

DA/FA mode decoder V3.5.1A(V0)
3 Jul 2021

Important: Thanks to some individuals on the Audiophile Style forum. These people do not endorse this program, but have contributed in one way or another – these are in no specific order – this list is incomplete and will add people if they remind me!!! Also, several other people both in the business, and not, who requested participation to be private.

@lucretius, @sandyk, @rando, @pkane2001, @jabbr, @skippack, @rexp. Extra special thanks to our late friend @sandyk.

Instead of writing about individual ‘differences’, this will describe general changes and behaviors, because the usage intent is very different. New info relative to previous release is in bold...

IMPORTANT: With all recent corrections, a few valid FA recordings will need modifying the `--coff` parameter.. The `--fw=classical` mode switch is still very operative. Perhaps 20-30% of pop recordings need ‘`--fw=classical`’ and over 2/3 of the ‘jazz’ recordings will need ‘`--fw=classical`’ also. Some ancient recordings might need `--lfeq`, which tries to be a good generic EQ for those recordings.

REALLY IMPORTANT: Many recordings appear to need a slight rolloff. This rolloff is a smaller increment than normally used for internal EQ. There is an ‘`--hfrl`’ switch that gives some control of the rolloff. If you want MORE HF rolloff, then start with ‘`--hfrl=1`’. If you force NO extra HF rolloff, then use ‘`--hfrl=0`’.

IMPORTANT USAGE CHANGES:

- Some recordings might need `--coff=-6` (Orchestrals)
- There are now: `--f9` and `--f10` switches in addition to the default `--fa` (same as nonexistent `--f8`.) A lot of recordings are definitely `--f10`, so beware the case where you think that there is more expansion available after `--fa`, because there likely is. (Note: `--f9` is same as `--fa=+9` and `--f10` same as `--fa=+10`) Sometimes `--f9` is better than `--f10`, even on `--f10` materials.
- ~~**WRONG/REMOVED:** Adding the `--vinyl` switch might have been premature because it appears that the more accurate equivalent might sometimes be ‘`--f10`’. (V3.5.0A ~~`--vinyl` changed to `--preFA` to avoid confusion with newer, FA vinyl.)~~~~
- `--lfeq=1` might be the norm when bass needs minor trimming. – you might have to use it more often than not. The decoder has very wide LF response, and perhaps TOO MUCH.
- `--lfeq` has become a complex monstrosity, but there is a ‘magic’ setting for those recordings with seriously too much bass. Just try ‘`--lfeq=99`’. It often appears to do the exact magic needed. Small additional changes can be made by using the ‘`--pvl`’ commands, but hopefully seldom really needed.
- A lot of the higher quality speed increases are less than expected because ‘`--f10`’ or ‘`--f9`’ might become the optimal norm. Once in a long, long while, the improvement by ‘`--f10`’ can be greater than using ‘`--xpp`’ over and above ‘`--fx`’. So, most of the time, given a choice, I would suggest ‘`--f9`’ in cases where a tiebreaker of using modes higher than ‘`--fx`’ vs ‘`--f10`’ over ‘`--fa`’.

Using '--f10' instead of '--f9', when '--f9' is indeed correct can create very subtle distortions. Therefore, it is almost always most safe to use '--f9' and be done with it.

BUGFIXES from V3.5.0A to V3.5.1A

Previous comments about the --vinyl/--preFA modes are all wrong. Capability removed. Additional minor adjustments to match almost 100% pre-FA materials though.

BUGFIXES from V3.4.3J to V3.5.0A

Important change for stereo image – causing vocals to be buried, width is now 1.0 for pop materials. Also, more work on the bass to flatten it further. Work and rework for the pre/de-emphasis, but decided on the brighter version that appears to match commercial recordings (both FA and non-FA) better.

BUGFIXES/TECH UPDATES from V3.4.0E to V3.4.3J

Note: A lot of these improvements come from access to some pre-FA materials. These materials provided a good baseline for adjustments in the decoder. Also, a lot of recordings appear to benefit from decoding at 10 layers!!! This causes a major slowdown but decoding at 10 layers can make improvements, and is optional. The default '--fa' command is usually good enough, but noticeable improvement is available at '--f9 or --f10' instead of '--fa'.

Many accuracy adjustments made. The inter-layer gain was ludicrously in error. This caused a gain drift when using different # of layers, but additionally caused a creeping increase in calibration errors to the point where some aspects of quality were impacted, and also made fewer useful layers available.. *The accuracy of the gain between layers is now at the 0.1% level.*

Because of the need for very fine gain accuracy when decoding at 10 layers, several minor dynamics improvement have been made in the Dolby A mechanisms. The attack/release speeds are more finely controlled. During FA decoding, any slight variance from the required attack/release characteristics becomes very audible as either 'too harsh' or 'too soft' sound. The current adjustment is slightly on the side of being 'too soft', but further improvement might be done before the release. This change in the attack/release shape has little to do with the actual attack/release timing characteristics, but instead keeps the slew rate minimized to avoid higher order sideband generation.

