

ECUMASTER WheelSpeedToCAN

Manual
Revision 2.0
(Firmware 16.x)

Contents

1. Copyright and trademarks.....	4
2. Introduction.....	4
3. Description.....	4
4. Specification.....	4
4.1. Mechanical drawings.....	4
4.2. Connector description.....	5
4.3. Specification table.....	5
5. PC connection.....	6
6. Light Client description.....	6
7. Software features.....	8
7.1. General information.....	8
7.2. Sensor input setup.....	9
7.3. Wheel speed setup.....	9
7.3.1. Front wheels setup.....	10
7.3.2. Rear wheels setup.....	10
7.3.3. EMU speed output.....	11
7.3.4. OEM speed output.....	11
7.3.5. Custom speed output.....	12
7.4. Frequency setup.....	13
7.4.1. Frequency output.....	13
7.5. Edge counter setup.....	14
7.5.1. Edge counter output.....	14
7.6. Encoder setup.....	15
7.6.1. Encoder output.....	15
7.7. Diagnostics.....	16
8. Use cases.....	18
8.1. Using VR sensors.....	18
8.1.1. Sensor description.....	18
8.1.2. Sensor type identification.....	18
8.1.3. Sensor connection.....	19
8.1.4. Sensor input configuration.....	19
8.2. Using HALL sensors.....	20
8.2.1. Sensor description.....	20
8.2.2. Sensor type identification.....	20
8.2.3. Sensor connection.....	20

8.2.4. Sensor input configuration.....	21
8.3. Using MR sensors.....	21
8.3.1. Sensor description.....	21
8.3.2. Sensor type identification.....	22
8.3.3. Sensor connection.....	22
8.3.4. Sensor input configuration.....	23
8.4. Using the module with working ABS systems.....	23
8.4.1. ABS with VR sensors.....	23
8.4.2. ABS with HALL sensors.....	24
8.4.3. ABS with MR sensors.....	24
8.5. Using the module with encoders.....	26
9. Document revision history.....	27

1. Copyright and trademarks

All trademarks, service marks, trade names, trade dress, product names and logos appearing on this document are the property of their respective owners.

2. Introduction

This document provides information about the ECUMASTER WheelSpeedToCAN module including a device description, specifications, available features and use cases.

Document created for firmware version 16.x. Description of all the features can differ from other firmware versions, so upgrade is recommended.

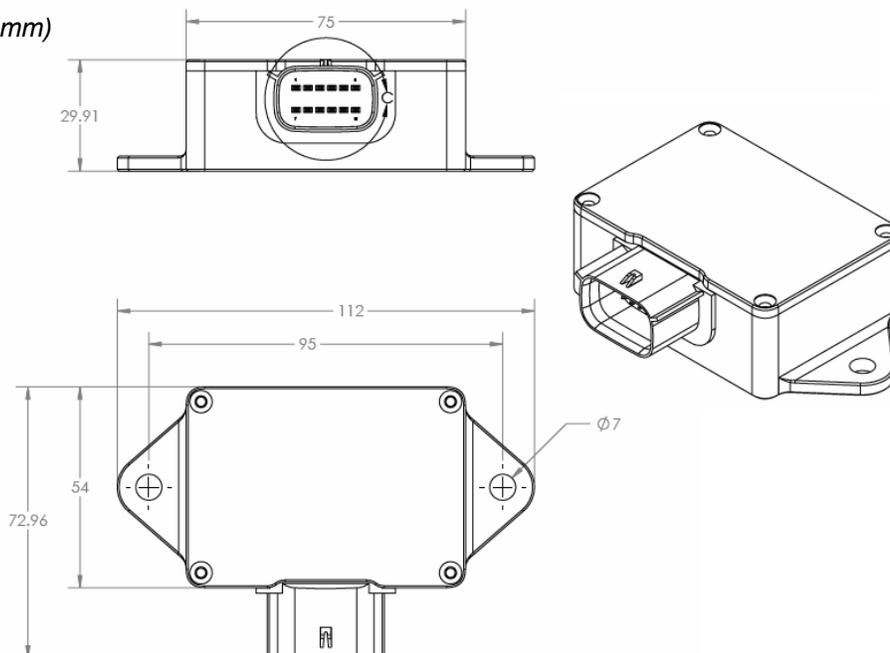
3. Description

WheelSpeedToCAN is a device designed to measure frequency-type signals from 4 separate sensors and provide that data over CAN bus. The device can measure signals from different types of sensors, or standard digital signals. Thanks to differential inputs, configurable logic threshold and pull up/down resistors, sensors can be connected directly to the device or parallel to a working ABS system. There are also additional features like sending the speed data in different OEM formats, edge counter or possibility to connect encoders with quadrature signal.

4. Specification

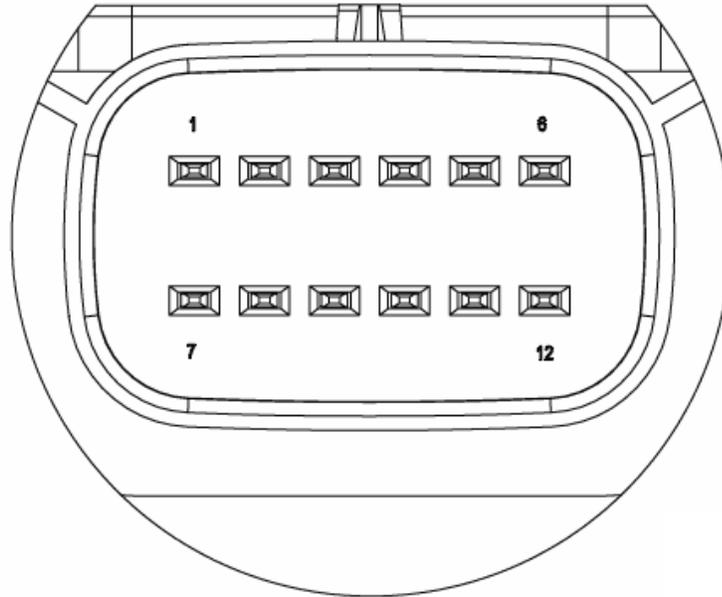
4.1. Mechanical drawings

(Dimensions in mm)



4.2. Connector description

Pin	Description
1	Sensor 1 positive
2	Sensor 1 negative
3	Sensor 2 positive
4	Sensor 2 negative
5	Sensor 3 positive
6	Sensor 3 negative
7	12V (after ignition)
8	Ground
9	CAN Low
10	CAN High
11	Sensor 4 positive
12	Sensor 4 negative



