

CANBed FD - Arduino CAN-FD Development Kit

SKU 102991442

This CANBed-FD adopts MCP2517FD CAN Bus controller with SPI interface and MCP2542FD CAN transceiver to achieve the CAN-BUS capability. With an OBD-II converter cable added on and the OBD-II library imported, you are ready to build an onboard diagnostic device.

CANBed FD is a CAN FD dev board with an **Atmega32U4** microcontroller inside, which has **32KB of Flash and 2.5KB of RAM**. The operating frequency is **16MHz**, which can meet most embedded applications. There is an Arduino Leonardo bootloader inside the Atmega32U4. If you have used the Arduino IDE, programming will not be a problem for you.

CANBed FD uses **MCP2517FD** as CAN controller and **MCP2542 (MCP2557)** as CAN receiver, which are high-performance CAN Bus chips, which can work in both CAN 2.0 and CAN FD protocols at the same time. There is a Micro USB connector on the board, through which you can program the board or supply power to the board. There is one I2C, one UART, one SPI interface, 3 analog input interfaces, and 8 digital IO on the board.

This CANBed-FD Arduino CAN-FD Development Kit adopts MCP2517FD CAN Bus controller with SPI interface and MCP2542FD CAN transceiver to achieve the CAN-BUS capability. With an OBD-II converter cable added on and the OBD-II library imported, you are ready to build an onboard diagnostic device.

What is more, the is an ATmega32U4 with Arduino Leonardo bootloader on the board, you can use Arduino IDE to program the board easily.

What is CAN-BUS?

CAN stands for *Controller Area Network*, it is used to allow microcontrollers and devices to communicate with each other within a vehicle without a host computer which allows for control and data acquisition. These devices are also called Electronic Control Units (ECU) and they enable communication between all parts of a vehicle.

Today, you can find up to 70 ECUs in a modern car. CAN is a serial communication bus designed for industrial and automotive applications. For example, they are found in vehicles, farming equipment, industrial environments, etc.

How does CAN-BUS work?

The fuel level, door sensors, odometer, and many more parts of a car have to communicate with each other somehow, and CAN BUS is what they used to do. These CAN-compatible components, which are called “nodes” are connected with a 3-string copper wire, with no central router to govern the flow of data. Every node can hear the messages of every other node.

Every node has an ID, where the ones with the higher priority ID can have the priority to “talk” first while the others “listen.” This is to ensure that there are never two nodes talking at the same time. The biggest benefit of CAN-BUS is to be able to just connect components without having to worry about signal routing.

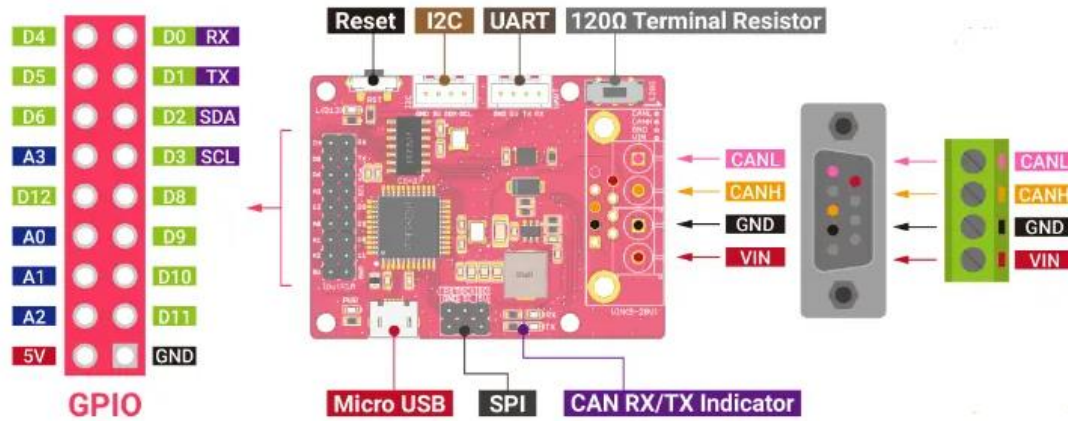
Features

- **Compact size design:** 56x41mm overall dimension
- **Support two CAN protocols:** CAN-FD and CAN 2.0
- **High-performance CAN Bus chip:** MCP2517FD as CAN controller and MCP2542 (MCP2557) as CAN receiver
- **Flexible CAN Interface:** Industrial standard 9-pin sub-D connector or 4-pin terminal, 2 x 4-Pin Grove connectors Compatible with the Grove ecosystem
- **Wide power input** ranges from 7-28V and **stable 5V/1A output**

Specifications

- MCU - ATmega 32U4 (with Arduino Leonardo bootloader)
- Clock speed - 16MHz
- Flash memory - 32KB
- SRAM - 2.5KB
- EEPROM - 1KB
- Operating voltage - 7~28V
- Input interface - Sub-D

Hardware Overview



1. GPIO - 9x2 I/O Pin OUT
2. Micro USB connector for programming
3. SPI - ICSP connector
4. CAN RX/TX Indicator
5. DB9 connector or Terminal for CAN Bus
6. Switch for the 120Ω terminal resistor for CAN Bus
7. Grove connector for UART
8. Grove connector for I2C
9. Reset

Part List



CANBed PCBA	1
Sub-D Connector	1
4-Pin Terminal	1
4-Pin 2.0 Connector	2
9x2 2.54 Header	1
3x3 2.54 Header	1

FAQ

I can't upload code to CANBed FD

- When the board is connected to the computer via the Micro USB cable, a new COM device will appear on the computer. If the new device does not appear on your computer, you can try a different USB cable or try another computer.
- If your PC recognize the COM port, please try pressing the reset button, then click on the Upload button in Arduino IDE, when the IDE shows compile done, release the reset button immediately.*

The RX/TX led light up and never turn off

- Check if the baudrate of CAN Bus is setting correct
- Try turning on/off the switch for the 120Ω terminal resistor
- Check if CANH and CANL is well connected