



0



ESP32 VFO 10 KHz-160 MHz with Retro Analog Scale

A variable frequency oscillator (VFO) is a necessary component in any tunable radio transmitter or receiver.

[Follow project](#)[Like project](#)[Join this project](#)[Public Chat](#)**6.5k**

views

0

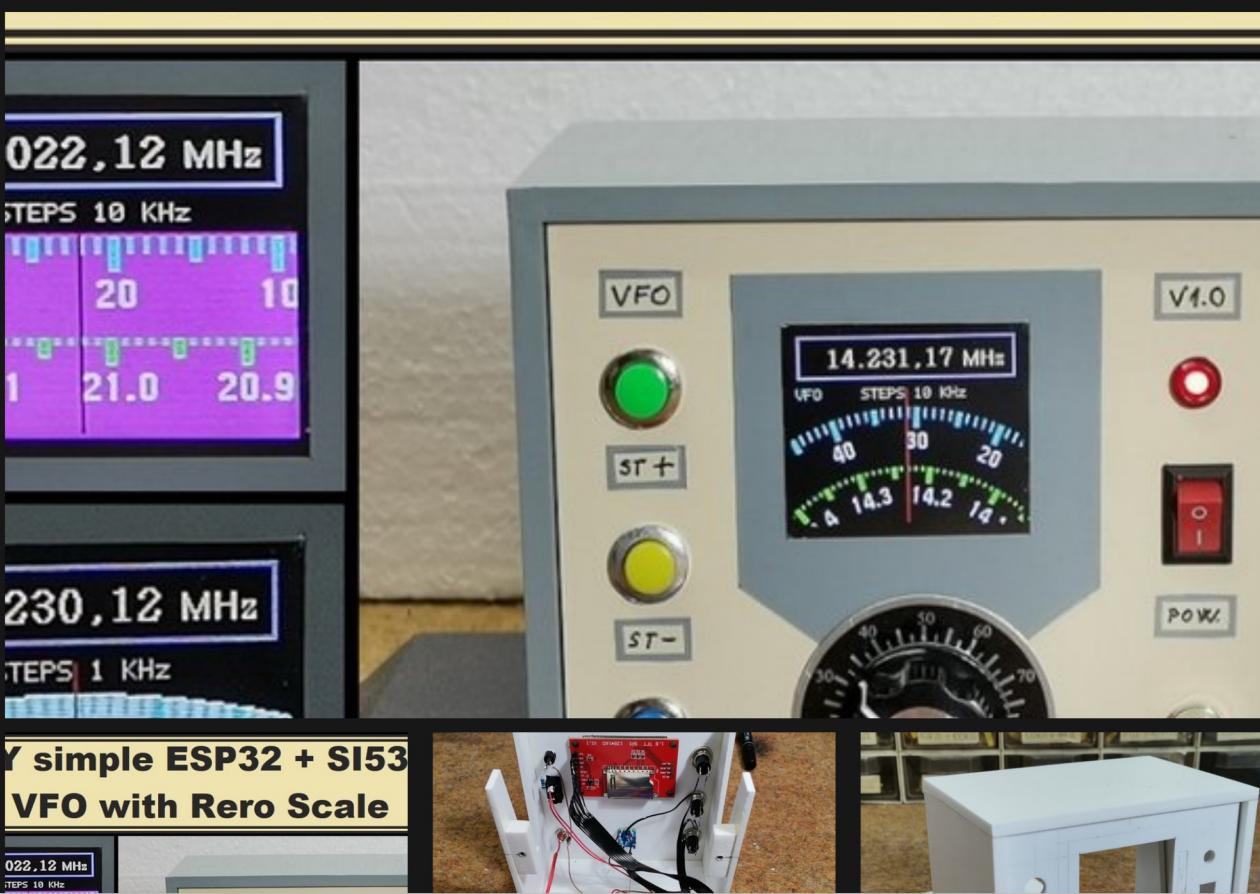
comments

19

followers

12

likes



Description Details Files 2 Components 12 Logs 0 Instructions 0 Discussion 0



[View Gallery](#)