4.3. Specification table

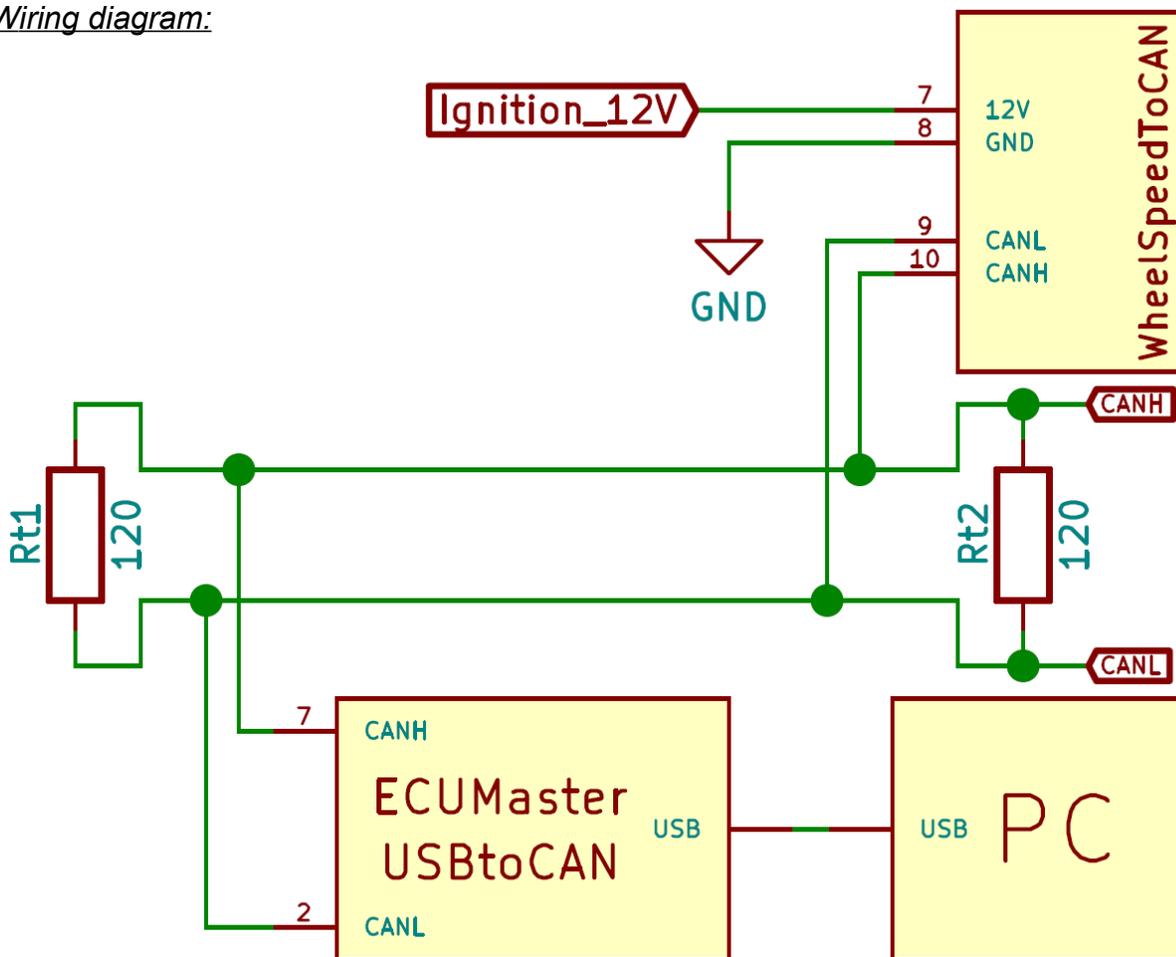
General	
Operating supply voltage	6-22 V, immunity to transients according to ISO 7637
Reverse polarity protection	Yes, internal up to 16 V
Temperature range	-40 to +85 °C
Enclosure	IP65, bespoke CNC machined aluminum
Size and weight	112x73x30 mm, 112 g
Connector	Molex MX120G 12P
CAN-bus bitrate	1 Mbps, 500 kbps, 250 kbps, 125 kbps
PC communication	Using USB to CAN interface (ECUMASTER USBtoCAN, PEAK, Kvaser)
Inputs	
Input type	Differential (2 pins per input)
Differential input range	0-7V
Max combined frequency	50kHz
Configurable pull up/down	None, 1k at positive input to 5V, 140, 220 or 390R at positive input to ground, 140, 220 or 390R at negative input to supply voltage (12V)
Configurable gain	Normal or double
Configurable logic threshold	Yes

5. PC connection

Communication with PC software is done using CAN bus. This requires a special interface which converts CAN bus communication to USB 2.0. Interface is an independent device and must be purchased separately. Supported interfaces:

- ECUMASTER USBtoCAN (driver and manual: <http://www.ecumaster.com/download/>)
- PEAK-System
- Kvaser

Wiring diagram:



