DFO: Dual Frequency Oscillator [Tutorial]

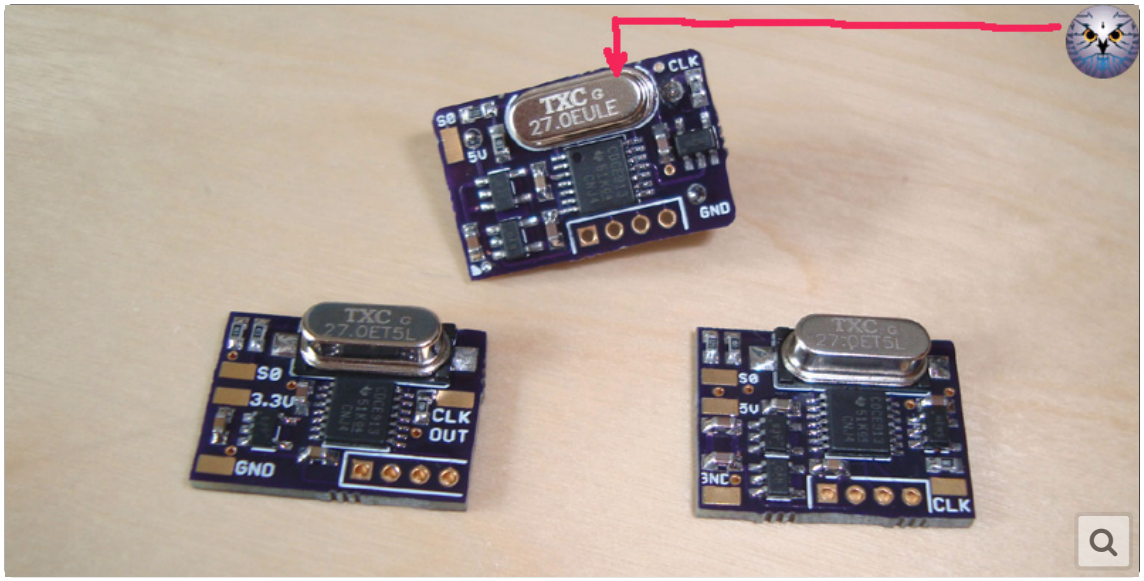
micro

DFO: Dual Frequency Oscillator

16. July 2015, 01:11

Your love CiBo stibs, for the sake of completeness, this project was also represented by me on the Circuit Board. Maybe one may need it.

DFO - dual frequency oscillator



What is it? What is it?

The DFO (dual frequency oscillator) is an oscillator circuit. A clock signal is therefore generated and output. The circuit is based on theCDCE913 grid IC (http://www.ti.com/product/cdce913)by Texas Instruments. The special thing is that the DFO is programmable. The frequency of the output clock signal can (almost) amount to any value in the range of 80 kHz to 230 Mhz.
The highlight of the whole thing is also the S0 input of the DFO. Depending on the applied level on S0, it can be switched between two different frequencies.

What do you need something for?

