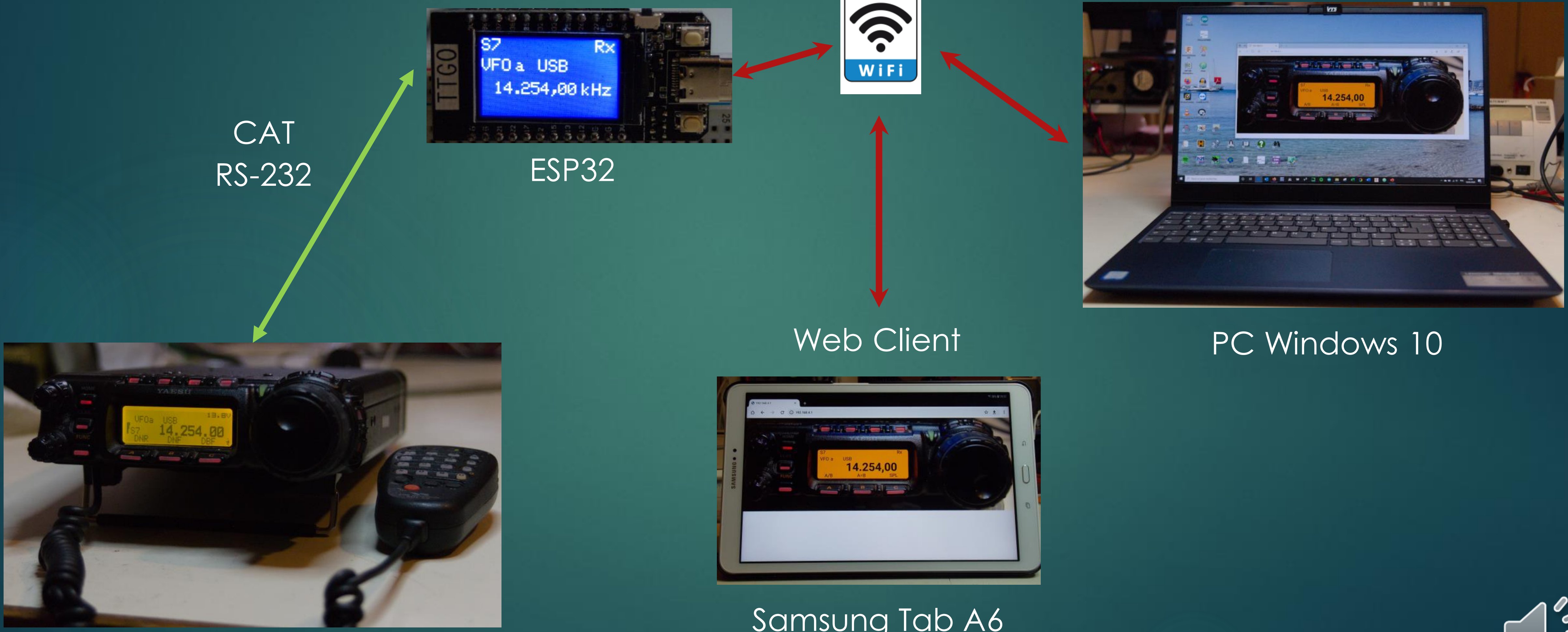
Summary

- Projects goals
- Test configuration
- Global architecture
- ESP32 Loop
- Web server
- Web browser display
- Web client application sequencing
- Software bench
- IDE : Source directories structure
- IDE : Libraries
- References

Project goals

- To implement a web App for an FT-857 CAT able to:
  - Have a realistic view of the radio front panel
  - Display the main radio parameters (freq, mode, S-Meter…)
  - Input frequency and mode
  - Simulate the VFO dial
  - Toggle the VFOs, the SPLIT and clarifier functions
- Implement the web server on an ESP32 connected to WiFi
- Take into account browsers on PCs, tabs or if possible on smartphone

