KVASER LEAF LIGHT HS V2

EAN: 73-30130-00685-0

The Kvaser Leaf Light HS v2 represents one of the easiest and lowest-cost methods of connecting a computer to a CAN bus network. With its USB 2.0 compliant connector and 9-pin D-SUB connector, the Leaf Light HS v2's sleek, ergonomically designed housing is both robust enough for everyday use and small and flexible enough to be used in space-constrained applications. Now with galvanic isolation as standard.

Warranty
2-year warranty. See our General Conditions and Policies for details.

Support
Free support for all products by contacting support@kvaser.com.

KVASER LEAF LIGHT HS V2

EAN: 73-30130-00685-0

Major Features

- 8000 messages per second, each time-stamped with 100 microsecond accuracy
- Supports both 11-bit (CAN 2.0A) and 29-bit (CAN 2.0B active) identifiers
- High-speed CAN connection (compliant with ISO 11898-2), up to 1 Mbit/s.
- Galvanic isolation, previously a more expensive option on Kvaser's original Leaf Light, now comes as standard on the Leaf Light v2, enhancing protection from power surges or electrical shocks.
- Low current consumption (90 mA) reduces power drain from your laptop.
- Local buffering and preprocessing results in high performance and a reduction of time critical tasks for the PC.

Technical Data

Bit Rate	40 - 1000 kbps
CAN Channels	1
CAN FD	No
Casing Material	PC-ABS
Connector	DSUB 9 Male
Current Consumption	90mA
Dimensions	35 x 165 x 17 mm
Error Frame Detection	Yes
Galvanic Isolation	Yes
Operating Temperature Range	-40 °C to +70 °C
PC Interface	USB
Timestamp Resolution	100 µs
IP Class	IP40
Weight	110g
Operating Systems	Windows, Linux

Software

Documentation, Kvaser CANlib SDK and drivers can be downloaded for free at www.kvaser.com/downloads.

Kvaser CANlib SDK is a free resource that includes everything you need to develop software for the Kvaser CAN interfaces. Includes full documentation and many program samples, written in C, C++, C#, Delphi, Visual Basic, Python and t programming language.

Kvaser CAN hardware is built around the same common software API. Applications developed using one device type will run without modification on other device types