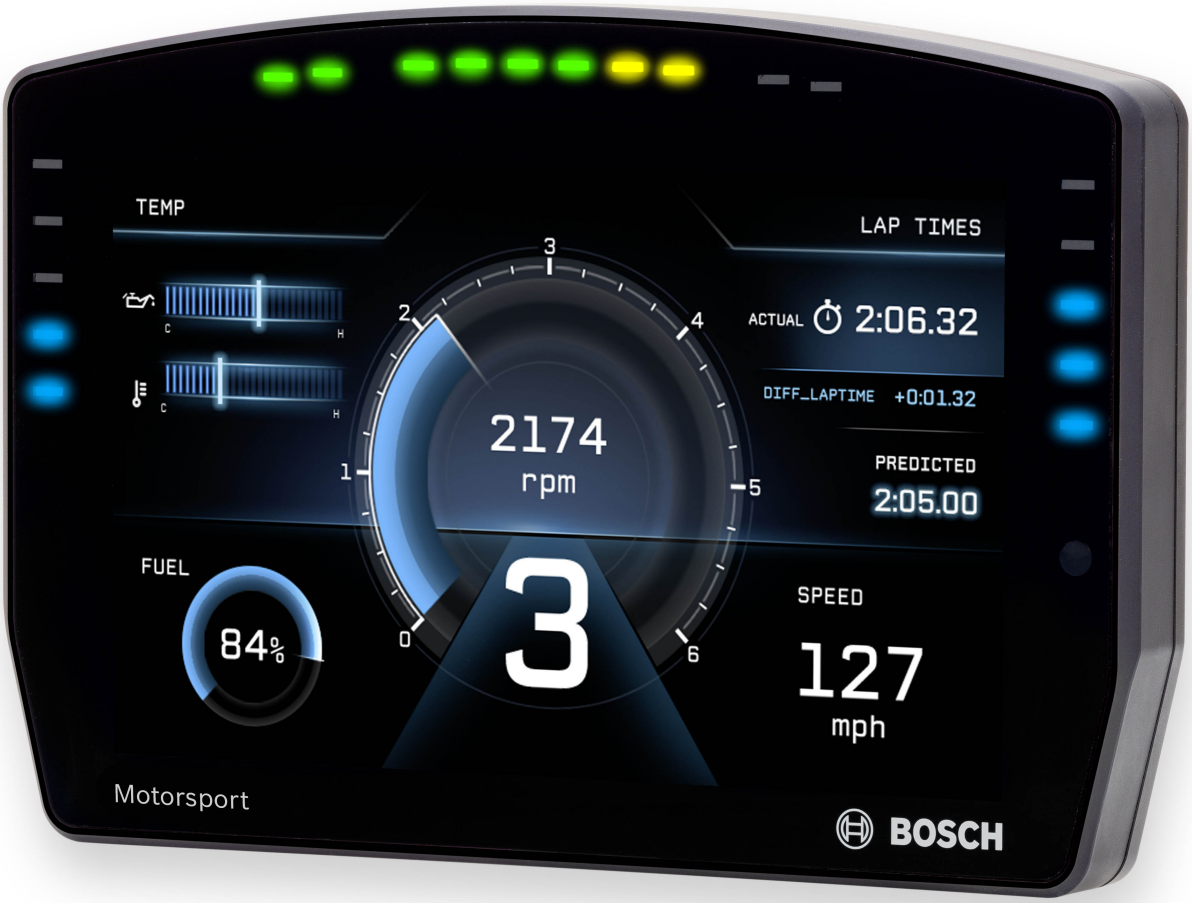
BOSCH



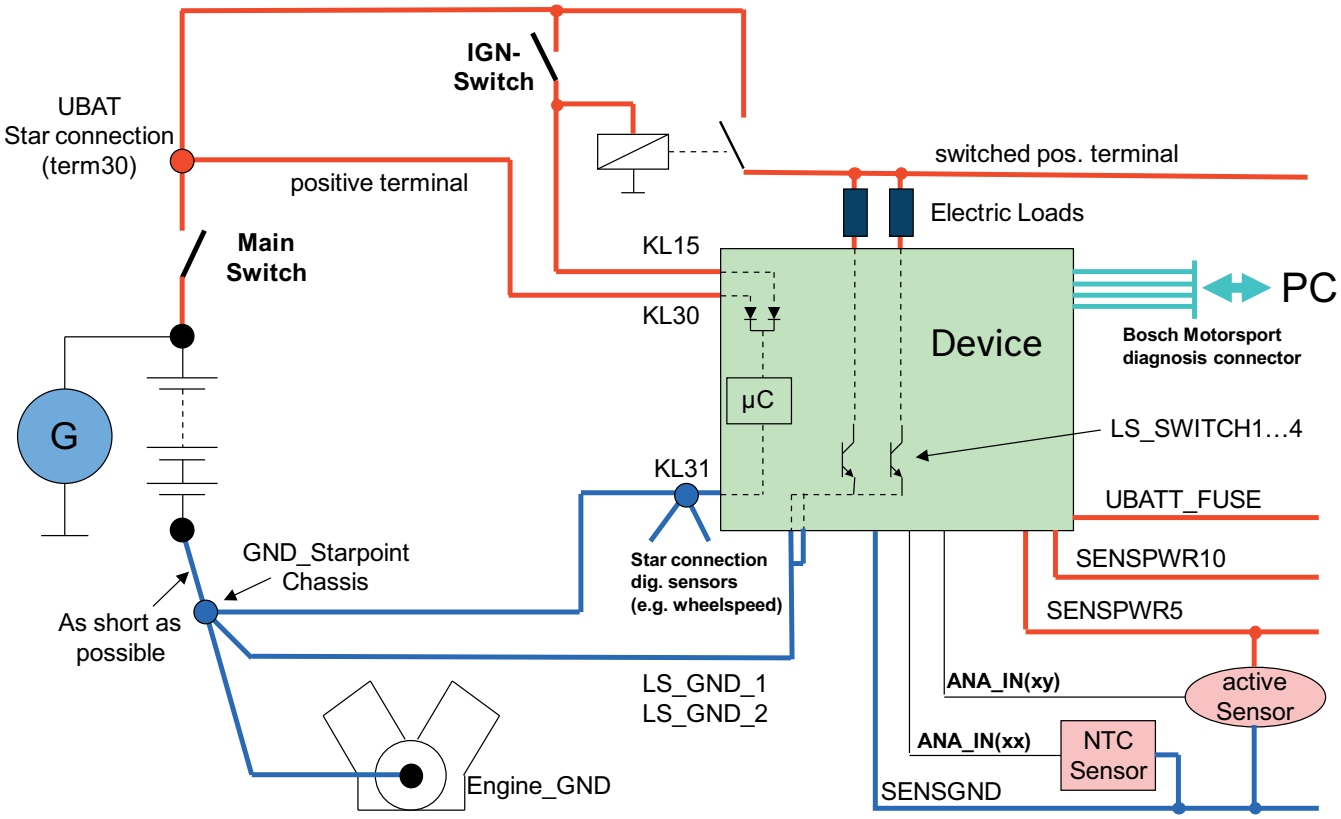
Display DDU 10

Manual

Version 1.0 18/03/2020

Content

1 Onboard Network Concept



NOTICE

This schematic is not device specific, please see the section “Technical Data for the specifications of your device.

2 Preparation

Use the DDU 10 only as intended in this manual. Any maintenance or repair must be performed by authorized and qualified personnel approved by Bosch Motorsport.

Operation of the DDU 10 is only certified with the combinations and accessories that are specified in this manual. The use of variant combinations, accessories, and other devices outside the scope of this manual are only permitted when they have been determined to be compliant from a performance and safety standpoint by a representative from Bosch Motorsport.

Read the manual carefully and follow the application hints step by step. Do not hesitate to contact us, contact data can be found on the last page of this document.

Important information on Electromagnetic Conformity

To avoid unwanted interference with the environment (people, animals, electronic devices) or unwanted harm to the environment, it is mandatory that the user of the DDU 10 carries out an appropriate analysis to determine the electromagnetic interaction the DDU 10 may have with its individual installation environment.

Disclaimer

Due to continuous enhancements, we reserve the rights to change any illustrations, photos and technical data within this manual.

Please retain this manual for your records.

NOTICE

In this document, all screenshots are created by way of example for a display. Please consider this and replace the product names with the name of your device.

3 Power Supply

Please ensure that you have a good ground installation. That means:

- A ground that has a solid, low resistance connection to the negative battery terminal
- Connection should be free from dirt, grease, paint, anodizing, etc.
- Use large diameter wire
- More metal-to-metal contact is better!

The following notations for power signals are used:

- KL 15 is a switched battery rail controlled by the IGN-switch
- KL 30 is an unswitched battery positive rail (same as battery positive terminal)
- KL 31 is an unswitched ground rail (same as battery negative terminal)

Be careful to observe current limits of wires and connector pins!

4 Starting Up

4.1 Before starting

Install the software required for the operation of the DDU 10 . It is developed for Windows system software. The following software versions are used in this manual:

- DDU 10 setup, configuration and calibration: RaceCon Version 2.6.
- Measurement data analysis: WinDarab V7

Set up the 100 Mbit ethernet connection to the DDU 10.

- The ethernet port has “cable auto crossover” functionality

4.1.1 Starting the unit

The DDU 10 powers up by turning on the ignition of the car. At startup the DDU 10 will display a Bosch logo.

After a moment the DDU 10 shows a display element screen.

4.1.2 Connecting the unit to RaceCon

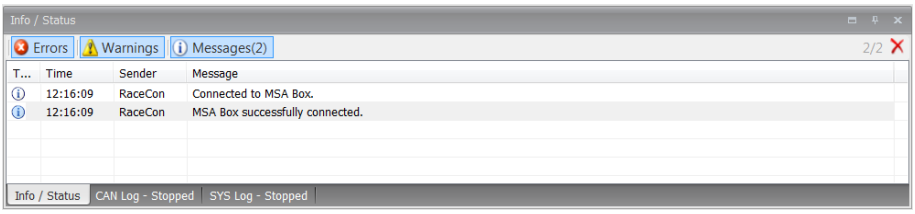
For testing new device configurations, you can connect the device to your computer via MSA-Box or ethernet cable.

Connection via MSA-Box

1. Reassure that the MSA-Box is installed properly on your computer. If needed, download the MSA-Box Diver from www.bosch-motorsport.com.
2. Connect an ethernet line of the device to the ethernet line of the MSA-Box.

   Please note, that the MSA-Box also requires power supply on the MSA-Box connector of your wiring loom.
3. Open RaceCon and connect the MSA-Box to the computer.

   In the ‘Info / Status’ Box of RaceCon you will receive messages that the connection was successful.



4. Reassure that the device is switched on.
5. ‘Link LED’ at the computer’s network adapter will illuminate.

   If the LED is off, check the wiring harness.

   After you created a RaceCon project with the device, the status icon of the device will switch from grey to one of the following colors: red, orange, green. For further information on how to set up a project, see the chapter “Setting up a new RaceCon Project [▸ 9]”. For the status color, see chapter “Color indication [▸ 22]”.

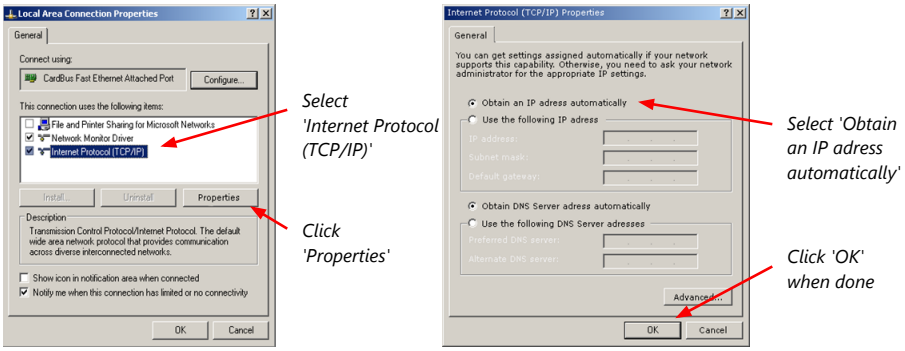
Connection via Ethernet Cable

1. Follow the steps above for a connection via MSA-Box.
2. Instead of connecting the ethernet line to the MSA-Box, Connect the ethernet directly to your computer.

Troubleshooting while setting up the network interface

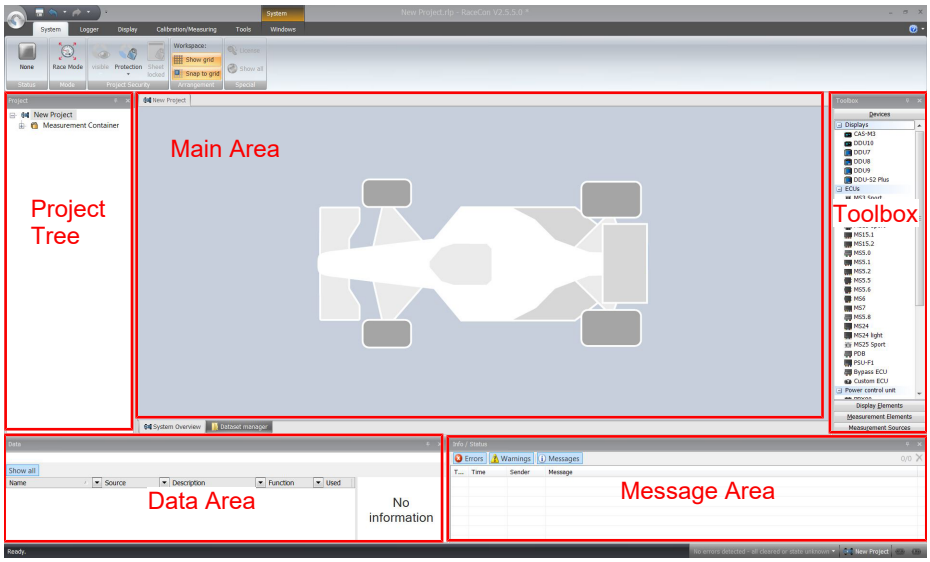
The DDU 10 contains a DHCP server, network addresses can be assigned automatically to the configuration PC. In case of problems during the network connection, please try the following steps:

1. Switch off the PC's firewall.
2. Reconfigure the PC or the MSA-Box network interface settings, to obtain an IP address automatically as shown in the pictures below.

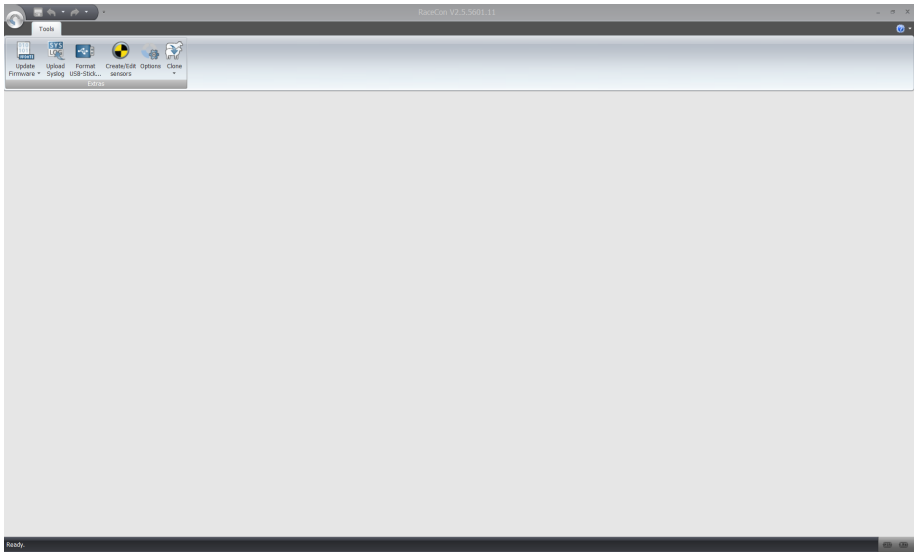


4.1.3 Setting up a new RaceCon Project

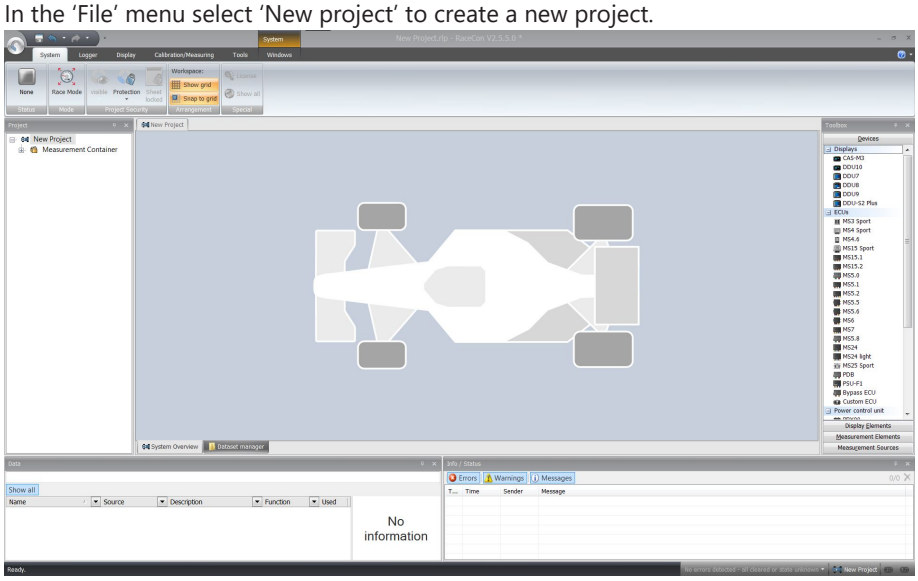
The following screenshot shows an overview of the RaceCon Main Screen with its areas. All (sub-) windows are resizable and dockable. You can find them under the 'Windows' tab.