Test configuration



Web Server

ESP32

CAT
RS-232

Web Client

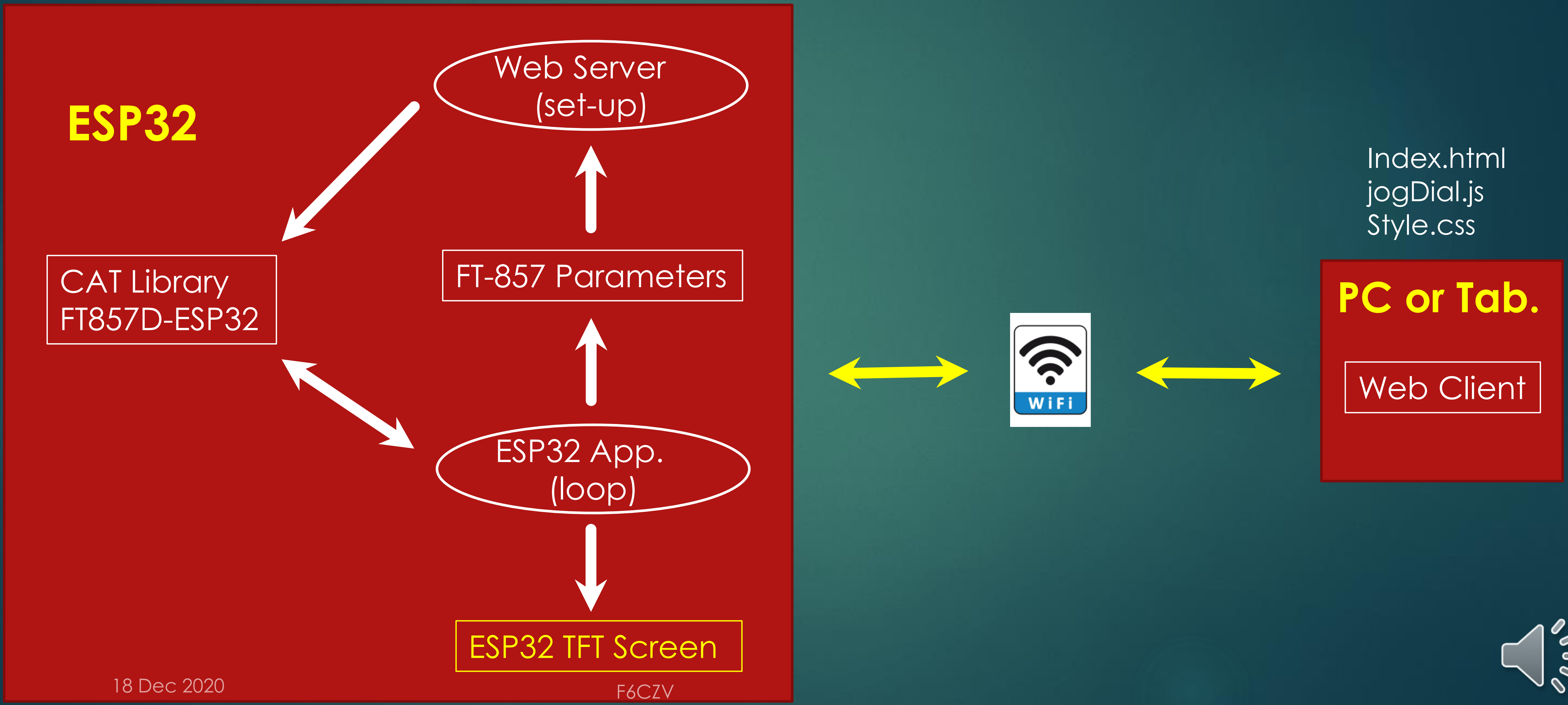
PC Windows 10

Web Client

Samsung Tab A6

Global Architecture

FT857-Web-browser-CAT-ESP32.ino



ESP32

Web Server (set-up)

CAT Library FT857D-ESP32

FT-857 Parameters

ESP32 App. (loop)

ESP32 TFT Screen

Index.html
jogDial.js
Style.css

PC or Tab.