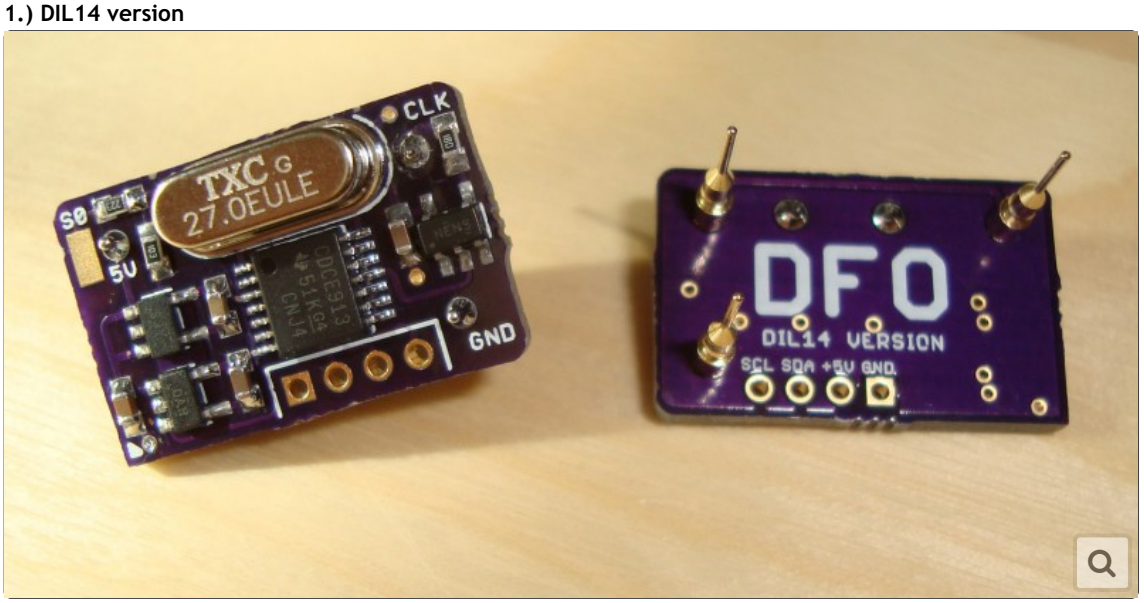
My motivation for the development of the DFO was that my PAL Megadrive 2 caused problems with the XRGB mini frame master in 60 Hz mode. The reason: In Pal MD consoles, oscillators with a frequency of 53.203425 MHz are installed, but in NTSC consoles, which are at 53.693175 MHz. The resulting frame rate of a PAL-MD in 60 Hz mode therefore differs slightly from that of a real NTSC-MD console. This deviation is enough to get the frame master out of the kick.
The first idea was to simply install an NTSC oscillator with 53,693175 MHz. However, as I found out, there are no more oscillators or crystals with this frequency to buy. I found only Aliexpress, but there was no clever specs alone a reasonable data sheet.
For this reason, I developed the DFO. The 50/60 Hz switch in Megadrive 2 is now connected to the S0 input of the DFO. The DFO supplies the megadrive of the appropriate frequency depending on the mode selected. The image problems with the frame master are therefore passé.

Another area of application for DFO would be e.g. Overclocking. Installed in a console or in an FX-SNES module, the DFO could be programmed with the maximum possible frequency. By applying 5V or mass at the S0 input of the DFO, it could then be switched back and forth between normal and overclocked operating mode.

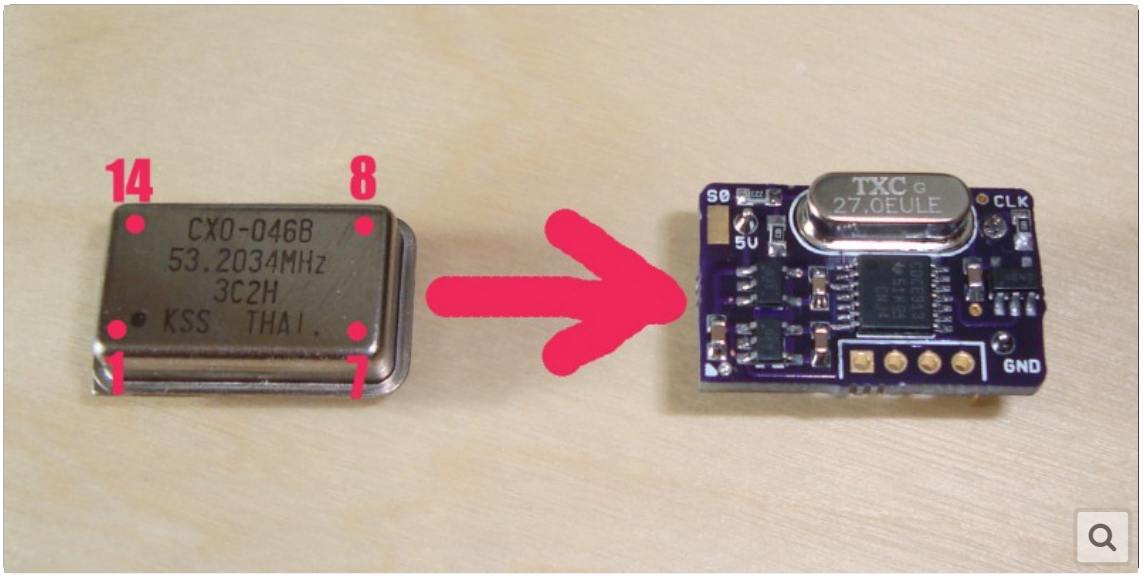
What versions of the DFO are available?

