

Technical Documentation

Table of Contents

A) Hardware.....	2
1) List of components.....	2
2) Preparing the parts.....	2
3) Connecting things together.....	3
4) Pin usage.....	3
B) Software.....	4
1) The repository.....	4
2) Configuration settings.....	4
3) Serial configuration commands.....	5

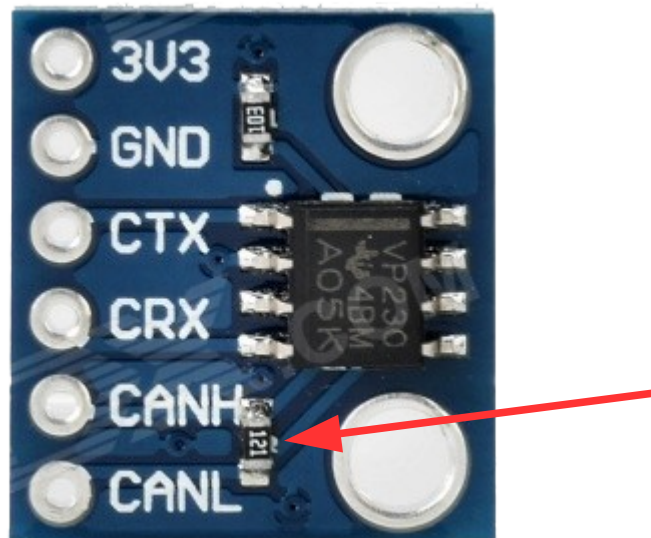
A) Hardware

1) List of components

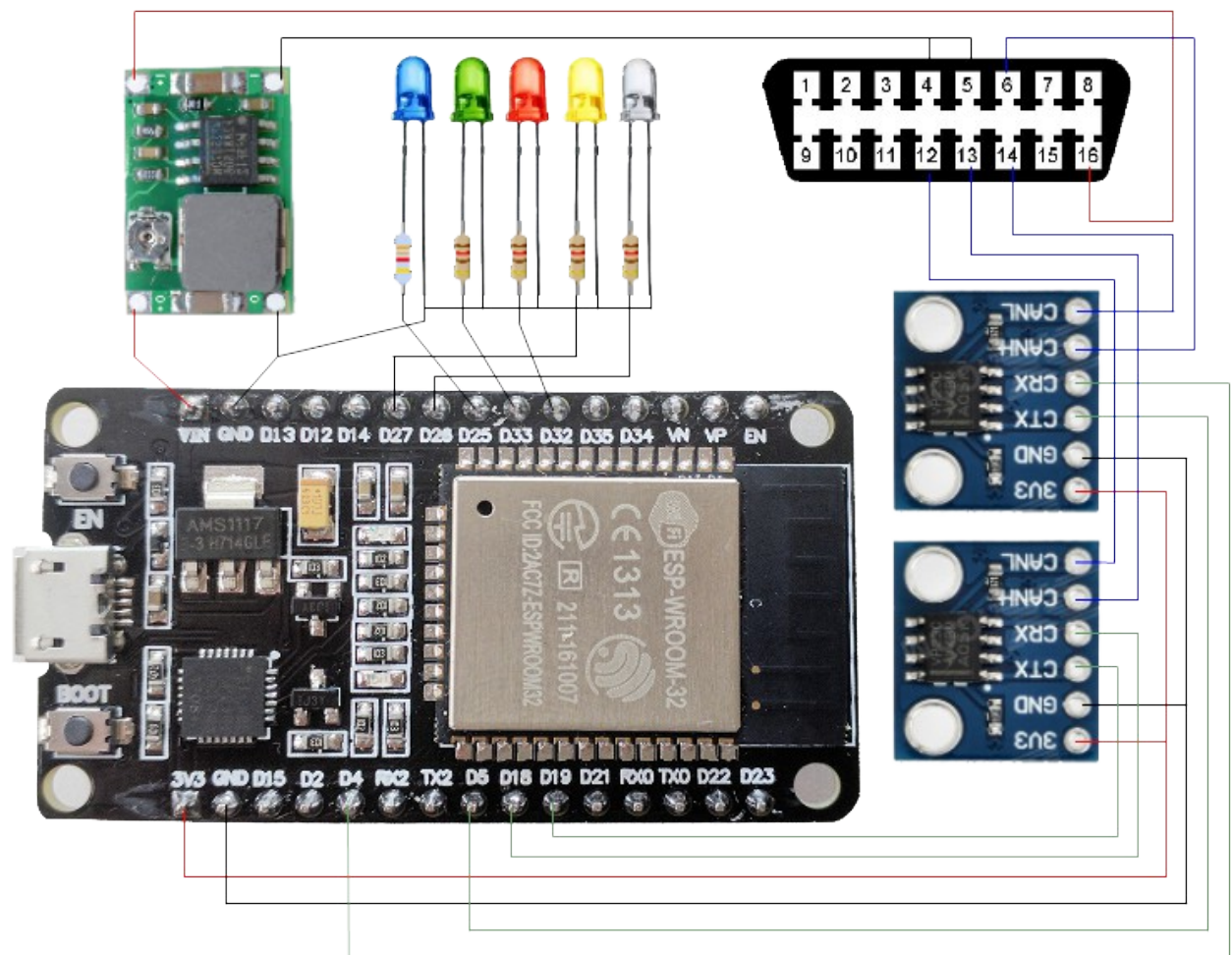
- 1x ODB case with connector
<https://www.aliexpress.com/item/CASE-ELM327-OB2-Connector-J1962m-Plug-with-Enclosure-16pin-Male-Connector/32827624441.html>
- 1x DOIT ESP32 DEVKIT or similar
<https://www.aliexpress.com/item/ESP32S-ESP-32S-ESP32-ESP-32-CP2102-Wireless-WiFi-Bluetooth-Development-Board-Micro-USB-Dual-Core/32842731763.html>
- 1x Mini DC-DC converter
<https://www.aliexpress.com/item/10Pcs-Mini-DC-DC-12-24V-To-5V-3A-Step-Down-Power-Supply-1-8V-2/32848669016.html>
- 2x SN65HVD230 CAN bus transceiver
<https://www.aliexpress.com/item/SN65HVD230-CAN-Bus-Transceiver-Communication-Module-Thermal-Protection-Slope-Control-for-Arduino/32851150816.html>
- 5x 3mm LED (blue, green, red, yellow, white)
- 4x 120 Ω resistor
- 1x 3.3 k Ω resistor