Web Client

ESP32 Loop

On timer (500ms):

- Parameter requests to the FT-857 through CAT commands
- Processing of CAT responses and update of global variables
- Update of the ESP32 display

Web server

At client init. :

- Sending from the SPIFFS memory zone of the contents of index.html, jogdial.js, style.css and of images

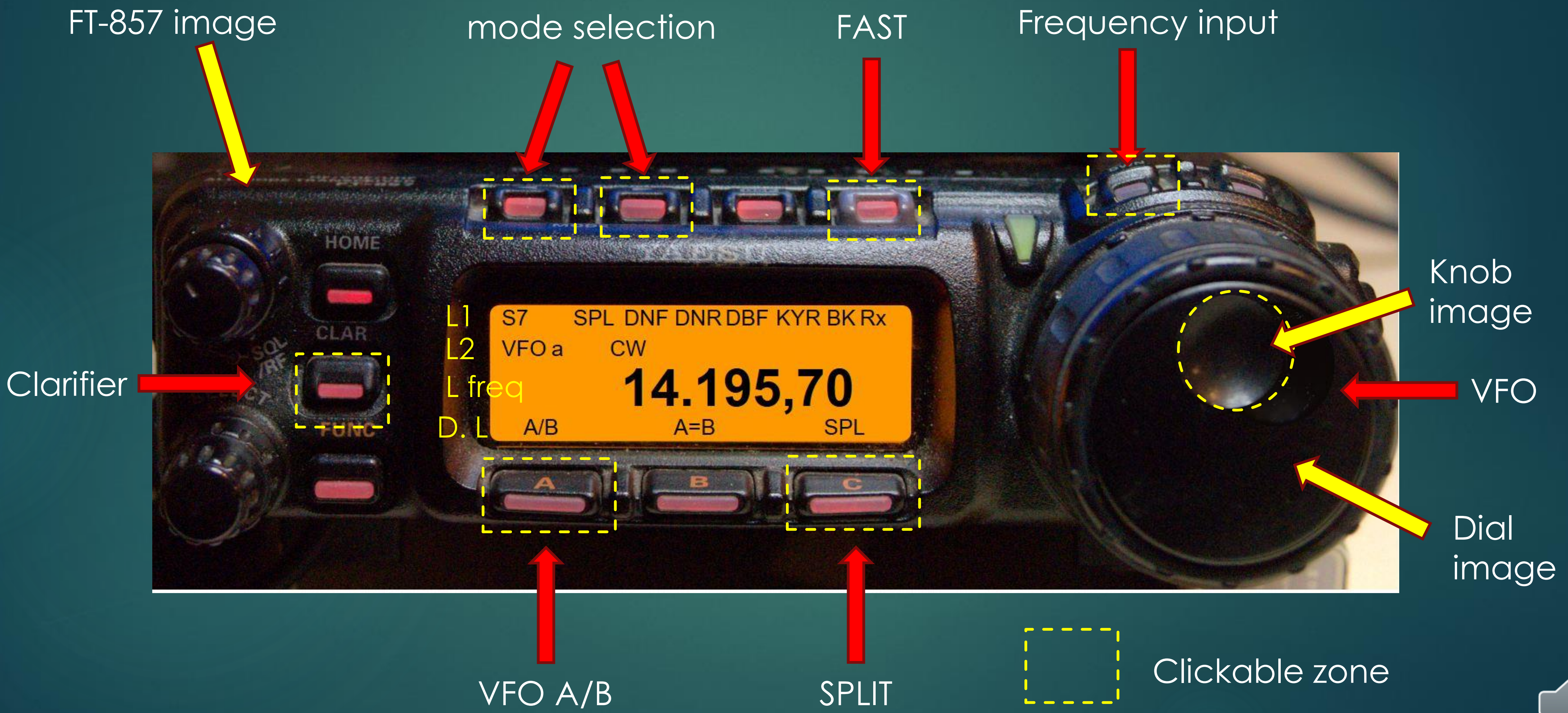
On FT-857 parameter request from the client:

