This topic ▾   SEARCH

nfg.forums

simplemachines® forum

June 09, 2024, 02:45:22 PM   News: **Forum Updated!**

Home   Search

DFO: dual frequency oscillator

Started by micro, February 01, 2015, 02:50:37 AM

Previous Topic - Next Topic

GO DOWN   Pages: **[1]** 2   PRINT

○ **micro**

☐☐☐☐
MassiveMember
Posts: 392
Logged

February 01, 2015, 02:50:37 AM
Last Edit: April 26, 2015, 11:51:54 PM by micro

Something not related to videogame controllers for a change...
😉

DFO: dual frequency oscillator
A programmable oscillator that let you switch between two different frequencies

Some months ago I got a X-RGB Mini device, also known as Framemeister. I quickly found out that my modded consoles don't work well with the Framemeister. Switching my PAL Megadrive 2 to 60 Hz or using NTSC games on my PAL PSX introduced massive stuttering, messed up black levels and other picture problems. It just wasn't fun to play the games this way.

The reason why these PAL consoles work fine on 50 Hz but don't on 60 Hz is that PAL and NTSC consoles use oscillators/crystals with different frequencies. Here's a little excerpt from the Megadrive 2 schematics:

OSC1 Value
NTSC: 53.693175MHz
PAL : 53.203424MHz
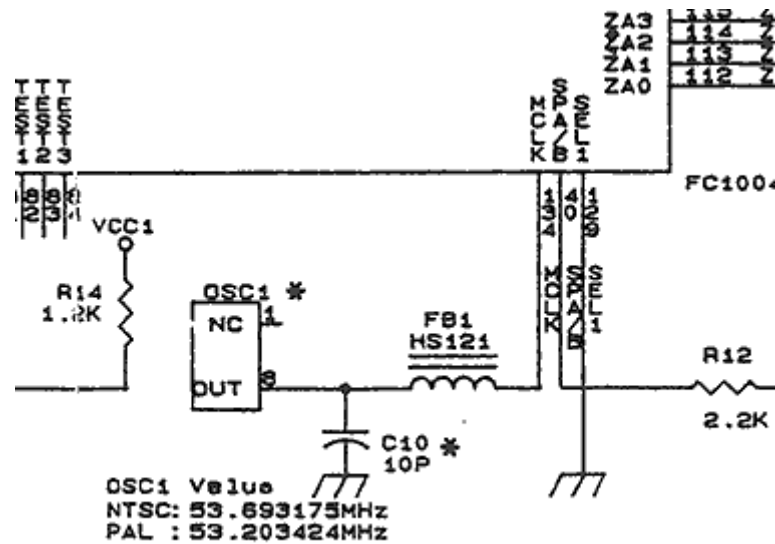
You can see the different oscillator frequencies in PAL and NTSC consoles.

NTSC: 53.693175 MHz (= 15 x NTSC subcarrier 3.579545 MHz)
PAL: 53.203424 MHz (= 12 x PAL subcarrier 4.3361875 MHz)

Because all other timings are derived from this master clock, it's easy to see what's the problem. A PAL Megadrive running on 60 Hz has a slightly different picture output rate than a genuine NTSC console. Back in the days this hardly was a problem because CRT TVs are quite tolerant regarding the output rate. But with modern devices like the X-RGB Mini Framemeister this can and will cause problems. This is a known issue, also mentioned in the Framemeister Wiki: http://junkerhq.net/xrgb/index.php/Sega_Genesis