Bass response is vastly improved, and might be too much. *Try --lfreq=1 to mitigate some of the excessive bass.* Simplified the bass EQ, enabled by earlier fixes, but didn't keep the bass EQ in sync with the improved capabilities of the MF/LF adaptation mechanism

The attack/release dynamics have been improved by a more accurate adjustment of the pre/de-emphasis. This was made much simpler and more accurate by access to the reference recordings noted above.

One more decoding layer is now available, where max number of layers was increased from 9 to 10. This might have some resource issues on slower machines, but shouldn't really be a major problem.

Useful high quality (\geq --fx) speedups by improving the 'filterpass' mechanism (this is the magical and complex gain multiplication code.) Some of these speedups are 'ate up' by the often greater number of desirable decoding layers.

Addition of potentially useful '--vinyl' mode, but the other fixes might have mitigated the usefulness. '--vinyl' is being kept just in case.

BUGFIX from V3.4.0A to V3.4.0E

Found a 'balance' problem on the LF – the wrong 'anti-distortion' parameter was used for 125Hz EQ. Fix stupid error where a part of the post-decoding EQ was missing. This should really tame the highs. Create a carefully crafted default setting for handling a slight rolloff at 21kHz and above, this can improve the smooth sound of the highs. Also, *added the '--bh' switch which does PRE decoding EQ*, mostly intended for correcting the HF rolloff on some recordings. **Using '--bh' is similar to '--pvdh', where just do: '--bh=18k,1.5' gives a 1.5dB boost BEFORE decoding at 18kHz.** Docs not ready, intended the '--bh' switch for V3.4.1 versions. Needed the '--bh' switch to get the best results for the 'classical' demos.

BUGFIX from V3.1.1Q to V3.4.0A

Open up to using 24kHz for the input EQ. This fixes a lot of HF problems, giving a good, clean sound. Finally, I think that the LF EQ is now close to accurate. This is so good that the development is moving to speed improvements, and only minor bugfixes, minor feature improvements. Don't look for major changes beyond this point for a while. Immediately, I only foresee an addition of input EQ capabilities, and minor refinement to the HF/LF EQ. The LF matters might be inadequate/excess lower bass, midrange mods. The HF is good, but might seem worse than it really is because of a revelation about '--coff' on some classical recordings.

BUGFIX from V3.1.1F to V3.1.1Q

Several critical bug fixes related to hearing troubles... Moved the upper bass EQ from 250Hz/150Hz to 250Hz/125Hz. Also, there is no other increase in LF gain below 125Hz. This increases the likelihood that the LF EQ is more likely flat. Also, found that the 2nd order EQ is also needed on the low side of LF. An additional 3X -3dB at 18.75Hz/Q=2.0 on top of the 3X +3dB at 75Hz/Q=2.0. Previously, attempts to avoid the 18.75 2nd order EQ created too strong middle bass. Also, my HF hearing is variable and strange, and the LF hearing is just as strange and variable. After exhaustive testing and retesting in different hearing conditions, there is now a better chance of the HF being correct along with the LF. The default -hfri value is now '2' instead of '1'. If you want stronger highs, try using '--hfri=1' or even '--hfri=0'. If highs are too strong, then try '--hfri=3'. A wider range of '--lfeq' is also added.

BUGFIX from V3.0.10L to V3.1.1F

Many, many bugfixes. First, most important, is an ongoing problem with the HF EQ. Those problems resulted from my own assumption that my headphones were correct, which they were not. There was an apparent -6dB or so rolloff at 9kHz and even worse at higher freqs. The EQ ended up being very hot on the high end. Further improvement for the pre-emphasis and de-emphasis, which improves the HF dynamics. Another VERY important matter – some needed EQ between 500Hz and 2.5kHz was totally missing. This new correction improved the ‘buried vocals’. A lot of apparent compatibility problems have been resolved. (REMOVED LOTS OF BUGFIX MESSAGES)

BUGFIX from V3.0.10E to V3.0.10L

A couple of problems overlooked/corrected and a true, final mitigation for the overly metallic sound. The ‘metallic’ sound finally was found to be an embarrassing pre/de-emphasis EQ error. There was way, way too much super-HF pre-emphasis, which made any compensation impossible. Also, perhaps just as important and just realized – the anti-distortion mechanism can not be used in the in-between layers EQ. The anti-distortion scheme works very well, but slight errors in my version of it can creep up when using the scheme for more than a few EQ in series. Since the in-between HF EQ is done 7 times, the small percent-level error in my implementation ends up being noticeable in the 0.5dB to 1dB level of total errors. Such errors are simply not tolerable. This series of V3.0.10 releases has been stuttered too many times, and hopefully this is the last fix for a while.