I have three different DFO versions on offer. The designs are hosted at Oshpark and can be ordered directly there (at a very attractive price):**oshpark.com/profiles/micro** (https://oshpark.com/profiles/micro)
I can and will not offer a fully assembled DFOs, please do not ask about it. The man is himself!

1.) DIL14 version



The DIL14 version of the DFO has the same pinout and also in roughly the same dimensions as traditional oscillators in the DIL14 metal housing:



The supply voltage may be up to 5 V (maximum 5.5V), therefore the DIL14 version of the DFO can be installed directly in the Megadrive as an oscillator replacement.

Components & assembly plan:

2.) 5 V-SMD version

The 5 V-SMD version is based on the same circuit as the DIL14 version. It can also be supplied with up to 5 V (maximum 5.5 V). However, the underside is flat, since only SMD components are used on the top. The pinout is also different: On the left side are the inputs (voltage supply & S0 to switch the frequency), on the right side the output clock signal.
Possible applications: consoles that expect a clock signal with a Vpp value of up to 5V (SNES vllt?). However, this DFO version does not have to be operated with 5 V, it also works with a supply voltage of 3.3 V.

Components & assembly plan:

3.) 3.3 V-SMD version

If it is known in advance that the DFO is to be used in a 3.3 V environment (maximum 3.6 V!), the 3.3 V-SMD version is recommended. Looks very similar to the 5 V-SMD version, but requires some components less and is therefore easier and cheaper to build.
Applications: PSX and Neogeo MVS MV-1C e.g.

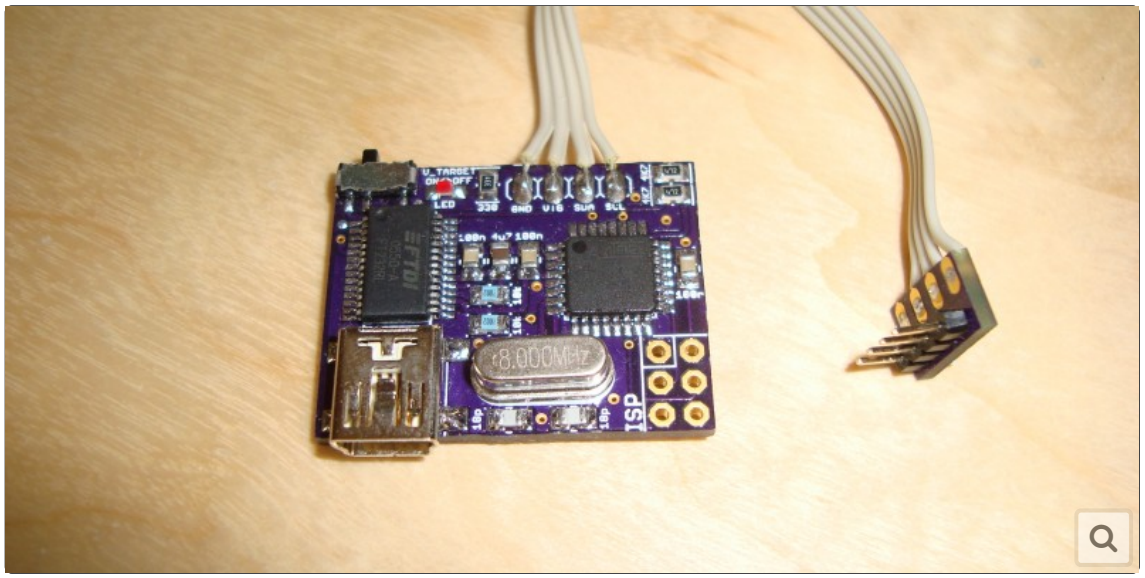
Components & assembly plan:

So, that's it for today. The rest is coming! (programming device, programming software etc.)