So to restore compatibility with the Framemeister for both 50 and 60 Hz modes, we need an oscillator with two different frequencies to choose from. I've designed such an oscillator using the CDCE913 clock generation ICs by Texas Instruments (CDCE925 in the previous design). 😃
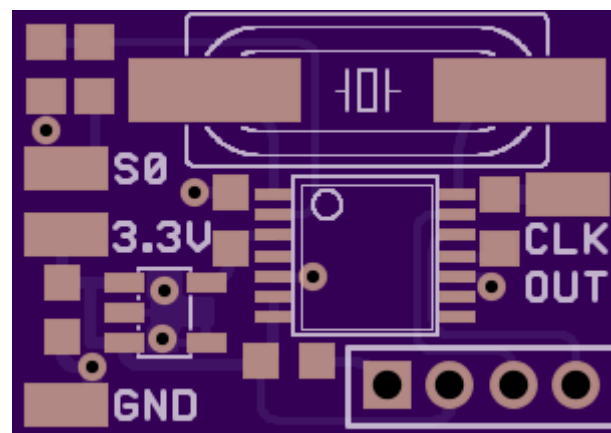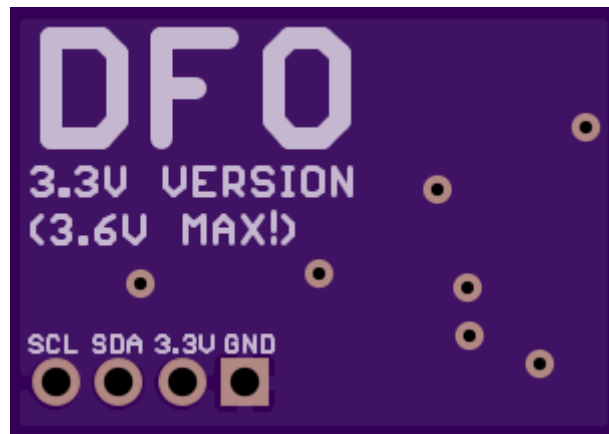
The CDCE913 on the dual frequency oscillator (DFO) has an built-in PLL (phase-locked loop). It can be programmed to output clock signals with (almost) any desired frequencies between 0.8 and 230 MHz. By tying the s0 input of the CDCE913 to either a low or a high logic level, it's possible to change the frequency of the clock output signal.

This pic shows the required hard- and software to successfully program the DFO:
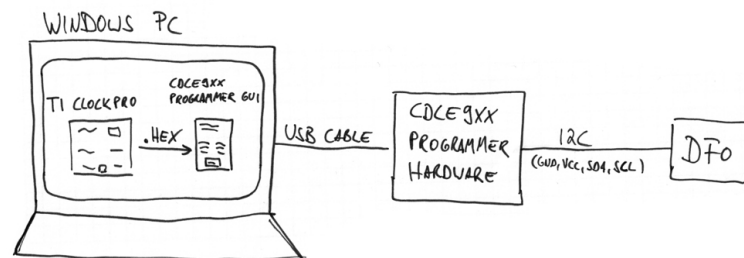
The first step is use the Clock Pro software by Texas Instruments. With Clock Pro you can create a setup for the CDCE913 IC by specifying the desired frequencies. Clock Pro will then automatically calculate the register values for your configuration. The register values can be saved into an Intel Hex file.

To program the CDCE913 on the DFO PCB you will need a programmer hardware called "CDCE9XX Programmer" (see below). It is connected to your PCB with a Mini USB cable. You also need to connect the programmer with the DFO through 4 wires. (I2C -> 4 wires: GND, VCC, SDA & SCL).

On the PC you open the "CDCE9XX Programmer GUI" that I've written (see below). Load the .hex file you've created with Clock Pro before and program the CDCE913 on the DFO by hitting the program button on the GUI.

Get your own DFO:

I've designed 3 different DFO PCB's. You can order the PCB's at oshpark: https://oshpark.com/profiles/micro

a) DFO 5V DIL14 version
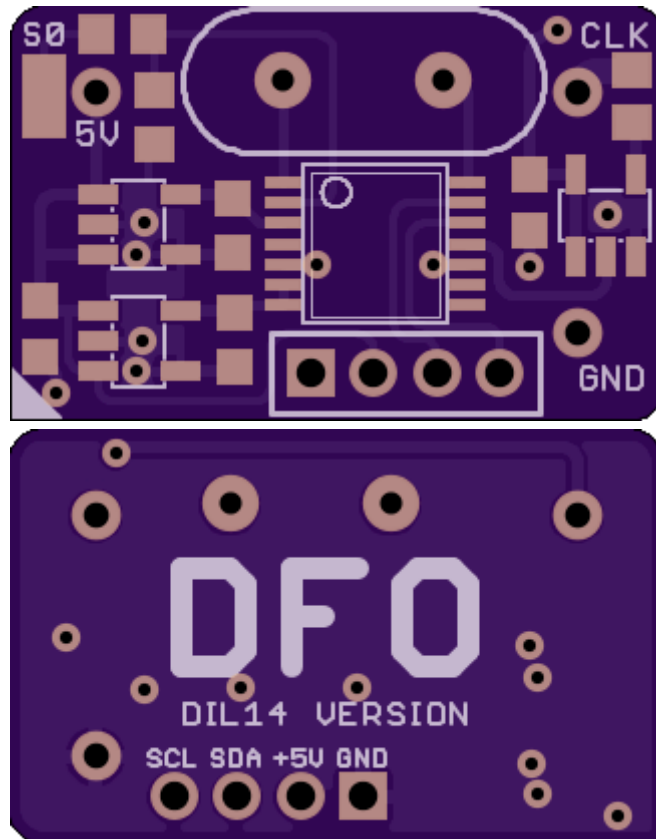
Schematic: http://www.mediafire.com/download/ p0lhr1nq79y6mq3/DFO_DIL14_5V.pdf

