

# Cyphal/DroneCAN application

This repo introduces a ready-to-use Cyphal/DroneCAN application for [RL Mini v2](#) and [RL Mini v3](#) nodes.

Please, refer Wiki for details:

- [Cyphal template application](#),
- [DroneCAN template application](#).

## 1. KEY FEATURES

- ☒ Both Cyphal/CAN and DroneCAN protocols or one of them in a single firmware,
- ☒ You can try the software in SITL mode (don't need to have any hardware),
- ☒ Small firmware image and software is friendly for customization,
- ☒ Redundant CAN interface (node v3),
- ☒ MPU-9250 IMU (node v3).
- ☒ CAN terminator resistors control (node v3).

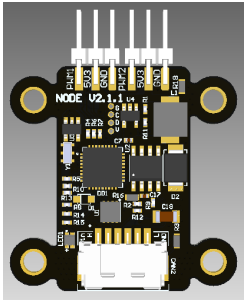
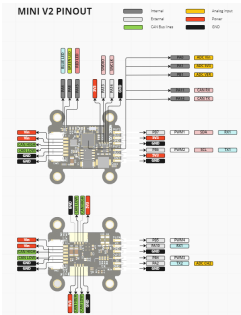
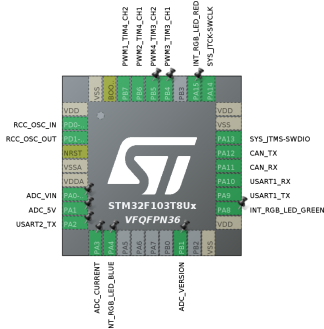
Not supported or tested yet / In Roadmap:

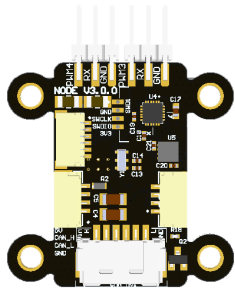
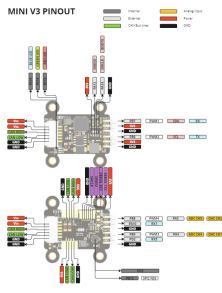

- ☐ Cyphal/DroneCAN Bootloader,
- ☐ Dynamics Node Allocation (DNA),
- ☐ Vibration Analysis: estimate dominant frequency and magnitude of vibrations for diagnosing mechanical issues or ensuring smooth operation in drones or robotic platforms in real time,

## 2. SUPPORTED HARDWARE

The software supports RL v2 nodes (stm32f103, 128 KBytes flash) and v3 nodes (stm32g0b1, 512 KBytes flash).

PROF

Name	View	Pinout	STM32CubeMX
Mini v2 stm32f103 128 KBytes flash			

Name	View	Pinout	STM32CubeMX
Mini v3 stm32fg0b1 512 KBytes flash			

The following table outlines the compatibility of the project with different operating systems, modes, and support timelines.

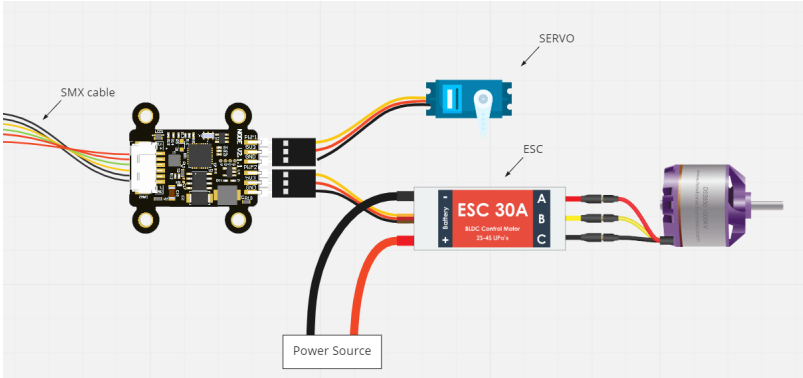
Operating System	DroneCAN Support	Cyphal Support	Notes
Ubuntu 24.04	✔ Supported	✔ Supported	Latest Ubuntu LTS version.
Ubuntu 22.04	✔ Supported	✔ Supported	Fully supported for both modes.
Ubuntu 20.04	✔ Supported	✘ Not Supported	Will be deprecated in April, 2026.
Windows (2022)	✔ Supported	✔ Supported	Current "latest" Windows version.