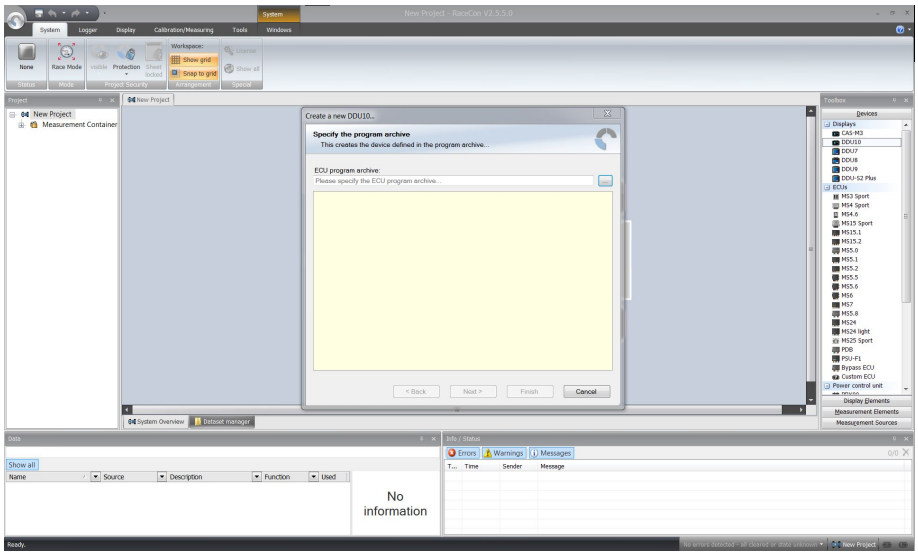
1. Start the RaceCon software.



2. In the ‘File’ menu select ‘New project’ to create a new project.



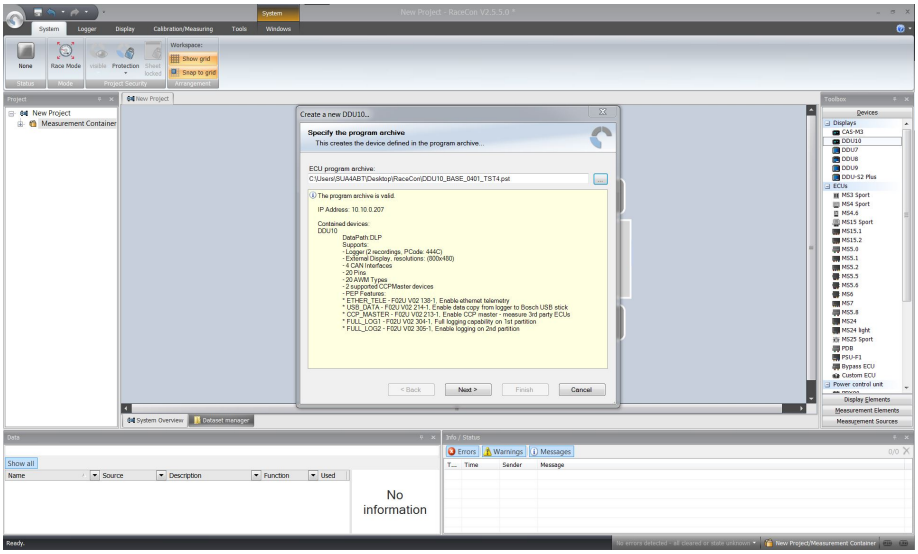
3. In the Toolbox select the DDU 10 and drag it into the Main Area. A pop up window to specify the DDU 10 program archive appears.



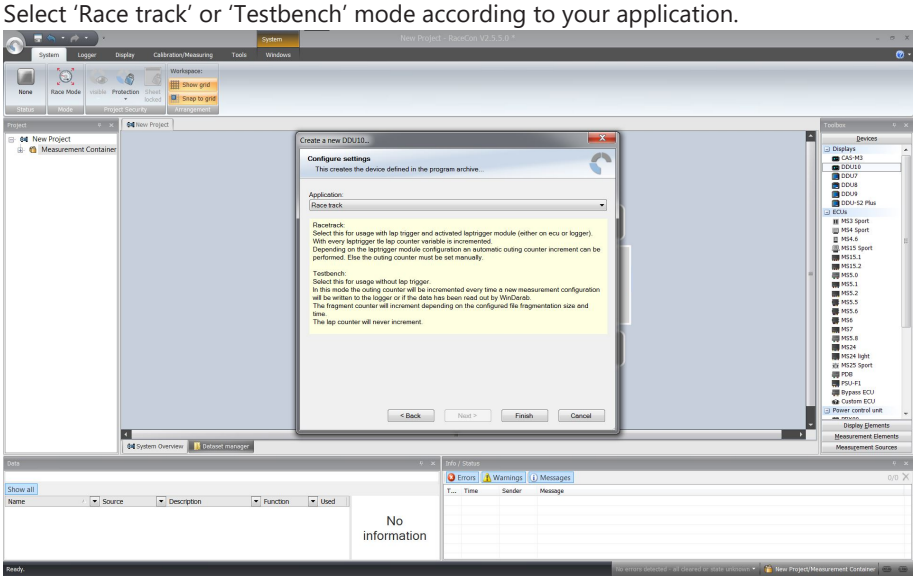
4. Download the firmware for the DDU 10 from www.bosch-motorsport.com.

   An information shows if the archive is valid or not.

5. Click ‘Next’.

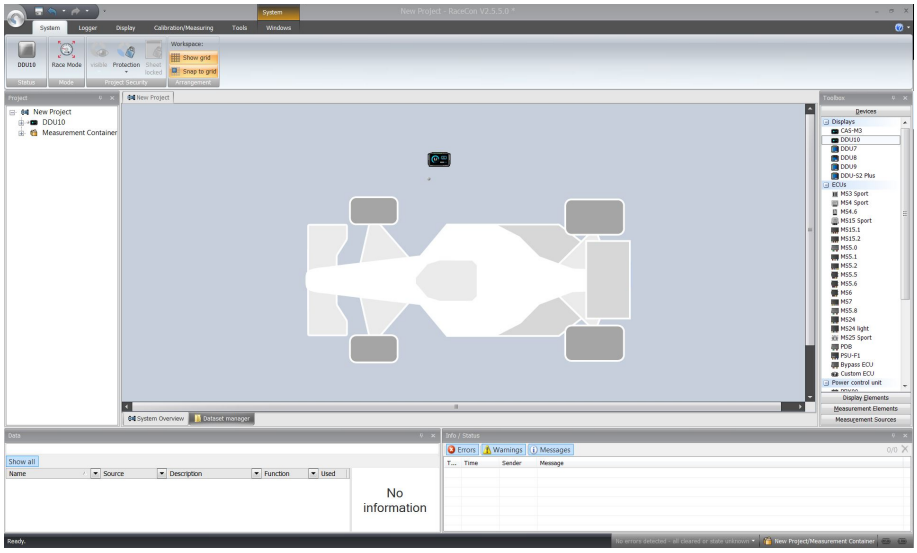


6. Select ‘Race track’ or ‘Testbench’ mode according to your application.



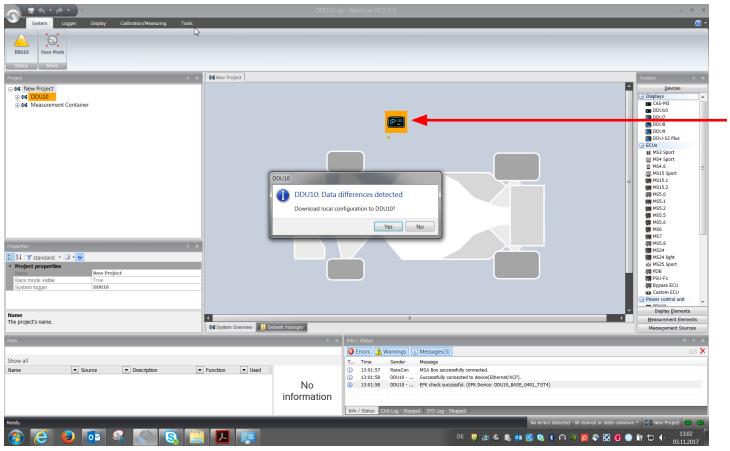
7. Click ‘Finish’.

   The DDU 10 is inserted into the project and RaceCon tries to connect to the device.



RaceCon detects configuration differences between the DDU 10 and the RaceCon project and asks for permission for data download.

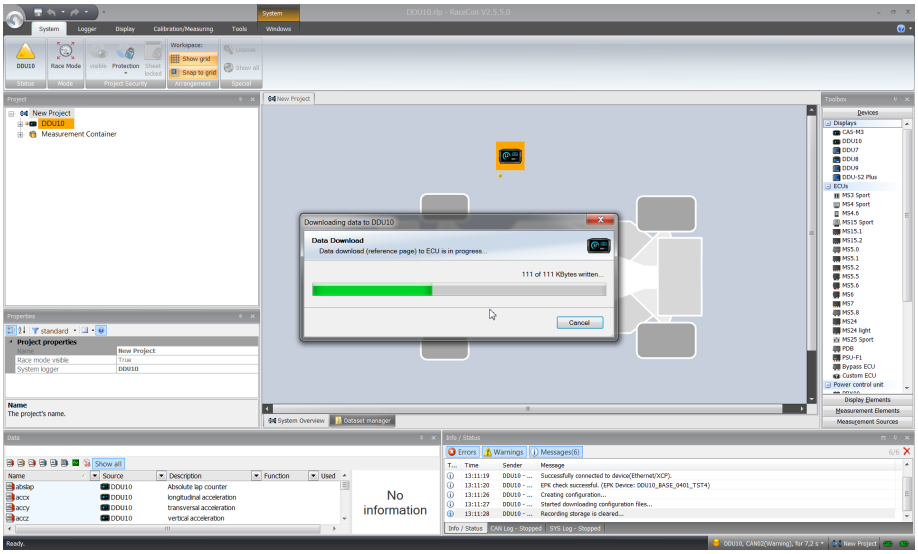
8. Click ‘Yes’ to download the configurations to the device or ‘No’ to continue without downloading the data.



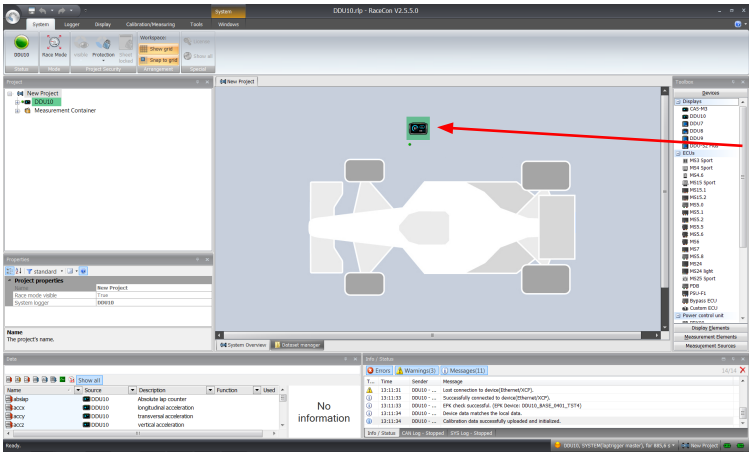
Successful ethernet connection, DDU "talks" to PC

If the device turns red, you might need to do a firmware update on the device. For more information see chapter “Firmware update [▶ 177]”.

The download starts and the DDU 10 carries out a reset.



After the reset, RaceCon reconnects to the DDU 10. Local configuration on both the PC and DDU 10 match (indicated by green background and dot). The DDU 10 is now connected to RaceCon.



Green background and dot indicate matching configuration

For further information on the color indication, see chapter “Color indication [▶ 22]”.

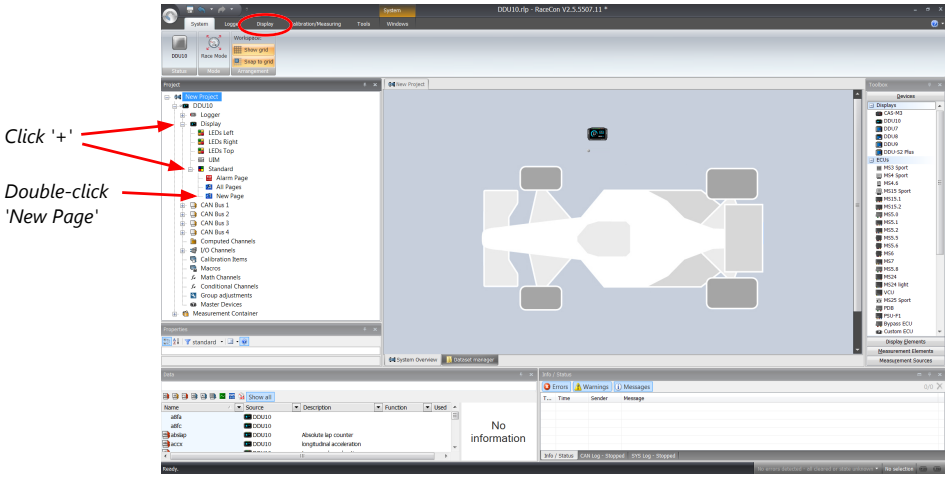
4.2 First display configuration (Quick Start)

This chapter explains the configuration of a display element showing the battery voltage.

See chapter Display element configuration for a detailed instruction to configure display elements.

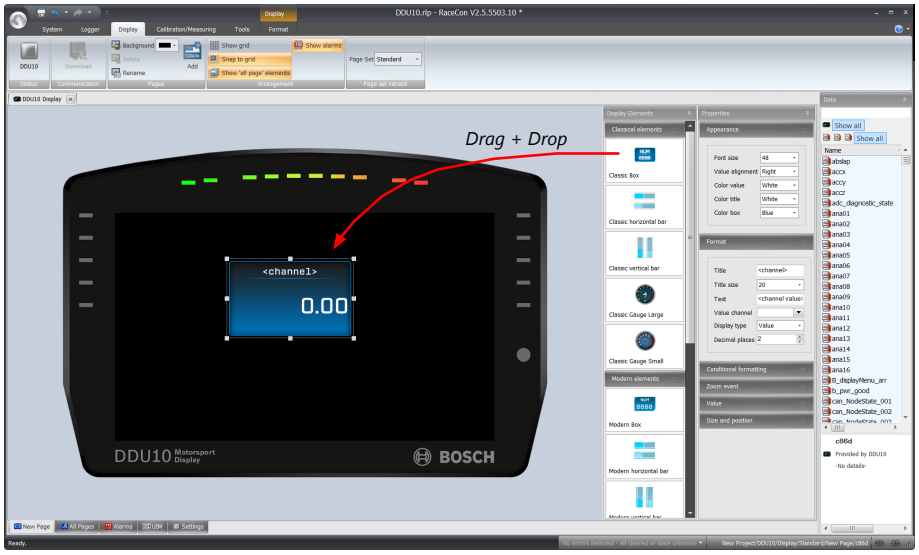
1. Click on ‘+’ to expand the DDU 10 project tree.
2. Click on ‘+’ to expand ‘Display’.
3. Double-click on ‘New Page’, or click on the ‘Display’ tab.

   RaceCon changes to the page ‘Display’ to open the DDU 10 display configuration area.

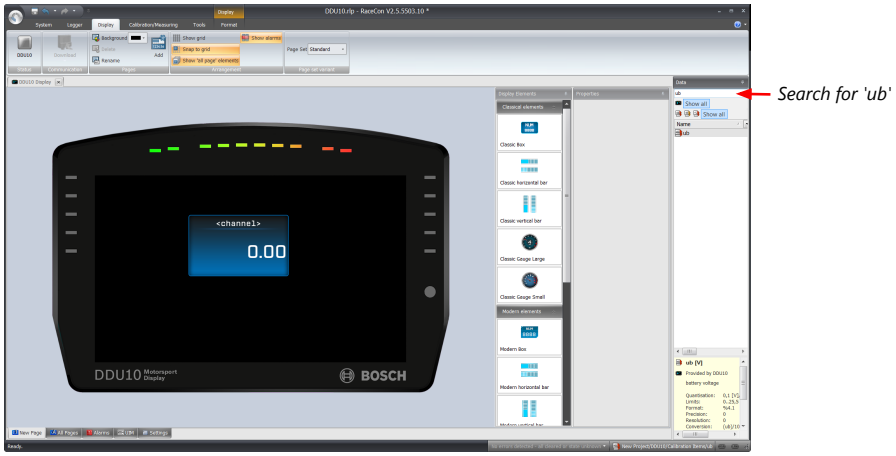


4. Drag any display element from the Toolbox and drop it on the display page.

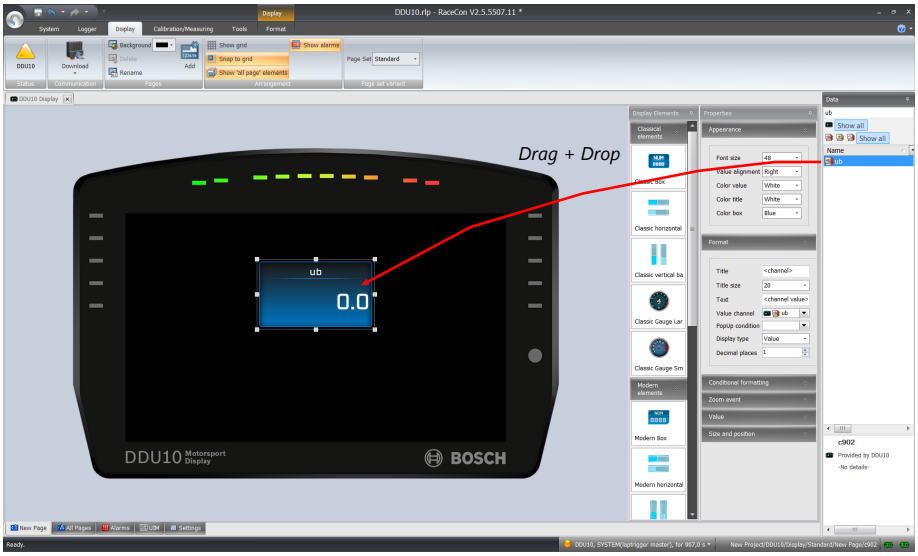
   The status signal in the upper left corner switches from green to orange, because the configuration in the tool differs from the configuration of the device.