ADJUSTMENT from V3.0.10D to V3.0.10E

Still not happy with the V3.0.10D change where 2nd order EQ was used. The result was too soft/‘rounded’ kind of rolloff. Didn’t result in a ‘clean’ sound. Therefore changed the 2nd order EQ (totaled as -3dB at 21kHz) to be a single first order rolloff of -3dB at 21kHz. The resulting sound is much more clean, but the rolloff actually starts lower in frequency... Kind of odd that more of a lower frequency rolloff results in more clean, precise sound, eh? This EQ stuff is strange and sometimes counter-intuitive.

- **The default (built-in) –coff value is ‘-2’.** Most recordings appear to be happy with that setting, but NOT ALL. Some recordings, e.g. Nat King Cole Story, some classical/high quality Jazz and many other POP recordings do need ‘--coff=-4’, appears to be the prominent secondary value. *UPDATE* – some classical recordings apparently need ‘--coff=-6’. If you notice a slightly grainy sound, then using ‘--coff=-6’ might be a good thing to try.
- Some older recordings have too much bass. Try ‘--lfeq=1’ or ‘--lfeq=2’ for minor decreases for bass. Try ‘--lfeq=3’ for a profound decrease in bass.

- The `-fa` mode again has some options, needed once in a while to optimize decoding for certain recordings. The '`--fa=6` or '`-fa=7`' modification is needed much less often than before. Apparently, the more correct pre/de-emphasis helped to mitigate a bug that required some modification of the FA layering. **With the current decoder corrections, I have found NO NEED for the '`--fa=6`' or '`-fa=7`' variants of the '`--fa`' command, and this is a very good thing.**
- Sometimes, using `-xp` instead of `-fz=max` or `-fx=max` can improve quality. Much of the time, the quality will be better when using the '`--xp`' series of quality commands, and the decodes will be faster. The best practical quality can be attained by '`--xp=max`', but for people with 'Threadripper' CPUs, it might be helpful to use '`--xpp=max -dp=yes`'.
- IMPORTANT: when using a 24bit signed integer output file, it is very important to consider using '`--outgain=-3`' or even more safe '`--outgain=-6`' so that the output doesn't clip. Even when using FP output files, you might save yourself grief by using the '`--outgain=-6`' because some software doesn't handle values greater than '+1' even in FP format.
- All versions of the decoder can do the precision sensitive calculations in DP mode, thereby giving a slightly higher decoding quality in the higher quality modes (marginally noticeable.) Currently, `--dp=off` is always the default, but can be enabled or disabled on any version by using the '`--dp=on`' or '`--dp=off`' switches. Using '`--dp=on`' on AVX2 or SSE3 machines will be very significantly slower, mostly not worth it. The decoding speed will be about 30% slower when enabling double precision mode (`--dp=on`.) I usually leave DP mode when doing massive decodes, but would use DP mode very seldom, when the very best quality is desired. If you have a choice between using `-fz=max` or `-dp=on`, then `-fz=max` will make a bigger improvement in quality. Use of `-dp=on` on non-AVX512 versions of the program is too costly in most cases.
- Some recordings benefit more than expected from using a higher quality (and time consuming) decoding mode. This can be manifest because the recording might be damaged in previous DA encoding/decoding cycles, and the decoder removing more DA 'fog' than usual. Slightly changing the internal HF EQ at the highest frequencies is being considered to mitigate this problem at lower quality decoding modes.
- I know that this seems like a long, random walk – be patient, there are NO SPECS for FA. I know that there was a bass/treble heavy response balance, and I tried everything until a 10dB equalizer instead of just 1.5, 3, 6, 9 12. 10dB seems to have fixed some problems. (Each equalizer has a VERY different sound, depending on how it is used.) 9dB usually sounds very different from 10dB which sounds very different from 12dB. It is so odd that only 1dB makes such a big difference, but it really does. The use of -10dB really simplified the LF equalizer.
- Commands use integer offsets for things like calibration and stereo image – no need to remember any magic numbers
- Use the `-stw=-2,-1,0,1,2` for stereo image. The default is 0 for `-fw=classical`, and 1 for the default `-fw=wpop`. If the stereo image seems 'narrow', just try a larger `-stw` switch. I suggest that sometimes POP that needs the `-fw=classical` or `-fw=wclassical` switch, an additional `-stw=1` will be very helpful. ADD-ON: sometimes using `--stw=1.414` or `--stw=1.5` has been helpful. Some recordings that need the `-fw=classical` mode might even benefit from `-stw=2`. '`--stw`' is *VERY NEEDED VERY SELDOM*.
- More intelligent utilization of computer memory for FIR filter coefficients. (good for small machines)

- The `-pvdL` instead of `-pvl` EQ and `-pvdH` instead of `-pvh` EQ will often produce better sound quality, even though the EQ is similar. Sometimes it is critical to use the ‘normal’ `-pvh` or `-pvl` though.
- AVX512 versions appear to better utilize the advanced instructions much better than before (probably cache locality improvements), so there is approx 30% speed improvement when the AVX512 instruction set is available. Windows AVX512 version supplied, but cannot test it. Linux AVX512 is one of the primary test versions, so should work fine.
- I found that the single precision Hilbert transforms leave a small amount of distortion in the signal, so I have added a `-dp` switch. The default precision for non-AVX512 versions is ‘single precision’, and if you want the higher quality of double precision at a cost dependent on CPU type, etc – then the `-dp` switch is available to try. The default precision for AVX512 versions of the software is ‘double’ but can be disabled by `--dp=off`. Some time in the future, I’ll use a different method of choosing `-sp` or `-dp`, perhaps based on AVX2 and # of cores.