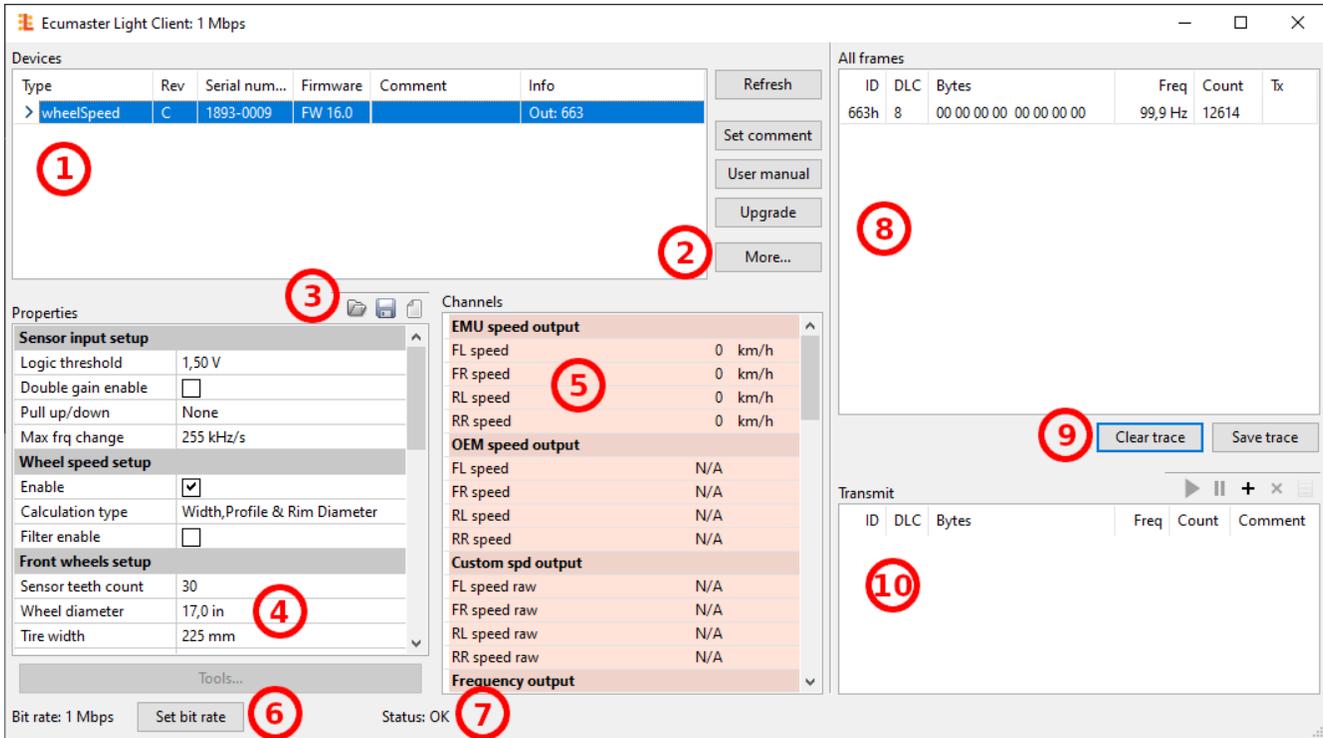
6. Light Client description

ECUMASTER Light Client is the configuration software for many ECUMASTER products as well as a tool for CAN bus monitoring. Light Client can be used to check available devices on the CAN bus, display channel values, change CAN bus bitrate and device specific properties. Monitoring of CAN bus is possible thanks to list of frames grouped by ID, saving traffic trace file and sending custom messages on the CAN bus.

Software and manual can be downloaded from:

<https://www.ecumaster.com/products/light-client/>

Light Client with WheelSpeedToCAN connected:



User interface description:

- 1) List of devices compatible with the Light Client that are available on the CAN bus. General information about the device. Double click a row to select the device and display its channels and properties.
- 2) Devices control buttons:
 - Refresh – refresh devices list,
 - Set comment – set a comment for the device,
 - User manual – view the online user manual in a web browser,
 - Upgrade – the device firmware upgrade, online or local file,
 - More:
 - Load project – open previously saved project file,
 - Save project – save project file of the currently selected device,
 - Restore project to default – restore default properties of the currently selected device,
 - Mount device from file... – open a virtual device from local file and append it to list of devices,
 - Mount device from ECUMASTER.com... – open a virtual device from Internet and append it to list of devices,
 - Export CAN definition .CANX... – export to .CANX file for importing in other ECUMASTER products,
 - Export CAN definition .DBC... – export to .DBC file for importing in other products,

- About – application information.
- 3) Device properties management toolbar:
 - Load project – open previously saved project file,
 - Save project – save project file of the currently selected device,
 - Restore project to default – restore default properties of the currently selected device.
- 4) List of properties for the selected device.
- 5) List of channels sent over the CAN bus.
- 6) Button for changing bit rate of all the devices compatible with Light Client.
- 7) CAN bus status information.
- 8) List of received frames on the CAN bus, grouped by ID.
- 9) Clear trace button removes all received frames from history and Save trace button saves that history to a text file.
- 10) Transmit window allows you to set custom frames which will be sent over the CAN bus.

7. Software features

7.1. General information

Properties:

Device properties are settings that can be changed by the user to configure the device for a specific use case. Properties can be viewed and changed using the ECUMASTER Light Client software and compatible CAN bus interface.

Channels:

Device channels are output values sent over CAN bus. Channels are sent as a raw values, which means that obtaining values with correct units requires doing below calculations:

$$Value[unit] = \frac{Value[raw] * Multiplier}{Divider} + Offset$$

Channels are described by the tables with CAN frame information, conversion factors, etc. All channel values are sent in Big Endian format.

7.2. Sensor input setup

Category for setting up the input properties. Each property in this category sets the same value for all signal inputs.

Properties:

- Logic threshold** – voltage threshold for comparator to switch between high and low state.
- Double gain enable** – enable double gain of input signal.
- Pull up/down** – select different pull up/down configurations. Example value of “Positive, 1k, 5V” means that every sensor input will have 1k ohm resistor connected between positive input and 5V.
- Max frq change** – maximum frequency change in kHz per second. Type of filtering, which can be used to remove fast spikes in the input signal. Lower value means stronger filtering.

7.3. Wheel speed setup

Category for setting up wheel speed calculation. There are 3 different types of calculation which require information about different physical dimensions of the wheels. There are also settings for wheel speed filter, which can have separate strengths for low and high speed with configurable threshold.

Properties:

- Enable** – enable wheel speed calculation.
- Calculation type** – select calculation type. Available types are “Width, Profile & Diameter”, “Tire Diameter” or “Tire Circumference”.
- Filter enable** – enable filtering of the wheel speed value.
- Filter strength** – wheel speed signal filter strength. Value from 0 to 9. Higher value means stronger filtering.
- Low speed thr** – low speed threshold in km per hour. Below that threshold “Low speed strength” will be used as a filter strength.
- Low speed strength** – low speed filter strength. Overrides “Filter strength” value when wheel speed is below the “Low speed thr”.

7.3.1. Front wheels setup

Category for setting up properties required for speed calculation of the front wheels. Required properties depend on the selected “Calculations type” in “Wheel speed setup” category. This category will be displayed after enabling “Wheel speed setup”.

Properties:

- Sensor teeth count** – number of teeth that will be read in one wheel rotation.
- Wheel diameter** – wheel diameter in inches. Required for “Width, Profile & Diameter” calculation type.
- Tire width** – tire width in mm. Required for “Width, Profile & Diameter” calculation type.
- Tire profile** – tire height aspect ratio in % of “Tire width”. Required for “Width, Profile & Diameter” calculation type.
- Tire diameter** – tire diameter in mm. Required for “Tire Diameter” calculation type.
- Tire circumference** – tire circumference in mm. Required for “Tire Circumference” calculation type.
- Effective radi ratio** – effective tire radius ratio. This is multiplier for the tire radius, which depends on the amount of tire deformation. Tire deformation depends on the tire type, rubber compound, pressure and load. Higher deformation corresponds with lower ratio values. This value should be used to calibrate the calculated speed.
- Dynamic corr** – dynamic correction in % per 1000 rpm of the wheel. This value depends on the tire expanding with increasing rotational speed. Bigger expansion of the tire corresponds with higher value of dynamic correction. Calibration of this value should be done at high speed, after calibrating with “Effective radi ratio” at low speed.

7.3.2. Rear wheels setup

Category similar to the “Front wheels setup”, but for rear wheels.

Properties:

- Same as front** – can be selected to use the same dimensions as for the front wheels.

**Rest of the properties are the same as in the “Front wheels setup”, but for rear wheels.*

7.3.3. EMU speed output

Category for setting up wheel speed data output for ECUMASTER EMU. It will be displayed after enabling “Wheel speed setup”.

Properties:

- Enable** – enable wheel speed output data for ECUMASTER EMU.
- Output ID** – CAN frame ID for the output data.
- Update frequency** – channels sending frequency.

