Blackbox Logging And Usage

Black Box Logger

RCGroups Thread with links to OpenLogger and firmware: https://www.rcgroups.com/forums/showthread.php?2299805-Blackbox-flight-data-recorder-feature-for-Baseflight-Cleanflight

Notion: for 4.0 and later version, OpenLogger is not recommended.

Black Box OpenLager

This has an SPI interface for faster logging (same as when the SD card is integrated onto an FC board). https://github.com/d-ronin/openlager/wiki Available from: https://store.myairbot.com/accessory/openlager.html and: http://www.readytoflyquads.com/high-rate-f4-data-logger-openlager

BlackBox Viewer

The Latest Viewer and source is here: https://github.com/betaflight/blackbox-log-viewer

Videos and tutorials

Joshua Bardwell just started a new video series which he calls BlackBox 101: https://www.youtube.com/playlist?list=PLwoDb7WF6c8l0-ABslcnJt1FyhX9MBoVW

RCG threads on learning to analyze BB logs:

Joshua's Blackbox Log Video Responses Blackbox log analyzation\help thread

Notes on new Black Box Viewer

- If you open a Betaflight Log, then the logo shown is the Betaflight Logo and the colour scheme is orange;

- If you open an iNav log, then the logo changes to iNav and the colour scheme is blue;
- If you open a Cleanflight log (or if it can't tell what kind of firmware you were running e.g. an old version of Betaflight perhaps) then the logo shows the Cleanflight logo and the colour scheme is green....

Some features are automatically disabled depending upon the firmware you are running that creates the log.

Do click on all the buttons to learn what they do and '?' for the Keyboard Short cuts.

Betaflight blackbox explorer Version 3.7

Blackbox Explorer MinMax control Blackbox Explorer: Power spectral density spectrums charts Blackbox Explorer: Power spectral density spectrums comparison

How to obtain data to evaluate noise frequency for setting the Notch filter.

Super simple visual explanation of the gyro data sequence through the filters: raw gyro->(debug gyro here)->soft lpf->(debug notch here)->notch1->notch2

Betaflight Version 3.4

As a part of the filtering overhaul, the names of the debug modes available to log filtering / tuning data have been improved NOTCH (gyro data after scaling, before filtering) is now GYRO_SCALED, GYRO (gyro data after all filtering has been applied) is now GYRO_FILTERED

e.g `set debug_mode = gyro_scaled` or `set debug_mode = gyro_filtered`

Changes were made in BetaFlight 3.0 & 3.1 along with a newer BB Viewer (see debug_mode on the 3.1.x release notes).

1. CLI DEBUG_MODE now can be GYRO or NOTCH. This will log all three axis but only for Pre-LPF or Pre-Notch Filter.
2. Still Add a Custom Graph then select the debug Pre-filter.
3. Analysis is the same.

BetaFlight Ver2.X

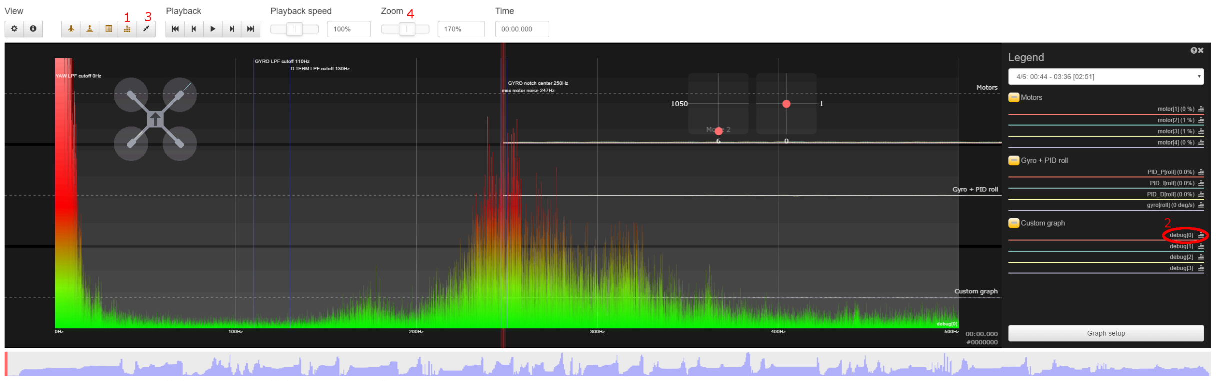
1. Use this CLI setting: `set debug_mode = notch` Make sure your blackbox logging rate is at least 1khz. The logging rate is based on pid-loop so 1/4 for 4k pid loop