5. Use the search bar in the 'Data' window, to search for 'ub' (measurement channel for battery voltage).

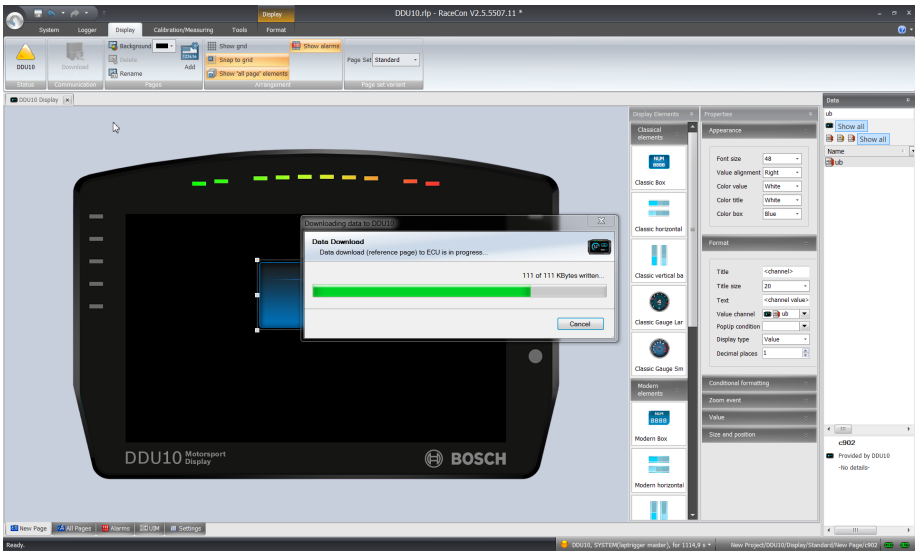


6. Drag the 'ub' measurement channel from the 'Data' window and drop it on the display element.



7. Click on the ‘Download’ button in the upper left corner.

   The configuration download starts and the DDU 10 carries out a reset. The status signal in the upper left corner switches to green.

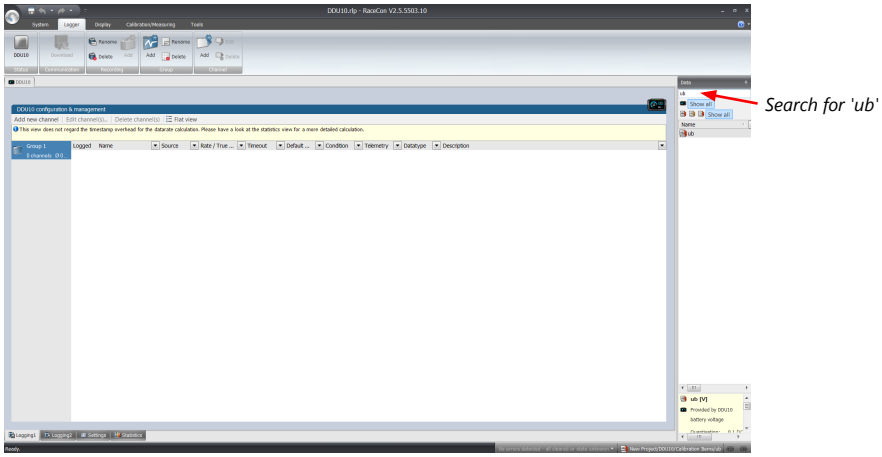


   The value of the battery voltage is displayed on the DDU 10.

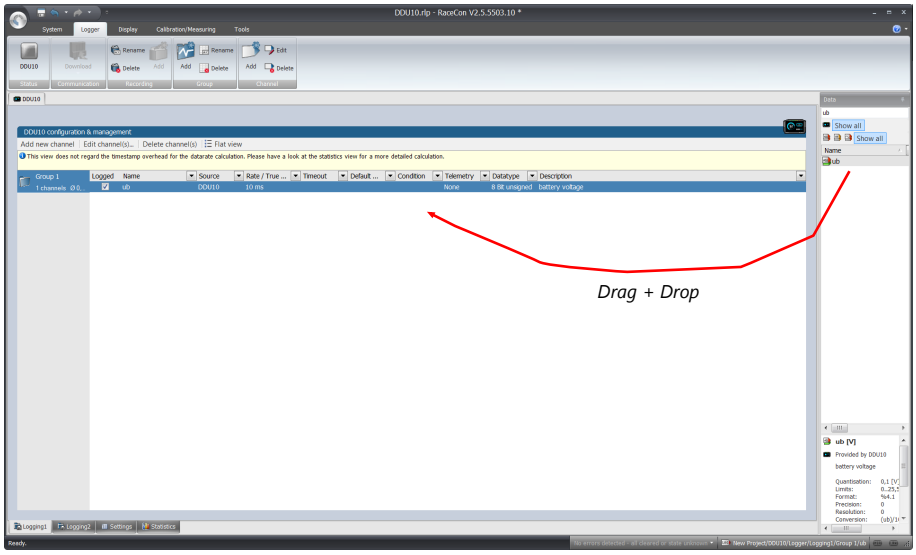
4.3 First recording (Quick Start)

This chapter explains the configuration of the recording of the battery voltage channel. See chapter ‘Recording’ for a detailed instruction to configure recordings.

1. Click on the ‘Logger’ tab to go to the page ‘Logger’.
2. Use the search bar in the ‘Data’ window, to search for ‘ub’ (measurement channel for battery voltage).



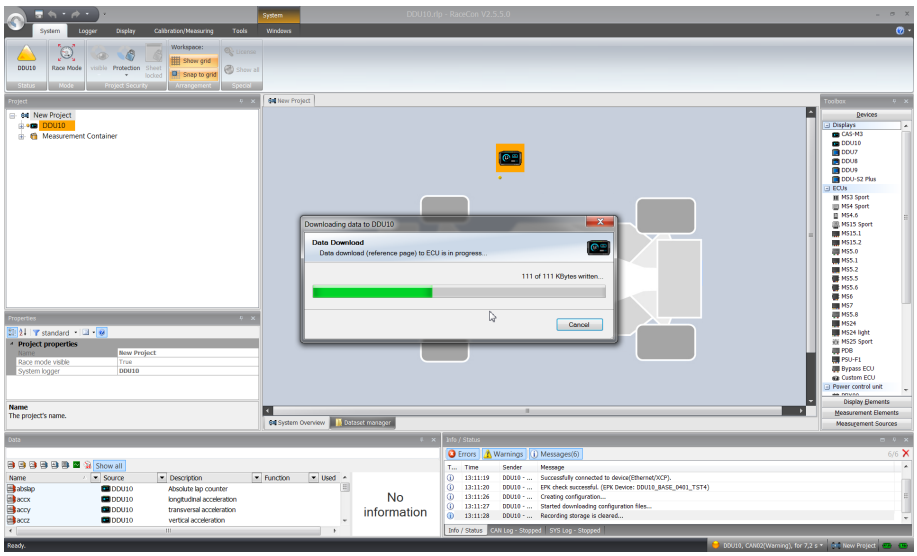
3. Drag and drop the ‘ub’ measurement channel into the recording area.



4. Click on the ‘Download’ button in the upper left corner.

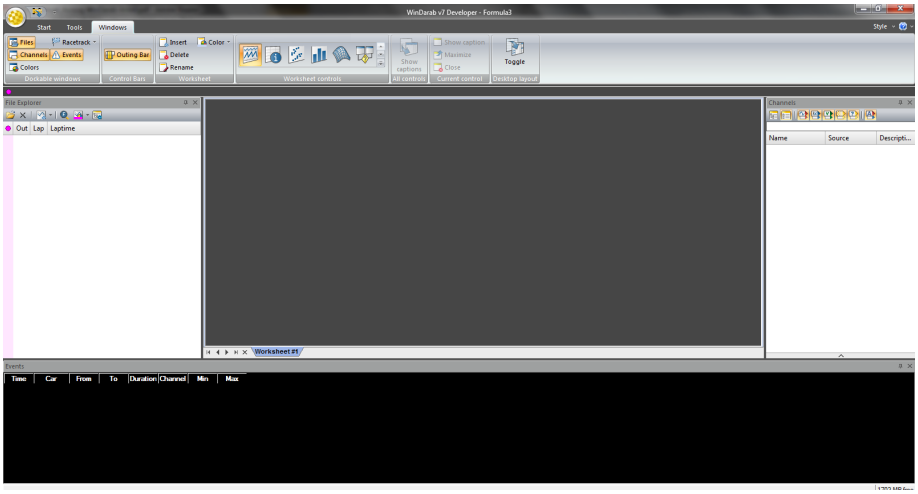
   The configuration download starts and the DDU 10 carries out a reset.

   Now you can find the ‘ub’ measurement channel in the ‘Data Area’.



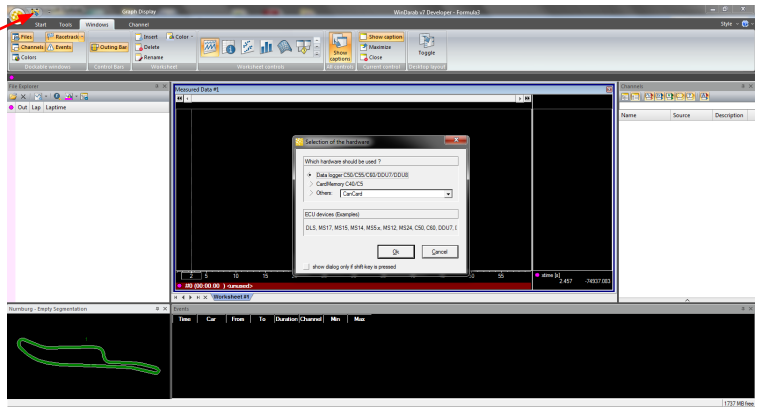
   As we did not define global start conditions, recording starts immediately.

5. Start the WinDarab software.



6. Disconnect the DDU 10 network cable.
7. Click on the ‘Read Data from Logging Device’ icon.
8. Choose your logger and click ‘OK’ when done.

Click 'read data from logging device'

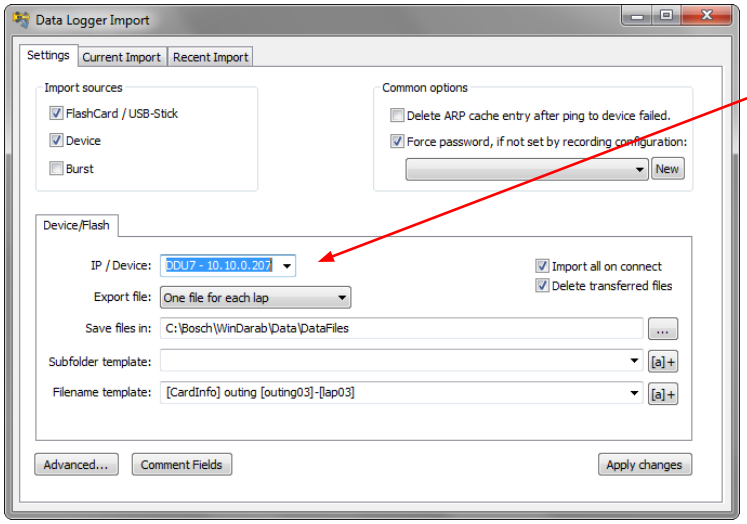


The ‘Data Logger Import’ dialog opens.

NOTICE

Refer to the WinDarab V7 manual for instructions on how to use the ‘Data Logger Import’ dialog and for more detailed descriptions and instructions.

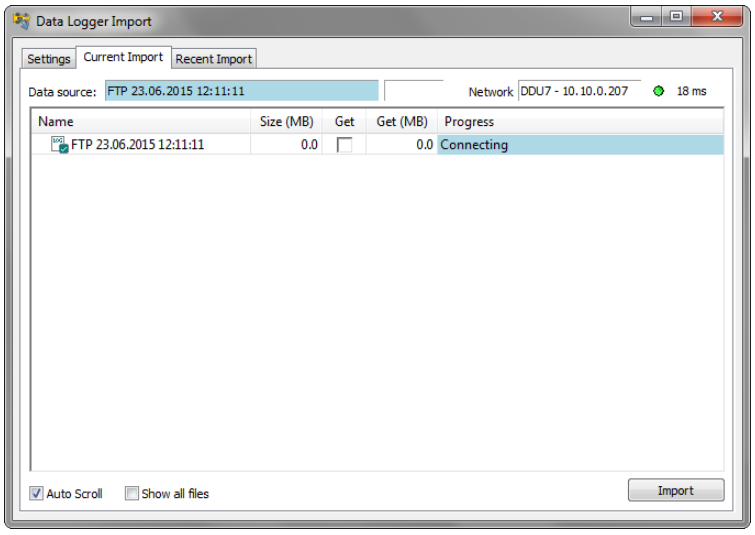
9. Choose the device and the IP address for the device.
10. Click ‘Apply changes’ when done.



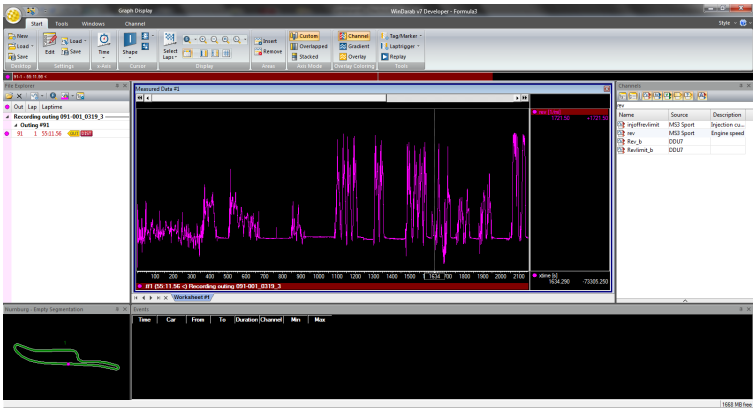
Choose your DDU from dropdown list

11. Connect the DDU 10 network cable.
12. Click on the ‘Current Import’ tab.
13. Click on ‘Import’ in the lower right corner.

    If the ‘Import all on connect’ box is checked, the data transmission from the DDU 10 starts automatically. Measurement files are stored automatically in the folder defined under ‘Settings’.



14. Click on ‘Close’ when the transmission has finished.
15. Click on the Start button and choose ‘Open measurement file’.
16. Select the measurement files from the storage folder.
17. Click on ‘Open’.
18. Click on ‘New Desktop’ to open a new measurement data window.
19. Drag the ‘ub’ measurement channel from the channel list and drop it into the measurement data window. The ‘ub’ measurement channel’s graph is displayed.

