DHNRDS FA mode decoder V2.2.0F(V1)
(More & more accurate distortion cancellation)
(MICRO level change over V2.2.0A & B)

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!!!
@lucretius, @sandyk, @rando, @pkane2001, @jabbr, @skippack, @rexp.

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...

- The new command set is still trivial. GOOD NEWS – '--fa' alone should work on everything now. I have decided to disable ALL submodes and helper modes. The left over variations in recordings are best handled by the –pvdl/--pvdh and the –pe equalizers. It REALLY IS a good idea to get used to using the EQ for personal mastering purposes. The 'helper modes' are mostly just confusing. However, there is a 'M' mode which really helps EQ the bass under certain cases. Some recordings are made thin, this appears to be a STANDARD EQ in some cases.
- 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.**
- **BASS EQ PERFECTED/CORRECTED!!! AGAIN – got serious help from some contributors above.**
- More intelligent utilization of memory for FIR filter coefficients.
- The –pvdl instead of –pvl EQ and –pvdh instead of –pvh EQ will often produce better sound quality, even though the EQ is very similar.
- **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.
- Only 7 layers is now supported. Almost all recordings appear to be 7 layers, and supporting more layers adds overhead. The support for 8 or more layers has now been disabled.

- **Relative to V2.2.0A – corrected an EQ mechanism that does the anti-distortion (minor bug/imbalance.)**
- Relative to many previous versions, the Q of the 80Hz band split is set to 1.25 from 1.07. Much less sense of distortion, and some of my Dolby A master tapes reproduce the bass more similarly to a 363 and cat22. Basically, on FA vocals, you should notice less of that 'modulation' effect. FA is much much sensitive because of the 7 layers all in one.
- USAGE SUGGESTION: I found that in some classical material, it is better to use '--fw=wclassical' instead of just '--fw=classical'

CAVEATS about this version:

- The bass has been troublesome to equalize. It seem to be perfected. ***Tell me what you think...***

- 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.**

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=-2 –fa –fw=classical**

For typical relatively recent pop or recent remasters:
\>> **da-avx –input=in.wav –overwrite –output=out.wav –info=1 –coff=0** (or -1 or -2) **–fa**

Alternative mode for pop material:
\>> **da-avx –input=in.wav –overwrite –output=out.wav –info=1 –coff=0** (or -1 or -2) **–fa –fw=wclassical**

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 COMMANDS

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.

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.

With this version of the decoder, there is **NO** need for submodes or different post-decoding EQ.   You might want to do your own EQ, but that isn't part of the decoding procedure. The options below give some flexibility without needing to remember magic numbers of any kind.

'H': disables the post decoding HF EQ correction, needed for some recordings like Linda Ronstadt that sound 'dead'.   This mostly appears to be an HF boost, but is more than that.

'T', 'TT', 'TTT': do HF rolloffs: '**T' is 12kHz@-6dB**, **'TT' is 9kHz@-6dB**, and **'TTT' is 9kHz@-6dB & 12kHz@-3dB**

'M', 'MM', 'MMM": an add 'bass' convenience feature.   This appears to do a 'standard' kind of additional bass.   A few recordings benefit, but only a few. Perhaps 10% can benefit.

Usage example: to disable the post decoding HF EQ correction and disable the LF tilt: --fa=H-M.

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.)

THE AMOUNT/DEPTH OF DECODING

(read the below with the caveat that only the correct number of layers will be precisely correct.)
Stricken out to demonstrate that the information might be passively useful, but not important anymore.

The command: --fa does support arguments, but the default will most likely give the very best performance that the decoder is capable. **Do NOT normally adjust the number of layers, unless there is a very strong reason.**

Sometimes, you'll need quicker decoding, so there is a way to change the *depth* of decoding, while still cleaning up the recording a little.