Public Chat 

 6.5k  0  19  12

TEAM (1)

 mircemk
Join this project's team

 COMPLETED PROJECT

 HARDWARE

VARIABLE FREQUENCY OSCILLATOR ARDUINO VFO ARDUINO SW RADIO

ESP32 VFO PROJECT LOCAL OSCILLATOR FOR RADIO SI5351 OSCILLATOR ESP32

ARDUINO TFT VFO ST7735 TFT DISPLAY ARDUINO SIMPLE ESP32 PROJECT

128X160 1.8' TFT DISPLAY ST7735 TFT ON ESP32 ROTARY ENCODER VFO SIMPLE VFO CIRCUIT

RELATED LISTS

 Devrelerim

Description Details Files 2 Components 12 Logs 0 Instructions 0 Discussion 0

THIS PROJECT IS SUBMITTED FOR

- Hackaday.com Tip Line

This project was created on 11/19/2022 and last updated 3 years ago.

DESCRIPTION

This time I will show you how to make a VFO that covers the range from 100 KHz to 160 MHz and uses a minimum number of components.

DETAILS

A variable frequency oscillator (VFO) is an oscillator whose frequency can be tuned (i.e., varied) over some range. It is a necessary component in any tunable radio transmitter or receiver, and controls the frequency to which the radio is tuned.