CAVEATS about this version:

- The bass has been troublesome to equalize. It seem to be perfected. *Tell me what you think... I have been getting help/opinions on this matter. The bass has been vastly improved as of V3.1.1F, but might still not be 100% correct.*
- In earlier versions, the decoder would work similarly no matter the number of layers that you specify. After the decoder has become VERY PRECISE, and the layers match almost perfectly, there is no looseness or imprecision (spread between layers) that help to hide errors. For efficiency reasons, the decoder now supports up to 7 layers, and no more.
- **Very important – the calibration (`--coff=` or `-tone=`) must be specified BEFORE the `-fa` command switch. This is needed because of the dependencies of the `-fa` calculations on the `-tone=` or `-coff=` values. The decoder WILL complain if you use the `-coff=` switch after the `-fa` switch.**
- There is a ‘focusing’ mechanism in the decoder, and the default is to be very dependent on the `-coff=` value. Some dependency can be relaxed by using the `--ss=0` switch. The `--ss` switch is not well documented, but specifies the amount of ‘focusing’ or ‘anti-distortion’ mechanism that is enabled. If the `-coff=` value isn’t accurate enough, and the built-in `-ss` value is used, sometimes the highs and sibilance can sound distorted. THERE IS NEVER A REASON TO TURN OFF THE ANTI-DISTORTION CANCELLATION/FOCUSING MECHANISM, and in fact even with `-ss=0`, it is only partially turned off. There be dragons when using modes higher than `--ss=1`, so don’t do it.

Using the decoder is a simple command line, like this:

For classical/instrumental:

```
>> da-avx -input=in.wav -overwrite -output=out.wav -info=1 -coff=-4 (or -2 or -6) -fa -fw=classical
```

For typical relatively recent pop or recent remasters:

```
>> da-avx -input=in.wav -overwrite -output=out.wav -info=1 -coff=-2 (or -4) -fa (or -f9 or --f10)
```

Alternative mode for pop material:

```
>> da-avx -input=in.wav -overwrite -output=out.wav -info=1 -coff=-2 (or -4) -fa (or -f9 or --f10) -fw=classical
```

NOTE IN THE ABOVE, the items in parenthesis are intended to show alternatives, NOT part of the command syntax...

There are variants that are a mishmash of the above, but the above are very typical

If using pipes, one can 'skip' the -input, --overwrite, --output switches, and pipe the input and output.

The input files are 16bit, 24bit and floating point .wav. The output files will be 24bit or floating point .wav. You can force 24bit output by using '--intout' switch, and force floating point by -floatout.

Sample rates can be 44.1k, 48k → 96k, 176k → 192k, 352k → 384k. Output will be 2X higher than input rate for 44.1k and 48k inputs.

THE ALL IMPORTANT FA COMMAND

The ‘--fa’ command-line switch changes the decoder mode from DA to FA modes. DA mode is intended for professionals and is almost totally useless for consumers. All of the consumer interest is in the ‘--fa’ mode.

(note that the –f9 and –f10 modes are added. This allows for more complete decoding, with small quality improvement, but slower decoding. --f10 is so severe that small errors in the decoder might make the less complete ‘--f9’ a better choice.)

The ‘-fa’ mode decoding reverses the compression that is commonly used on commodity consumer entertainment audio recordings. The decoder is NOT perfect, but has been improving until now it is very usable. We have found that not all recordings use the same EQ, but the dynamics processing (the engine of the decoding) appears to be almost or exactly the same between recordings.

Useful switches that are best remembered. Perhaps the most important of these:

- Stereo image manipulation (e.g.: --fw=classical, --stw=1, --stw=1.5, etc.)
- Real-time play (e.g.: --play or --splay)
- Run-time information display. (e.g.: --info=1, --info=3, etc.)

Simple, Generic Decoding Commands

These examples are reasonable 'starter' multi-layer decoding commands. These generally gives improved results on perhaps 2/3s (or even more) of FA material.

There are two general command-types that could be deemed 'generic' decode commands. The 'N' value is the number of layers.

(Generic command #1)

da-avx --input=infile.wav --overwrite --output=outfile.wav --info=1 -fa"

(Generic command #2)

da-avx --input=infile.wav --overwrite --output=outfile.wav --info=1 --coff=-2 -fa --fw=classical"

--coff=-4 might sound better than the default --coff=-2. You'll need '-4' if there is some gating in the result. You might even need '-6' if the sound is 'grainy' when using '--coff=-4'.