~~You can either partially decode a recording or fully decode it.* To more completely decode most recordings, you would use the '+' version of decoding. For partial decoding, just omit the '+' sign.~~

> ~~* **The current version of the decoder does VERY POORLY on partial decodes.** Before, I had tuned the decoder (incorrectly) to produce similar results for different layers. The new versions work much more precisely, and therefore, you REALLY want to use the correct number of layers!!!~~

~~For partial decoding, I suggest a command like: --fa=+5, or for more complete decoding, --fa=+7 appears normally to be full decoding, but the decoder can do up to '+9'. The '+' version does a decode with layers with sequences like: '5/2', while the partial decode with 6 would do a set of sequences like '4/2/2'. It appears on many pop recordings, '+7' seems to be the optimal number of layers to decode, and full, wide dynamic instrumentals might benefit from '+9'.~~

~~**I suggest using the defaults # of layers until you understand what is going on (further documentation is coming.)**~~

~~If you want to do a quicker decode, I suggest something like a '+5' type decoding. Explaining the arguments for the depth of decoding is not important here, but here are good suggestions:~~

~~--fa=+7 (the default, and probably best)~~
~~--fa=+5 (will still leave some compression)~~
~~--fa=4 (will leave compression and soft sound)~~

~~You can specify depths (layers) between +1 to +9, where the bigger number is slower. There is also decoding without the ‘+’, but it will not fully decode, no matter the number. A good try for non-’+’ decodes – try ‘6’.~~

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”

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.)

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', or '--fz' 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'. 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

New mode: --fx=opt, --fz=opt…   These are an ideal mix of ‘max’ and the default, depending on which layer, automatically.

Absolutely the best quality *for normal usage* is obtained by the –fz=max mode, runs full anti-MD on all layers.

There are even higher modes, DO NOT TRY TO USE THEM (--xpp=max or somesuch), unless you have more than 20 cores.   I suggest using “–fx”, “--fz” or “--fx=opt” or “–fz=opt”, in this order of decreasing speed.   If you are intrepid, only running 6 layers or less, and have a very fast computer, then you might try ‘--xpp=max’ or just ‘--xpp’ alone.   The higher modes are generally more beneficial for layers > 4.   For 4 layers, --fx is probably good enough to clean up the highs a little bit.

> *** The higher quality modes like –xp and --xpp are intended for Dolby A decoding ONLY.**   You’ll be happiest with –fx and –fz modes, trust me!!!

I have a computer with 10 cores, and –fz=max is just barely tolerable, but I avoid it. If you need to clean the highs, then try –fz=plus. If you just want a generally more clean result than –fz (my own default), then use –fz=opt.

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 listening 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.

Mode/Tuning switches:

--fa
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.**
.

***--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 '0' 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 interations.

Pre-decoding EQ switches: (pEQ)

These switches are mostly useful when NOT in auto mode. **You probably don't normally need to use these.** Perhaps the –b18 or –b15 series of switches might be helpful on recordings that have an HF rolloff, but unlikely. **Most useful pre-decoding EQ selections are built-in to the –fcs command complex now.**

Using these switches are normally an integral part of FA decoding when NOT in auto mode, and the –b9k version is usually the minimum necessary. The –b18k version is especially useful for compensating the loss of HF from legacy HW. Normally, using the auto mode will mitigate most of the usage of these switches, but -b15k and -b18k are still useful for bandwidth compensation. This form of EQ is NOT operational in '--equalizer' mode.

--b3k6k, --b3k9k, --b3k18k, --b3k22k, --b3kxxk
--b9k18k, --b9k22k, --b9kxxk
--b12k18k, --b12k22k, --b12kxxk
--b15k18k, --b15k22k, --b15kxxk
--b18k22k, --b18kxxk

> These 1st order shelving boost EQ are placed immediately before each layer where you specify them. If in auto mode, then will automatically be placed at the beginning of each sequence of layers. The normal usage is for beginning of each set of layers like this for 6 layers:
>
> > <pEQ> <calibration -44.5> <calibration -34.5> <calibration -24.5> <calibration -14.5>
> > <pEQ> <calibration -54.5> <calibration -44.5>
>
> (Note: when you see something like –b9kxxk, that means that the 1st order shelving boost goes from 9k to near-inf, which happens to be 60k)

--b3k, --b9k, --b12k, --b15k, --b18k

> These are the same as the fully specified <pEQ> as above, but the decoder automatically selects the best version of each <pEQ>. The choice is based upon the type of <FA initiator>. For fce, the correct version is <--bYYk18k>, for fcf, the correct version is <--bYYk22k>, then for fcc, the correct version is <--bYYkxxk>.

--cde

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.

--wide=<bw-mult>

Specifies the bandwidth of the processing. This bandwidth is the ‘bw-mult’ times the internal sample rate. Normally, for 44.1kHz input, the internal sample rate is 66.15kHz. So, if the bw-mult is specified to be 0.30 on a normal CD file input, then the bandwidth will be 66.15kHz * 0.30 or 19.845kHz.