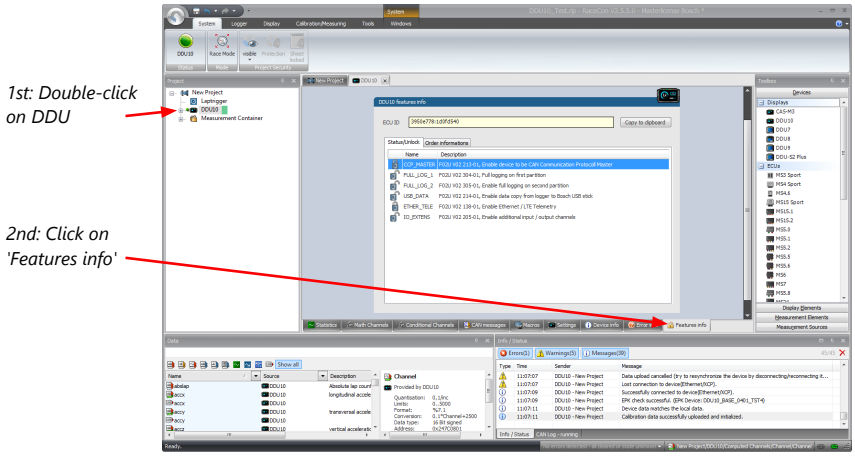


4.4 Feature activation

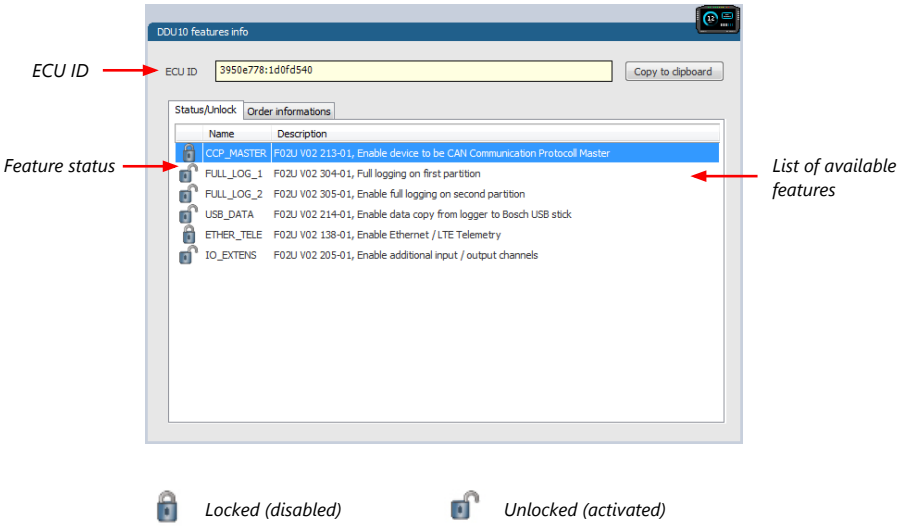
- Optional software feature packages are available for the DDU 10 .
- All software feature packages can be purchased prior to delivery or after you have received your device.
- If you have purchased an optional software feature package, it must be activated before it becomes operational.
- The feature activation status is stored permanently in the device and requires activating once only.
- As the activation key is device specific, a key delivered with one DDU 10 does not work on any other DDU 10 .
- When purchasing a software feature package, you have to tell Bosch the ECU ID code.

  The ECU ID code is device specific and can be found in the ‘features info’ window, shown in the screenshots below.
- If you have not purchased an optional software feature package, the next steps can be skipped.

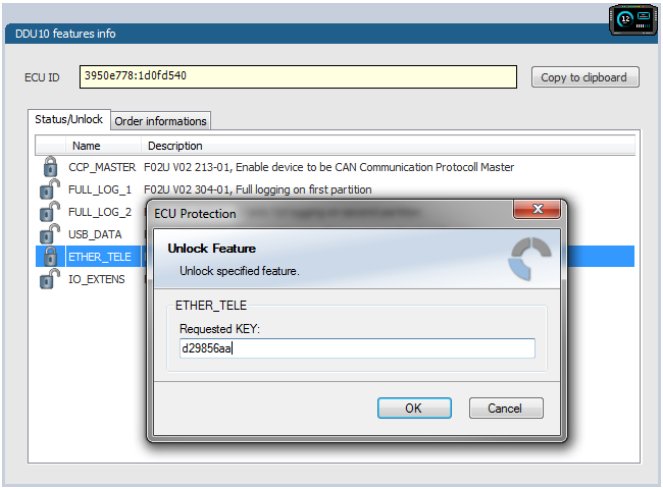
1. Ensure a connection to the device.
2. To activate a feature, double-click on ‘DDU 10’ in the Project Tree.
3. Click on the ‘Features info’ tab in the Main Area.



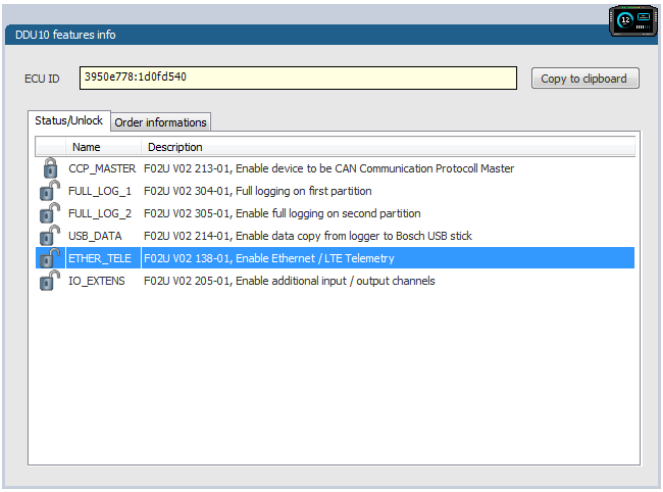
The ‘DDU 10 features info’ window appears.



4. Double-click on the feature you want to activate. A feature unlock window appears.



5. Enter the activation key you received for this feature on this device and click 'OK' when done. The feature's status changes to 'unlocked'.



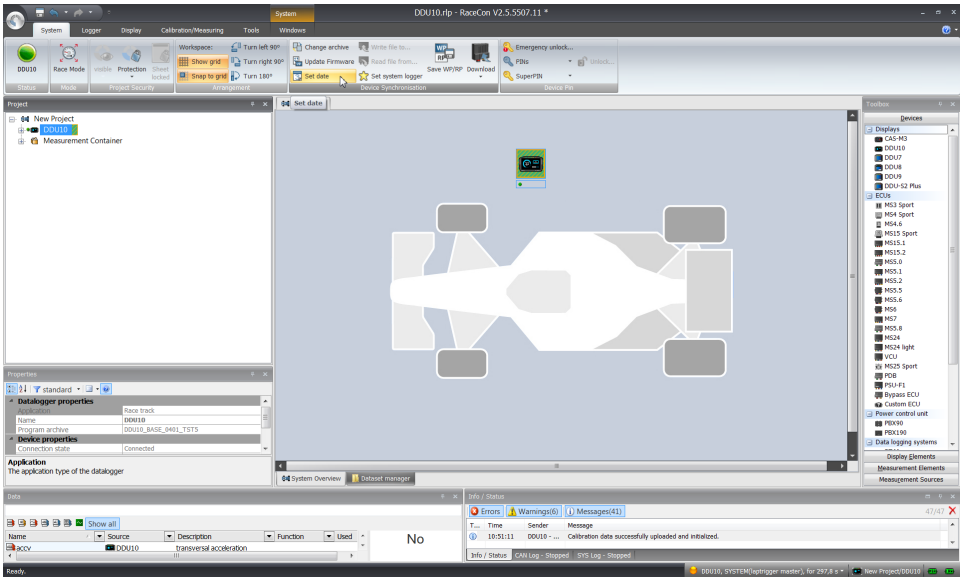
6. Perform these steps to activate other features you purchased.
7. Switch the car's ignition off and on again to cycle the power of DDU 10.

4.5 Set time and date

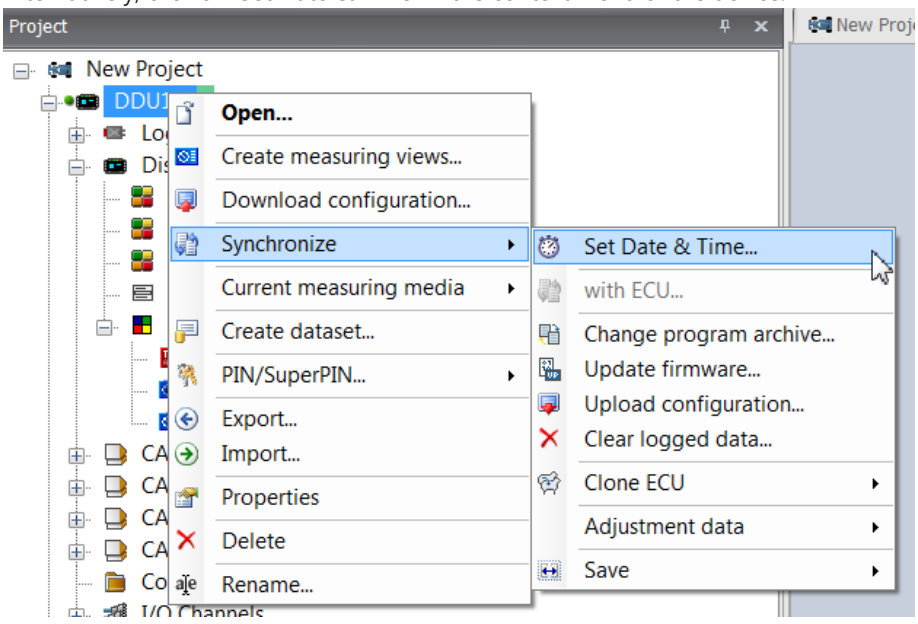
The DDU 10 is equipped with a real time clock which is supplied by an internal accumulator. Once this accumulator is charged correctly by 12 V supply of the display, 'Date & Time' can be programmed by RaceCon.

Reassure that the time is set correctly, if the device has not been used for more than two weeks.

1. Connect the DDU 10 to the PC.
2. Click on the 'Set date' button in the 'System' tab menu.



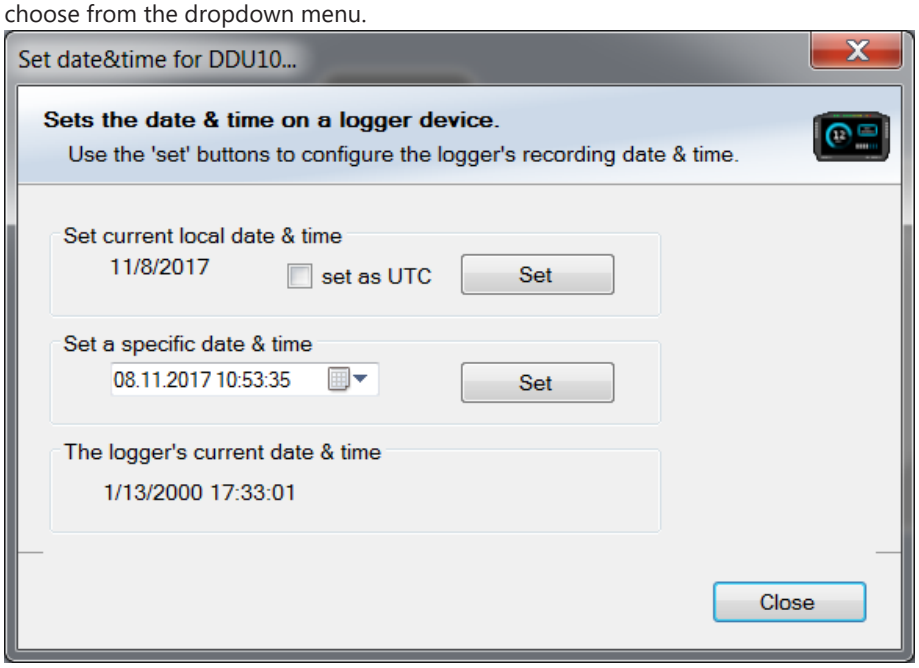
3. Alternatively, click on 'Set Date & Time' in the context menu of the device.



A 'Set Date & Time' menu opens

4. Set the current local date and time as coordinated universal time.

5. At ‘Set a specific date & time’ click and type on the value you want to change, or choose from the dropdown menu.

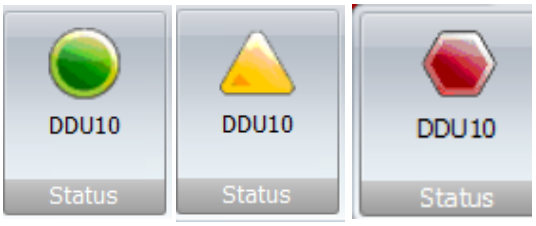


4.6 Color indication

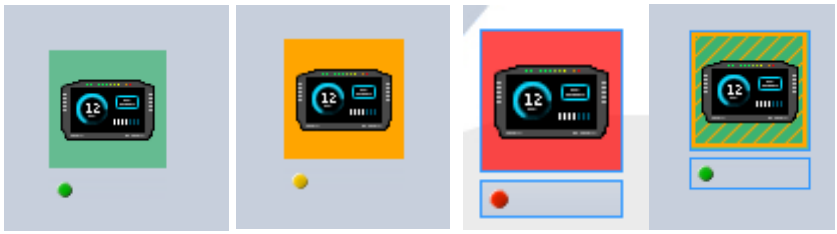
The color indication in RaceCon visualizes different messages, such as differences between tool and device, status of the device configuration or the accrual of errors.

Visible color indications:

– In the status area in the upper left corner.

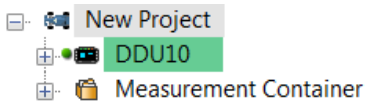


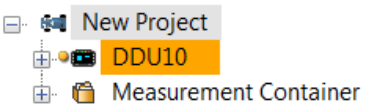
– As a background, as well as a little dot around the display icon in the ‘System window’.

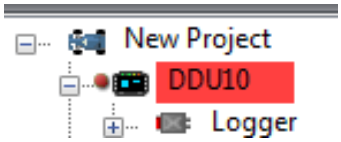


– As a colored stripe beside the device name in the project tree.

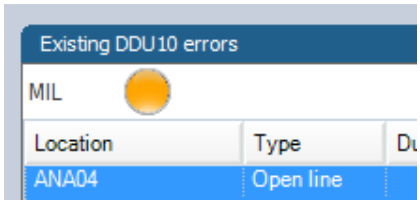
– As a colored background around the device name in the project tree.

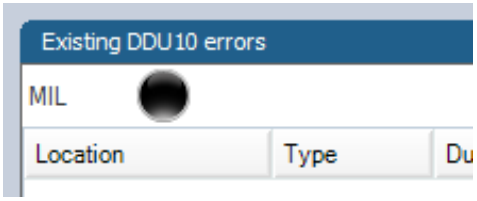




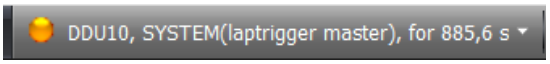


– As a colored MIL in the “Error Info” window.





– As a colored dot in the error memory at the bottom.



The colors and their meaning:

– Grey: No connection with the device.

– Green: Matching configuration and firmware between device and project.

– Orange: A different configuration between device and project.

– Red: A different firmware between device and project.

– Colored background with orange stripes: Matching configuration with stored (inactive) errors in the device.

– Blinking colored background with orange stripes: Matching configuration with active errors in the device.

– Black MIL: No errors.

– Orange MIL: Inactive Errors (Error entries existing, but not longer active).

– Blinking MIL (orange/black): Active Errors.

For further information, see chapter Error Memory Properties [▶ 57].

5 Export and Import in RaceCon

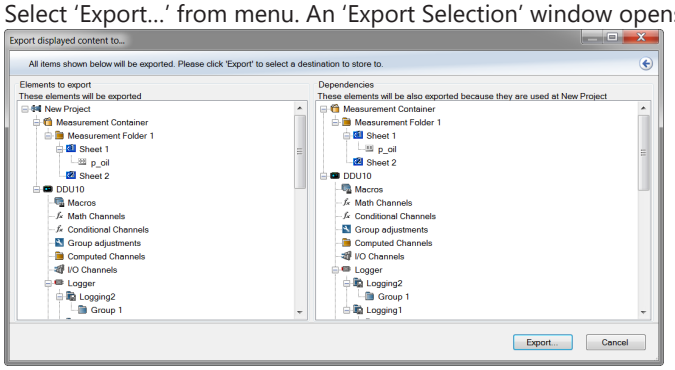
You can perform an export or an import on almost any level in the project tree.

5.1 Export in RaceCon

You can choose to export the whole project or you can export specific parts of the project.

Proceed with the following steps to perform an export:

1. Click with the right mouse button on an item in the project tree.
2. Select ‘Export...’ from menu. An ‘Export Selection’ window opens.



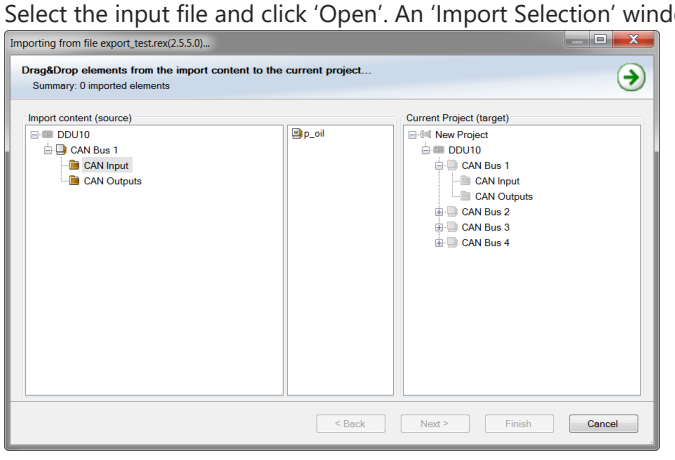
3. Click on ‘Export’ to select a destination to store.
4. Specify the filename.
5. Click ‘Save’ when done.

5.2 Import in RaceCon

You can choose to import into the whole project or you can import into specific parts of the project.

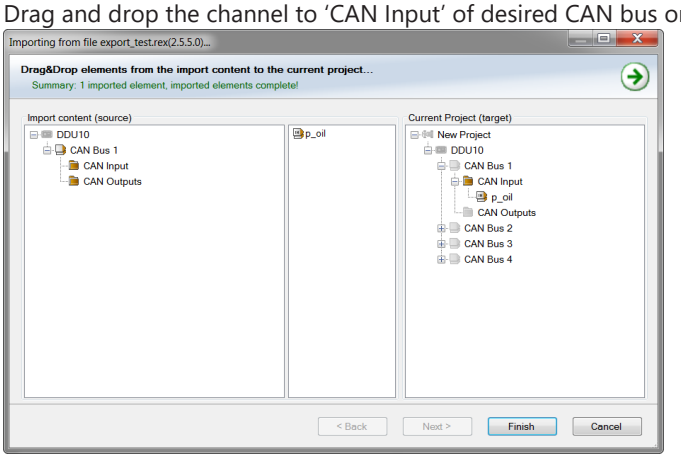
Proceed with the following steps to perform an import:

1. Click with the right mouse button on any item in the project tree.
2. Select ‘Import...’ from menu. A file browser opens.
3. Select the input file and click ‘Open’. An ‘Import Selection’ window opens.