The selected frequency is displayed on a beautifully designed retro scale in the style of ham radios from the previous century. About 2 years ago, In one of my previous videos:



I made a small shortwave radio receiver with this VFO, but the current version has additional functions for changing the tuning step, as well as storing frequencies. These options drastically simplify the operation of the device.

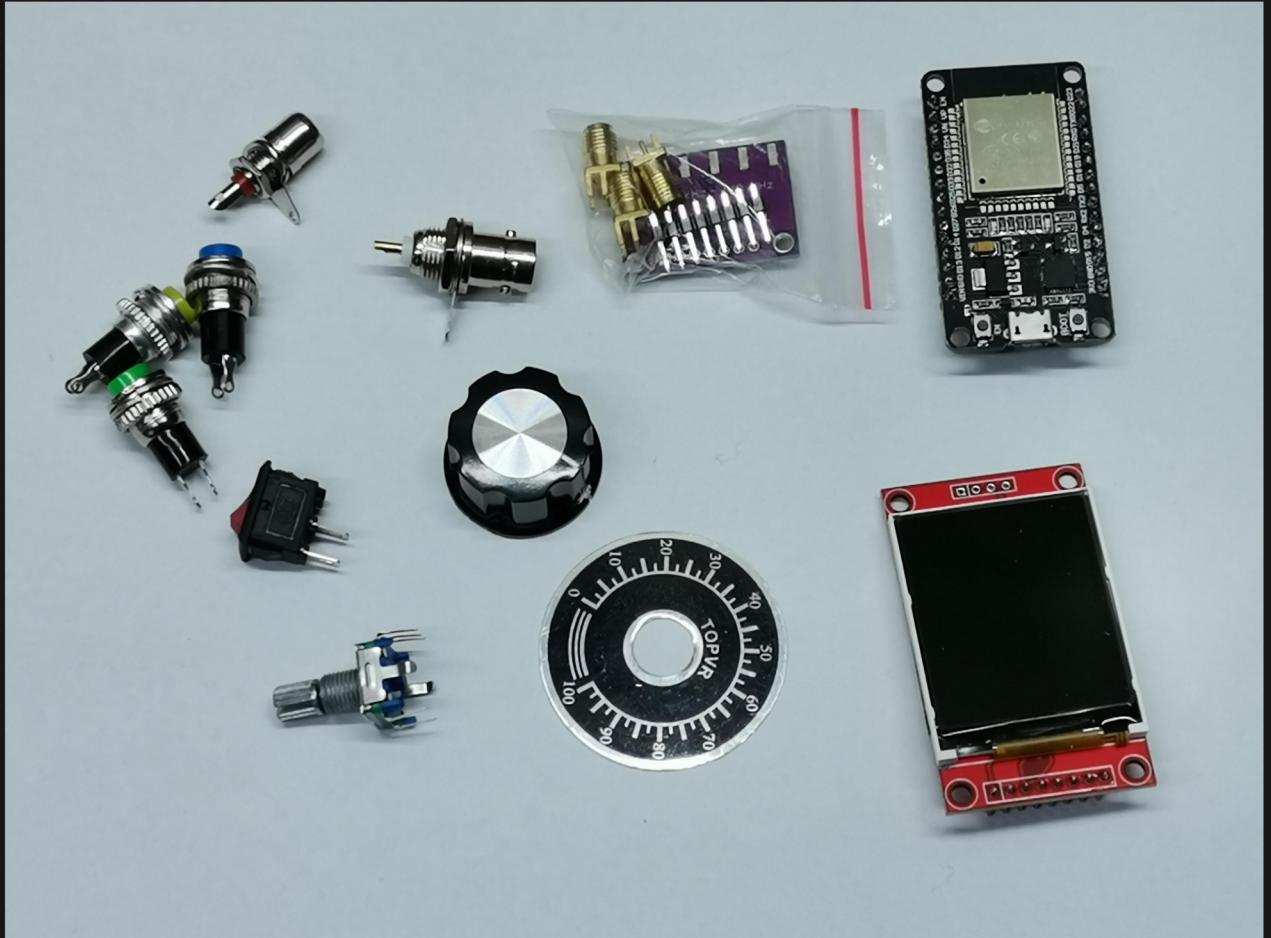
Otherwise, the device was originally designed by Japanese radio amateur JF3HZB, and the version that I present to you is modified by DJ7OO. I got the code from my friend Branko Zupan (<http://users.triera.net/zupanbra/>), on whose page you can find many projects and useful tips, especially in the field of Radio technology. For this project he has designed a custom PCB as well as an add-on Direct Conversion Receiver with the famous NE612 mixer IC. Also a different version of the PCB for this project is on DJ7OO's site.