--f9 or --f10 are more than likely the most accurate mode for many recordings.

--fa=6 (sometimes --fa=7 is better) might be needed if --coff=-4 doesn't work alone. '--fa=6' can also mitigate a 'tight' almost 'crunchy' sound in vocals.

Use the --fw switch now for specifying the 'classical' stereo image. 'Pop' is the default, but there is also 'wpop' (wide pop), 'npop' (narrow pop) and 'mix' for something in between classical and pop for the stereo image. (there is also nclassical, wclassical, wmix, nmix.)

Important notes for some 'bright' sounding results

Some decoding results can lack a 'high end'. This comes from a need for post decoding EQ. Try using '--hfrl=1' or '--hfrl=2' instead of the built-in default of '--hfrl=0' first. If the highs are too intense, then try '--hfrl=3' before doing the EQ yourself. There are some variants on the theme, but most of these can be corrected by 1st order post-decoding EQ. Almost all of the necessary post-decoding EQ involves 3kHz, 6kHz, 9kHz, 12kHz, 18kHz, and 24kHz. The EQ steps are usually +- 3dB and/or +- 6dB. Sometimes the results are best if you use two +-3dB at a single frequency, sometimes a single +-6dB is best. Here are some examples (exact album names aren't listed here – some of my info isn't online):

For excessive LF, try '--lfeq=1' for minor decrease, '--lfeq=2' for a slightly larger decrease, or '--lfeq=3' for a more profound decrease in LF. Try these before doing the LF EQ yourself. The '--lfeq' command is actually much more intricate than documented here. The correct settings beyond the simple ones above haven't been determined. Therefore, if these aren't 'good enough', then MAYBE try adding '10' to the numbers above. After that, then give up with '--lfeq', and do the LF EQ yourself.

* The examples below are NOT prescriptive, but are simply examples. Other values might be better, and some 'Linda Ronstadt' values might be best for 'Carpenters' or vice versa. Most of the time, the problem is excessive LF. The magic frequencies are usually: 500Hz, 75Hz, 37.5Hz, 18.75Hz. Always use gain values like 6dB, 3dB, 1.5dB. Sometimes, -9dB is helpful. The combo of different plus and minus levels of dB at the same frequency do NOT cancel. There are very profound differences in shape between the various gains.

--pvl=500,-3 --pvl=75,-3

--pvl=75,-3 --pvl=18.75,-6

--pvl=75,-6 --pvl=75,3 --pvl=18.75,-6

--pvl=75,-9 --pvl=75,-6 --pvl=18.75,-6

You can imagine the other variants, but the first three above will get 90% of the 10% of the recordings that need help.

About 'Higher Quality' modes

About higher quality modes. More than likely, they aren't needed. Most recordings have already been damaged by NR encoding/decoding, and the normal higher quality modes only help a little. On pristine material, the improvement is tremendous. Simply add the '--fx', '--fz', '--xp' or '--xpp' switches for slightly improved quality at the cost of a LOT of CPU. The amount of CPU usage is dependent on the number of layers, and can be very very slow when using especially '--fz', '--xp' and '--xpp'. Frankly, I am not patient enough to use '--fz' unless for testing purposes.

Here are the 'higher quality' modes in order of increasing quality:

--fx, --fx=plus, --fx=highs, --fx=max, fx=opt
--fz, --fz=plus, --fz=highs, --fz=max, fz=opt

--fx, --fz, --fx=plus, --fz=plus, --fx=highs, --fz=highs, --fx=max, --fz=max, --xp, --xpp, --xp=plus, --xpp=plus, etc.

New mode: --fx=opt, --fz=opt, --xp=opt, --xpp=op... These are an ideal mix of 'max' and the default, depending on which layer, automatically.

In the comments below – every time that '--fz' is mentioned as being better, then --xp and --xpp are even better yet.

Sometimes, if it seems like a recording is very high quality, but also using '--fz', then using '--fz=opt' or '--fz=max' can sometimes bring the highest highs up to be more crystal clear. Using the higher '--fz=max' mode, or even the '--fz=opt' mode can cause a long decoding time on slower, older computers. '--fz=max' is much less unpleasant to use on newer 9th/10th/11th general Intel or most recent Ryzen multi-core computers. **If you have an 'AVX512' capable machine, and you want more quick decoding speed, then use the 'da-avx512' binary and the --dp=off switch. This combo means, take full advantage of the AVX512 capabilities of the machine, but only use AVX2 sized data. This speeds up decoding by about 30% over and above using AVX2 decoder version alone.**

Absolutely the best quality *for normal usage* is obtained by the --fz=max mode, runs full anti-MD on all layers. Sometimes better quality can be had by using --xp or --xpp instead of --fz=max mode. '--xp' and '--xpp' are likely much faster than '--fz=max'. *'--xp' is better sometimes when '--fz=max' isn't the best. For general purpose 'best', I would tend to use '--xp' nowadays, and if there is especially congested material, then perhaps --fz=highs or --xp=highs.*

* I almost always use --xp nowadays, but I have a 10 core CPU. Otherwise, try '--fz', which is NOT bad at all. Sometimes --fz=max is better than --xp because of a congested sounding recording. However, in *by far* most cases, just '--xp' is best all around.