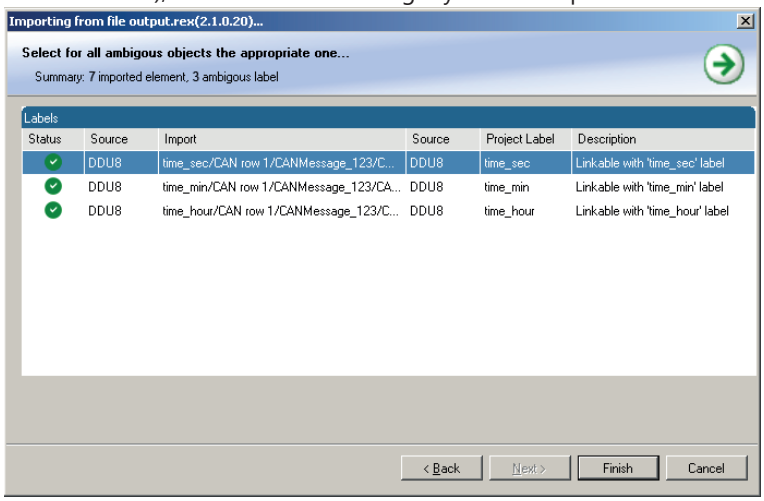


4. Select channels to import.

5. Drag and drop the channel to ‘CAN Input’ of desired CAN bus on right hand side.



6. Click ‘Finish’. If a measurement channel belongs to more than one source (e.g. DDU 10 and MS 5.1), the ‘Solve Label Ambiguity’ window opens.



7. Assign the ambiguous channels to the desired source.
8. Click ‘Finish’.

6 Analog and Frequency Inputs

Analog inputs

- 0 to 5 V
- 12 bit A/D converter
- Switchable 3.01 kOhm pull-up resistor
- 10 kHz acquisition rate, up to 1 kHz recording rate
- Linear phase digital filter

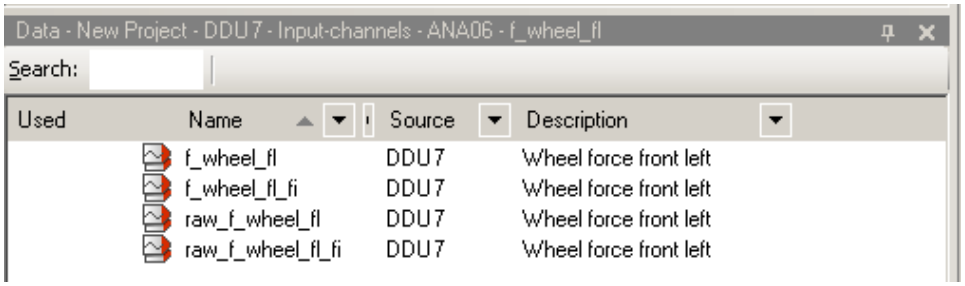
Frequency inputs

- 5 V Hall-effect type, 2.5 V trigger level (DF11 input with current interface or 5 V Hall-effect input with 2.5 V trigger level)
- 20 kHz max. frequency
- 10 ms measurement window

6.1 Analog inputs

6.1.1 Measurements channels

For each analog channel, several ‘subchannels’ are available.



Measurement labels with the characters ‘raw’ show the exact values in mV.

Measurement labels with the characters ‘_fi’ show filtered values.

The word ‘name’ in the table is a placeholder for the channel’s name.

Measurement label	Function
raw_name	mV value of sensor
raw_name_fi	Filtered mV value of sensor
name	Physical value of sensor
name_fi	Filtered physical value

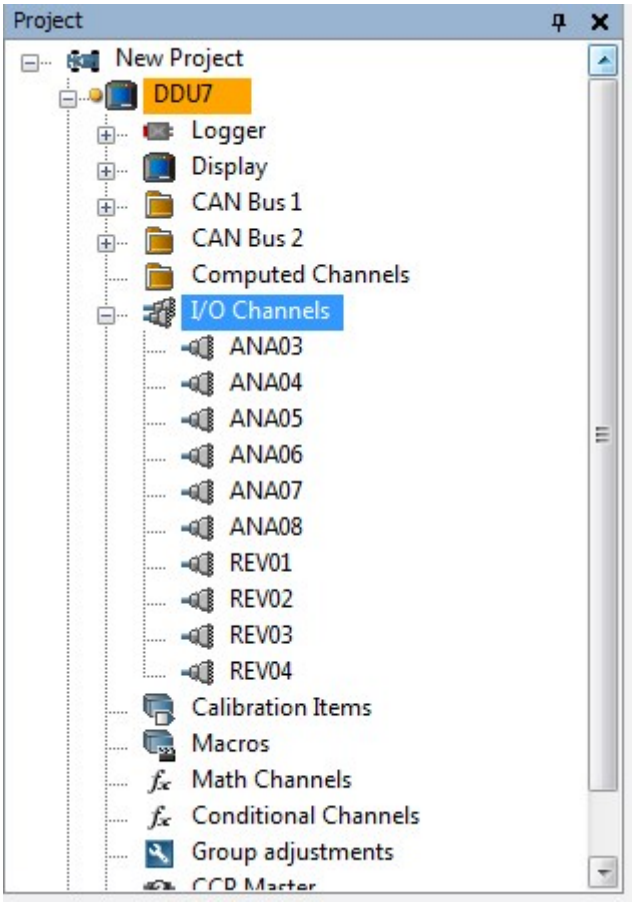
Filtered channels are routed through digital low pass filters:

- DDU 10 uses A/D converter oversampling and digital filtering to recording rate
- Digital filters eliminate ‘out-of-band’ noise
- Cut-off frequency automatically adjusted to recording rate
- Linear phase – no signal distortion
- Latency compensation – no filter delay in recorded data

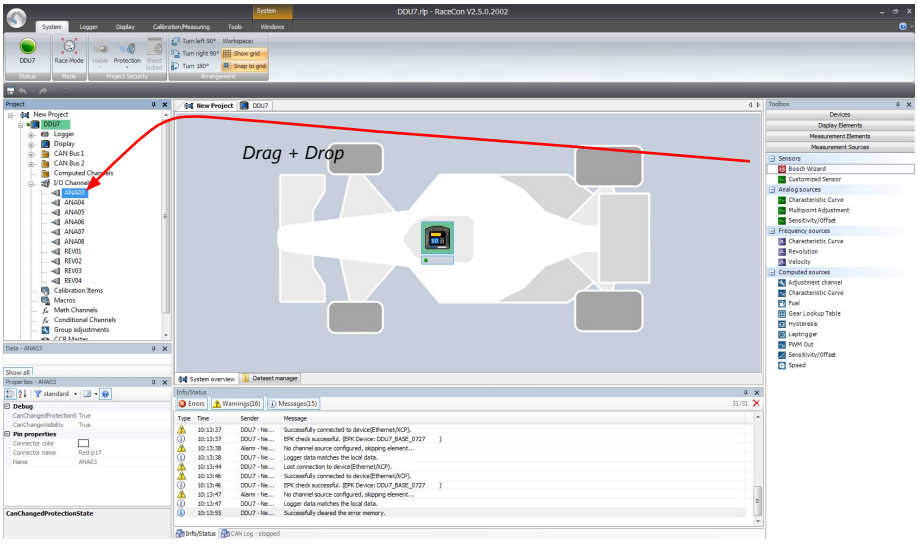
6.2 Configuring inputs

6.2.1 Configuring a predefined Bosch sensor with the 'Bosch Sensor Wizard'

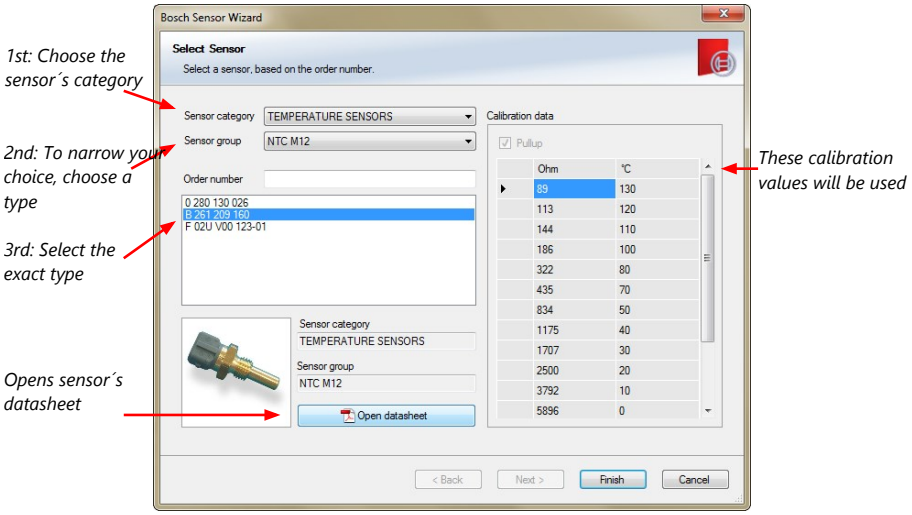
1. Click on ‘Measurement Sources’ in the Toolbox.
2. To expand the list of ‘I/O Channels’, click on ‘+’ in the DDU 10 Project Tree.



3. Drag the “Bosch Sensor Wizard” from the Toolbox and drop it on the desired analog input channel in the DDU 10 Project Tree.

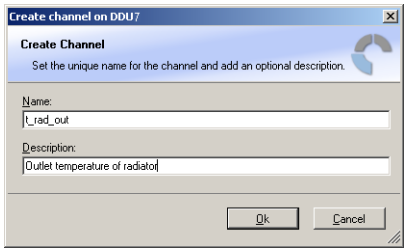


The “Bosch Sensor Wizard” opens.

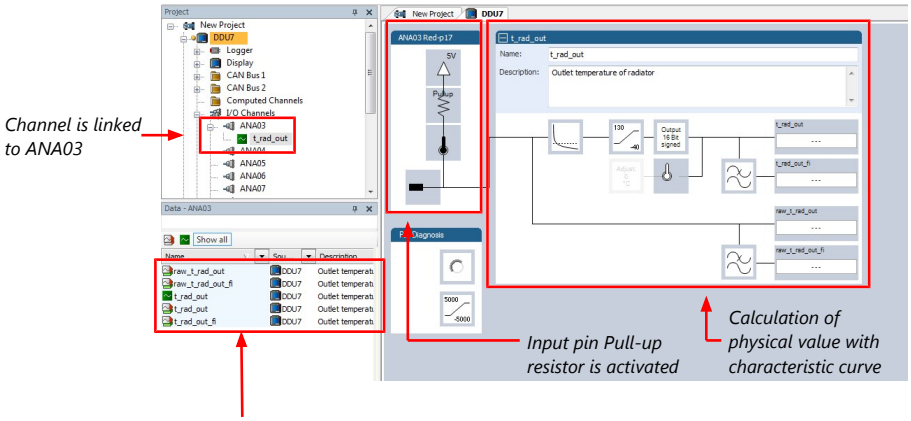


4. Click ‘Finish’ when done.
   The “Create channel” window opens.

5. Enter the channel name and description.



6. Click ‘Ok’ when done.
   The channel is inserted into the DDU 10 Project Tree.



Available measurements for channel:

Measurement label	Function
raw_name	mV value of sensor
raw_name_**fi**	Filtered mV value of sensor
name	Physical value of sensor
name_**fi**	Filtered physical value

6.2.2 Configuring a generic linear sensor

Example: Acceleration sensor 5 g

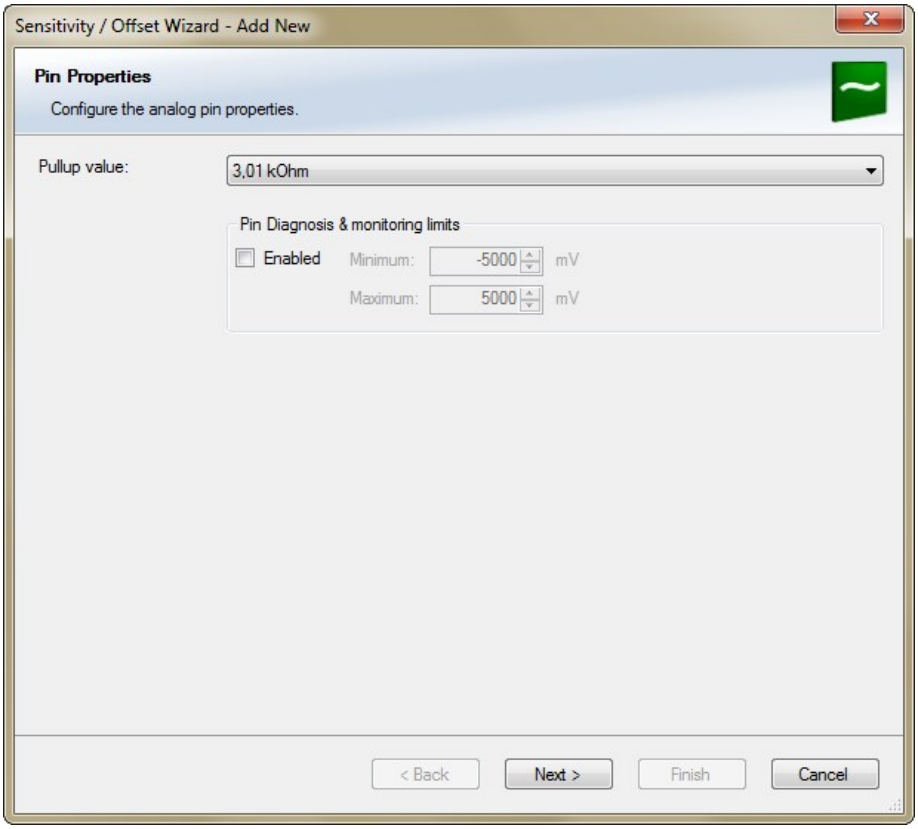
– From sensor data sheet - operating characteristics:

Output Signal					
Zero g (T_A = 25°C, V_{DD} = 5.0 V)(4)	V_{OFF}	2.25	2.5	2.75	V
Zero g (V_{DD} = 5.0 V)	V_{OFF}	2.0	2.5	3.0	V
Sensitivity (T_A = 25°C, V_{DD} = 5.0 V)(5)	S	380	400	420	mV/g
Sensitivity (V_{DD} = 5.0 V)	S	370	400	430.1	mV/g
Bandwidth Response	f_{-3dB}	42.5	50	57.5	Hz
Nonlinearity	NL_{OUT}	–1.0	—	+1.0	% FSO

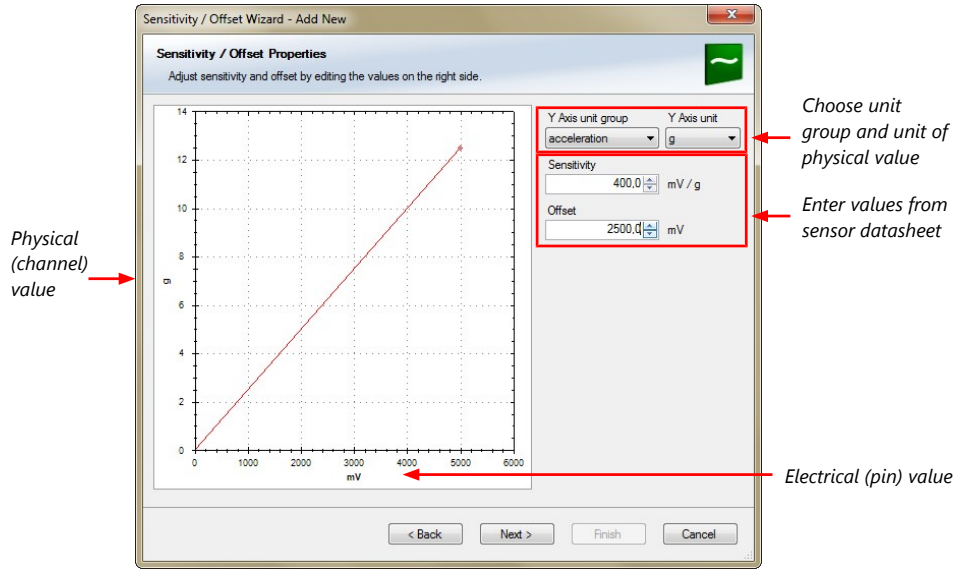
– Sensitivity 400 mV/g, Offset 2,500 mV

– The sensor has a linear output signal with sensitivity and offset

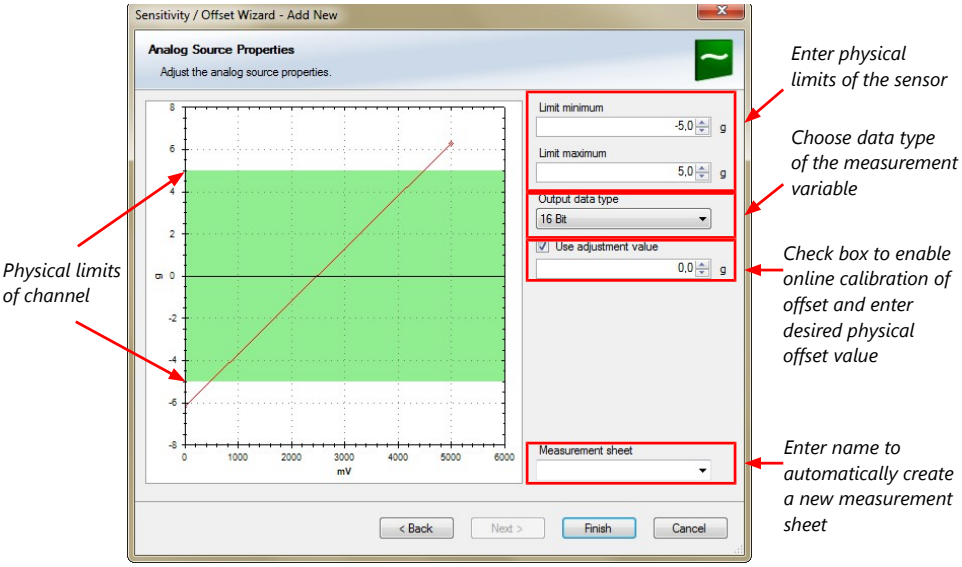
1. Click on ‘Measurement Sources’ in the Toolbox.
2. To expand the list of ‘I/O Channels’, click on ‘+’ in the DDU 10 Project Tree.
3. Drag the “Sensitivity/Offset” analog signal source from the Toolbox and drop it on the desired analog input channel in the DDU 10 Project Tree. A “Sensitivity/Offset Wizard” opens.
4. To activate the internal pullup-resistor, check the box. The internal pullup-resistor is used to get a 5 V signal at the analog channel of the DDU 10. It allows you to use a push-button. The fixed value of the internal pullup-resistor is 3,010 Ohm. If using an additional external pullup-resistor, set up the overall resistance.



