CANnect

CAN to CAN interface adapter

Revision: 1.0A

1. Features

- **Two FDCAN interfaces**
- **DBC file integration and message mapping**
- **5V CAN interfaces**
- **Baud rate bridge**
- **CAN ID translation**
- **5V analog input interface**
- **Compact size: 35mm x 12mm**
- **Wide operating supply voltage**

2. Applications

- Automotive prototyping and testing
- Legacy system integration
- CAN network simulation
- Sensor data conditioning
- Protocol translation between ECUs

3. Description

The CANnect board is a powerful and flexible tool designed to bridge and manipulate Controller Area Network (CAN) messages between two independent CAN interfaces. It enables seamless communication between devices operating at different baud rates or using different message formats, making it ideal for automotive, industrial, and embedded systems applications. The CANnect provides in a small form factor PCB, with a wide operating supply voltage between 9 - 28 V, two CAN interfaces fully compliant with the FDCAN protocol.

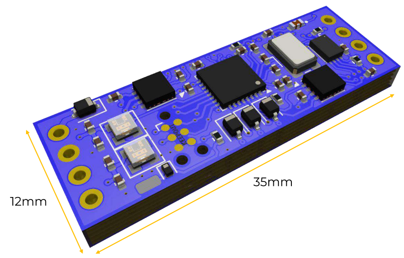

Figure 1: CANnect 1.0a

4. Technical Characteristics

PARAMETERS	MIN	TYP	MAX	UNIT
Voltage Supply	9	12	28	V
Supply Current	14	21	32	mA
Operating Temperature	-20	25	105	°C

Table 1: Electronics characteristics

5. Board pin-out

The board comes with four pre-soldered wires. The table below shows the corresponding pin-out based on the color of the cable.

CABLE COLOR	SIGNAL
Red	VCC (Power)
Black	GND (Ground)
White	CAN-H
Blue	CAN-L

Table 2: Cable color to signal mapping

Different customization options are available on request. The board can be equipped with customer-defined connectors.

6. Application information

6.1. Overview

Modern embedded systems often require integration between multiple CAN-enabled devices that may differ in communication speed, message structure, or protocol. The CANnect board addresses this challenge by acting as a real-time, bidirectional CAN message adapter. It not only relays messages between two CAN interfaces but also allows for dynamic manipulation of message content, structure, and timing. The main task of the CANnect is to adapt and manipulate CAN bus payloads, acting as a loopback between two CAN interfaces. These two CAN interfaces might have different baud rates. The system therefore is able to read CAN messages on one interface, modify the received data and schedule the transmission of the manipulated data on the other interface. Both the CAN interfaces are able to receive and transmit messages and they can both manipulate the messages before transmission.

6.2. System capabilities

CANnect features two fully independent CAN interfaces (CAN1 and CAN2), each capable of:

- Receiving and transmitting CAN messages.
- Operating at distinct baud rates: 1 Mbps, 500 kbps, 250 kbps or 125kbps.
- Manipulating message content before transmission.

This dual-interface design allows CANnect to function as a protocol bridge, message translator, or data conditioner. The core functionality of CANnect is to receive messages on one CAN interface (RX), apply user-defined transformations, and transmit the modified messages on the second interface (TX). This process occurs with minimal latency, enabling real-time communication between devices.

Example 1 - Baud Rate Bridging

CAN1 is connected to a CAN sensor that communicates at 500kbps, CAN2 is connected to a device that only communicates at 250kbps. CAN1 is the RX interface and CAN2 is the TX interface. Data coming from the sensor are received on CAN1, CANnect modifies the message (e.g., changes CAN ID, applies scaling and offset) and retransmit the new data over CAN2 with the different baudrate.

Example 3 - CAN ID Translation

CAN1 is connected to a CAN sensor that communicates at 1Mbps on CAN ID 0x100, CAN2 is connected to a device that only communicates at 1Mbps on CAN ID 0x105. CAN1 is the RX interface and CAN2 is the TX interface. The data coming from the sensor are received on CAN1, manipulated by CANnect (the CAN ID to which the data is sent is 0x105) and transmitted over CAN2.

Example 2 - Bidirectional Communication

Messages received on CAN1 are transformed and sent on CAN2., simultaneously, messages received on CAN2 are transformed and sent on CAN1. Each direction can have its own transformation rules and message definitions.

6.3. DBC file integration and message mapping

CANnect leverages DBC (Database CAN) files to define and interpret CAN messages. Each interface can be assigned a unique DBC file, enabling:

- Parsing of incoming messages based on signal definitions;
- Construction of outgoing messages with modified structure or content.

The system supports message mapping between the two DBCs, allowing:

- CAN ID remapping;
- Signal scaling (multipliers);
- Signal offsetting;
- Endianness configuration;
- Custom transmission intervals.

This DBC file is assigned to the CAN interface that receives the data from an external device (sensor, ECU or any other device that communicates over CAN bus). The CANnect is configurable via a GUI that provides an easy and user friendly interface for managing the system. The GUI therefore allows the user to assign the DBC to one of the two interfaces (CAN1 or CAN2) and to create a second DBC file that bridges the two CAN interfaces. The second DBC contains the same data structure of the first one but the user can:

- Change the TX CAN ID of each message;
- Introduce an offset to each variable inside the CAN frame;
- Introduce a multiplier to each variable inside the CAN frame;
- Set the TX transmission rate of each message;
- Set the endianness of each variable inside the CAN frame.

The GUI allows the user to set the CAN baud rate of each CAN interface independently. The allowed CAN baud rates are 1Mbps, 500kbps, 250kbps, 125kbps.

6.4. Configuration via GUI

The GUI is designed to fully customize the parameters of the CANnect firmware. In the GUI the user is able to load a DBC file containing the channels that should be manipulated by the CANnect, providing full control over system behavior. Key features include:

- Loading and assigning DBC files to CAN1 and CAN2
- Creating a bridge DBC that defines how messages are transformed between interfaces.
- Setting transformation rules for each signal: change CAN ID, apply offset and multiplier, configure endianness, set transmission rate;
- Independently configuring baud rates for each CAN interface.

This GUI-based approach ensures that even complex configurations can be implemented quickly and accurately, without requiring firmware-level changes.

7. Mechanical drawings

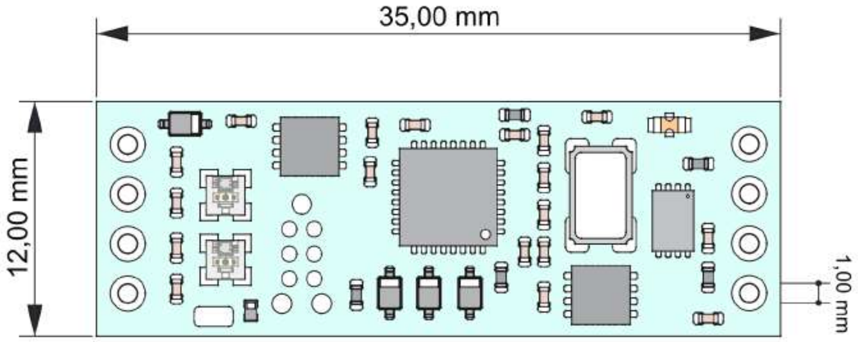

Figure 2: CANnect mechanical drawing