



[Home](#)[Blog](#)[Ecosystem](#)[API](#)[Forum](#)[Mission](#)[Downloads](#)[Wiki](#)

RetroArch, Libretro core license violations by

Hyperkin's Retron5

Posted by [thetoadking](#) on Sep 19, 2014 in [Blog](#), [GPL](#), [Libretro](#), [Non-commercial](#), [RetroArch](#) | 1,709 Views | [2 Comments](#)

We have an open-source project called RetroArch. It has a development interface called libretro that allows for the easy creation of emulators and games that can plug straight into this program called RetroArch. This development interface is open to others so that they can run these pluggable emulator and game cores also in their own programs or devices. You can find this project on Github. (github.com/libretro). We also have a website – <http://www.libretro.com>. We started this project in late 2010 and we have been steadily increasing in popularity. We are on over 15 different hardware platforms right now that you can run RetroArch on, including Android (which this Retron5 device is based on).

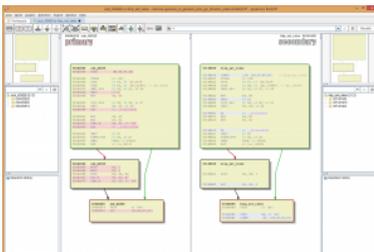
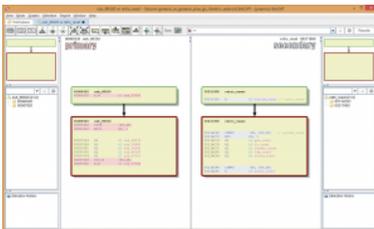
These open-source programs are covered under certain licenses. Several of the emulators are covered under non-commercial licenses, which means they cannot be sold or profited from.

We have discovered that Retron5 is in violation of the licenses of several projects:

Genesis Plus GX

1. It uses the open-source emulator 'Genesis Plus GX' by author Eke-Eke for its Sega Genesis/Mega Drive module (<https://github.com/ekeke/Genesis-Plus-GX>). This core has been licensed under a non-commercial license. It can therefore not be part of a commercial product. Furthermore, having this piece of software on the same device that has software covered under other licenses would be another license violation.

Proof is in the accompanied screenshots at the bottom of this post. None of the authors were contacted about their code's use in the Retron5 hardware.



SNES9x Next

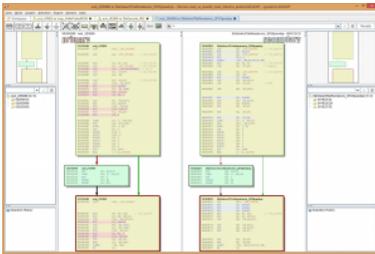
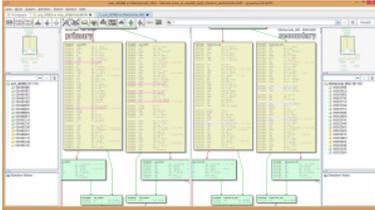
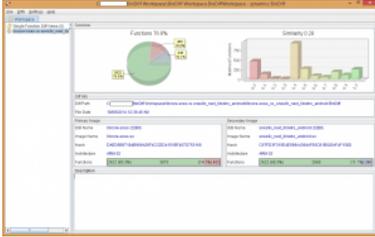
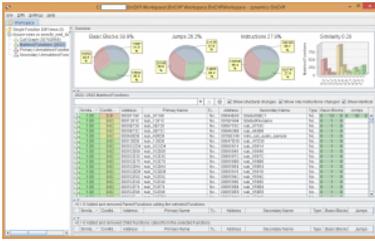
2. It uses the open-source emulator 'SNES9x Next', which is itself a derivative of SNES9x (<https://github.com/libretro/snes9x-next>). I (Squarepusher) personally made this version of SNES9x. It has a few differences compared to normal SNES9x. It has SuperFX overclocking code and it has certain game speed hacks that make games run faster on slower hardware. This comes at the expense of some graphics inaccuracies though.

We could tell it was the SNES9x Next core because the exact same strings for variables to do with the speed hacks and the SuperFX overclock code popped up in their SNES core.