micro

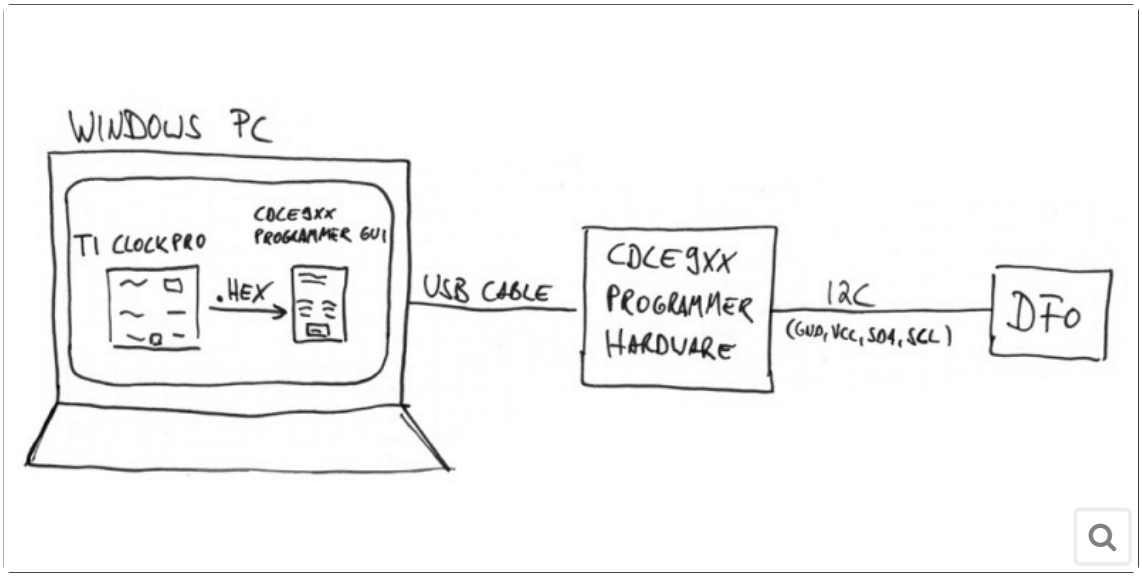
16. July 2015, 01:11

CDCE9XX Programmer Hardware



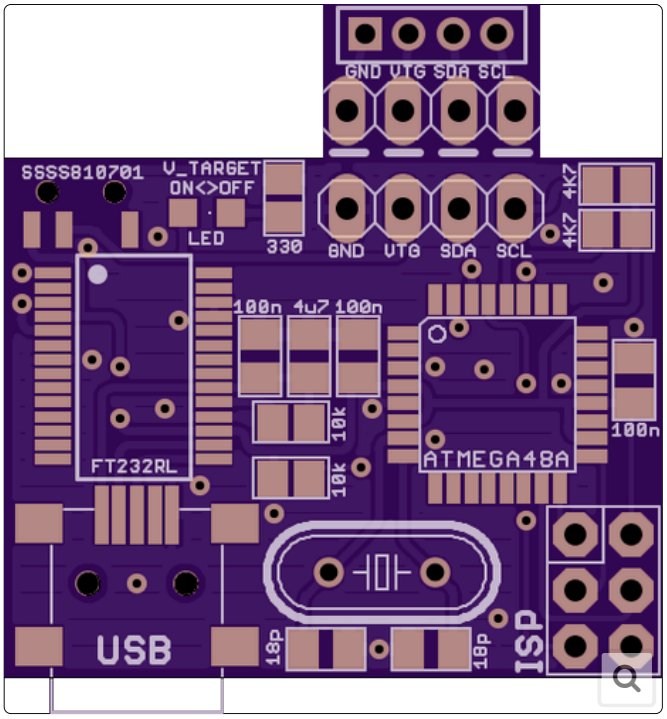
As the name suggests, this is the programmer with which the CDCE9XX ICs from TI can be programmed. This includes the CDCE913 on the various DFO versions, but also the "thick" ICs (CDCE925, CDCE937, CDCE949) can be programmed with it.

Concept



As the above image makes clear, the CDCE9XX programmer is the connector between the DFO and a PC, on which a graphical user interface for the programr's response runs (more on this later). The CDCE9XX programmer is connected to the PC using a mini USB cable. The connection to the DFO is made via 4 strings (GND, V-Target aka VCC, SDA & SCL).