This version has the same dimensions and pinout as usual DIL14 metal can oscillators. Supply voltage is 5V (5.5V max). Therefore this DFO version can be used as a drop-in replacement for the metal can oscillator found in consoles such as the Megadrive or the Amiga.
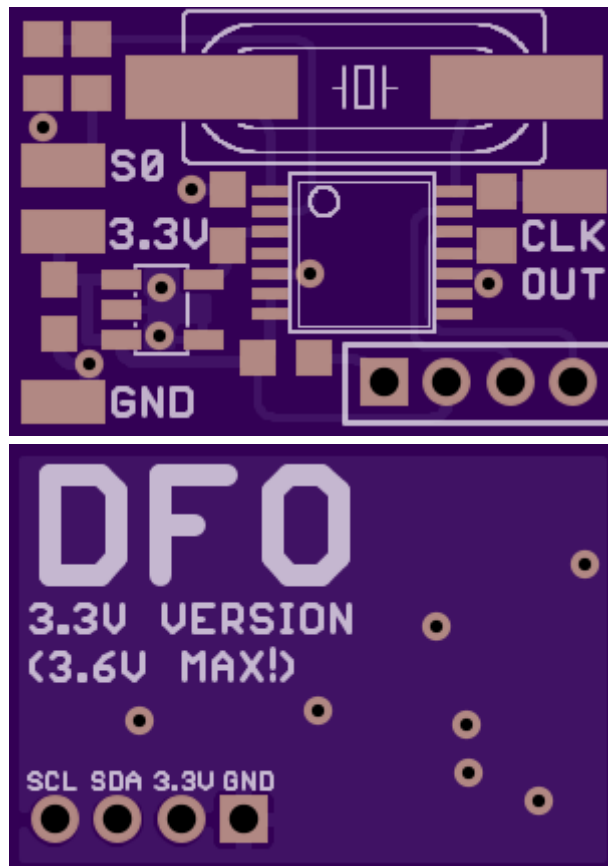
Parts list:

Quantity	Description	mouser.com part name
1x	CDCE913 clock generation IC	595-CDCE913PWR
1x	buffer/level shifter	595-SN74LV1T126DBVR
1x	3.3V LDO	595-TPS78233DDCR

| 1x	1.8V LDO	595-TPS78218DDCT
1x	27 MHz crystal	774-ATS270B
3x	1 uF capacitor	81-GRM188R61E105KA12
1x	2.2 uF capacitor	81-GRM188R61E225KA2D
1x	18 Ohm resistor	71-CRCW0603-18-E3
1x	10 kOhm resistor	71-CRCW0603-10K-E3
1x	22 kOhm resistor	71-CRCW0603-22K-E3
1x	thin pin header	AW 122/20
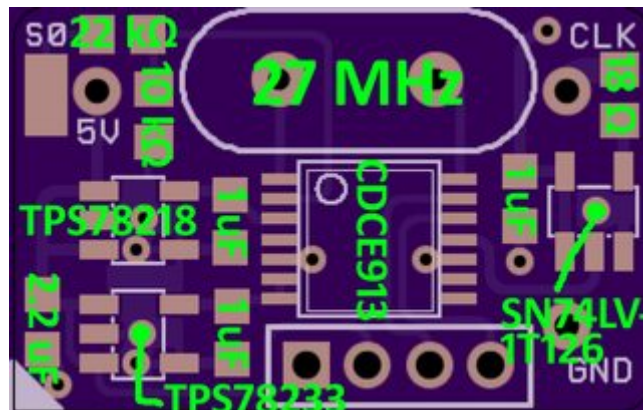
b) DFO 3.3V SMD version

Schematic: http://www.mediafire.com/download/uyiftm3m5u3moh0/DFO_SMD_3.3V.pdf

This version is more suited for consoles that have no metal can

oscillators inside. The input supply voltage is 3.3V (3.6 max). This version is suited for consoles such as the PSX or the MVS MV-1C.