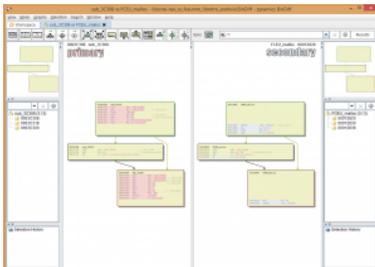
SNES9x is licensed under a non-commercial license. Like Genesis Plus GX, it can therefore not be a part of a commercial product.

What also bears pointing out is that SNES9x Next has never been released in any other version than the libretro version. Libretro is the development interface of RetroArch if you remember. We will return to this later on.

Proof is in the accompanied screenshots (see below). None of the authors were contacted.

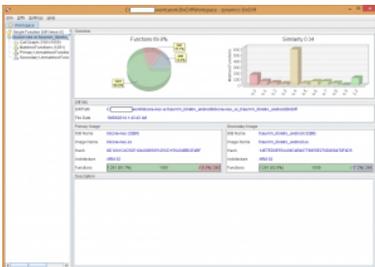


FCEUmm



3. It uses the open-source emulator ‘FCEUmm’ for its NES module, which is itself a derivative of FCE Ultra. FCEUmm is licensed under the GPLv2. Technically they would have been allowed to sell this IF they would have made sure none of the other software on their device were incompatible with GPLv2. Unfortunately, this turns out not to be the case as we’ll find later on – since they are using GPLv3 code as well which is technically incompatible with this license.

Proof of it being FCEUmm is in the accompanied screenshots (see below).

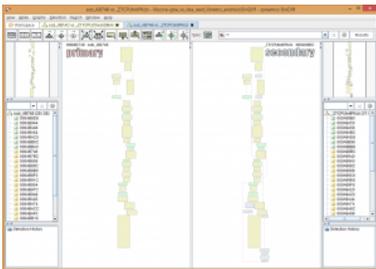
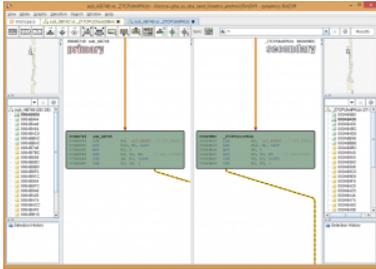
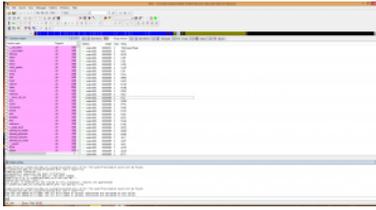


VBA Next

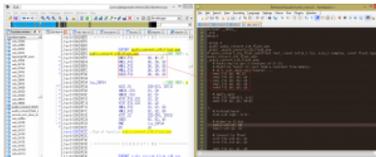
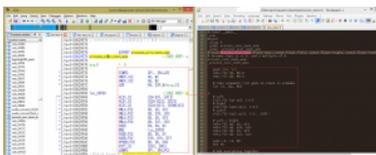
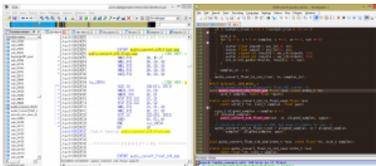
4. It uses the open-source emulator ‘VBA Next’ for its GBA module. VBA Next is a derivative of another emulator called VBA-M. I (Squarepusher) made this

version specifically and I could recognize it was this version because of the fact that I have built-in a game database into this emulator. The game ID strings that are used to identify the ROMs appeared in Retron's GBA module as well. As for the rest of the code, it is undoubtedly VBA. The screenshots showing the code flow of operation will illustrate this clearest.

VBA Next is licensed under the GPLv2. None of the authors were contacted. Proof is in the accompanied screenshots (see below).



RetroArch



5. We found obvious bits of RetroArch's sourcecode inside their frontend. Now the reason we could identify these snippets is because it is inlined Assembly code that is hard to obfuscate. The relevant parts are the ARM NEON-optimized sinc resampler code and the audio integer to float conversion routines. If you want photographic evidence, I refer you to the second link I posted below.