5. Click ‘Next’ when done.
   The second part of the “Sensitivity/Offset Wizard” opens.



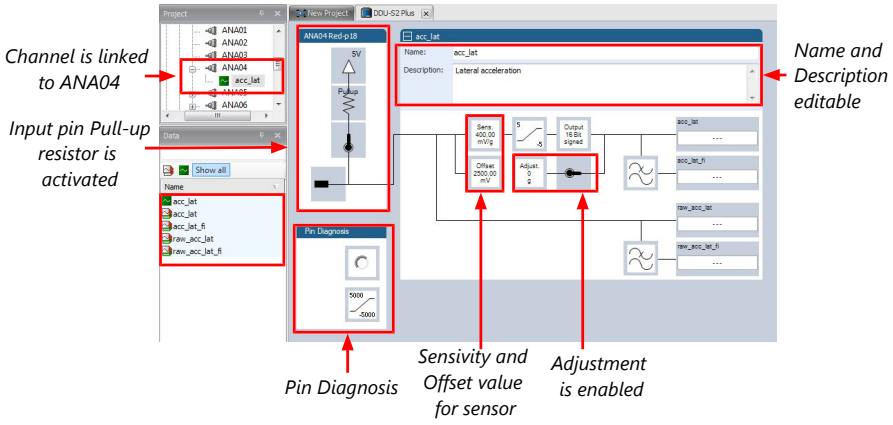
6. Click ‘Next’ when done.
   The third part of the “Sensitivity/Offset Wizard” opens.



NOTICE

Working with automatically created measurement sheets is explained in chapter ‘Setting up an online measurement [▷ 46]’.

7. Click ‘Finish’ when done.
8. Enter a channel name and a description.
9. Click ‘OK’ when done.
   The channel is inserted into the DDU 10 Project Tree.



Available measurements for channel:

Measurement label	Function
raw_name	mV value of sensor
raw_name_**fi**	filtered mV value of sensor
name	physical value of sensor
name_**fi**	filtered physical value

6.2.3 Configuring a generic nonlinear sensor

Example: Thermistor 5 kOhm

– From sensor data sheet - resistance values over temperature:

PART NR.: 2381 640 502
HTCLE100E3502

Toper [°C]	RT [Ω]
-40	166 047
-35	119 950
-30	87 600
-25	64 643
-20	48 179
-15	36 250
-10	27 523
-5	21 078

Toper [°C]	RT [Ω]
0	16 277
5	12 669
10	9 936
15	7 849
20	8 244
25	5 000
30	4 030
35	3 267

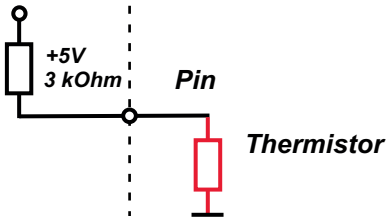
Toper [°C]	RT [Ω]
40	2 685
45	2 166
50	1 903
55	1 494
60	1 245
65	1 024
70	876
75	740

Toper [°C]	RT [Ω]
80	628
85	535
90	457
95	399
100	338
105	292
110	251
115	221

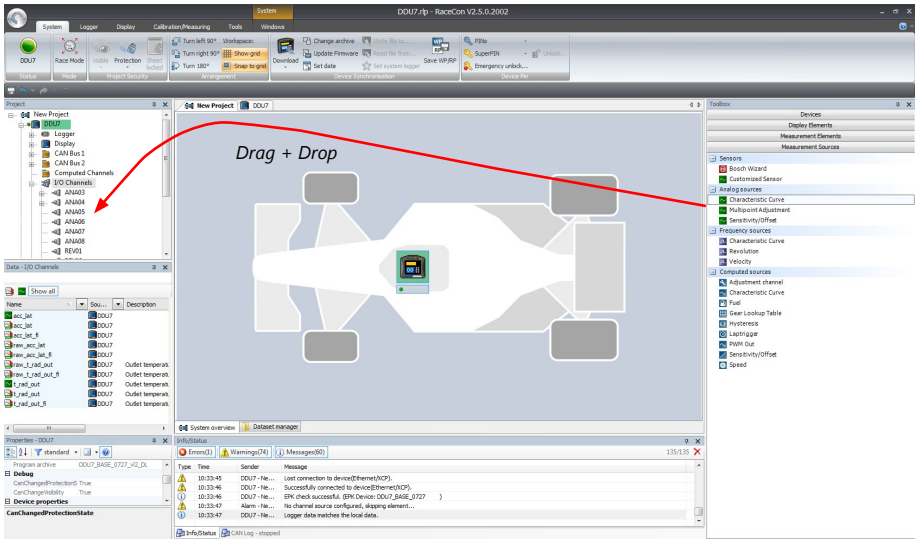
– The sensor has a nonlinear behavior

– Use characteristic curve for linearization

– Input voltage is the ratio between pull-up resistor and thermistor

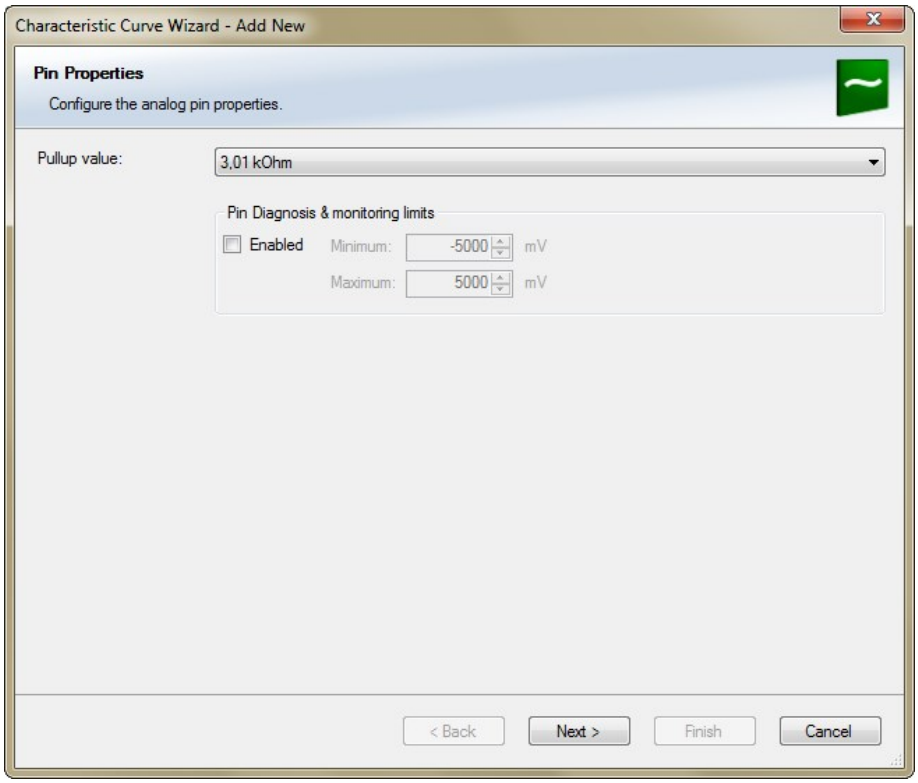


1. Click ‘Measurement Sources’ in the Toolbox.
2. To expand the list of ‘I/O Channels’, click on ‘+’ in the DDU 10 Project Tree.
3. Drag the “Characteristic Curve” analogue signal source from the Toolbox and drop it on the desired analogue input channel in the DDU 10 Project Tree.

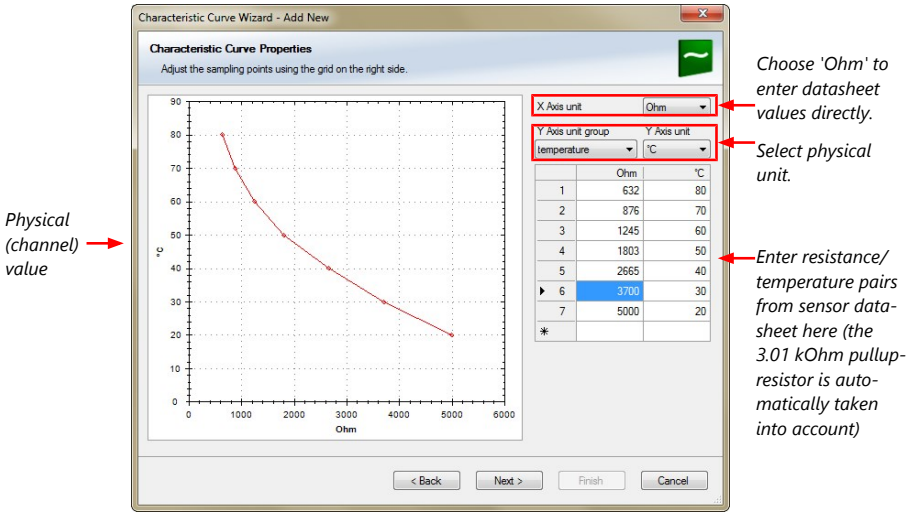


A “Characteristic Curve Wizard” opens.

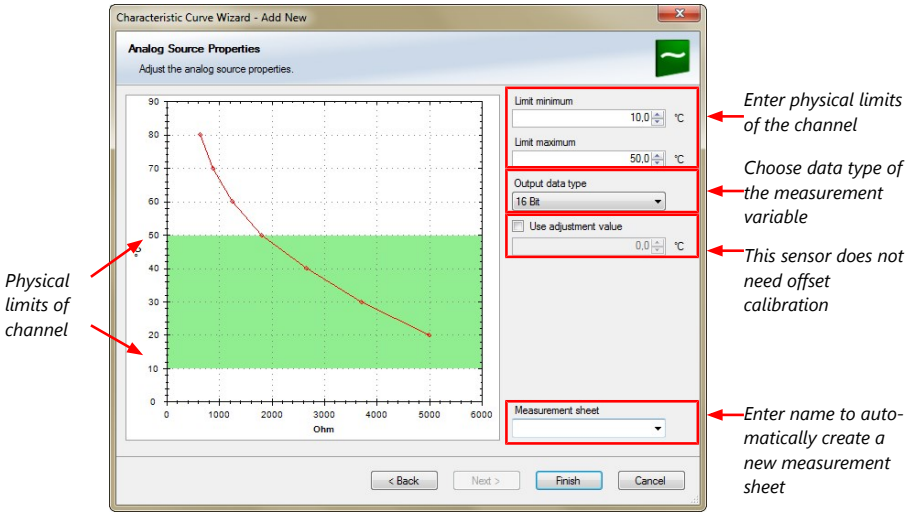
4. To activate the internal pull up-resistor, check the box. The DDU 10 pull up-resistor is used to get a 5 V signal at the analogue channel of the DDU 10. It allows you, to use a push-button. The fixed value of the internal pull up-resistor is 3,010 Ohm. If using an additional external pull up-resistor, set up the overall resistance.



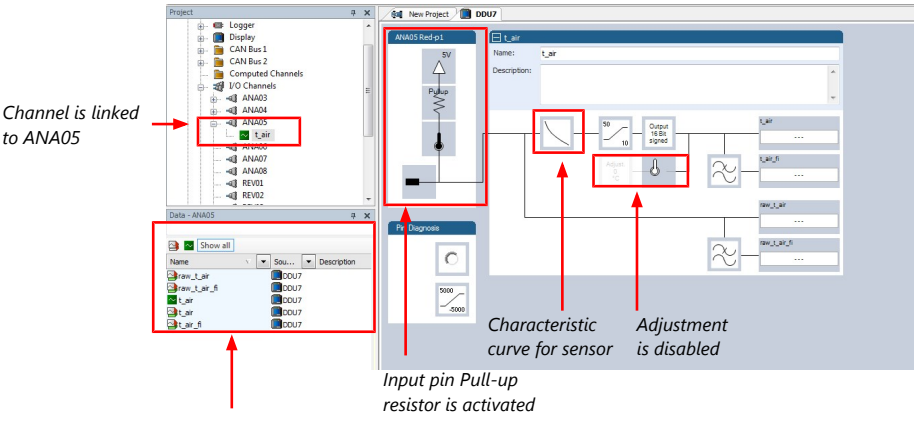
5. Click ‘Next’ when done.
The second part of the “Sensitivity/Offset Wizard” opens.



6. Click ‘Next’ when done.
   The third part of the “Characteristic Curve Wizard” opens.



7. Click ‘Finish’ when done.
8. Enter channel name and description.
9. Click ‘OK’ when done.
   The channel is inserted into the DDU 10 Project Tree.



Available measurements for channel:

Measurement label	Function
raw_name	mV value of sensor
raw_name_**fi**	filtered mV value of sensor
name	physical value of sensor
name_**fi**	filtered physical value

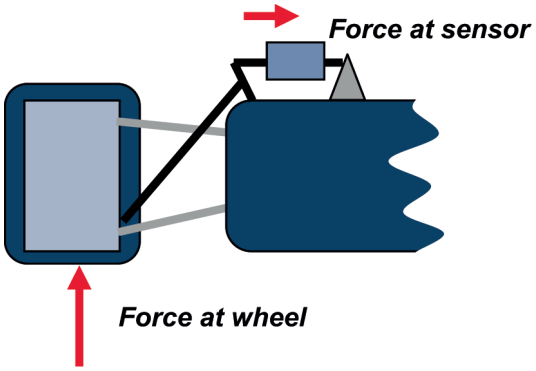
NOTICE

Working with automatically created measurement sheets is explained in chapter 'Setting up an online measurement [▷ 46]'.