- Sending of the value of the corresponding global variable

On update request (mode, frequency) or action request (VFO A/B, SPLIT, …) :

- Direct call to the FT-857 CAT library

Web browser display



Web client application sequencing

On click :

- Mode or frequency :
  - data is input from a form and the client sends a GET request to the web server
- Clarifier, Split, VFO A/B :
  - The client sends a specific GET request to the web server
- FAST : local change of the frequency delta calculation step
  - Step = 3 ou 0,8 (FAST or not)
  - Display of the FAST animated GIF

On timer (600ms):

- If the VFO dial delta is not zero, the client reckons the frequency delta and sends it with a GET request to the web server
- The client sends a GET request to the web server for each of the FT-857 parameters
- On data receipt updates the display

Web client application sequencing

Managed by the jogDial.js module

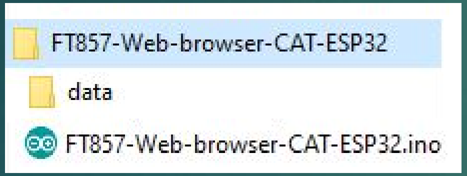
On VFO knob drag (rotation) :

- The jogDial function counts the number of degrees since the init
- Storage of the current-rotation number

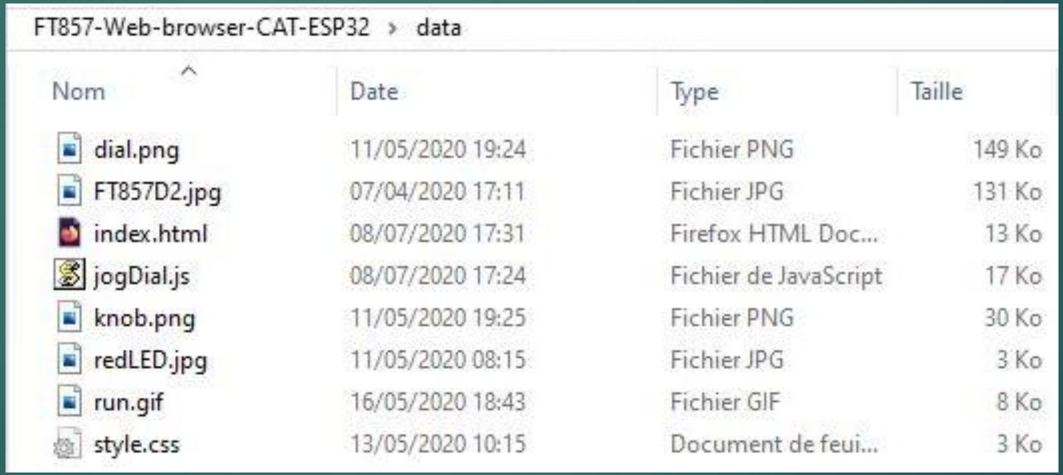
Software bench

- **IDE Arduino for the ESP32 application**
  - **Use of « ESP32 Sketch Data upload » for the SPIFFS memory upload (HTML, CSS, jogDial.js and web client images)**
- **ATOM editor for the web files (CSS, HTML, Javascript)**
- **Firefox, Edge and Samsung browser for the tests**