So, if you want to make a PCB for this project, or for any other electronic project, PCBway is a great choice for you. PCBway is one of the most experienced PCB manufacturing company in China in field of PCB prototype and fabrication. They have a large online community where you can find a Open Source projects, and you can also share your project there. From my personal experience I can tell you that on this community you can find many useful projects with already designed PCBs, from where you can place an order directly.

The PCBWay logo, featuring the word "PCBWay" in a bold, green, sans-serif font. A thick orange swoosh underline starts under the letter "P" and ends under the letter "y".

As I mentioned before, the device is extremely simple to make and contains only a few components:

- ESP32 Dev. kit board
- 1.8" TFT Display with 128x160 dots resolution (ST7735 chip)
- SI5351 programmable clock generator board
- Rotary encoder with push button
- three buttons
- and Output connectors



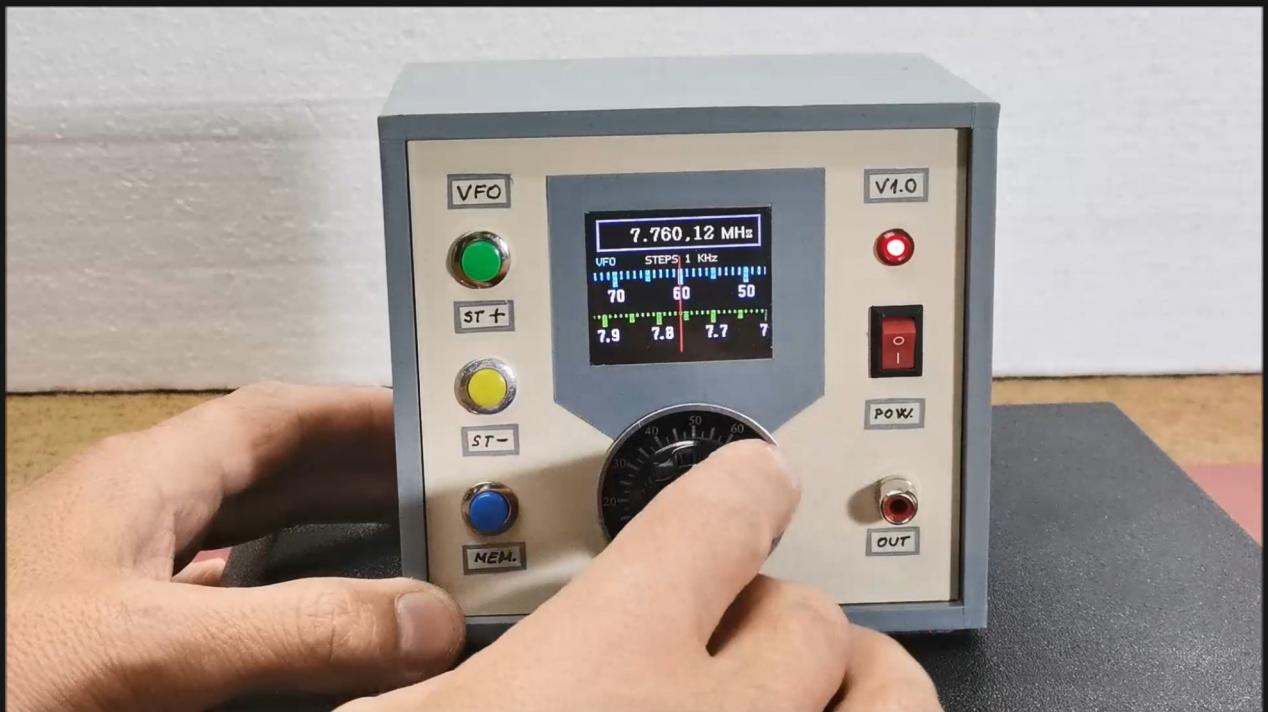
Now let's see how the device works in reality. Immediately after switching on, the start frequency that is previously defined in the code appears on the screen. Turning the rotary encoder changes the frequency. The tuning step, and thus the scale rotation speed, is controlled by the two buttons, and ranges from 10 Hertz to 1 Megahertz. With the the rotary encoder button we memorize the current frequency, and after the next power-on of the device, it will be the initial frequency. The oscillator also has the ability to memorize previously defined frequencies in the code, that we can call up with the MEM button.





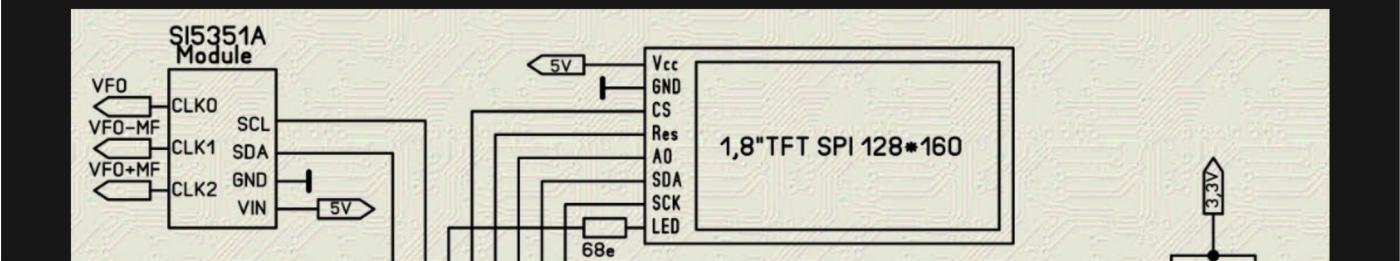
For the first few megahertz the signal has an approximately rectangular shape. Then by increasing the frequency, the shape of the signal becomes sinusoidal. Even at the highest frequencies, the shape is almost triangular and the amplitude of the signal decreases significantly.