Board & assembly



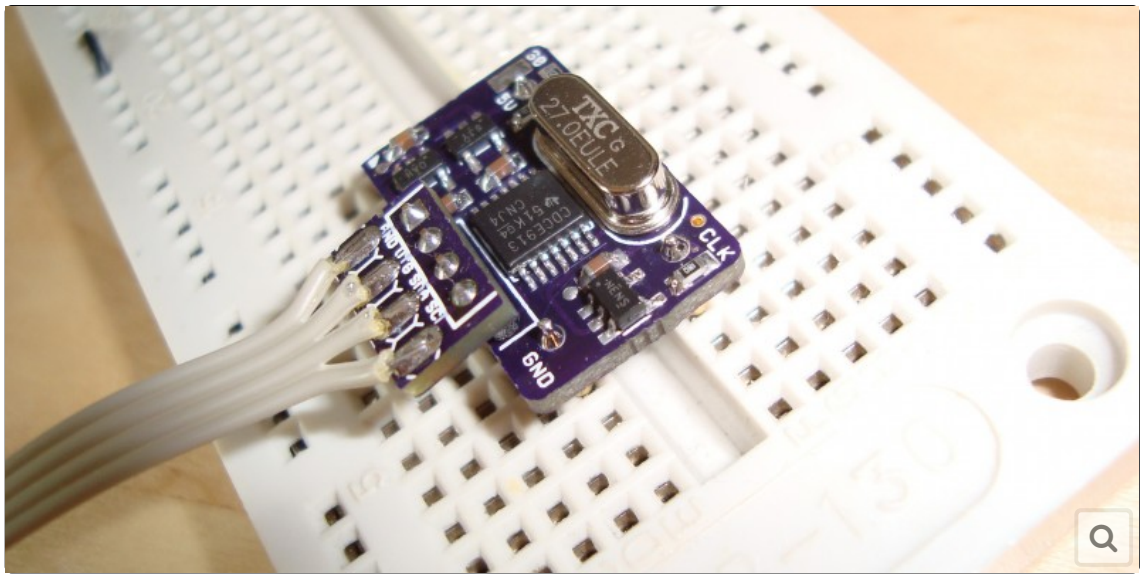
The board of the CDCE9XX programmer is also available for ordering on oshpark:oshpark.com/profiles/micro (https://oshpark.com/profiles/micro)

List of required components:

Tip for loading: what does the mini USB jack**before**the FT232RL, otherwise it will be really hairy!😊

The board has a small corner at the upper end, which can be tumbled if necessary and can extend using a ribbon cable or strand, as I did in the very top of the picture. This is not a must, but it increases flexibility.

In the list of components, a pin header with a 2 mm grid size is provided, which is then soldered into the extra sip slip and establishes the connection to the DFO. Currently I always put the pin headers in the DFO, edge it for a good contact and leave it on until the programming process is completed:



But you can also solder a suitable jack strip into the DFO, then you do not need to keep your finger on it. It would also be conceivable to soldering in the rightPogo-pins (http://www.ebay.de/sch/i.html?_from=R40&_sacat=0&_nkw=pogo+pins&LH_PrefLoc=2)in the extra zip, there are many possibilities to realize a connection to the DFO...

Firmware for the programmer

In order for the CDCE9XX programmer to work as desired, the microcontroller of the type ATmega48A located on it must first be flashed with the appropriate firmware.

Firmware:mediafire.com/download/tn6kw1g...E9XX-PROPROMMER-FW-v1.zip
(http://www.mediafire.com/download/tn6kw1grj65s761/CDCE9XX_PROGRAMMER_FW_v1.zip)
Source:mediafire.com/download/n6yon47...MMER-FW-v1-SOURCECODE.zip
(http://www.mediafire.com/download/n6yon471o85ehan/CDCE9XX_PROGRAMMER_FW_v1_SOURCECODE.zip)

The six-pin AVR ISP interface ("ISP") is located in the right-hand right-down on the programmer board. The pinout ensticht thecommon standard (http://www.mikrocontroller.net/articles/AVR_In_System_Programmer#ISP). Pin 1 is the one that is surrounded by the square. (Who would have thought?)

I do not even want to go into flashing at this point, because it has already been dealt with here on the board several times and the actual flash process can differ, depending on which AVR ISP programmer is used for this. For questions/problems simply report here, no thing!
(Do we actually have an AVR Flash tutorial? I thoughtÃ sanniwould have written down this something about.)