Channels:

Byte	Channel	Data type	Range	Multiplier	Divider	Offset	Unit
Output ID (default: 0x663)							
0	FL speed	16bit unsigned	0 – 4095	1	16	0	km/h
2	FR speed	16bit unsigned	0 – 4095	1	16	0	km/h
4	RL speed	16bit unsigned	0 – 4095	1	16	0	km/h
6	RR speed	16bit unsigned	0 – 4095	1	16	0	km/h

- FL speed** – front left wheel speed.
- FR speed** – front right wheel speed.
- RL speed** – rear left wheel speed.
- RR speed** – rear right wheel speed.

7.3.4. OEM speed output

Category for setting up wheel speed data output for OEM vehicles. CAN frames with “Preview ID” are sent to display the calculated speed in Light Client. There will also be frames sent at the vehicle specific IDs. This category will be displayed after enabling “Wheel speed setup”.

Properties:

- Enable** – enable wheel speed output data for an OEM vehicle.
- Preview ID** – CAN frame ID for the preview data.
- Update frequency** – channels sending frequency.
- Vehicle** – select a vehicle for which speed data will be sent. Available vehicles are “Mitsubishi Lancer” and “Dodge Charger Gen I”. With sufficient data we can add new vehicles to the list at the user’s request.

Channels:

Byte	Channel	Data type	Range	Multiplier	Divider	Offset	Unit
Preview ID (default: 0x648)							
0	FL speed	16bit unsigned	0 – 4095	1	16	0	km/h
2	FR speed	16bit unsigned	0 – 4095	1	16	0	km/h
4	RL speed	16bit unsigned	0 – 4095	1	16	0	km/h
6	RR speed	16bit unsigned	0 – 4095	1	16	0	km/h

- FL speed** – front left wheel speed.
FR speed – front right wheel speed.
RL speed – rear left wheel speed.
RR speed – rear right wheel speed.

7.3.5. Custom speed output

Category for setting up wheel speed data output with custom conversion factors from a raw value. It will be shown after enabling “Wheel speed setup”.

Properties:

- Enable** – enable custom wheel speed output data.
Output ID – CAN frame ID for the output data.
Update frequency – channels sending frequency.
Multiplier – multiplier for a conversion from a raw value.
Divider – divider for a conversion from a raw value.

Channels:

Byte	Channel	Data type	Range	Multiplier	Divider	Offset	Unit
Output ID (default: 0x649)							
0	FL speed raw	16bit unsigned	0 – 65535	1	1	0	-
2	FR speed raw	16bit unsigned	0 – 65535	1	1	0	-
4	RL speed raw	16bit unsigned	0 – 65535	1	1	0	-
6	RR speed raw	16bit unsigned	0 – 65535	1	1	0	-

FL speed raw	– front left wheel speed raw value.
FR speed raw	– front right wheel speed raw value.
RL speed raw	– rear left wheel speed raw value.
RR speed raw	– rear right wheel speed raw value.

7.4. Frequency setup

Category for setting up frequency calculation. There are settings for frequency value filter, which can have separate strengths for low and high speed with configurable threshold.

Properties:

Enable	– enable frequency calculation.
Filter enable	– enable filtering of the frequency value.
Filter strength	– frequency filter strength. Value from 0 to 9. Higher value means stronger filtering.
Low speed thr	– low speed threshold in Hz. Below that threshold “Low speed strength” will be used as a filter strength.
Low speed strength	– low speed filter strength. Overrides “Filter strength” value when frequency is below the “Low speed thr”.

7.4.1. Frequency output

Category for setting up frequency data output with custom conversion factors from a raw value. It will be shown after enabling “Frequency setup”.

Properties:

Output ID	– CAN frame ID for the output data.
Update frequency	– channels sending frequency.
Multiplier	– multiplier for a conversion from a raw value.
Divider	– divider for a conversion from a raw value.

Channels:

Byte	Channel	Data type	Range	Multiplier	Divider	Offset	Unit
Output ID (default: 0x64A)							
0	Sensor1 freq raw	16bit unsigned	0 – 65535	1	1	0	-
2	Sensor2 freq raw	16bit unsigned	0 – 65535	1	1	0	-
4	Sensor3 freq raw	16bit unsigned	0 – 65535	1	1	0	-
6	Sensor4 freq raw	16bit unsigned	0 – 65535	1	1	0	-

Sensor1 freq raw – sensor 1 frequency raw value.

Sensor2 freq raw – sensor 2 frequency raw value.

Sensor3 freq raw – sensor 3 frequency raw value.

Sensor4 freq raw – sensor 4 frequency raw value.

7.5. Edge counter setup

Category for setting up edge counter of the input signal.

Properties:

Enable – enable edge counter.

Edge type – select which edges should be counted. Can be selected from “Rising”, “Falling” or “Both”.

7.5.1. Edge counter output

Category for setting up edge counter data output with custom conversion factors from a raw value. It will be shown after enabling “Edge counter setup”.

Properties:

Output ID – CAN frame ID for the output data.

Update frequency – channels sending frequency.

Multiplier – multiplier for a conversion from a raw value.

Divider – divider for a conversion from a raw value.

Channels:

Byte	Channel	Data type	Range	Multiplier	Divider	Offset	Unit
Output ID (default: 0x64B)							
0	Sensor1 count raw	16bit unsigned	0 – 65535	1	1	0	-
2	Sensor2 count raw	16bit unsigned	0 – 65535	1	1	0	-
4	Sensor3 count raw	16bit unsigned	0 – 65535	1	1	0	-
6	Sensor4 count raw	16bit unsigned	0 – 65535	1	1	0	-

Sensor1 count raw – sensor 1 counted edges raw value.

Sensor2 count raw – sensor 2 counted edges raw value.

Sensor3 count raw – sensor 3 counted edges raw value.

Sensor4 count raw – sensor 4 counted edges raw value.

7.6. Encoder setup

Category for setting up encoders connected to the module. Incremental (quadrature) encoder signal consist of 2 separate frequency signals with phase offset between them. Two separate encoders can be connected simultaneously.

Properties:

Enable – enable encoder reader.

Encoder 1 dir invert – invert direction information for the first encoder.

Encoder 2 dir invert – invert direction information for the second encoder.

7.6.1. Encoder output

Category for setting up encoder data output. It will be displayed after enabling “Encoder setup”.

Properties:

Output ID – CAN frame ID for the output data.

Update frequency – channels sending frequency.

Channels:

Byte	Channel	Data type	Range	Multiplier	Divider	Offset	Unit
Output ID (default: 0x64C)							
0	Encoder 1 dir	8bit unsigned	0 – 1	1	1	0	-
2	Encoder 2 dir	8bit unsigned	0 – 1	1	1	0	-

Encoder 1 dir – direction value of the first encoder.

Encoder 2 dir – direction value of the second encoder.