--**maxfreq**=<bandwidth>

Specifies the bandwidth of processing in ‘Hz.’ There freq response can never be higher than about 42kHz. The actual maximum frequency response can never be greater than ‘internal clock rate’ * 0.45. This would be about 41-42kHz for 96k based frequency sources, and 38kHz for 88.2kHz based frequency sources. This is useful for dealing with noisy SACDs where there is a lot of hash above 24kHz.

Mode/Tuning switches (cont'd):

--fx, **--fz**

These are the anti-MD mode specifiers. The old specifiers are still available and theoretically provide higher quality at GREAT cost of CPU. 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 VERY VERY CPU expensive, esp when running multiple layers.

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)

--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.

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 nornally need to be used.)

--fw=<stereo-width-name>

This swtich 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, not for decoding per-se.

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)

Simple 1^{st} order pi1k (bass), pi3k (treble), pi6k (treble), pi9k (treble), pi12k switches

These used to be necessary for accurate decoding. Now, these might be useful for tweaking the sound. These filters below can be used multiple times on a command line, and will sum up to the total number (- and + cancel, just like any math operations.)

These are also all operational in –equalizer mode.

--pi500=<0 to 3>	500Hz 3dB BASS boost	(For recordings that appear ‘thin’ sounding)
--pi750=<0 to 3>	750Hz 3dB BASS boost	(For recordings that appear ‘thin’ sounding)
--pi1k=<0 to 3>	1kHz 3dB BASS boost	(Needed on most decoding operations)
--pi2k=< -6 to 6>	2kHz to **9kHz** TREBLE cut or boost	
--pix2k=< -6 to 6>	2kHz to 21.5kHz TREBLE cut or boost	
--pif2k=< -6 to 6>	2kHz to 100kHz TREBLE cut or boost	
--pi2p25k=< -6 to 6>	2.25kHz to **9kHz** TREBLE cut or boost	
--pix2p25k=< -6 to 6>	2.25kHz to 21.5kHz TREBLE cut or boost	
--pif2p25k=< -6 to 6>	2.25kHz to 100kHz TREBLE cut or boost	
--pi2p5k=< -6 to 6>	2.5kHz to **9kHz** TREBLE cut or boost	
--pix2p5k=< -6 to 6>	2.5kHz to 21.5kHz TREBLE cut or boost	
--pif2p5k=< -6 to 3>	2.5kHz to 100kHz TREBLE cut or boost	
--pi2p75k=< -6 to 6>	2.75kHz to **9kHz** TREBLE cut or boost	
--pix2p75k=< -6 to 6>	2.75kHz to 21.5kHz TREBLE cut or boost	
--pif2p75k=< -6 to 6>	2.75kHz to 100kHz TREBLE cut or boost	
--pi3k=< -6 to 6>	3kHz to **9kHz** TREBLE cut or boost	
--pix3k=< -6 to 6>	3kHz to 21.5kHz TREBLE cut or boost	
--pif3k=< -6 to 6>	3kHz to 100kHz TREBLE cut or boost	
--pi3p25k=< -6 to 6>	3.25kHz to **9kHz** TREBLE cut or boost	
--pix3p25k=< -6 to 6>	3.25kHz to 21.5kHz TREBLE cut or boost	
--pif3p25k=< -6 to 6>	3.25kHz to 100kHz TREBLE cut or boost	
--pi4p5k=< -6 to 6>	4.5kHz to **9kHz** TREBLE cut or boost	
--pix4p5k=< -6 to 6>	4.5kHz to 21.5kHz TREBLE cut or boost	
--pif4p5k=< -6 to 6>	4.5kHz to 100kHz TREBLE cut or boost	
--pi6k=< -6 to 6>	6kHz to **9kHz** TREBLE cut or boost	
--pix6k=< -6 to 6>	6kHz to 21.5kHz TREBLE cut or boost	
--pif6k=< -6 to 6>	6kHz to 100kHz TREBLE cut or boost	