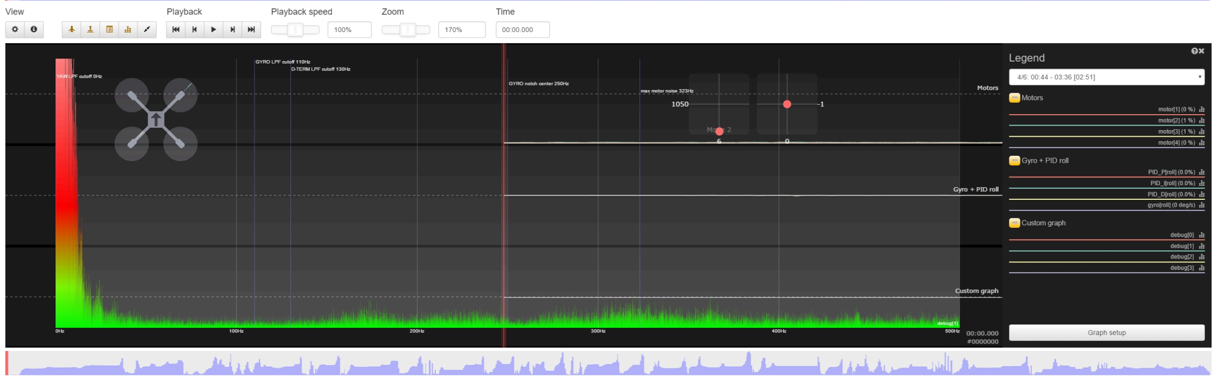
would be enough.
2. Fly as usual
3. Open your log in blackbox viewer
4. Add all debug options to your graph setup
5. Click on debug[0] Note: This action of clicking on the graph section traces to the right will show the analyzer screen :D
6. Make the graph fullscreen (next to playback options)
7. You can now see where your motor noise is most significant

The debug setting will log additional data to debug[0]-debug[3]:

- debug[0] is unfiltered and raw gyro data on roll axis.
- debug[1] is only notch filtered gyro data on roll axis.
- debug[2] is unfiltered and raw gyro data on pitch axis.
- debug[3] is only notch filtered gyro data on pitch axis.

Two images showing how to view the spectrum and the result of the notch filter.





If the notch filter is disabled 0/1 and 2/3 will be identical. Otherwise you can directly see what the filter does. More details on phase shift for example can be found here: https://github.com/betaflight/betaflight/pull/668

Post by r.a.v. You can view earlier BB logs (pre BF 3.0) but the analyzer won't know at which rate it was recorded. This will probably result in a correct looking spectrum but with incorrect frequency labels. Clicking on the traces on the Graph section to the right will show the analyzer screen.

Post by ctzsnooze: Remember the spectrum is a *relative* comparator of the frequencies in the noise spectrum. It won't tell you about the absolute amount of noise. It can only be compared to other traces, and even then only when

- A fixed and comparable amount of vertical gain is set
- The horizontal scale is set to be comparable to other traces
- *most important* the duration of the data to be analyzed is fixed. ie 30s. use the i and o keys to select a 30s realm for example.

The most important thing is to scroll through and look at the motors traces for how big that noise is. The spectrum is useful for analyzing what you see there, but the motor trace and how it looks is what matters. See Gyro and Filters for more info.

The Spectral data is accumulative which means the longer the time frame of a log the higher the noise in the spectra;l display will be. This makes it impossible to compare the noise from two logs if they are different lengths of time. Therefore, you must compare spectras of same length - say 30 sec on each log, else it has zero meaning. To do it use "I" and "O" keys to mark start and end for analysis.

So by disabling all notches you mean, gyro_notch_1, gyro_notch_2, AND dterm_notch... correct?

Won't set debug_mode=notch do the same thing or am I misunderstanding? Answer from ctzsnooze: The filtering sequence for P is gyro lpf -> gyro notch 1 -> gyro notch 2 -> P. For D it is gyro lpf -> gyro notch 1 -> gyro notch 2 -> Dterm LPF -> D notch -> D

Disabling all three notch filters means that they are gone. The gyro data goes into the PID loop without any attenuation from the notches. Frequencies that might otherwise have been filtered out by the notches will now be amplified by the PID loop. What you see in the gyro, P, D and motor spectrums is how your quad actually behaves when they aren't there at all. Set debug_mode = notch keeps the notch filters active, but records a version of the gyro trace from just after the gyro LPF - before the notch filters are implemented. The P, D and motor traces will be affected by the notches, because they are still active. Because they remain active, PID related noise amplification at those frequencies will be attenuated, reducing the amount of noise in those regions even in the pre-notch gyro trace. It just doesn't show what it would be like with no notch filters. Disabling a notch filter completely is the only way to really know what the noise state of the machine is like without it. Set debug_mode = notch isn't nearly as useful, that's why I don't recommend using it for this purpose.