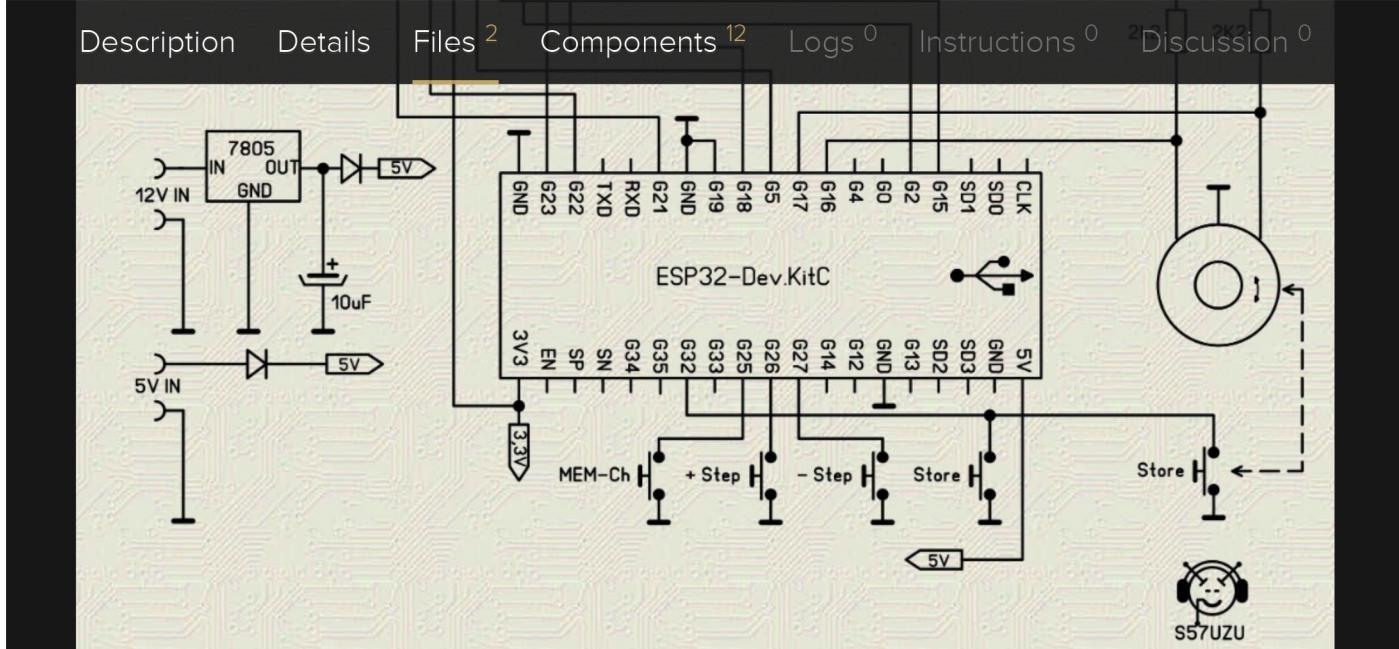
It is interesting to note that as part of the code, in the "dial_prm.h" header file, many parameters can be changed, starting from the colors, through the dial radius, up to the definition of Length and Width of Dial pointer, space between Main and Sub Dial etc...



I plan to use this VFO in the construction of very simple AM and SSB radio receivers (mostly Direct Conversion receivers) that I will present to you in some of the following videos.

And finally, the whole device is embedded in a suitable box made of PVC material with a thickness of 3mm and 5mm, and coated with colored self-adhesive wallpaper.





FILES

VFO_CTRL_V5.zip



Arduino Code

Schematic.jpg



Circuit diagram

COMPONENTS

1 × ESP32 Development Board

1 × ESP32 Development Board

1 × Si5351a clock generator module

1 × Si5351a clock generator module

1 × TFT Display 128x160 (ST7735 chip) 1.8"

[Description](#) [Details](#) [Files 2](#) [Components 12](#) [Logs 0](#) [Instructions 0](#) [Discussion 0](#)**ENJOY THIS PROJECT?**

Share

**DISCUSSIONS***Log In or become a member to leave your comment*[Log In/Sign up to comment](#)**Does this project spark your interest?**

Become a member to follow this project and never miss any updates

 Your email address Password Enter captcha[Become a member](#)

Going up?

[About Us](#)[Contact Hackaday.io](#)[Give Feedback](#)[Terms of Use](#)[Privacy Policy](#)[Hackaday API](#)

© 2025 Hackaday