--pi7p5k=<-6 to 6>	7.5kHz to **9kHz** TREBLE cut or boost
--pix7p5k=< -6 to 6>	7.5kHz to 21.5kHz TREBLE cut or boost
--pif7p5k=< -6 to 6>	7.5kHz to 100kHz TREBLE cut or boost
--pi9k=< -6 to 6>	9kHz to **18kHz** TREBLE cut or boost
--pix9k=< -6 to 6>	9kHz to 21.5kHz TREBLE cut or boost
--pif9k=< -6 to 6>	9kHz to 100kHz TREBLE cut or boost
--pi10p5k=<-6 to 6>	10.5kHz to **9kHz** TREBLE cut or boost
--pix10p5k=< -6 to 6>	10.5kHz to 21.5kHz TREBLE cut or boost
--pif10p5k=< -6 to 6>	10.5kHz to 100kHz TREBLE cut or boost
--pi12k=< -6 to 6>	12kHz to **18kHz** TREBLE cut or boost
--pix12k=< -6 to 6>	12kHz to 21.5kHz TREBLE cut or boost
--pif12k=< -6 to 6>	12kHz to 100kHz TREBLE cut or boost
--pi13p5k=<-6 to 6>	13.5kHz to **9kHz** TREBLE cut or boost
--pix13p5k=< -6 to 6>	13.5kHz to 21.5kHz TREBLE cut or boost
--pif13p5k=< -6 to 6>	13.5kHz to 100kHz TREBLE cut or boost
--pi15k=< -6 to 6>	15kHz to **18kHz** TREBLE cut or boost
--pix15k=< -6 to 6>	15kHz to 21.5kHz TREBLE cut or boost
--pif15k=< -6 to 6>	15kHz to 100kHz TREBLE cut or boost
--pi18k=< -6 to 6>	18kHz to **18kHz** TREBLE cut or boost (noop)
--pix18k=< -6 to 6>	18kHz to 21.5kHz TREBLE cut or boost
--pif18k=< -6 to 6>	18kHz to 100kHz TREBLE cut or boost
--pi1k=<0 to 3>	1kHz 3dB BASS boost

New 1st order *variable* EQ commands

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)
(--limiter and compressors will likely be removed – --as will likely stay)

--limiter=-NN.NN (dB)

This limiter switch attempts to limit the output to the specified level. The behavior is not quite accurate (perhaps +-1.0dB) because of the simplified anti-distortion DA style detector. The default value is equivalent to using -3dB. The actual limiting will be at a dB or so higher than what you specify. The default -9dB should be *relatively* safe for 0dB clipping, but no guarantee. This function now works very well. .

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

This anti-sibilance mode is primitive but is insidiously subtle when used correctly. It is implemented both by parallel and cascade notch filters at different frequencies. This is a ‘soft’ control, and a good default value to use might be 5 to 15. Most often, I use –as=9.5, which often gives a good balance. You might have to adjust the setting, and is mostly useful for very loud sibilance, and is also very effective against ‘fake sibilance’ where they used an approx 6kHz peaking filter to brighten the vocals. Be careful, because the setting is not effective over a wide dB range, and also can create a weird kind of dynamic compression at high frequencies. REALLY WORKS WELL!!!!

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”

DEMO SETTINGS FOR MORE COMPLETE ‘DECODES’

```
/music/Simon*/Disco*/Book*/10* --coff=-1 --fa
/music/Simon*/Disco*/Book*/11* --coff=-1 --fa
/music/Simon*/Disco*/Wed*/*Sound* --coff=-1 --fa
/music/*FLAC/1971*/05*  --coff=0 --fa --stw=1
/music/*FLAC/1971*/06*  --coff=0 --fa --stw=1
/music/*FLAC/1971*/09*  --coff=0 --fa --stw=1
/music/*FLAC/1975*/04*  --coff=0 --fa --stw=1
/music/*FLAC/1976*/01* --coff=0 --fa --stw=1
/music/*FLAC/1970*/04* --coff=-1 --fw=classical --stw=1
/music/*FLAC/1970*/06* --coff=-1 --fw=classical --stw=1
/music/*FLAC/1970*/08* --coff=-1 --fw=classical --stw=1
/music/Elton/*1/04*  --coff=-1 --fa
/music/Elton/*1/05*  --coff=-1 --fa
/music/Elton/*2/01*  --coff=-1 --fa
/music/Elton/*2/12*  --coff=-1 --fa
/music/Petula/SinglesP1/01* --fa
/music/Nat/*ry/*1/01*  --coff=-1 --fa --stw=1.414
/music/Nat/*ry/*1/11*  --coff=-1 --fa --stw=1.414
/music/Nat/*ry/*1/15*  --coff=-1 --fa --stw=1.414
/music/olivia/CD1/*05*  --coff=-1  --fa
/music/olivia/CD2/*05*  --coff=-1  --fa
/music/olivia/CD2/*14*  --coff=-1  --fa
/music/olivia/CD2/*23*  --coff=-1  --fa
/music/xmusic/Al*S*/1976*/09* --coff=0 --fa
/music/Herb/H*/04*  --coff=-1 --fa
/music/Herb/H*/07*  --coff=-1 --fa
/music/Herb/S*/02*  --coff=-1 --fa
/music/Carly/old/02* --coff=1 --fa
/music/Carly/old/05* --coff=1 --fa
/music/ABBA/j/ABBA/*01* --coff=-1 --fa
/music/ABBA/j/ABBA/*02* --coff=-1 --fa
/music/ABBA/disco/Waterloo1988/*12* --coff=-1  --fa
/music/Bread/*02* --coff=-1 --fa
/music/xmusic/*Mason*/07*Gas*.flac  --coff=-2 --fa
/music/London/CD1/08* --coff=-2 --fa --fw=classical
/music/Linda/1977*/*06*  --coff=-2 --fa
/music/Linda/orig*/*07*  --coff=-2 --fa
/music/Anne*/15*/01* --coff=-1  --fa --fw=classical
/music/Anne*/15*/10* --coff=-1  --fa --fw=classical
```