Alternative to the programmer

There is an alternative to the programmer described above, which I have not tried myself, but I do not want to withhold you from. Users"Unseen, who is now also up to mischief here in the CiBo, has created a script with which the CDCE9XX ICs can be programmed using a Raspberry PI.

So far, however, only the CDCE925, which was used in the very first DFO version, is supported. But who knows, vllt. Unseen is still drilling the thing...

Link to GitHub:github.com/ikorb/cdceprog (https://github.com/ikorb/cdceprog)

Files

CDCE9XX PROPROMMER-FW-v1.zip
(1.63 kB, **464** times downloaded, last: 5. June 2024, 09:03)

CDCE9XX PROGRAMMER-FW-v1-SOURCECODE.zip
(6.35 kB, **391** times downloaded, last: 5. June 2024, 09:03)

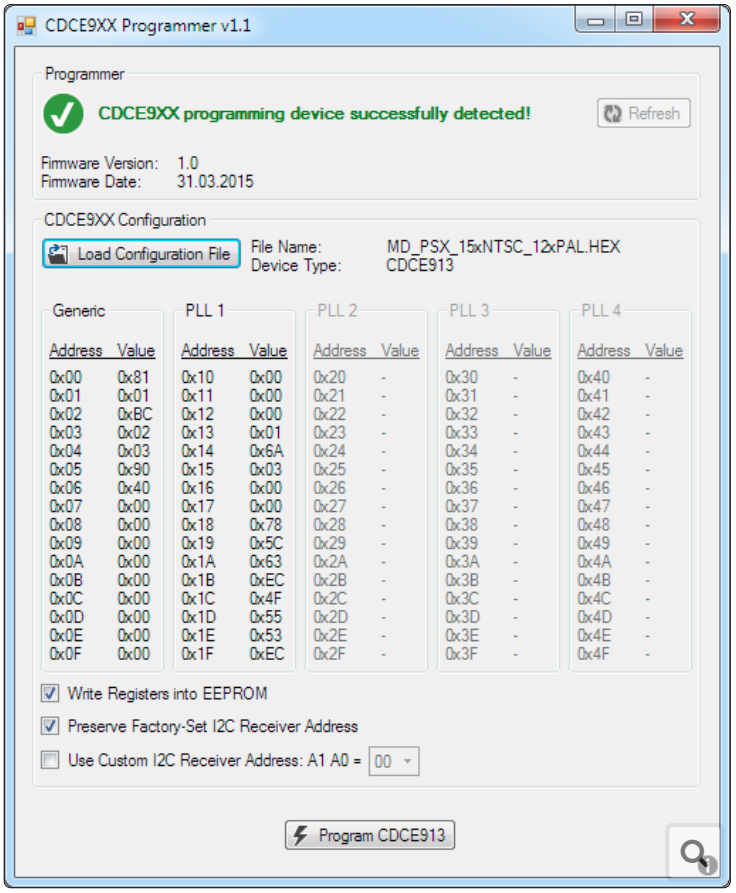
This post has been edited 1 times, last edit bymicro(16. July 2015, 20:09)

micro

16. July 2015, 01:12

Graphical user interface for programming the DFO

For the appeal of the so-called CDCE9XX programmer and thus the programming of the DFO there is a separate graphical user interface (GUI):



Download:mediafire.com/download/dx4k1yt...X-Programmer-GUI-v1.1.zip
(http://www.mediafire.com/download/dx4k1yt349tb7ct/CDCE9XX_Programmer_GUI_v1.1.zip)
Prerequisite: Windows PC with .NET Framework 4.5

Display & control elements

Let's simply go through the top to the bottom...

1.) "Programmer" info window

This signals whether a CDCE9X programmer connected to the PC has been recognized. The firmware version of the program is also displayed. (At the moment the v1 firmware is up to date.)
If a connected programmer is not recognized, simply click on the "Refresh" button. If the CDCE9XX programmer is still not recognized, the probability is high that the CDCE9XX programmer is not occired (error when picking up, not flashed with the firmware, etc.)