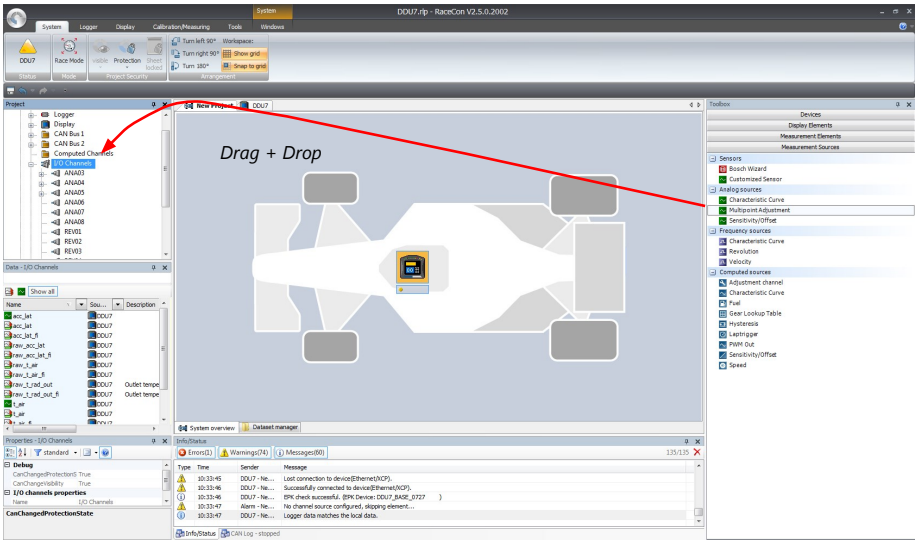
6.2.4 Configuring a multipoint adjustment

Example: Measurement of wheel force

- Physical property 'wheel force' not directly measureable
- Load transfer through suspension kinematics
- Physical value at sensor position defined by vehicle
- Curve definition by online adjustment at vehicle

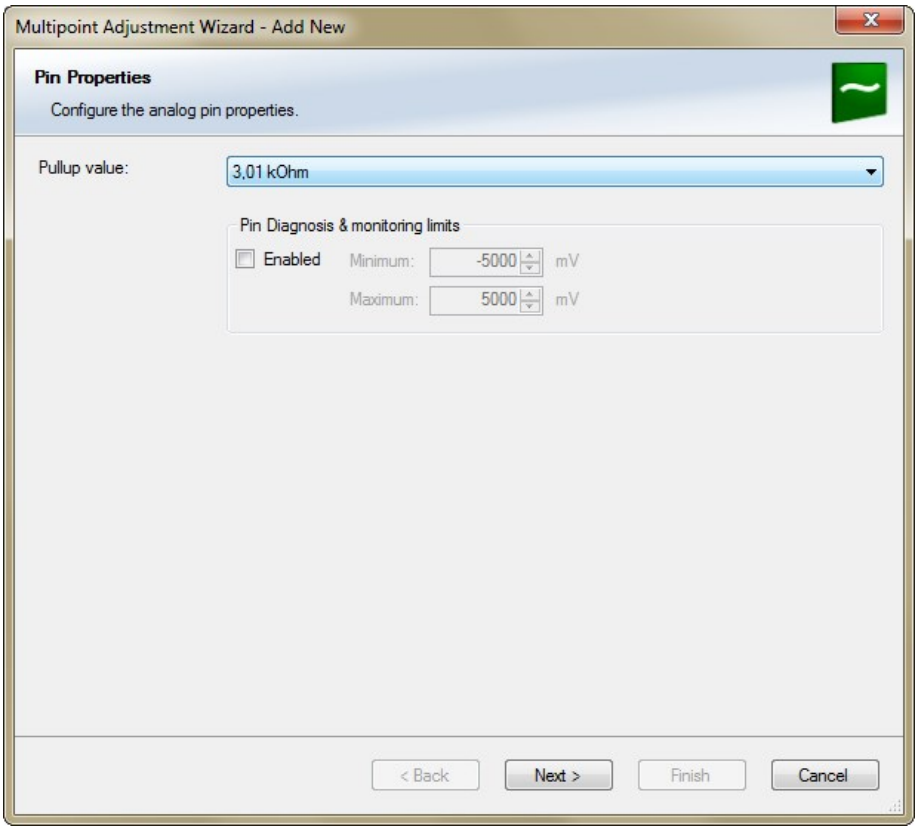


1. Click on 'Measurement Sources' in the Toolbox.
2. Expand the list of 'I/O Channels' by clicking on '+' in the DDU 10 Project Tree.
3. Drag the 'Multipoint Adjustment' analog signal source from the Toolbox and drop it on the desired analog input channel in DDU 10 Project Tree.

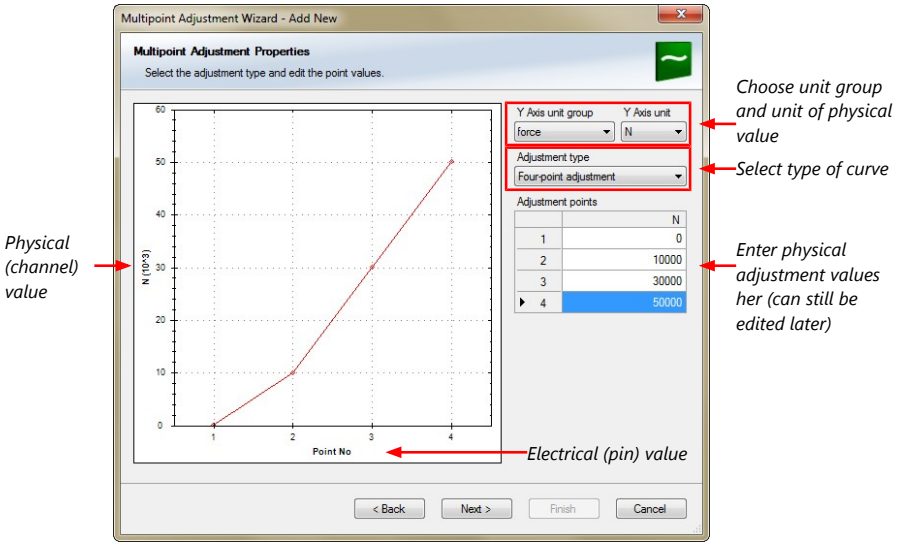


A ‘Multipoint Adjustment Wizard’ opens.

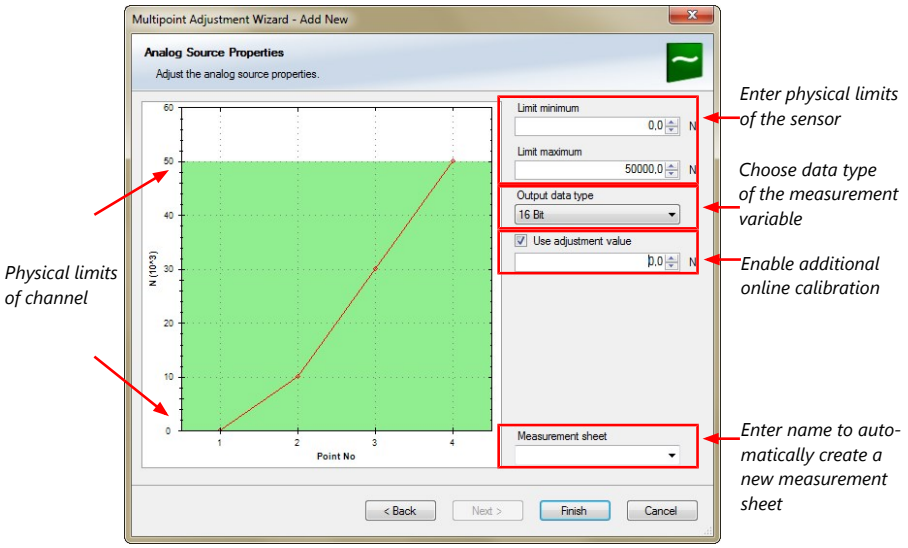
4. To activate the internal pullup-resistor, check the box. The internal pullup-resistor is used to get a 5 V signal at the analog channel of the DDU 10. It allows you to use a push-button. The fixed value of the internal pullup-resistor is 3.01 kOhm. If using an additional external pullup-resistor, set up the overall resistance.



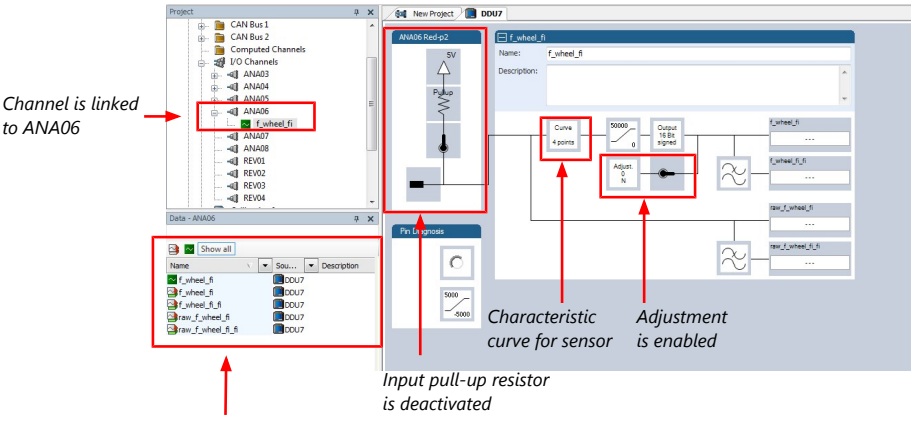
5. Click ‘Next’ when done.
   The second part of the ‘Multipoint Adjustment Wizard’ opens.



6. Click ‘Next’ when done.
   The third part of the ‘Multipoint Adjustment Wizard’ opens.



7. Click ‘Finish’ when done.
8. Enter channel name and description.
9. Click ‘OK’ when done.
   The channel is inserted into the DDU 10 Project Tree.



Available measurements for channel:

Measurement label	Function
raw_name	mV value of sensor
raw_name_**fi**	filtered mV value of sensor
name	physical value of sensor
name_**fi**	filtered physical value

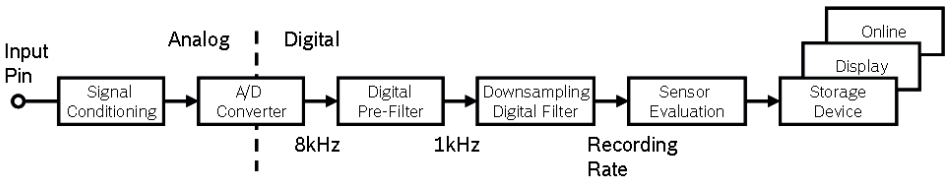
Online definition of the curve is covered in the chapter 'Online calibration of measurement channels [▶ 51]' of this manual.

NOTICE

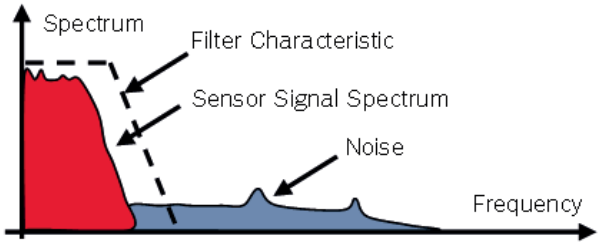
Working with automatically created measurement sheets is explained in chapter 'Setting up an online measurement [▶ 46]'.

6.2.5 Digital filter details

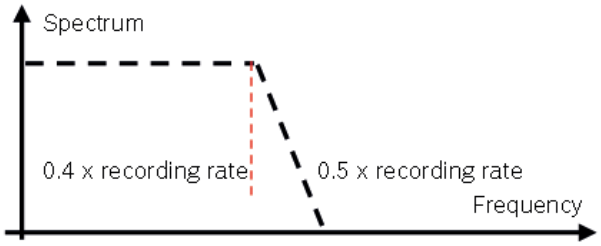
DDU 10 uses A/D converter oversampling and digital filtering to recording rate.



Digital filters eliminate 'out-of-band' noise



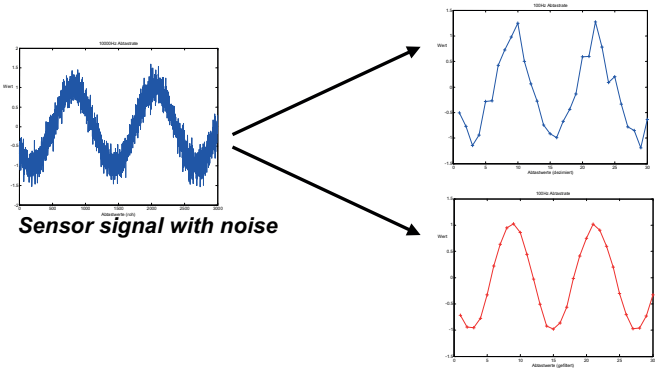
Cut-off frequency automatically adjusted to recording rate



Example:

- 100 Hz recording rate (10 ms)
- <40 Hz pass band (>99 %)
- >50 Hz stop band (<1 %)

Linear phase – no signal distortion



Latency compensation – no filter delay in recorded data

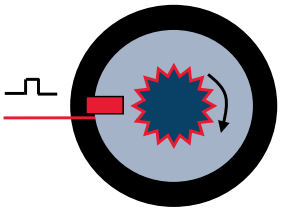
- Filtering is (smart) averaging over several samples
- Filtered signal is delayed with respect to real time signal

- DDU 10 filters have constant, frequency independent delay
- Delay (e.g. 22 samples at 10 ms) is corrected during recording
- No delay filtered vs. unfiltered in recorded data
- Correction is (of course) not possible for real time data (display, online, PWM out)
- Use filtered data for recording, use unfiltered data for real-time

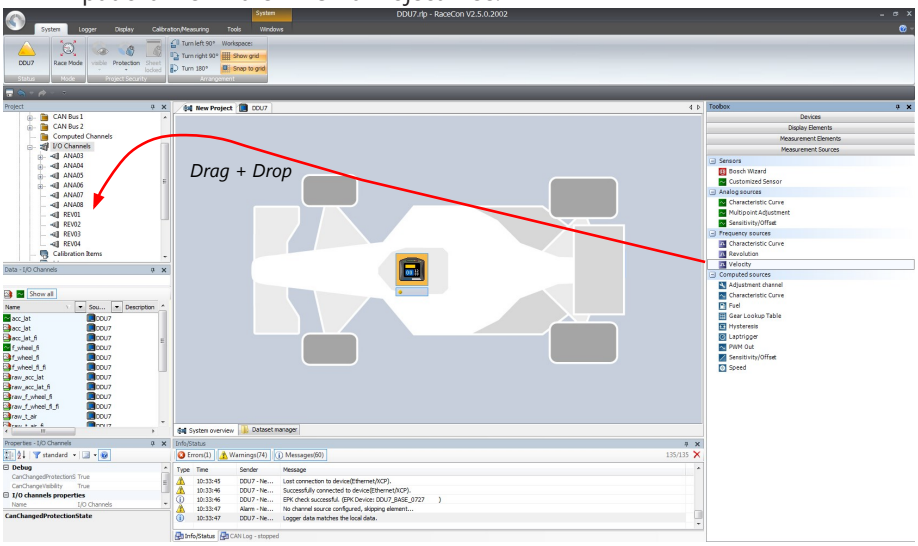
6.2.6 Configuring a frequency input

Example: measurement of wheel speed

- Pulse wheel attached to wheel
- Each passing tooth of pulse wheel triggers hall sensor
- Calculation of wheel speed with wheel circumference

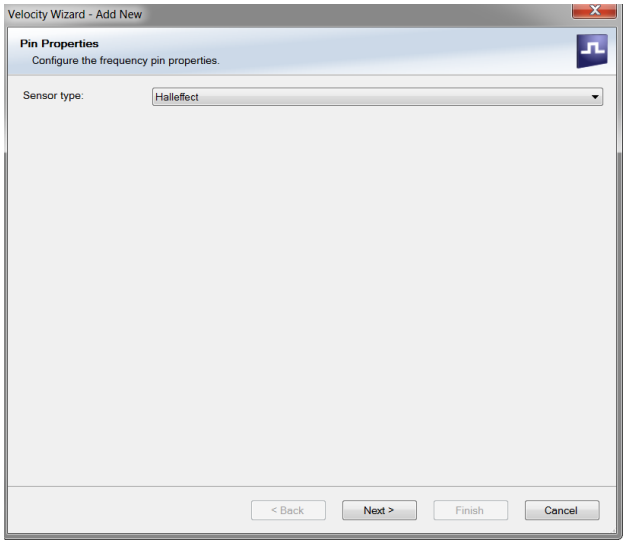


1. Click on ‘Measurement Sources’ in the Toolbox.
2. To expand the list of ‘I/O Channels’, click on the ‘+’ in the DDU 10 Project Tree.
3. Drag the ‘Velocity’ digital signal source from the Toolbox and drop it on the desired ‘REV’ input channel in the DDU 10 Project Tree.

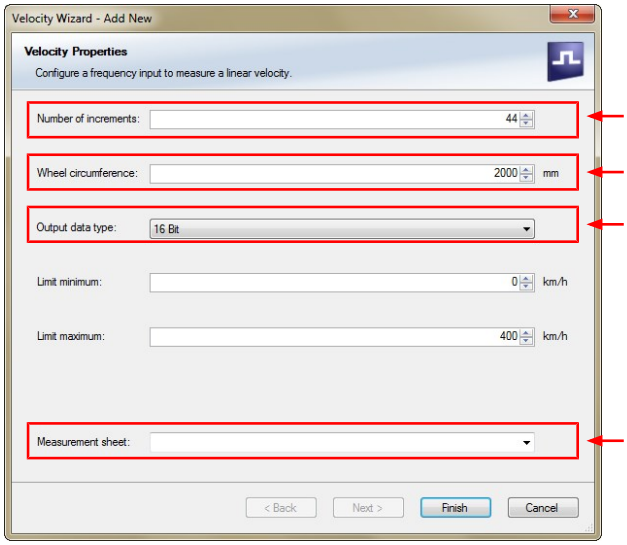


The ‘Velocity Wizard’ opens.

4. Select the sensor type. The DDU10 works with Halleffect and DF11 sensors.



5. Click ‘Next’.
6. Define the settings for the sensor.



Number of teeth on the pulse wheel

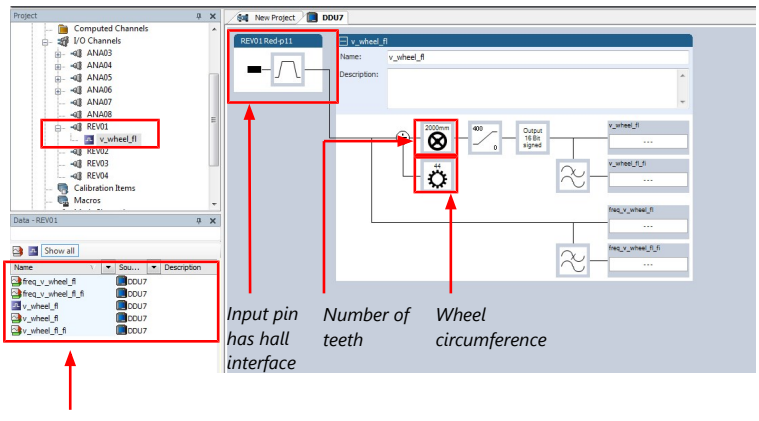
Circumference of wheel for speed calculation

Choose data type of the measurement variable

Enter name to automatically create a new measurement sheet

7. Click ‘Finish’ when done.
8. Enter the channel name and description.
9. Click ‘OK’ when done.
   The channel is inserted into the DDU 10 Project Tree.

