**Wraith32 - 32bit BLHELI ESC (35A)**

Features:

Build in Current sensor

On-Board RGB LED

2-6s LIPO input

35A designed

15\*30mm

------------------------------------------

**Wraith32 Plus - 32bit BLHELI ESC (50A)**

Features:

Build in Current sensor

On-Board RGB LED

2-6s LIPO input

50A designed

20\*35mm

------------------------------------------

**Wraith32 MINI- 32bit BLHELI ESC (25A)**

Features:

2-6s LIPO input

25A designed

10\*20mm

BLHeli\_32 is the third generation BLHeli code, following base BLHeli and BLHeli\_S.

BLHeli\_32 runs on an ARM 32bit MCU, initially it will be on a Cortex-M0 running at 48MHz, but there are MCUs out there that can run a lot faster.

So what can the increased MCU speed do?
First of all, it can run input signals with lower latency at faster rates.
Dshot up to at least Dshot1200, and at rates up to at least 32kHz is now supported.

Secondly it allows packing of more functionality.
- Like programmable pwm frequency, up to 48kHz, that can run motors even smoother, and also allows for moving of small but potentially disturbing humps in the throttle response. All ESCs have these bumps, with BLHeli\_32 they can now be moved in the rpm range, to a place where the system has low sensitivity to them.

- Like auto timing. Which is not the option to choose if you’re after max power. But if you want the most efficient running with very good robustness against desync, it is the perfect choice.

- Like voltage and current limiting. Which is implemented in the code, and will be supported on some ESCs. Voltage limiting is desirable for fixed wing crafts. And current limiting adds protection against ESC failure.

- Like programmable brake on stop force. Which allows control of the braking of fixed wing props.

- Like improved direction change in bidirectional mode.

- Like more to come in the future. Such as setup from the FC by using Dshot commands. Such as telemetry functionality. There is room to grow performance and functionality.