Parts list:

Quantity	Description	mouser.com part name
1x	CDCE913 clock generation IC	595-CDCE913PWR
1x	1.8V LDO	595-TPS78218DDCT
1x	27 MHz SMD crystal	774-ATS270BSM-1
3x	1 uF capacitor	81-GRM188R61E105KA12
1x	18 Ohm resistor	71-CRCW0603-18-E3
2x	10 kOhm resistor	71-CRCW0603-10K-E3
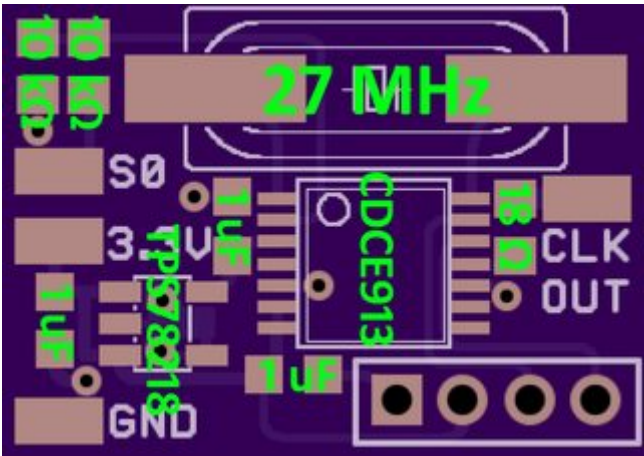
b) DFO 5V SMD version
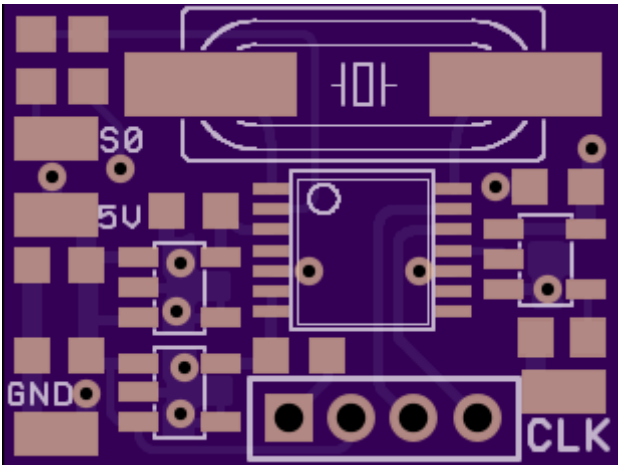
Schematic: http://www.mediafire.com/download/h5xblii4zit2sk2/DFO_SMD_5V.pdf

This version is has the same rectangular shape as the 3.3V SMD version, but it's specified for a supply voltage of 5V. It's suited for systems that got a 5V clock circuit (SNES maybe?). It's also possible to use this version in a 3.3V enviroment such as the PSX but for these consoles I'd rather recommend the 3.3V SMD version above because it needs less parts and is significantly cheaper. ^^

Parts list:

