Building the ZL2PD SugarCube VFO.

Background.
Maybe it helps if you know how this came about.
Andrew wanted to build something very small – which is absolutely what I need for my tiny radios for tramping.
His first offering needed all parameters programmed into it – and it used BASCOM as the programming environment. While a free version was available it would not allow compiling a programme over 4k.
At my suggestion Andrew was soon able to produce a version which held the variables such as startup and i.f.frequencies on the EEPROM. He was able to design a spreadsheet which produced the hex numbers to enter into a hex file – and then used the same programmer cable to enter both the flash memory and the EEPROM hex data.

Building the Sugar Cube
If you google ZL2PD you will find his amazing array of vfos. Many of them, including the sugarcube, are based on the Si5351 PLL chip. It is controlled by the Attiny85 - which holds the programme you put on it. Because the Attiny85 only has 8k of flash memory, the programming needed to be very tight!

A version of a sugar cube

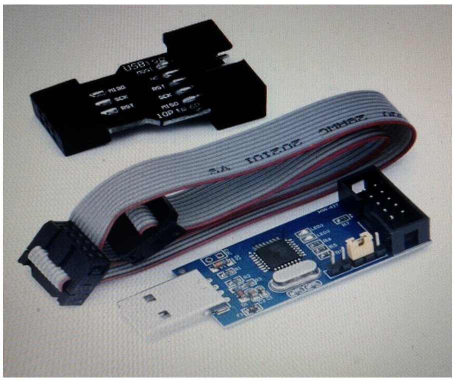
So what do you need?
Well all the instructions and circuit diagrams are on Andrew's website. But here is an overview.

You need:
1. A computer with USB port
I have used WinXP, Win8, and Win10. I have now managed to get a Linux running some of the software - but so far I have not managed to run Extremeburner. Read comments back in article 2 if you have trouble getting your computer to talk to the USBASP programmer board.
I can't tell you what Win11 might do.

2. A programme downloaded to enable your computer to write the files to the ATtiny85. I find eXtremeburner to be easiest. I have also used Khazama and AVRDUDESS.
See Andrew's site.

3. You need cables to connect your USB port to the ic. This is a USBASP programmer and a few other little boards. Again see ZL2PD. These bits are cheap from the Asian sellers and Andrew has instructions to make a board to hold the Attiny85 if you want, though at a dollar of two you might as well buy one.

4. You also need to download the relevant files from Andrew's website.
Read everything carefully even if you don't understand it at first.



USBASP cables. Get the set of three

5. Finally you have to assemble the circuit. You will need to contact Andrew if you need a genuine SugarCube board. But if you feel the Si5351 soldering is beyond you - you can buy a generic (usually purple) board from Asia or elsewhere for less that $5. That can be used with the Attiny85 but you will need to assemble them in some way. I will try to find some pics of ones I have assembled. Or there may be my circuit in the downloads.

Of course you will need ATtiny85s plus Si5351s if you are using them and probably some 3.3v 100ma 3-terminal regulators. The 128x32 oled is prescribed but a 128x64 also gives a larger readable display (BUT -check what I have written in another article about Asian OLED displays).You will also need some tactile switches, a rotary encoder, 8-pin ic socket, circuit board pins and bits from the junk box.

What now?
This is where the fun or hair pulling really begins.
Perhaps it was just my old computer, but for at least a week I could not get my computer and the USBASP to talk to one another. Eventually I downloaded a little programme for my USB port called ZADIG and then it started to know the USBASP was plugged in!!
Even WITH Zadig it does not always work at first so see what I have written in article 2 on this site about checking the working of the USBASP plugged into the USB port.
Next you have to get your head around putting the .hex file on the flash memory in the chip - though this is relatively easy.

Then you have to learn how to write the configuration file for the EEPROM using the spreadsheet programme that Andrew supplies. Read carefully what he says - and write down in full the hex numbers that are generated to be put into the EEPROM. Maybe at first write a simple frequency generator without bfo frequencies etc.
Finally we need to write the fuses. These are really a set of parameters that tell the attiny85 how to operate.
BUT BE WARNED!!
Andrew decided he wanted as much use from the Attiny85 as possible; so in setting the fuses he used pin1 in such a way that, once set with his fuse settings, it does not allow the chip to be reprogrammed easily (you will need a high voltage programmer).
For this reason - as I really didn't need that feature (a bar graph) I changed the fuse setting to allow ready reprogramming of the chip - at least to begin with. Andrew talks about this on his website.

If you get to this point and are able to get the sugarcube vfos running - that's great. And it may be all you need.

But there's more !!
Having reached this stage, I was able to get the hex file for a different programme written for an Atmega328 chip (ie the chip on an Arduino). It too was written to run the Si5351 and I had the circuit. Knowing very little I unplugged the 6pin header from my programmer cable and plugged it onto an Arduino Nano. Not having the 328 on the list of chips - I chose another as close as possible (the 324) and asked to programme it. The computer replied " Wrong chip! Proceed anyway?" I clicked "yes" - and it DID! And it runs perfectly.
So skills I learned in making the SugarCube have turned out to be transferable.
There are many more things that could be said, but you can contact Andrew if you have problems with the sugarcube. Or feel free to contact me if needed.

Eric Sears ZL2BMI. sears@xtra.co.nz