7.7. Diagnostics

Category for setting up diagnostic information output. This additional information about the measured signal can help with setting up or diagnosing sensors.

Properties:

Enable – enable diagnostic data output.

Output ID – CAN frame ID for the output data.

Update frequency – channels sending frequency.

Channels:

Byte	Channel	Data type	Range	Multiplier	Divider	Offset	Unit
Diagnostics ID (default: 0x700)							
0	S1 input voltage	8bit unsigned	0 – 8.4	100	3035	0	V
1	S2 input voltage	8bit unsigned	0 – 8.4	100	3035	0	V
2	S3 input voltage	8bit unsigned	0 – 8.4	100	3035	0	V
3	S4 input voltage	8bit unsigned	0 – 8.4	100	3035	0	V
4	S1 comparator voltage	8bit unsigned	0 – 3.3	1	77	0	V
5	S2 comparator voltage	8bit unsigned	0 – 3.3	1	77	0	V
6	S3 comparator voltage	8bit unsigned	0 – 3.3	1	77	0	V
7	S4 comparator voltage	8bit unsigned	0 – 3.3	1	77	0	V

Byte	Channel	Data type	Range	Multiplier	Divider	Offset	Unit
Diagnostics ID+1 (default: 0x701)							
0	S1 comparator min	8bit unsigned	0 – 3.3	1	77	0	V
1	S2 comparator min	8bit unsigned	0 – 3.3	1	77	0	V
2	S3 comparator min	8bit unsigned	0 – 3.3	1	77	0	V
3	S4 comparator min	8bit unsigned	0 – 3.3	1	77	0	V
4	S1 comparator max	8bit unsigned	0 – 3.3	1	77	0	V
5	S2 comparator max	8bit unsigned	0 – 3.3	1	77	0	V
6	S3 comparator max	8bit unsigned	0 – 3.3	1	77	0	V
7	S4 comparator max	8bit unsigned	0 – 3.3	1	77	0	V
Diagnostics ID+2 (default: 0x702)							
0	S1 frequency	16bit unsigned	0 – 6553.5	1	10	0	Hz
2	S2 frequency	16bit unsigned	0 – 6553.5	1	10	0	Hz
4	S3 frequency	16bit unsigned	0 – 6553.5	1	10	0	Hz
6	S4 frequency	16bit unsigned	0 – 6553.5	1	10	0	Hz
Diagnostics ID+3 (default: 0x703)							
0	S1 period	32bit unsigned	0 – 4.29*10 ⁸	1	10	0	µs
4	S2 period	32bit unsigned	0 – 4.29*10 ⁸	1	10	0	µs
Diagnostics ID+4 (default: 0x704)							
0	S3 period	32bit unsigned	0 – 4.29*10 ⁸	1	10	0	µs
4	S4 period	32bit unsigned	0 – 4.29*10 ⁸	1	10	0	µs
Diagnostics ID+5 (default: 0x705)							
0	S1 low time	32bit unsigned	0 – 4.29*10 ⁸	1	10	0	µs
4	S2 low time	32bit unsigned	0 – 4.29*10 ⁸	1	10	0	µs
Diagnostics ID+6 (default: 0x706)							
0	S3 low time	32bit unsigned	0 – 4.29*10 ⁸	1	10	0	µs
4	S4 low time	32bit unsigned	0 – 4.29*10 ⁸	1	10	0	µs
Diagnostics ID+7 (default: 0x707)							
0	S1 high time	32bit unsigned	0 – 4.29*10 ⁸	1	10	0	µs
4	S2 high time	32bit unsigned	0 – 4.29*10 ⁸	1	10	0	µs
Diagnostics ID+8 (default: 0x708)							
0	S3 high time	32bit unsigned	0 – 4.29*10 ⁸	1	10	0	µs
4	S4 high time	32bit unsigned	0 – 4.29*10 ⁸	1	10	0	µs

SX input voltage	– sensor voltage at the input pins.
SX comparator voltage	– voltage at the internal comparator input, comparator switches between high and low states depending on the “Logic threshold” setting.
SX comparator min	– minimum voltage at the internal comparator input.
SX comparator max	– maximum voltage at the internal comparator input.
SX frequency	– sensor signal frequency.
SX period	– sensor signal period.
SX low time	– sensor signal low time.
SX high time	– sensor signal high time.

**X is the device sensor input number from 1 to 4*

8. Use cases

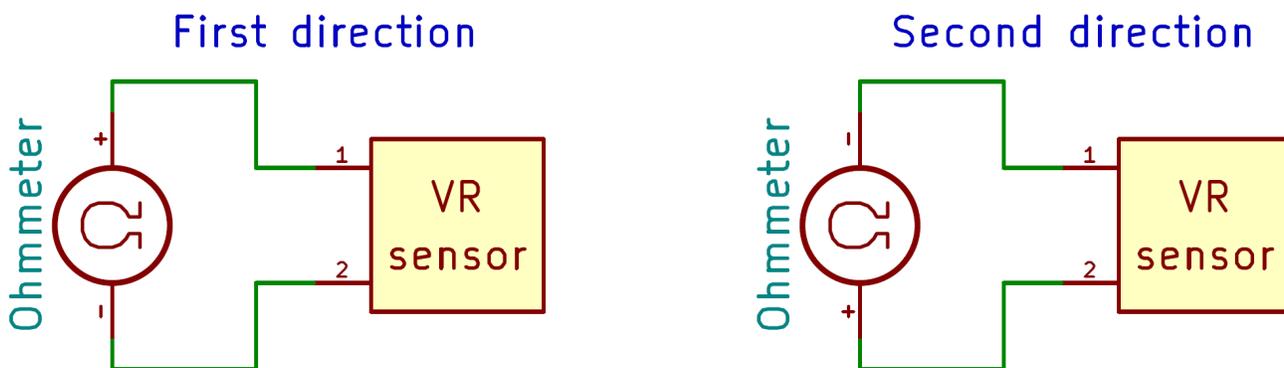
8.1. Using VR sensors

8.1.1. Sensor description

VR sensor is an abbreviation for a variable reluctance sensor. They are also sometimes called magnetic or inductive sensors. VR sensor consists of a wire coil on a permanent magnet or ferromagnetic core. This coil is directly connected to a 2 pin connector. Changes in the magnetic field around the coil generate voltage which can be measured. Output voltage is similar to sine wave with negative and positive amplitude. The main disadvantage is that signal voltage is proportional to the frequency, which creates problems with measuring slow signals. This sensor type is also the most susceptible to electrical interference.