Quantity	Description	mouser.com part name
1x	CDCE913 clock generation IC	595-CDCE913PWR
1x	buffer/level shifter	595-SN74LV1T126DBVR
1x	3.3V LDO	595-TPS78233DDCR
1x	1.8V LDO	595-TPS78218DDCT
1x	27 MHz SMD crystal	774-ATS270BSM-1
5x	1 uF capacitor	81-GRM188R61E105KA12
1x	18 Ohm resistor	71-CRCW0603-18-E3
1x	10 kOhm resistor	71-CRCW0603-10K-E3
1x	22 kOhm resistor	71-CRCW0603-22K-E3
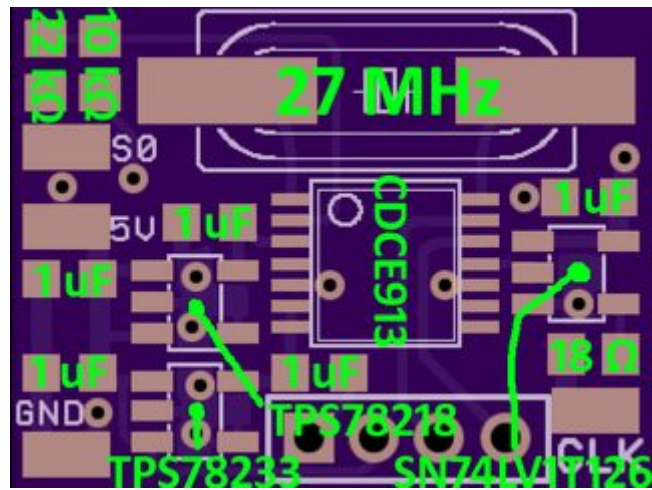
Programmer:
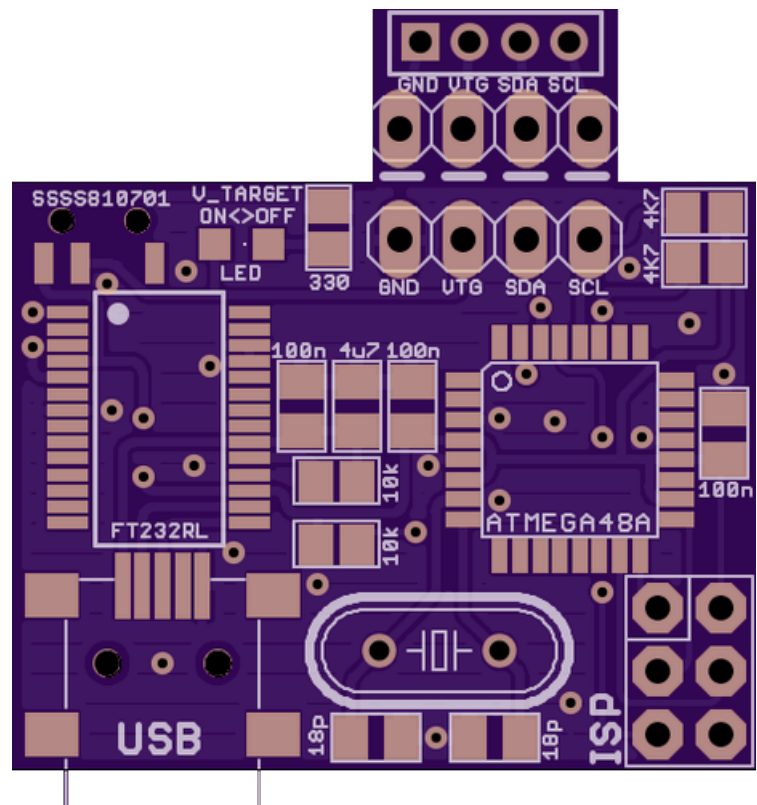
The hardware for the programmer is finished. You can order the PCB's at https://oshpark.com/profiles/micro (CDCE9XX Programmer)

Parts list:

Quantity	Description	mouser.com part name
1x	FT232RL USB UART interface IC	895-FT232RL
1x	ATMEGA48A microcontroller	556-ATMEGA48A-AU
1x	Mini USB socket	587-690-005-299-143
1x	Sliding switch	688-SSSS810701
1x	Pin header	855-M22-2010405
1x	Crystal	774-ATS080B
3x	100 nF capacitor	80-C0805C104K5R
1x	4.7 uF capacitor	581-08053D475K
2x	18 pF capacitor	77-VJ0805A180GXJCBC
1x	330 Ohm resistor	71-CRCW0805-330-E3
2x	10 kOhm resistor	71-CRCW0805J-10K-E3
2x	4.7 kOhm resistor	71-CRCW0805-4.7K-E3
1x	LED	720-LSR976-NR-1

Firmware for the programmer: http://www.mediafire.com/
download/tn6kw1grj65s761/
CDCE9XX_PROGRAMMER_FW_v1.zip
Source: http://www.mediafire.com/download/n6yon471o85ehan/
CDCE9XX_PROGRAMMER_FW_v1_SOURCECODE.zip

Programming software for Windows PC:
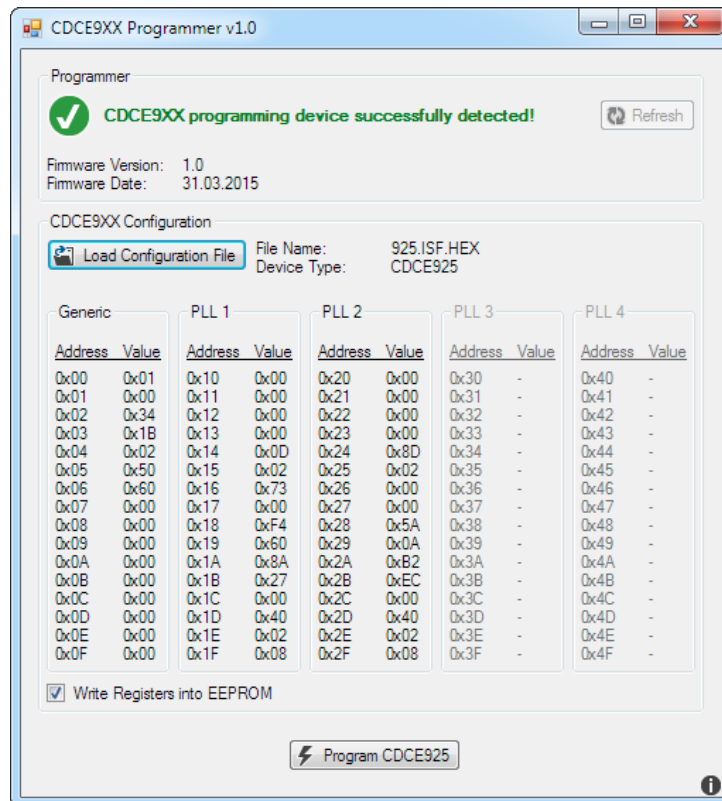
Download link: http://www.mediafire.com/
download/55gu3975qu2kmux/CDCE9XX_Programmer_GUI.zip

