LEVCAN GitHub page

Intro

This is open-source CAN protocol used in our devices. LEVCAN is licensed under the MIT License. Written in GNU ISO C11 Some ideas taken from J1939 and CanOpen. Uses 29bit CAN network (1 MBps default).

Features:

- Two transmission modes, 'TCP' (controlled reception and data order) and 'UDP'
- Multiple nodes supported for one device
- Dynamic network address
- Configurable parameters for devices
- Simple file i/o with file server
- Static and dynamic memory supported
- Up to 125 devices on bus
- 10 bit message ID + length matching
- Broadcast and addressed messages

Why?

Nucular electronics (controller, bms, display) was designed to be independent but same time complement each other. For example - single controller can regulate total battery voltage, but in connection with BMS it allows you to control every cell voltage and temperature. Display can show parameters from any device on bus, it collects some information by device type and show common values on main screen. Also it can be control-center for all your controllers, there is a lot of IO ports inside. Connect multiple controllers and configure each separately - just one system wire (and a bunch of power cables huh). Since device configuration stored on every node, it allows to update software separately and get new features. New software updater works over simple file transfer protocol, for end-developer it is abstracted to usual f_read, f_open functions.

Developers friendly (well... most of time)

It is easy to integrate any new device to existing system, just connect CAN cable, initialize your LEVCAN node and you can change your own device parameters from display or interact with controllers. Standard message objects described in levcan_objects.h. Some drivers for STM32-CAN already included in repo. You can check examples folder, I'm trying to keep them up-to-date.

Just if you want to know everything…

What happens with your EV, you can collect all data from devices. This feature is WIP for display now (data-logger on microSD card).

From:
https://docs.nucular.tech/ - **Nucular Electronics**

Permanent link:
https://docs.nucular.tech/doku.php?id=en:levcan&rev=1602170147

Last update: **2020/10/08 17:15**