8.1.2. Sensor type identification

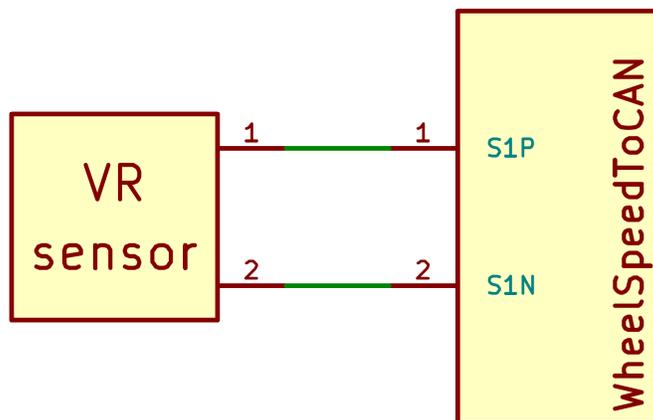
VR sensors will have 2 pin connector and relatively big size, since they are constructed from wire coil and metal core. Existence of the wire coil can be checked using ohmmeter. Coil resistance should be in the range of several hundred to several thousand ohms. Measure the resistance between the 2 pins in both directions. If values are the same in both directions and in the plausible range it probably is VR sensor. The best way to confirm the sensor type is to check the signal on an oscilloscope.



Both measurements have to show the same value between several hundred and several thousand ohms.

8.1.3. Sensor connection

These sensors must be connected directly to the positive and negative input pins (direction does not matter).



8.1.4. Sensor input configuration

- Logic threshold** – lower value allows lower speeds to be registered, but at some point noise will disturb the reading. Should be as low as possible while maintaining stable reading at low speeds.
- Double gain enable** – more gain allows to measure slower signals with low amplitude, but will also amplify the noise.
- Pull up/down** – choose “None”, as VR sensors do not require pull resistors.

8.2. Using HALL sensors

8.2.1. Sensor description

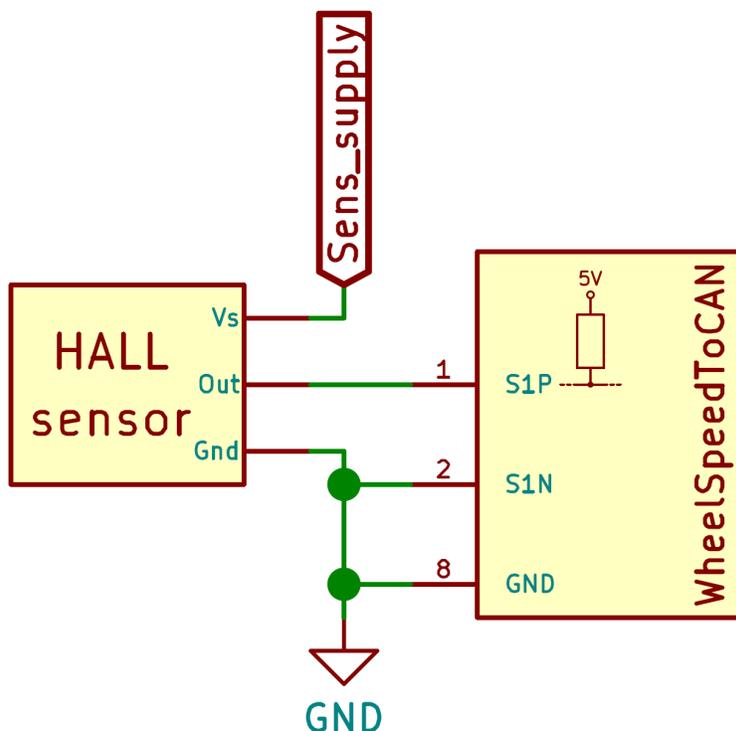
HALL sensor name comes from Hall effect that is used to detect changes in the magnetic field. HALL sensor consists of electronics which measure the hall effect over a conductor and provide binary output that depends on the applied magnetic field. Most common signal output type is an open collector/drain, which means the output is either not connected to anything (open) or connected to ground (closed). This requires pull up resistor between output and some voltage source (typ. 5V or 12V) to create square wave with two distinct voltage levels. Permanent magnet can also be a part of the sensor, which will then react to ferromagnetic materials or have no magnet and react to something magnetic. Connector will have 3 pins with power supply (typ. 5V or 12V), ground and output.

8.2.2. Sensor type identification

There is no special identification needed since this sensor has 3 pins, compared to other types of sensors which are 2 pin. Sensors with 3 pins will not necessarily use Hall effect, but in most cases they are active sensor, so 2 pins are for providing power and one pin is for signal output. Pinout description must be found for your specific sensor.

8.2.3. Sensor connection

Proper power supply for the specific sensor must to be connected to the power and ground pins. Output of the sensor should be connected to positive input pin of the module. Grounds between the sensor and the WheelSpeed module should also be connected. If the sensor output is open collector/drain and there is no pull up resistor in the circuit, there is internal pull up resistor in the module which can be enabled through software. Negative input pin of the module should be connected to ground.



8.2.4. Sensor input configuration

- Logic threshold** – check comparator min/max in diagnostic information while providing the signal. Set the threshold to the middle value between min and max.
- Double gain enable** – not needed. Set to disabled.
- Pull up/down** – if the sensor output is open collector/drain and there is no pull up resistor in the circuit, select “Positive, 1k, 5V”. Otherwise select “None”.

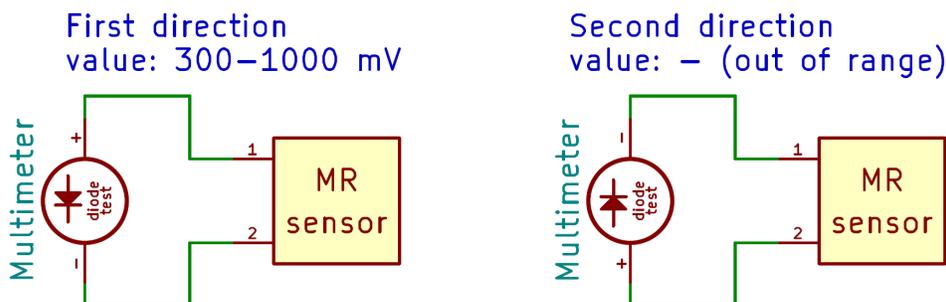
8.3. Using MR sensors

8.3.1. Sensor description

MR sensor is an abbreviation for magneto-resistive sensor. It is based on a material that changes resistance due to a magnetic field. Inside the sensor is electronic circuit that behaves like a current source, which is connected to a 2 pin connector. This current source outputs 2 different current values (typ. 7 and 14 mA) depending on the magnetic field. Output voltage is measured across the external resistor connected in series with the sensor. ABS systems usually use this type of sensor to measure wheel speed.