The software runs on Windows PC's and it needs the .NET
framework 4.5.

## Outdated! Instructions for the three new DFO's coming soon.

How to program the DFO:
1. Obtain a configuration file

To program the DFO you'll need a suitable configuration .hex file.
You can use one of my example files. If you want/need different
frequencies, you should download and install the Clock Pro
software by Texas Instruments: http://www.ti.com/tool/
CLOCKPRO

Start Clock Pro and you will see this window:
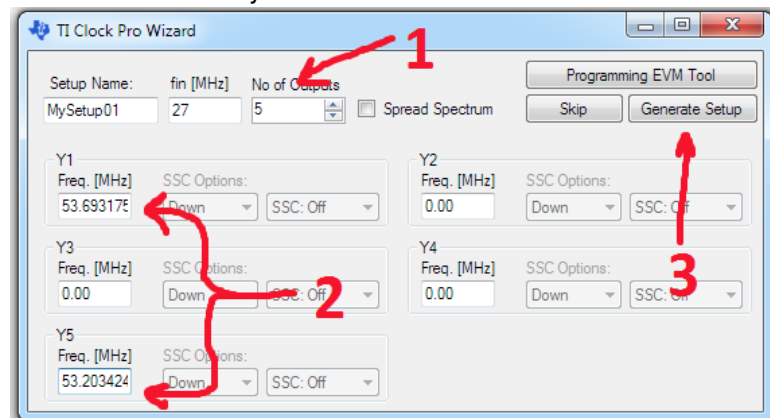
First set the number of outputs to 5. Then enter both the frequencies you desire in the the Y1 and Y5 window. Y1 will be selected if the CLK1/!CLK2-pad of the DFO is tied to a high level (or just left open). Y5 will selected if you tie the CLK1/!CLK2 pad to GND.
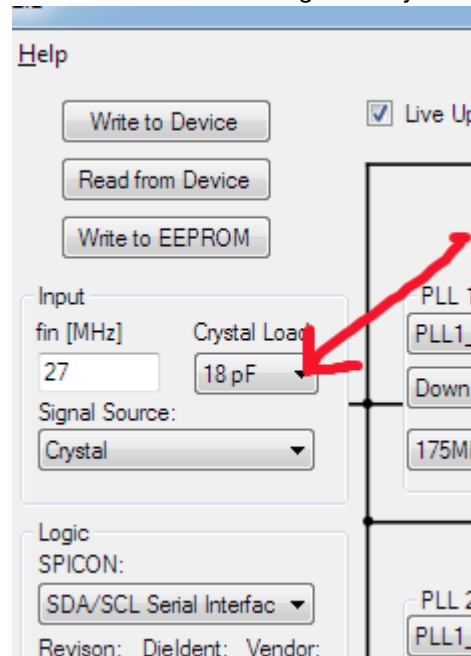After you have entered your frequencies, click on "Generate Setup".