If you just want the best 'reasonable' quality, I suggest: --fx, --fz, --xp, --fx=highs, --xpp. For the very best quality, you can use --xpp=max, but you'll need a threadripper to use that mode!!!

Bottom line, the very best quality can be had at '--xpp=max', but it is very very slow. For the 'best' sort of quality, then try: --fz, --xp, --xpp, --xp=max, --xpp=max. These should be in approximately increasing slowdown order, and equivalently increasing quality. (Sometimes, --xp=max might not be quite as good as just '--xpp'. The various advantages of using '=max' are between almost none, to noticeably increase in detail.

Other switches that you'll likely need

For this document, other than the simple examples above, we need to depend on the older, broken documentation. That IS bad, and I am planning an update soon. However, in the interim, I am listing in a simple form, the switches that you'll need. This is admittedly VERY primitive – I haven't even updated the man page yet, because I haven't chosen the commands to add and remove.

These are the switches that I normally use, and I have been working on the most difficult material for testing reasons!!!

Input/Output switches:

--input=<infile.wav>

Specifies audio file to read/decode.

--overwrite --output=<outfile.wav>

Switch combination that writes the output file.

--info=1 or --info=2 (--info=3, info=11, etc supplies some more useful details.)

Gives real-time display of program activity. I STRONGLY suggest using one of these switches.

At --info=3, the EQ used at each step, and the calibrations for each step are displayed.

--outgain=<xxdB>

This is a general purpose facility to support additional/less output gain. Since the output level can be higher than the input level, it might be necessary to use less than 0dB gain on output. This switch gives that capability. --outgain values now add together.

--ingain=<xxdB>

This is a general purpose facility to support additional/less input gain. Be very careful with this switch – it is important to maintain 0dB gain from digital source material, or the calibration numbers will be wrong. --ingain values now add together.

--fgain=<xxdB>

Final gain after all compressors/limiters/etc. Adds to --outgain when a compressor/limiter is not used.

--floatout

For greater dynamic range, the decoder can produce floating point .wav files. Some consumer programs choke on FP files though.

--intout

Force creating 24bit integer output files

--chsw

Switch output left and right channels.

--dp=off (on the AVX512 program version)

Use normal SP variables with the AVX512 capable machine, gaining about 30% faster decoding. If 'double precision' is not enabled, then there is a little less decoding accuracy. --dp=on can be used on AVX2 machines with a significant slow down in decoding.

Mode/Tuning switches:

--fa, --f9 and --f10

This changes the decoder from a decoder that can process Dolby A materials to the consumer mode. There ARE arguments to the --fa command, but since this is a quick start, I'd suggest considering only --fa, --fa=+7, or --fa=4. **NO LF EQ modes should be needed. '--f9' and '--f10' are simply more complete versions of '--fa'**

--tone=<calibration level>

This specifies the calibration level used, and is almost standardized. *You don't need to use this switch for FA decoding.*

--coff=<calibration offset>

This is the best equivalent to '--tone' when doing FA decodes. The correct values are: -2, -1, 0, 1, 2. The correct --tone level will be generated based upon the selection that you provide. The default value is '-2' and will mostly work even if incorrect.

--equalizer (--skip)

This supports running the post-decoding EQ (e.g. --pe, --pi), compression (--c1, --c2, --c3) and anti-sibilance (--as), and/or stereo image width manipulation (--wof) without doing a decoding operation. This is very useful so that the time consuming decoding might be done once, since it is easier now, then do the more tricky post decoding EQ in quicker iterations.

--hfrl=<n>, n=0,1,2,3,4,5

These are used to 'soften' the sound by 1st order EQ HF rolloff. If '--hfrl' is never used on the command line, then the same effect as '--hfrl=1' is enabled. If you want NO rolloff, then use '--hfrl=0'. Most recordings need the slight rolloff. These are preferable to the 2nd order versions, except when trying to remedy sibilance. The default value without arguments is '--hfrl=2'. Currently, the higher modes are not well documented, but you can experiment when you need stronger softening.

--lfeq=<n>, n=0,1,2,3 (or 99)

This is intended to correct older recordings with too much bass. This is a generic bass rolloff solution that usually works, but for more precise correction, just try using the '--pvl' 1st order EQ. '--lfeq=99' is a general purpose EQ for older recordings with very excessive bass.