2.) "CD9XX Configuration" window

Here you should load the .hex configuration file that suits your application. To do this, click on the "Load Configuration File" button and select a suitable .hex file. I provide pre-built .hex files, see the following post! Of course, you are also free to create your own configuration files. 😊

After loading the .hex file, the recognized CDCE9XX Device Type is displayed (for the DFOs, this is always the CDCE913). In addition, all register values contained in the .hex file are also displayed.
Below are 3 checkboxes that have special options activated or deactivated. If you have no idea, then you better leave it at the default settings. Here the functions areexplains:

3.) "Program CDCE913" button

This will program the DFO.😊
The button is only available if the CECE9XX programmer has been recognized and you have loaded a valid .hex configuration file.

(In the other case, the button is greyed out.)

After pressing the button, you will receive a feedback whether the programming process was performed successfully (we hope) or whether an error occurred. Possible errors would be e.g.:

- I2C Error
Wrong I2C address? Do the 4 strings from programmer to DFO (GND, V-Target, SDA, SCL) have no clever contact?

- Verification Error
Register values of the DFO not match the written register values.

- FTDI Error
Programmer pulled out of the USB jack during the programming process?

Files

CDCE9XX-Programmer-GUI-v1.1.zip
(51,64 kB, **530** times downloaded, last: 5. June 2024, 09:03)

This post has been edited 1 times, last edit bymicro(18. July 2015, 10:53pm)

micro

16. July 2015, 01:12

.hex configuration files

As can be seen from the previous posts, the DFO must be programmed with a .hex configuration file. This file determines the frequency of the output clock signal of the DFO at the end.

You can create your own configuration files using the Texas Instruments ClockPro program or you simply take my pre-made sample files🙂

Pre-made configuration files

I offer pre-made configuration files whose functionality is checked and confirmed.

1.) MD-PSX-15xNTSC-12xPAL.HEX

Download:MD-PSX-15xNTSC-12xPAL.zip

S0 - high > 53.693175 MHz (Ã...15x NTSC subcarrier)
S1 Ã low > 53.203425 MHz (Ã...12x PAL subcarrier)

Suitable for:

- Megadrive 1/2(DIL14 version of DFO recommended)

- PS1/PSOne(3.3V-SMD version of DFO recommended)

2.) SAT-4xNTSC-4xPAL

Download:SAT-4xNTSC-4xPAL.zip

S0 - high > 14.31818 MHz (Ã...4x NTSC subcarrier)
S1 low > 17.734475 MHz (Ã...4x PAL subcarrier)

Suitable for**PAL and NTSC Saturn consoles**(5V SMD version of DFO recommended)

3.) SAT-4xNTSC-PAL CORRECTION

Download: SAT-4xNTSC-PAL-CORRECTION.zip

S0 - high > 14.31818 MHz (Ã...4x NTSC subcarrier)
S1 low > 14,22148... MHz

Suitable for **NTSC-Saturn consoles** (5V SMD version of DFO recommended)

4.) DFO-for-NeoGeo (THX to Ã...mi213 & Ã...borti4938)

Download: DFO for NeoGeo.rar

S0 - high (3.3V not 5V!) -> originale 24.00 MHz (original refresh rate of 59.18Hz)
S1 - low - > 24.3059 MHz (refresh rate of 59.94 Hz)

Tested with an MV-1C with the 3.3V version of the DFO (SMD version recommended; possibly also includes 3.3V on the Ostillator for other MVS versions?)

Create your own configuration file with ClockPro

With the ClockPro (http://www.ti.com/tool/clockpro) software tool from Texas Instruments (http://www.ti.com/tool/clockpro), you can create your own .hex files with the frequencies you need for your application. For most people, this will probably not be important, so the whole stuff is hidden in the spoiler for a better overview in the thread.

This post has been edited 17 times, last edit byn00b(19. November 2023, 17:51)

micro

16. July 2015, 01:13

How to install DFO into unknown console?

With the Megadrive 1/2, the installation of the DFO is very easy. Simply removing the old quartz oscillator and inserting the DFO (ideally in the DIL14 version) - done. However, most consoles have no oscillator installed as a separate component. Instead, a oscillating quartz is usually used and an oscillator circuit is built on it.

Using the example of Saturn, it will now be shown how such oscillator circuits are to be analysed in order to be able to install the DFO.

This post has been edited 1 times, last edit bymicro(27. July 2015, 20:33)

xadox

16. July 2015, 7:43:

I will do that to my PAL MD1. Can this also be operated in a PAL PS1?