Clock Pro should propose a CDCE925 device for your task. Click on "View Setup".
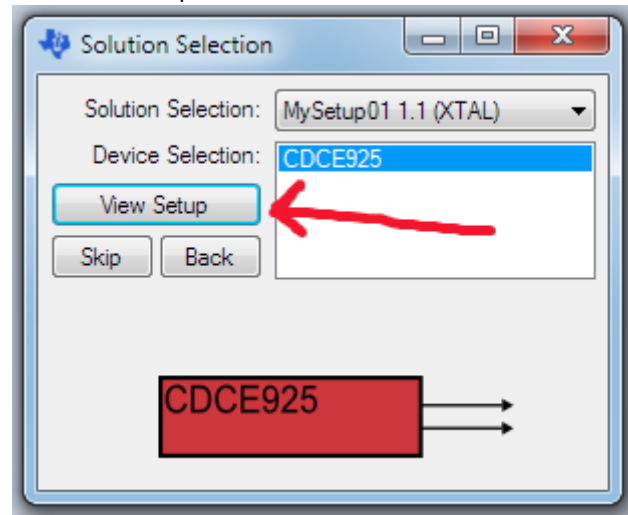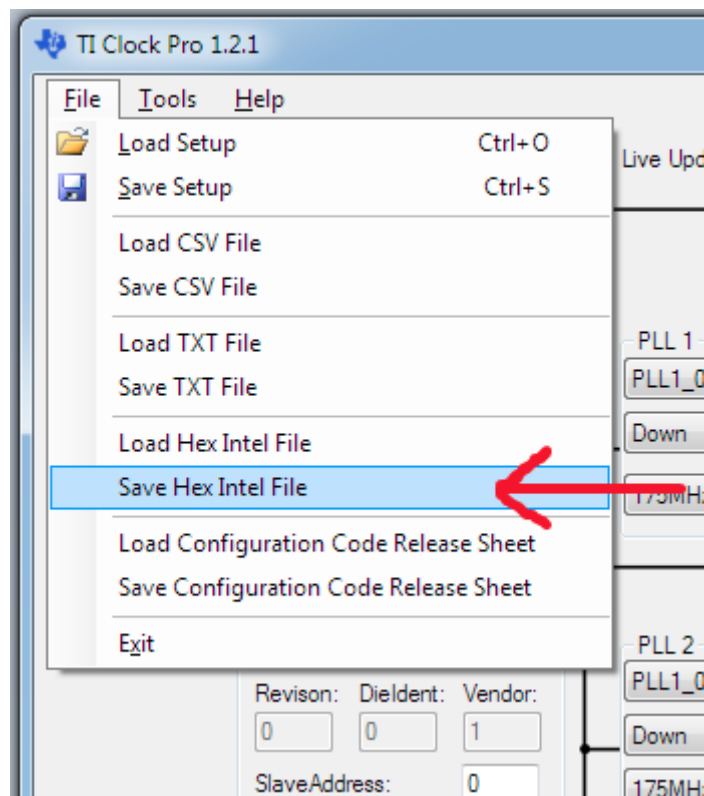

(If it proposes a CDCE913 instead, go back and enter some frequencies for Y2, Y3 and Y4 too.)

In the next window change the crystal load to 18 pF:

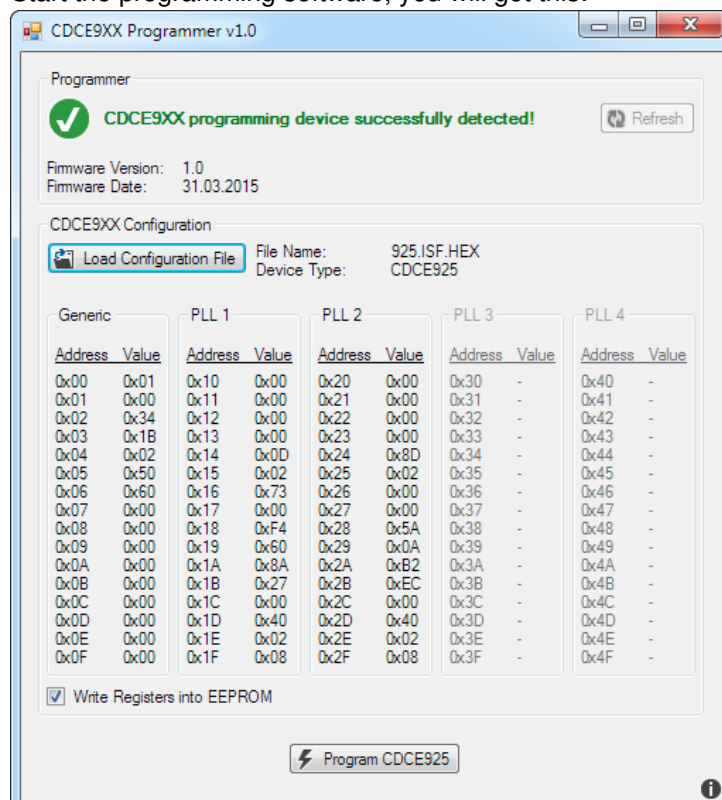Now click on File->"Save Hex Intel File"

That's it, now you got the .hex file for your DFO. 😊

2. Program the DFO with your .hex file

Plug your assembled CDCE9XX programmer hardware into your PC with an USB cable. It's important that you've also flashed the programmer with the firmware I've provided above.