Tuning Tips using BB logs

ctzsnooze:

Clicking the little 'i' at top left of the blackbox viewer, shows all your settings. On that quad, both gyro and D lowpass are PT1 (not what you thought it was). I'm suggesting putting the D lowpass to BiQuad and cutting its low pass to 70Hz. You could, as R.A.V. suggests, leave it at 100, but I doubt you'd notice any difference, and lower would be better in your case.

Flight performance generally does not deteriorate very much if you filter D more heavily; all that happens is that D sort of disappears from high frequencies. This is good. On the other hand, it's best to try and filter gyro/P the least possible.

To get an idea of whether overall noise is a problem, look at the motor traces. Less than 1-2% noise is great, 5% marginal, any more than that is not so good.

To find out what your gyro filters are doing, set the spectrum to the P trace. P trace includes the effect of gyro lowpass and both gyro notches.

To find out what your D filters are doing, set the spectrum to the D trace. Remember that first the gyro low-pass and the gyro notch filters are applied to the gyro data, then D is calculated (amplifying the fast noise), then the D filters are applied. So if you apply a notch to gyro, you probably won't find anything useful by applying the same filter again to D.

To directly compare the overall contribution of P compared to D, leave the spectrum scaling alone and click on P then D then P then D. The relative height of the peaks tells you which of P or D is contributing most tot he noise.

I don't change the slider positions from default.

Note that the spectrum is calculated, by default, from the entire log. Longer logs will make a 'taller' spectrum than shorter logs because there is more data to build the spectrum from. To directly compare spectrums log to log, always select the same amount of time. To do that, scroll to a start point, press 'i', then scroll to an end point, press 'o'. Now the spectrum is created only from the data in the smaller selection. This allows direct comparison of equal time length bits of different logs. Very short selections can also look at small bits eg just a short full throttle period to analyze that period alone.

Also in the graph setup, be sure to set smoothing to zero and expo to 100 on all traces.

After that, you can save the graph setup by pressing shift-1. Then pressing 1 will recall it. Same for all the number keys on your keyboard. This is very useful, you can save different setups for roll, pitch and yaw separately, and recall them quickly, with smoothing and expo as you set it.

ctzsnooze comments on a log posted:

PS seems to me like a bit more D on roll might be better. Also the log shows that the motors seem a bit slow to develop the requested amount of thrust. Lighter props might give better handling. mtfinger22: Can you explain what you are looking at in the logs that show that? I'm currently looking at different props for my recent build, and I would like to have an educated guess on what I should do without buying 10 different sets of props and trying them all out. ctzsnooze: The time taken between the RC input signal and the resulting gyro change when doing a fast roll or flip tells you how long it took your motors to turn the quad. In the log posted, the motors go to full 100% signal to turn it at the requested speed, but then it overshoots and the reverse pair have to speed up to slow down the overshoot. And there is a fair amount of delay, overall, between RC input and achieved actual roll. That kind of stuff implies either relatively weak motor power compared to rotational inertia, or over-propped motors for their kv, or battery current limitation, or heavy props, or any combination of those things. Light weight props allow motors to spool up faster and generate thrust more quickly, so the delay becomes less, and the amount of overshoot becomes less as well. Also if you look at the absolute height of your P and D traces (make sure you have no expo on the PID traces in blackbox), you can see that D is relatively small compared to P. That means it is relatively ineffective in controlling your P overshoot. As a very rough 'rule of thumb', a good 'ratio' is D about half the amplitude of P during roll stops. Ensure you don't have expo on the blackbox P and D traces, or comparing heights of P and D is meaningless! You could add more D, which most likely would allow more P, making for a crisper overall response without any more overshoot. Additionally, if your D setpoint weight was set above 1, D will drive the motors harder at the initial part of a stick input, which may help a little in scenarios like this. I have quite a few underpowered quads like this. They need a lot of P and a lot of D to handle as crisp as the more powerful ones, but with tuning they fly great, though they cannot control propwash nearly as well. With a more sensitive throttle they can actually feel much the same in ordinary flight, they just don't go so fast at the top end. These questions are really just basic P and D tuning questions not really betaflight related.

Youtube

Joshua Bardwell

Ivan Efimov

Community

Discord

IntoFPV

Links

Oscar Liang

VitroidFPV

Feeds

RSS

Atom

JSON