2) Preparing the parts

The transceivers do need some special treatment before you can use them. It has installed by default a termination resistor, which should not be there if connected to a car, so you really need to remove it or your dongle won't be able to communicate with the car.



3) Connecting things together



The DC-DC converter must be adjusted to an output of 5V. **Don't mix up CAN connections on the OBD2 connector, as you may break your car's bus system if improperly connected!**

4) Pin usage

Pin	Connection
4	RX of main CAN bus
5	TX of main CAN bus
18	RX of multimedia CAN bus
19	TX of multimedia CAN bus
25	blue LED
26	white LED
27	yellow LED
32	red LED
33	green LED

B) Software

1) The repository

The repository of the CANSee code is located on Gitlab

<https://gitlab.com/jeroenmeijer/cansee>

2) Configuration settings

%

3) Serial configuration commands

Command	Explanation																		
a	Show all buffered free CAN_frames																		
z	Reboot (needed for all configuration changes except debug mode)																		
r	Reset configuration to default and reboot.																		
gxxx	Get free frame with id xxx																		
ixxx,yyyy	Get ISO-TP frame with id xxx, PID yyyy																		
nxxx,config	<p>set EEPROM configuration xxx is determined by this table</p> <table> <tr> <td>100</td><td> <p>Set the flags configuration as hex as ssbbwwlldd [01010200ff]</p> <p>ss: Serial over USB 00 for off, any other for on</p> <p>bb: Bluetooth 00 for off, any other for on</p> <p>ww: WiFi 00 for off, 01 for Station mode, 02 for Soft Access Point mode. For station mode a DHCP server is assumed.</p> <p>ll: Use leds 00 for off, ff for on</p> <p>dd: Debug 00 for off, ff for on. A bit pattern can be used and added 01: show CANbus receive free CAN_frame_std 02: show CANbus receive ISO-TP CAN_frame_std 04: show Commands issued 08: show free frame Commands 10: show ISO-TP Commands</p> </td></tr> <tr> <td>200</td><td>Set the Bluetooth name [CANSee]</td></tr> <tr> <td>201</td><td>Set the Bluetooth pin [1234, not implemented by the API!!]</td></tr> <tr> <td>300</td><td>Set the WiFi ssid for Soft Access Point mode [CANSee]</td></tr> <tr> <td>301</td><td>Set the WiFi password for Soft Access Point mode [CANSeeMe]</td></tr> <tr> <td>400</td><td>Set the WiFi ssid for Station mode [Home]</td></tr> <tr> <td>401</td><td>Set the WiFi password for Station mode [Password]</td></tr> <tr> <td>500</td><td> <p>Set CANbus parameter for CAN0 in hex as sprxtx</p> <p>sp: speed in 25 kbps increments. [0x14 for can0, 0x0a for can1]</p> <p>rx: Rx pin of transceiver [4 for can0, 0x12 for can1]</p> <p>tx: Tx pin of transceiver [5 for can0, 0x13 for can1]</p> <p>Note that this is a very advanced setting and selecting pins that are internally used on your board (notably 6-11) will crash the board. Also pins 34 and up are input only.</p> </td></tr> <tr> <td>501</td><td>Set CANbus parameter for CAN1 in hex as sprxtx (see above).</td></tr> </table>	100	<p>Set the flags configuration as hex as ssbbwwlldd [01010200ff]</p> <p>ss: Serial over USB 00 for off, any other for on</p> <p>bb: Bluetooth 00 for off, any other for on</p> <p>ww: WiFi 00 for off, 01 for Station mode, 02 for Soft Access Point mode. For station mode a DHCP server is assumed.