DEMO SETTINGS for DEFAULT PARTIAL RECORDING CLEANSING ONLY

```
/music/Simon*/Disco*/Book*/10* --fc
/music/Simon*/Disco*/Book*/11* --fc
/music/Simon*/Disco*/Wed*/*Sound* --fc
/music/*FLAC/1971*/05*  --coff=2 --fc --stw=2
/music/*FLAC/1971*/06*  --coff=2 --fc --stw=2
/music/*FLAC/1971*/09*  --coff=2 --fc --stw=2
/music/*FLAC/1975*/04*  --coff=2 --fc --stw=2
/music/*FLAC/1970*/04* --coff=2 --ff --fw=classical --stw=2
/music/*FLAC/1970*/06* --coff=2 --ff --fw=classical --stw=2
/music/*FLAC/1970*/08* --coff=2 --ff --fw=classical --stw=2
/music/Elton/*1/04* --fc --stw=2
/music/Elton/*1/05* --fc --stw=2
/music/Elton/*2/01* --fc --stw=2
/music/Elton/*2/12* --fc --stw=2
/music/Nat/*ry/*1/01*  --coff=-1 --fd
/music/Nat/*ry/*1/15*  --coff=-1 --fd
/music/olivia/CD1/*19* --fc --fw=classical
/music/olivia/CD1/*05* --fc --fw=classical
/music/olivia/CD2/*05* --fc --fw=classical
/music/olivia/CD2/*14* --fc --fw=classical
/music/olivia/CD2/*23* --fc --fw=classical
/music/Herb/H*/04*  --fd
/music/Herb/H*/07*  --fd
/music/Herb/S*/02*  --coff=-2 --fd --fw=classical --stw=2
/music/Carly/old/02* --coff=-2 --fc --fw=mix --stw=-1
/music/Carly/old/05* --coff=-2 --fc --stw=-1
/music/Carly/old/11* --coff=-2 --fc --stw=-1
/music/ABBA/j/TheVi*/*07* --coff=1  --fb
/music/ABBA/j/ABBA/*01* --coff=1 --fb
/music/ABBA/j/ABBA/*02* --coff=1 --fb
/music/ABBA/disco/Waterloo1988/*02*  --coff=1 --fb
/music/ABBA/disco/Waterloo1988/*12*  --coff=1 --fb
/music/ABBA/j/RingRing/*08* --coff=1 --fb
/music/xmusic/Yes*/*1971/*01* --fc
/music/tay/06*   --fc
/music/xmusic/*Mason*/07*Gas*.flac  --fc  --fw=classical
/music/London/CD1/08* --coff=-2  --fft --fw=classical
/music/Linda/1977*/*06*  --coff=-2 --ff
/music/Linda/orig*/*07*  --coff=-2 --ff
/music/Anne*/15*/01*  --coff=-2 --fc --fw=classical
/music/Anne*/15*/10*  --coff=-2 --fc --fw=classical
```