8.3.2. Sensor type identification

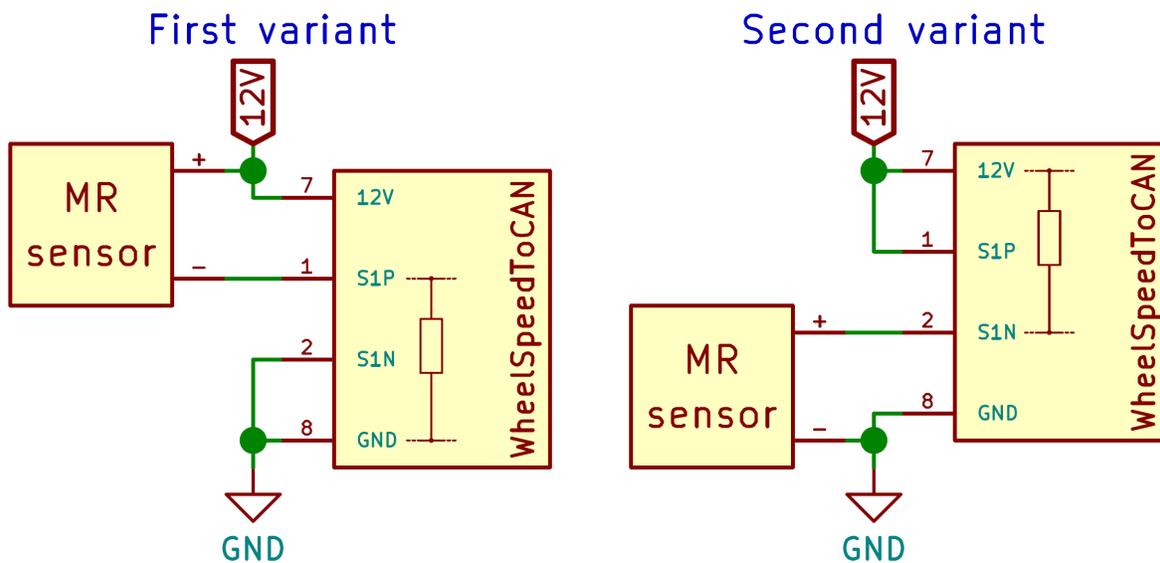
MR sensors have polarity, so positive and negative pins have to be identified. Some sensors have markings on a connector case (like a “+” sign). If there are no markings, polarity can be checked using multimeter with diode test. These sensors are protected against reverse polarity with a diode. Use diode test in both directions and the direction which shows diode voltage out of range is the correct one.



First measurement shows value between 300 and 1000 mV, which means this is the direction with protection diode. Second measurement shows value out of range so this is the correct polarization. In this case pin 1 is negative and pin 2 is positive.

8.3.3. Sensor connection

Power supply is typically car battery voltage (12V). Sensor must be connected in series with the resistor between power supply and ground. There are 2 different variants how the sensor can be connected. First variant is with sensor connected to power supply and resistor to ground. Second has sensor and resistor swapped. Output voltage is measured across the resistor. Voltage across the resistor can be calculated from ohms law. With typical sensor current values of 7 and 14mA and 220 ohm series resistor, voltage across the resistor changes between 1.54 and 3.08V. If the module is not connected to an existing circuit, the internal resistors can be used.



8.3.4. Sensor input configuration

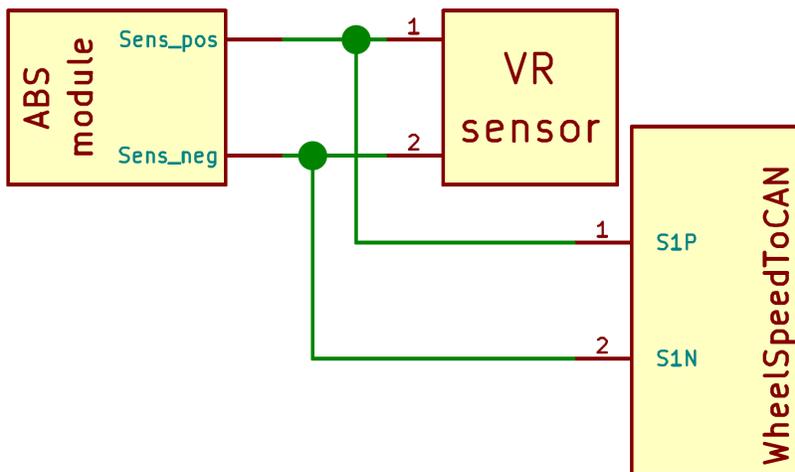
- Logic threshold** – check comparator min/max in diagnostic information while providing the signal. Set the threshold to the middle value between min and max.
- Double gain enable** – can be use to get bigger difference between voltage levels. Signal after gain can get saturated if original signal is too high.
- Pull up/down** – if the sensor will be used with the internal resistor, select “Positive, XR, GND” for first variant and “Negative, XR, 12V” for second variant. If connecting to an existing circuit select “None”.

8.4. Using the module with working ABS systems

This section describes how to connect the WheelSpeed module in parallel with a working ABS system to read the sensors and not disturb the original signal. There are different types of sensors used with ABS modules on the market, so the first step is to confirm the type used in this case (reference sections about using specific sensors). When using the WheelSpeed module to connect to already existing circuits, the pull resistor must be set to “None”.

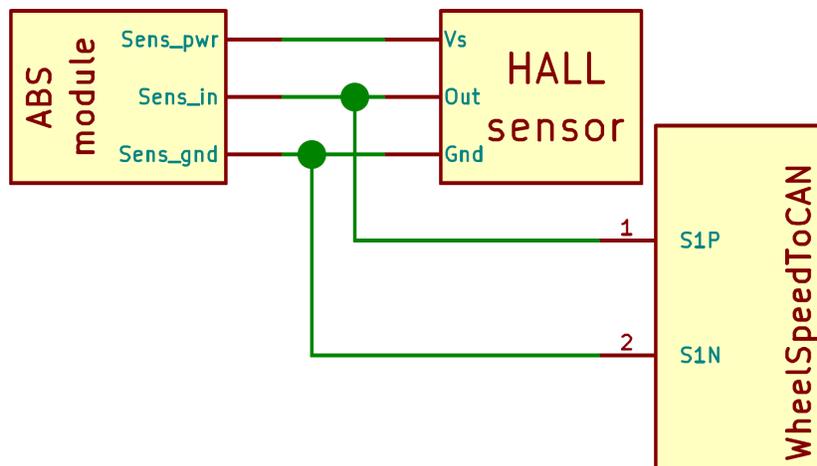
8.4.1. ABS with VR sensors

Sensor wires should be spliced into the positive and negative input of the module. Setup is similar to the setup of a directly connected sensor, so the section about using VR sensors should be referenced.



8.4.2. ABS with HALL sensors

Sensor output and ground wires should be spliced into the positive and negative input of the module. Setup is similar to the setup of a directly connected sensor, so the section about using HALL sensors should be referenced.