### 3. USE CASES

This node is a versatile device integrating PWM control, IMU sensing, and dual CAN bus communication, suitable for various applications in robotics, drones, and other embedded systems. Below are the detailed use cases for different scenarios.

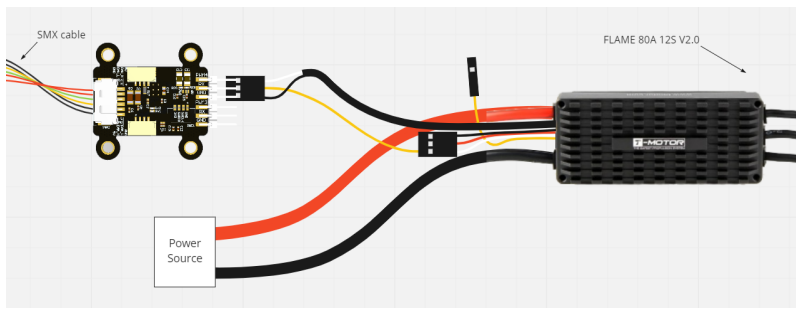
#### 1. Basic ESC and Servo Control

- Use PWM1-4 to control ESC via Cyphal or DroneCAN esc.RawCommand or actuator.ArrayCommand messages.
- Connect standard 5V servos to PWM1-PWM2 for controlling servo motors. Ideal for use cases requiring precise angular movement, such as robotic arms or UAV control surfaces.



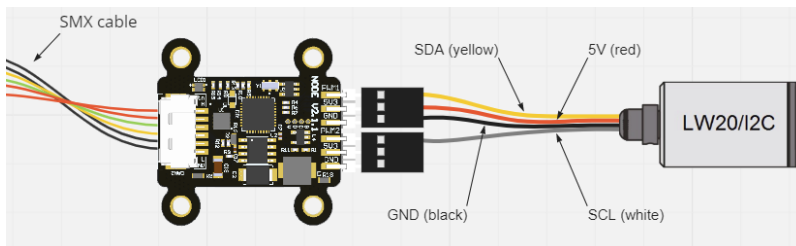
#### 2. Advanced ESC Control with Feedback

- Utilize PWM3 or PWM4 for controlling ESCs with UART or ADC feedback on the FB pin.



### 3. CAN-I2C Converter

- Configure PWM1 and PWM2 as I2C pins to act as a CAN-I2C converter.
- Use the node as an adapter to connect I2C peripherals like rangefinders or environmental sensors to a CAN-based network.



### 4. Cargo Gripper

- Use PWM1 or PWM2 to control servo-based cargo grippers via DroneCAN hardpoint.Command messages.
- Ideal for UAV payload management systems, enabling secure and precise handling of cargo.

### 5. Dual CAN Bus Interface for Protocol Conversion

- Leverage the node's dual CAN interfaces (CAN1 and CAN2) to convert between different CAN protocols.
- For example, connect CAN1 to an autopilot using DroneCAN or Cyphal/CAN, and CAN2 to a motor or servo using canopen or another protocol.
- This enables seamless integration between devices using different CAN standards, making it ideal for systems where components from various manufacturers need to communicate.

Supported only on Mini v3 node

### 6. Servo Position Estimation with I2C Encoder

- Connect a servo motor to PWM3 or PWM4 and an I2C encoder to PWM1/SDA and PWM2/SDC to estimate the angular position of the servo.
- Consider [AS5600 I2C sensor](#) encoder.

An example of this feature is not included in the repository yet...

### 7. Vibration Analysis with IMU

- Utilize the onboard IMU (MPU9250) to monitor vibrations in your system.
- Estimate dominant frequency and magnitude of vibrations, which is crucial for diagnosing mechanical issues or ensuring smooth operation in drones or robotic platforms.

Software doesn't support it yet...

#### 4. Q&A

If you are struggling with the software building, please refer to the build workflow [build.yml](#) for a hint. If it doesn't help, you can open [an issue](#).

#### 5. More examples

Consider the following projects as examples:

1. [RL UAV Lights node](#)
2. [IU VTOL PMU node](#)
3. [IU Airspeed+rangefinder+lights+PWM node](#)
4. [IU Fuel tank node](#)