IDE : Source directories structure



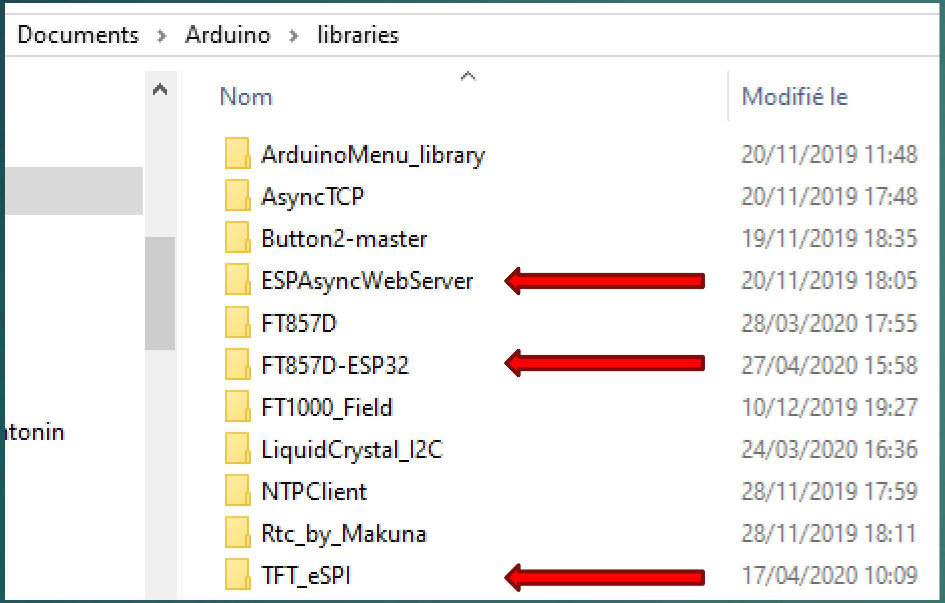
- FT857-Web-browser-CAT-ESP32
  - data
  - FT857-Web-browser-CAT-ESP32.ino



FT857-Web-browser-CAT-ESP32 › data

Nom	Date	Type	Taille
dial.png	11/05/2020 19:24	Fichier PNG	149 Ko
FT857D2.jpg	07/04/2020 17:11	Fichier JPG	131 Ko
index.html	08/07/2020 17:31	Firefox HTML Doc...	13 Ko
jogDial.js	08/07/2020 17:24	Fichier de JavaScript	17 Ko
knob.png	11/05/2020 19:25	Fichier PNG	30 Ko
redLED.jpg	11/05/2020 08:15	Fichier JPG	3 Ko
run.gif	16/05/2020 18:43	Fichier GIF	8 Ko
style.css	13/05/2020 10:15	Document de feui...	3 Ko

IDE : Libraries



Documents › Arduino › libraries

tonin

Nom	Modifié le
ArduinoMenu_library	20/11/2019 11:48
AsyncTCP	20/11/2019 17:48
Button2-master	19/11/2019 18:35
ESPAsyncWebServer	20/11/2019 18:05
FT857D	28/03/2020 17:55
FT857D-ESP32	27/04/2020 15:58
FT1000_Field	10/12/2019 19:27
LiquidCrystal_I2C	24/03/2020 16:36
NTPClient	28/11/2019 17:59
Rtc_by_Makuna	28/11/2019 18:11
TFT_eSPI	17/04/2020 10:09

Asynchronous web Server library

FT857D-ESP32 CAT library

TFT screen on SPI bus library

References

- **My site:** https://f6czv.fr/
- **Code on Github :** https://github.com/Phil-f6czv/FT857-Web-browser-CAT-ESP32
- **Projects and tutorials of all kind for ESP32:** https://randomnerdtutorials.com/
- **Reference site for web developments (HTML, CSS, Javascript …):** https://www.w3schools.com/default.asp
- **ATOM editor:** https://atom.io/
- **Asynchronous Web Server on ESP32:** https://github.com/me-no-dev/ESPAsyncWebServer
- **TFT e_SPI:** https://github.com/Bodmer/TFT_eSPI
- **Jogdial:** https://github.com/ohsiwon/JogDial.js