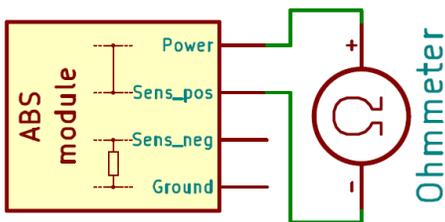
8.4.3. ABS with MR sensors

There are two variants on how the sensors can be connected (reference the section about using MR sensors). First step is to determine which variant is used in the ABS module and there are 2 methods to check this. One method requires pinout for the ABS module, the module to be disconnected and to make measurements with ohmmeter. The other method requires the whole system to be working, possibility to generate the signal (spin the wheels) and to make measurements with voltmeter.

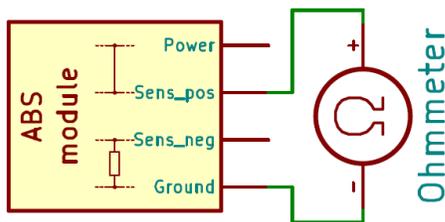
First method to determine the sensor connection variant:

This method comes down to directly finding the internal resistor for the sensor and if it is connected to power supply or ground. Prepare for making measurements by disconnecting the ABS module connector and finding pins for power supply, ground and sensor input. Next measure resistance from one of the sensor input pins to power supply and ground pin. Do the same for the second input pin. One of the input pins will be directly connected to power supply or ground and second pin will be connected to power supply or ground through a resistor. If the resistor is between input pin and ground it is first variant and with resistor from input to power supply it is second variant.

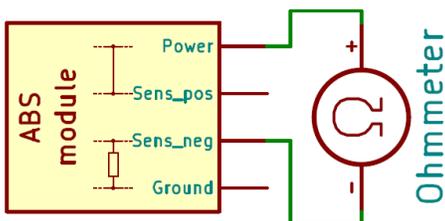
1st measurement
value: 0 Ω



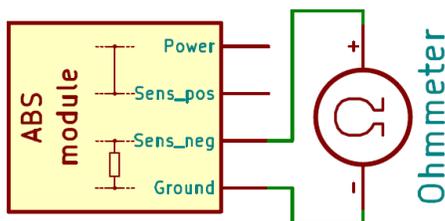
2nd measurement
value: - (out of range)



3rd measurement
value: - (out of range)



4th measurement
value: 100–500 Ω

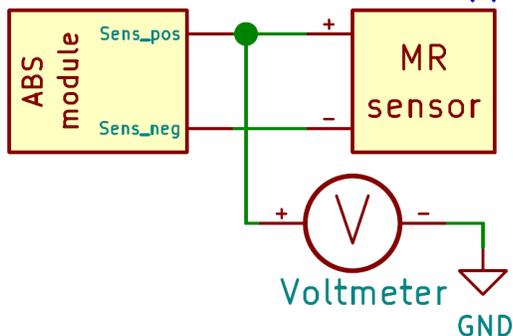


First measurement shows 0 Ω , so the positive input is connected directly to power supply. Second and third measurement doesn't show any connections. Fourth measurement shows 100–500 Ω , which means the resistor is between negative input and ground. That information describes variant number one.

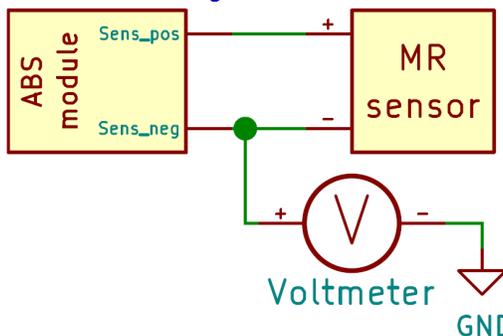
Second method to determine the sensor connection variant:

This method can show which variant is used by checking voltage levels at the sensor inputs. Prepare by turning on the whole system, having ability to spin the wheels (generate the signal) and making voltage measurements at sensor inputs in the ABS module. All voltage measurement must be done with respect to ground. One input will have constant voltage (power or ground) and the other one will have changing voltage levels while spinning the wheel (spin the wheel very slowly, so you can see the voltage change). If the changing voltage is closer to the ground level, it is first variant. If the changing voltage is closer to the supply level, it is second connection variant.

1st measurement
value: constant sensor supply



2nd measurement
value: changes between 1.5 and 3.0 V

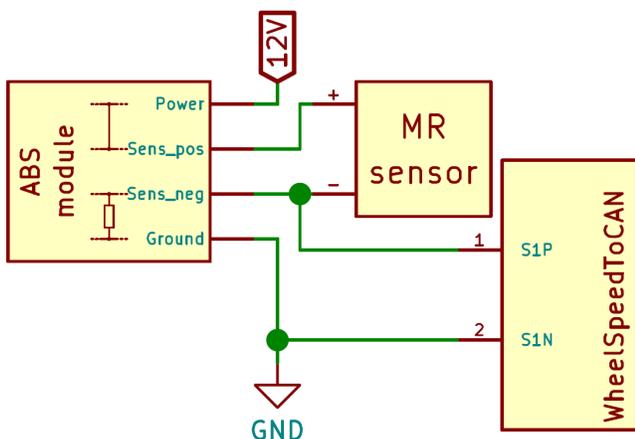


First measurement shows constant sensor supply voltage and second measurement shows voltage changing between 1.5 and 3.0 V. Changing voltage with values closer to the ground level means the resistor is connected to the ground, so this is a first variant.

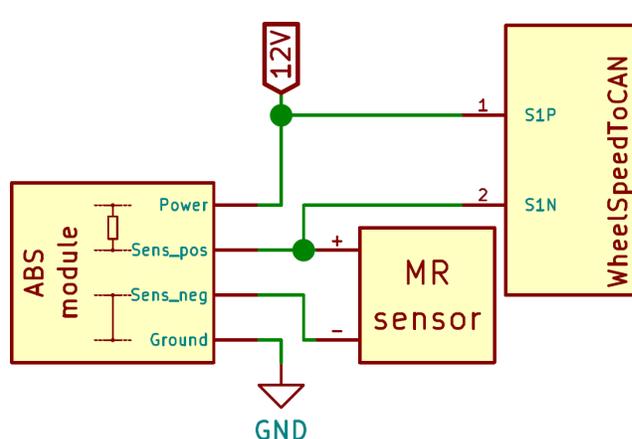
Connecting the WheelSpeed module to the ABS system with MR sensors:

After finding out which connection variant is used in the ABS system, the module can be connected like on the schematics below. Setup is similar to the setup of a directly connected sensor, so the section about using MR sensors should be referenced.

First variant



Second variant



8.5. Using the module with encoders

WheelSpeed module can read the direction of connected encoders. Two separate encoders can be read simultaneously. First encoder has to be connected to sensor inputs 1 & 2 and second encoder to sensor inputs 3 & 4.

9. Document revision history

Revision	Date	Changes
2.0	06-04-2022	- completely reworked manual for firmware version 16.x
1.0	14-06-2019	- official release, minor corrections
0.1	20-08-2018	- initial release