All of the other C code of RetroArch seems to have been obfuscated so it will take us some more time to identify these parts. What is evidently clear though is that they are already in violation of the GPL license that we covered this RetroArch code under. GPL version 3 specifically forbids TIVO-ization. Let me explain later what TIVO-ization is. It basically means that you use opensource software to make a locked-down hardware device that doesn't allow you the freedoms that the GPL generally provides to users and developers alike.

Since they have used our libretro cores evidently and since the only way to actually use these cores is through a libretro frontend implementation, and since actual RetroArch code has already been identified in their frontend, this raises serious questions as to how much of their frontend constitutes 'original work' and how much of it is just RetroArch. Either way, they are in the wrong for several reasons here:

- They didn't provide any patches to us, i.e. the parts that changed. They should have also made these publicly available for every user to download since that is

part of the rules and stipulations of using GPL code.

- They made a locked-down crippled hardware device based on open-source software. This is TIVO-ization and the GPL version 3 was specifically made to prevent this.

What is TIVO-ization?

<http://en.wikipedia.org/wiki/Tivoization>

Tivoization /'ti:vɔɪ,zɛɪʃən/ is the creation of a system that incorporates software under the terms of a copyleft software license (like the GPL), but uses hardware restrictions to prevent users from running modified versions of the software on that hardware. Richard Stallman coined the term in reference to TiVo's use of GNU GPL licensed software on the TiVo brand digital video recorders (DVR), which actively blocks users from running modified software on its hardware by design.[1][2] Stallman believes this practice denies users some of the freedom that the GNU General Public License (GNU GPL) was designed to protect.[3]

The GPL version 3 was specifically made because GPL version 2 did not provide enough safeguards against abuse like in the case of the TiVo digital video recorders. They would take from open source, not credit anybody and not give anything back either – and even create a closed platform around it where they would set themselves up as owners of the software (and in effect the hardware).

GPLv3 forbids you from building a gated community around open-source software like this and giving nothing back in return. The fact that they have used RetroArch's GPL version 3-licensed audio resampler code in a product that is running a locked-down, encryption-crippled version of Android is already bad enough. That they don't even provide to users the ability to run content on this device without any restrictions is another serious concern.

Anyway, as it stands right now in its current state the product is using parts of our software illegally. There were also some other things found that were legally questionable like a Microsoft-licensed Verdana font which is covered by a End-User License Agreement, so there are multiple license violations here at play.

Multiple license violations, multiple conflicting licenses, bad faith

The problems with this are many-fold, but for us it comes down to mixing non-commercial cores on this device with more permissively licensed cores, the infringement of the emulator authors' rights, the lack of credit paid where credit is due, the lack of freedom in the hardware device (which restricts the user in what he/she can do and makes him/her reliant on Hyperkin to serve as the gateway keeper since he/she can't uncripple this version of Android on their device without voiding their warranty), and the multiple conflicting licenses.

Open-source is not a matter of doing with it as you please. The license is there for a reason and it needs to be followed, and it dictates how you should go about your business when deciding to make a commercial product out of such software. GPL is known as a 'viral license' which means that the community behind this uses the viral nature of the GPL as an effective strategy to ensure more and more software gets licensed under the GPL, since every bit of GPL code that gets incorporated into another project needs to be made GPL or GPL-compatible as well otherwise it's a violation of the license. As it stands right now, the software for the Retron5 is very likely illegal to distribute.

Links:

1. <http://imgur.com/a/T6W4e> – This image gallery shows comparisons of the infringing derivative Retron code vs. the originals
2. <https://www.anonimg.com/img/5d807718b069e5dae8a4e4320fdda1ab.png> – This shows the RetroArch audio resampling and audio conversion routines in the Retron frontend
3. <http://i.imgur.com/81bnckH.png> – Another image of the audio resampling code. Originally from this tweet: <https://twitter.com/FioraAeterna/status/512790355591196673>
4. Regarding the MS Verdana font: <http://www.microsoft.com/typography/fonts/font.aspx?FMID=1817> – “Verdana is either a registered trademark or a trademark of Microsoft Corporation in the United States and/or other countries.”

(EDIT: A section here about the ARM Mali drivers was removed since it appears to not be related to these issues)