</p> <p>ll: Use leds 00 for off, ff for on</p> <p>dd: Debug 00 for off, ff for on. A bit pattern can be used and added 01: show CANbus receive free CAN_frame_std 02: show CANbus receive ISO-TP CAN_frame_std 04: show Commands issued 08: show free frame Commands 10: show ISO-TP Commands</p>	200	Set the Bluetooth name [CANSee]	201	Set the Bluetooth pin [1234, not implemented by the API!!]	300	Set the WiFi ssid for Soft Access Point mode [CANSee]	301	Set the WiFi password for Soft Access Point mode [CANSeeMe]	400	Set the WiFi ssid for Station mode [Home]	401	Set the WiFi password for Station mode [Password]	500	<p>Set CANbus parameter for CAN0 in hex as sprxtx</p> <p>sp: speed in 25 kbps increments. [0x14 for can0, 0x0a for can1]</p> <p>rx: Rx pin of transceiver [4 for can0, 0x12 for can1]</p> <p>tx: Tx pin of transceiver [5 for can0, 0x13 for can1]</p> <p>Note that this is a very advanced setting and selecting pins that are internally used on your board (notably 6-11) will crash the board. Also pins 34 and up are input only.</p>	501	Set CANbus parameter for CAN1 in hex as sprxtx (see above).
100	<p>Set the flags configuration as hex as ssbbwwlldd [01010200ff]</p> <p>ss: Serial over USB 00 for off, any other for on</p> <p>bb: Bluetooth 00 for off, any other for on</p> <p>ww: WiFi 00 for off, 01 for Station mode, 02 for Soft Access Point mode. For station mode a DHCP server is assumed.</p> <p>ll: Use leds 00 for off, ff for on</p> <p>dd: Debug 00 for off, ff for on. A bit pattern can be used and added 01: show CANbus receive free CAN_frame_std 02: show CANbus receive ISO-TP CAN_frame_std 04: show Commands issued 08: show free frame Commands 10: show ISO-TP Commands</p>																		
200	Set the Bluetooth name [CANSee]																		
201	Set the Bluetooth pin [1234, not implemented by the API!!]																		
300	Set the WiFi ssid for Soft Access Point mode [CANSee]																		
301	Set the WiFi password for Soft Access Point mode [CANSeeMe]																		
400	Set the WiFi ssid for Station mode [Home]																		
401	Set the WiFi password for Station mode [Password]																		
500	<p>Set CANbus parameter for CAN0 in hex as sprxtx</p> <p>sp: speed in 25 kbps increments. [0x14 for can0, 0x0a for can1]</p> <p>rx: Rx pin of transceiver [4 for can0, 0x12 for can1]</p> <p>tx: Tx pin of transceiver [5 for can0, 0x13 for can1]</p> <p>Note that this is a very advanced setting and selecting pins that are internally used on your board (notably 6-11) will crash the board. Also pins 34 and up are input only.</p>																		
501	Set CANbus parameter for CAN1 in hex as sprxtx (see above).																		

	<table> <tr> <td data-bbox="405 120 485 212"></td><td data-bbox="485 120 1484 212"> <p>Usage of the semi-secondary CAN bus by switching the single controller to another pin pair is not supported and basically fails spectacularly.</p> </td></tr> <tr> <td colspan="2" data-bbox="405 212 1484 282"></td></tr> </table>		<p>Usage of the semi-secondary CAN bus by switching the single controller to another pin pair is not supported and basically fails spectacularly.</p>		
	<p>Usage of the semi-secondary CAN bus by switching the single controller to another pin pair is not supported and basically fails spectacularly.</p>				