--pt=<n>, n=1,2,3,4

--pq=<n>, n=1,2,3,4

--ps=<n>, n=1,2,3,4

These commands provide a special kind of HF eq which helps to mitigate sibilance without dynamics processing. Each of '--pt', '--pq' and 'ps' are EQ in order of 'strength'. Basically, each one in sequence does EQ at more frequencies in the 6kHz to 18kHz range. The numbers specify the #of 0.375dB increments. The defaults in each case are the <n> value of 2. Most often, if this scheme helps control sibilance, just using '--pt' with the default value is the best first choice. If the sibilance persists, then just 'go all the way' and try '--ps'. In the end, if the EQ scheme is not sufficient, just punt and use the '--as' (anti sibilance) dynamics processing.

--as=<level> (anti-sibilance)

This anti-sibilance mode is primitive but is very helpful when used correctly. It is implemented both by parallel and cascade notch filters at different frequencies. 'levels' between one and 10 are usable. '1' has the least amount of anti sibilance, but is useful. I find that the values '1.5' to '2' is usually good enough when anti sibilance can be helpful.

--cdd

Does a CD de-emphasis before the decoder.

--comment or **--comment="msg"**

Normally the decoder will not overwrite the 'ICMT' list item if it already exists. Adding the '--comment' switch with no arguments will cause the decoder to overwrite the command line information into the 'ICMT' list item, even if the item already exists in the source .wav file. Using an argument supports a manually specified comment into the 'ICMT' list item. If the 'ICMT' already exists, and the '--comment' switch was not used, then the decoder will create a nonstandard 'FADC' list item.

Mode/Tuning switches (cont'd):

--fx, --fz, --xp, --xpp (--xppp for masochists)

These are the anti-MD mode specifiers. I suggest that if you want to use an anti-MD mode, then '--final' gives good first order improvements, and '--fx' is probably all that you really need. '--fz' is really good, and really does ferret out distortions, but is CPU expensive, esp when running multiple layers. '--xp' uses even more CPU, but can make great improvements.

If you want 'max' quality, it might be best to use the 'opt' version of the switches.

There are options that support incremental improvement over the gross-level anti-MD. The --fx and --fz switches support these options:

--fx=plus	(splits the two HF bands, mitigates some intra-band MD creation)
--fx=highs	(additional splitting of the HF bands, further mitigating the intra-band MD creation)
--fx=max	(splits the MF band also, with further MD mitigation.)
--fz=plus	(splits the two HF bands, mitigates some intra-band MD creation)
--fz=highs	(additional splitting of the HF bands, further mitigating the intra-band MD creation)
--fz=max	(splits the MF band also, with further MD mitigation.)
--fz=opt	(Better optimization of CPU time and quality than --fz=max)
--fx=opt	(Better optimization of CPU time and quality than --fx=max)

(or any of the above with '--xp' or '--xpp' instead of '--fx' or '--fz'. Avoid using '--xpp' unless you have an 18 thread CPU or more.

--ss=0 through --ss=6

These allow disabling parts of the timing correction. The default is full timing correction, most of the timing correction can be turned off by '--ss=0'. Normally, no adjustment is needed. DO NOT USE HIGHER THAN '--ss=1'.

Very useful information switch:

--df

This switch means 'dump filters'. This is very important because it also shows the calibration levels (--tone= values) for each layer. It can help if loosing track of what the tone= values are...

Stereo image switches: (only --fw and --stw normally need to be used.)

--fw=<stereo-width-name>

This switch is now used in-lieu of being contained in the FA initiator set of strings. The useful options are:

classical	the stereo width is not modified when decoding. (one of the normal 'classical' music modes)
wclassical	the stereo width is only widened when decoding. (possible 'normal' mode for classical)
pop (default)	the stereo width is modified during decoding
wpop	the stereo width is modified during decoding, and the resulting width is made slightly more wide.
npop	the stereo width is modified during decoding, and the resulting width is made slightly more narrow

mix the stereo image width is a hybrid between classical & pop

--wia=<lratio>

--woa=<Oratio>

--wof=<Eratio>

the --fw switch sets these settings based upon the type of stereo image. Sometimes the values must be modified manually. The only one of these that would commonly be used: --wof. I use --wof very regularly, but will probably more often use --fw=wpop and --wf=npop in the future. --wof is operative during --equalizer mode.

--stw=<stereo image index> (this is the FA version of --wof).

There are 5 correct values for --stw. **The most usual correct and default value is '1'.** For a wider stereo image, use '2'. For a more narrow stereo image, then try a value lower than '1'. Most likely if '1' isn't a correct value, then '0' will be. Once in a LONG while, or on certain older recordings, other values might be used.

Output/post-decoding equalization switches (Secondary, limited 'mastering' capability)

These should be used mostly for 'tweaking' the sound.

For the 2nd order 'peXX' switches:

mode = 1,2,3,4,K

mode 1 – simple filter, <optQ> IS operative

mode 2 – mixed Q=0.5/0.8409, <optQ> is not needed

mode 3 – mixed Q=0.5/0.707/0.8509, <optQ> is not needed

mode 4 – mixed Q=0.5/0.8409 spread over +-250Hz, <optQ> is not needed

mode K (5) – mixed Q=0.5/0.707/0.8409 spread over +-250Hz, <optQ> is not needed

Three (3) instances of each of the –pe equalizers can now be used. Also, these are operational in –equalizer mode.