Start the programming software, you will get this:

If the software doesn't recognize your programmer hardware, chances are you haven't flashed it with the right firmware and fuse bytes. (Or you might have made a mistake while assembling the PCB.)

Click on "Load Configuration File" and load the .hex file you've created with Clock Pro before. If the .hex file is valid, the "Program CDCE925" button on the bottom will be enabled.

Now you should connect the programmer hardware with the DFO you want to program. Just insert the pin header into the 4 approriate holes:
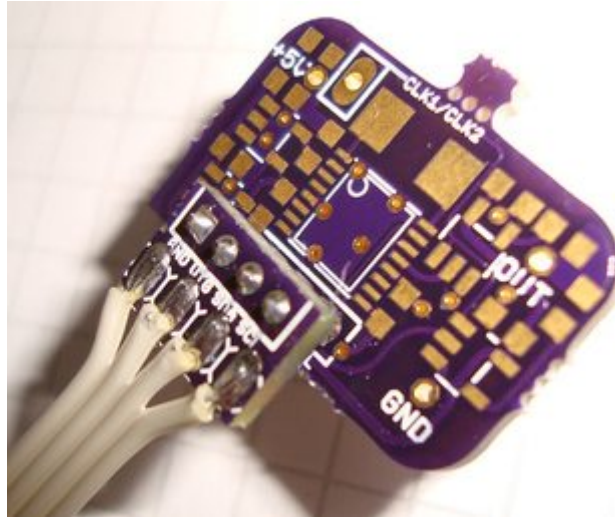

(Pic for demonstration purposes only. Of course you need to assemble the DFO itself too. 😁)

It's important to set the V_Target switch on the programmer hardware right. If you have the DFO sitting on your table and you want to program it, set V_Target to on. The LED should start to shine red.
If you want to program a DFO that's already installed in a system (let's say in a Megadrive for example), then turn V_Target off. Instead turn on the Megadrive itself. It will power the DFO.
If you mess this up you can damage your console, the programmer and even your PC!
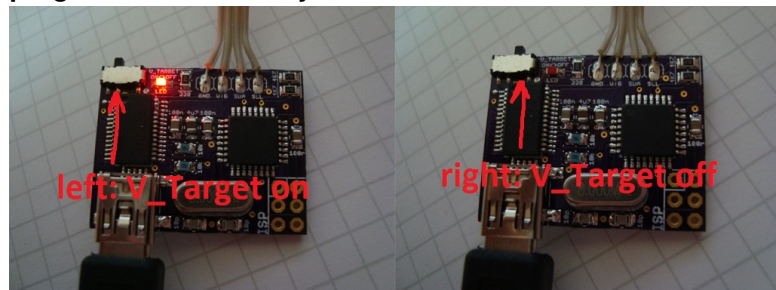
Now click on "Program CDCE925". The CDCE9XX programming software will tell you if the programming was successful or not. 😊

WTS/WTT: UWRC wireless controller PCB sets; looking for MVS, MD & Saturn games

---

○ albino_vulpix

Professional warranty voider

■■■■

MassiveMember

Posts: 190

Location: Melbourne, Australia

Logged

February 01, 2015, 09:56:29 AM

#1

Looks like an interesting project! Do keep us posted.

Strange though, my PAL MDII works fine at 60Hz on my Framemeister. I'm using firmware 1.11, SCART cable wired this way and no sync stripper.

○ micro

■■■■

MassiveMember

Posts: 392

Logged

February 04, 2015, 03:12:22 AM

#2

I guess this can be explained by the fact that oscillators got an initial error (something about 50 to 100 ppm I guess) but also a frequency drift (x ppm/year).

Switching my japanese Saturn to 50 Hz or playing NTSC games on my modded PAL Playstation consoles also result in picture problems on the Framemeister. But it's not as bad as my PAL MD2. I suspect that this is also caused by oscillators/crystals with (marginally) different frequencies used in PAL and NTSC consoles. These problems are also documentated in the X-RGB Mini Wiki:

http://junkerhq.net/xrgb/index.php/Sega_Genesis
http://junkerhq.net/xrgb/index.php/Playstation

Inside the Saturn there's no oscillator but a crystal. It should be possible to use my dual frequency oscillator in the Saturn, too. I just have to find out which of the crystal's two pins is clock in and clock out.