Each of the mixed modes do sound different. Sometimes mode 2 sounds better than mode 3 and vice versa. Same as mode 4 vs. mode 5 (K). It is tricky to tell beforehand which of the 2,3,4,K filters will sound best. Usually, it is best to mix the filter types if you are going to use EQ at nearby frequencies. For example, if you need an EQ at 6kHz, 9kHz and 12kHz, it might be best to use type '4' for 6kHz, type 'K' for 9kHz and type '2' for 12kHz.

The type 4 & 5 EQ aren't available at frequencies 120Hz and below. It makes sense, because one cannot create a negative Hz filter, and calculating 120Hz – 250Hz does create a negative frequency.

Very often, during decoding, a *first order* boost at 1kHz (--pi1k) is needed, and sometimes even another boost at 750Hz (--pi750), however I found that given the EQ on most recordings, *first order* filters aren't helpful at 500Hz or below. The pe375Hz filter is very very helpful to create a good bass balance, especially the –pe375=K,0.75 type filter – can be *wonderful*. For the deepest bass, a small change at –pe45=2,0.75 (or -0.75) can be helpful.

I used to use –pe1k for the needed lower MF boost on most decodes, but I found that the 1st order –pi EQ works better there. However, it is perfectly valid to mix both 1st order and 2nd order EQ at the same frequency.

Bass EQ:

--pe225=<mode>,<dBchange>,<optQ>	22.5Hz
--pe45=<mode>,<dBchange>,<optQ>	45Hz
--pe90=<mode>,<dBchange>,<optQ>	90Hz
--pe120=<mode>,<dBchange>,<optQ>	120Hz
--pe375=<mode>,<dBchange>,<optQ>	375Hz
--pe500=<mode>,<dBchange>,<optQ>	500Hz
--pe750=<mode>,<dBchange>,<optQ>	750Hz

--pe1k=<mode>,<dBchange>,<optQ> 1kHz

Treble EQ:

--pe3k=<mode>,<dBchange>,<optQ> 3kHz
--pe4p5k=<mode>,<dBchange>,<optQ> 4.5kHz
--pe6k=<mode>,<dBchange>,<optQ> 6kHz
--pe7p5k=<mode>,<dBchange>,<optQ> 7.5kHz
--pe9k=<mode>,<dBchange>,<optQ> 9kHz
--pe10p5k=<mode>,<dBchange>,<optQ> 10.5kHz
--pe12k=<mode>,<dBchange>,<optQ> 12kHz
--pe15k=<mode>,<dBchange>,<optQ> 15kHz
--pe18k=<mode>,<dBchange>,<optQ> 18kHz

Special MF EQ: (+ or – about 1.5dB between 1k & 3k), default is +1
--pemf=1 (default), **--pemf**=0 (old style), **--pemf**=-1 (for symmetry only)

1st order *variable* EQ commands
THESE ARE GENERALLY THE BEST EQ SCHEME, WHEN NEEDED

LEARNING TO USE THESE CAN BE TRICKY, BUT WELL WORTH IT!!!

These provide both 1st order both bass and treble:

HF shelf: --pvh=freq,#dB

LF shelf: --pvl=freq,#dB

Special anti-distortion shelving EQ:

HF ANTI-D shelf: --pvdh=freq,#dB

LF ANTI-D shelf: --pvdl=freq,#dB

The above shelving EQ utilize the same resources as the internal EQ required for FA decoding. These are very high precision EQ and have been vastly improved from before.

The anti-distortion shelving does some tricks to help cancel distortion that happens during the decoding process. All internal EQ uses this capability internally. The higher quality shelving is much more complex than just EQ.

**New auxiliary switches
(operational in –equalizer mode)**

DECODING EXAMPLES

The examples on the following pages are incomplete in that the entire filenames are not included. These are copied directly from two of my scripts, and I take advantage of wildcards in filenames. These are being included as exemplars that show the consistency of decoder settings.

About some of the settings...

The 'R' submode is a matter of my own choice – personally, I am irritated by intense high frequencies, so I use a 'single R' rolloff from time to time. I don't believe that such usage is canonically correct, but I do it. The rolloff is mild, starting at 18kHz.

Note that instrumentals/classical recordings take advantage of the 'T' or 'TT' submodes. In some cases 'B' or 'L' might be helpful. The 'T' and 'TT' submodes are very important mode changes. However, 'B' or 'L' might either be a matter of taste or true corrections.

The following files are very raw, and intended as a control file for creating demos. These are REALLY what I use. (The control files are used as input to create the full command line.) I have a script that creates the command line, which ends up looking something like this:

```
>> da-avx -input=in.wav -output=out.wav -info=1 -fz -tone=-55.00 -fcs="8,auto,fGg"
```