Channel is linked to REV01



Available measurements for channel:

Measurement label	Function
raw_name	mV value of sensor
raw_name_**fi**	filtered mV value of sensor
name	physical value of sensor
name_**fi**	filtered physical value

NOTICE

Measurement of ‘Revolution’ is similar.

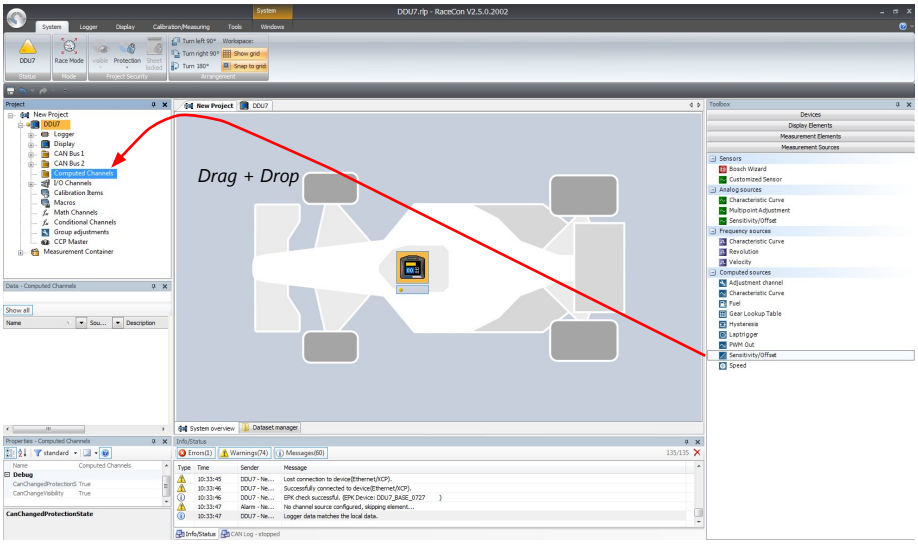
6.3 Configuring computed sources

Computed sources receive data from a measurement channel rather than an input pin.

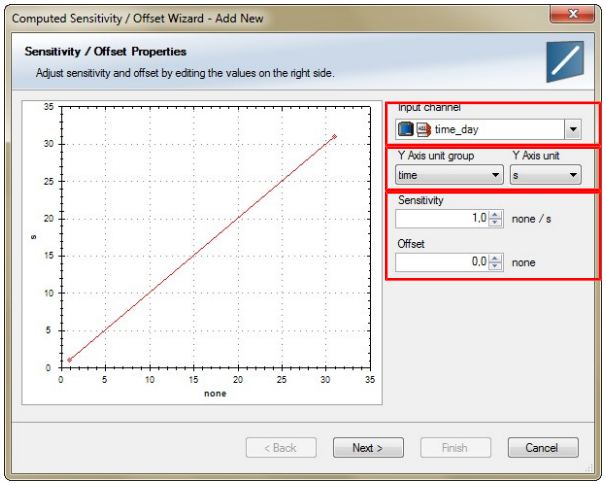
- Sensitivity/Offset calculation on input channel
- Characteristic curve calculation on input channel
- Computed vehicle speed
- Lap trigger (covered in a special separate section)

Example: Sensitivity/offset calculation on input channel

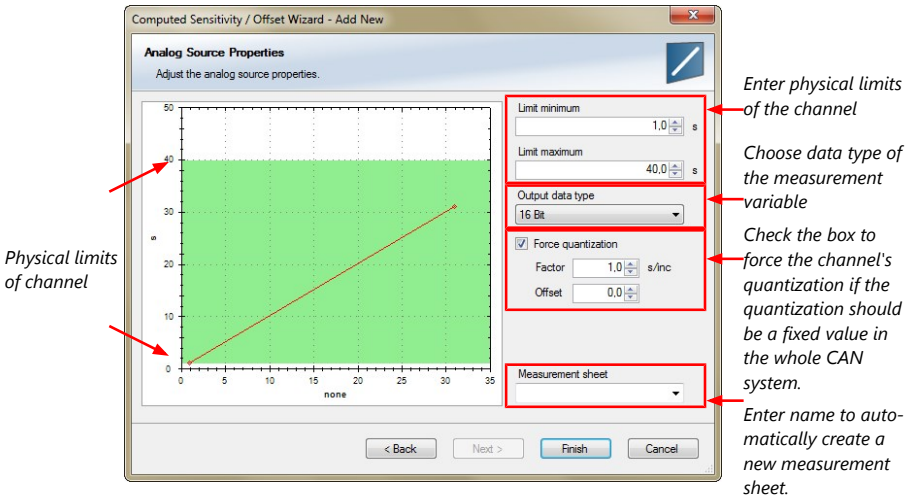
1. Click ‘Measurement Sources’ in the Toolbox.
2. Drag the ‘Sensitivity/Offset’ computed source from the Toolbox and drop it on ‘Computed Channels’ in the DDU 10 Project Tree.



A ‘Computed Sensitivity/Offset Wizard‘ opens.



3. Click ‘Next’ when done.
   The second part of the ‘Computed Sensitivity/Offset Wizard’ opens.



4. Click ‘Finish’ when done.
5. Enter channel name and description.
6. Click ‘OK’ when done.

   The channel is inserted into the DDU 10 Project Tree.

NOTICE

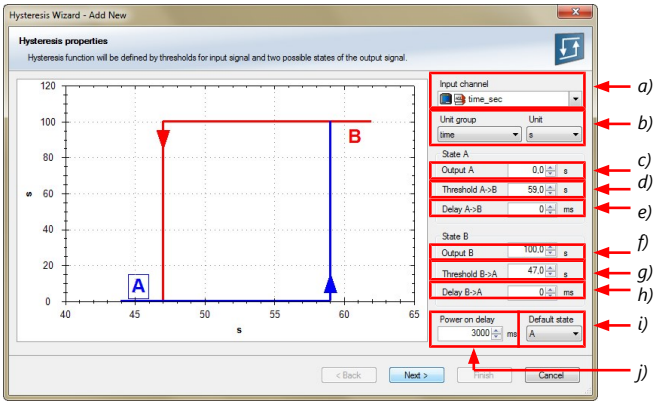
Working with automatically created measurement sheets is explained in chapter ‘Setting up an online measurement’.

6.4 Hysteresis

The hysteresis function avoids the high-frequent switchover of the measurement channel value. The hysteresis can be adjusted for each input measurement channel individually and can be used for further processing.

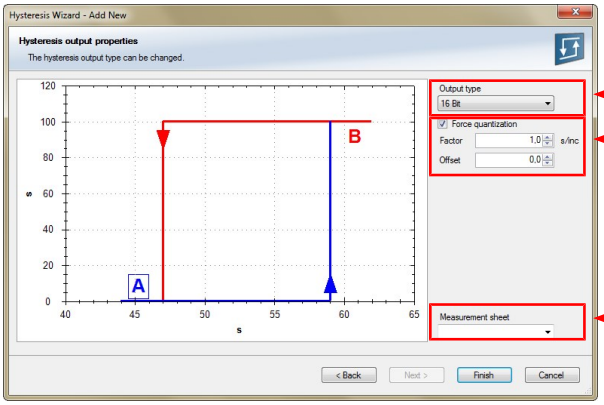
1. Click ‘Measurement Sources’ in the Toolbox.
2. Drag the ‘Hysteresis’ computed source from the Toolbox and drop it on ‘Computed Channels’ in the DDU 10 Project Tree.

   A ‘Hysteresis Wizard’ opens.

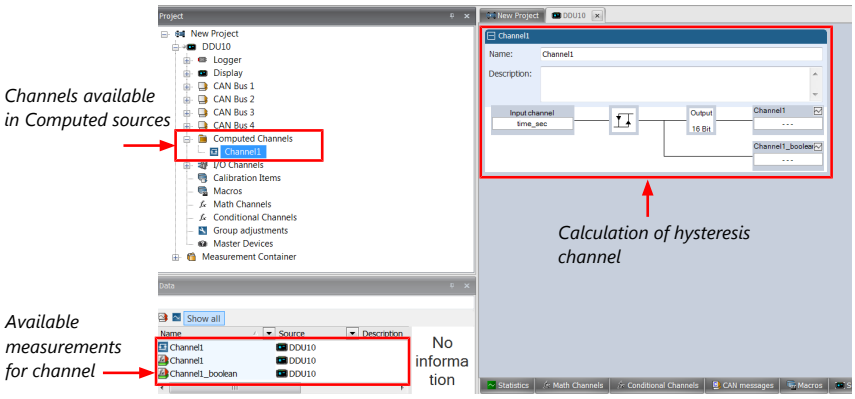


a) Choose input measurement channel.
b) Choose unit group and unit of output.
c) Enter output value of state A in the unit selected in b).
d) Enter threshold value when state changes from A to B.
e) Enter delay time when state changes from A to B.
f) Enter output value of state B in the unit selected in b).
g) Enter threshold value when state changes from B to A.
h) Enter delay time when state changes from B to A.
i) Enter time when the hysteresis function is activated after vehicle´s startup.
j) Enter the channel´s state (A or B) at startup.

3. Click ‘Next‘ when done.
   The second part of the ‘Hysteresis Wizard‘ opens.



4. Click ‘Finish‘ when done.
5. Enter channel name and description.
6. Click ‘OK‘ when done.
   The channel is inserted into the DDU 10 Project Tree.



6.4.1 Special functionality: Vehicle speed

This functionality allows:

- high performance vehicle owners to measure wheel spin under acceleration and wheel slip/lock under braking.
- calculating vehicle ‘speed over ground’.

Vehicle speed calculation function

- Calculating vehicle speed of 2 wheel drive: (Wheel speeds of non-driven axle as input)

  Calculated speed is average of both speeds if speed difference between wheels < limit.

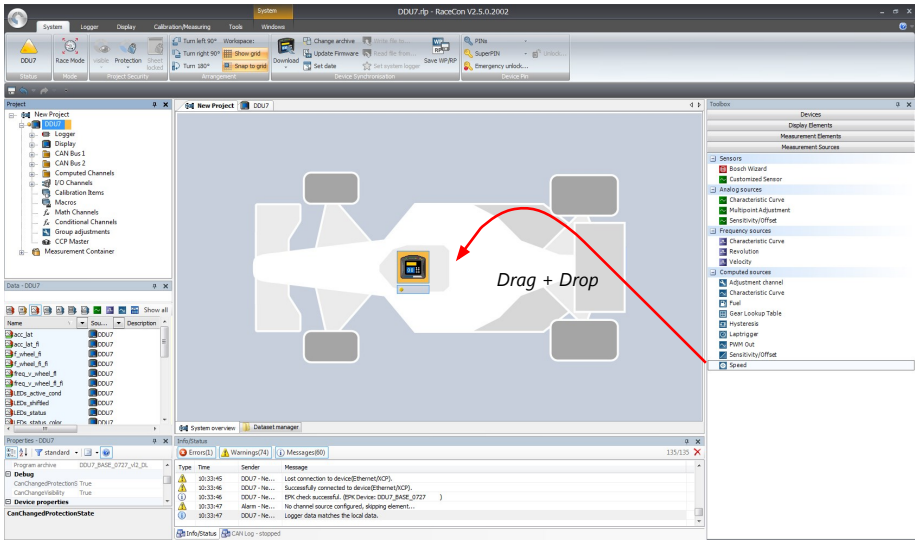
  Calculated speed is maximum of both speeds if speed difference between wheels > limit.

- Calculating vehicle speed of 4 wheel drive: (Wheel speeds of all wheels as input)

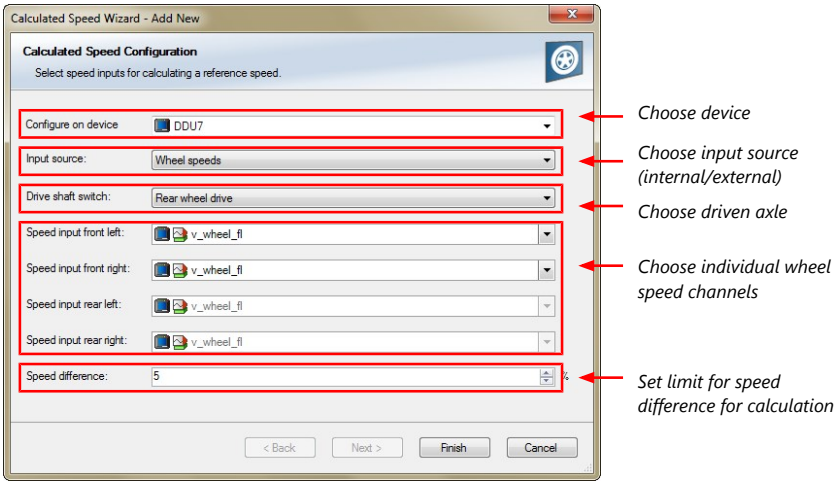
  Calculated speed is speed of 2nd fastest wheel.

6.4.2 Setting up calculated speed

1. Click on tab ‘System Overview‘.
2. Click on ‘Measurement Sources’ in the Toolbox.
3. Drag the ‘Speed’ computed source from the Toolbox and drop it on the project name in the DDU 10 Project Tree. Do not drop it on ‘DDU 10’!

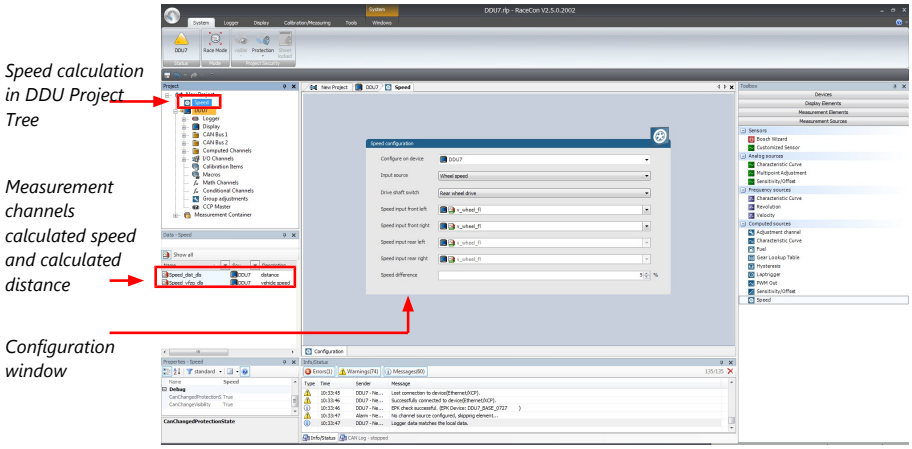


A ‘Calculated Speed Wizard’ opens.



4. Click ‘Finish’ when done.

   The speed calculation is inserted into the DDU 10 Project Tree.



7 Online Measurement

DDU 10 configuration

- System configuration (channel + display configuration, CAN I/O, etc.) is stored in the DDU 10
- Use RaceCon to create and download configuration from the PC to DDU 10 Communication interface: Ethernet
- Communication protocol: XCP

Online measurement + calibration

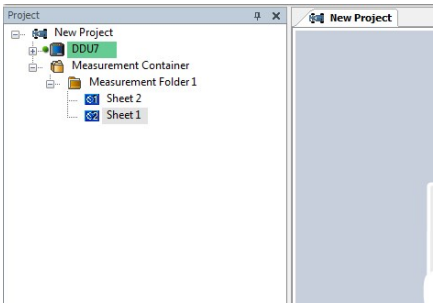
- System status and diagnosis
- Check and calibrate sensors in the vehicle
- Live display of sensor values on the PC
- Use RaceCon for diagnosis, online measurement and calibration
- Communication interface: Ethernet
- Communication protocol: XCP

7.1 Setting up an online measurement

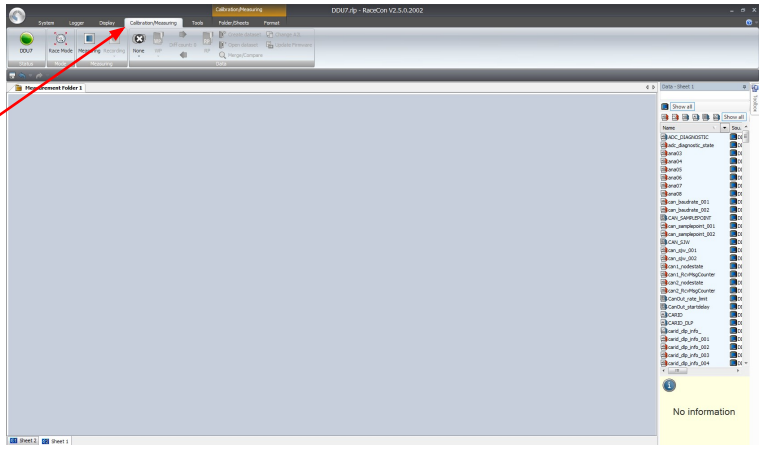
DDU 10 supports online measurement of sensor values and diagnostic variables.

1. Expand 'Measurement Container' and 'Measurement Folder 1' in the Project Tree and double-click on 'Sheet1'. Alternatively, click on the 'Calibration/Measuring' tab to open the window directly.

   'Sheet 1' opens in a new 'Calibration/Measuring' window.

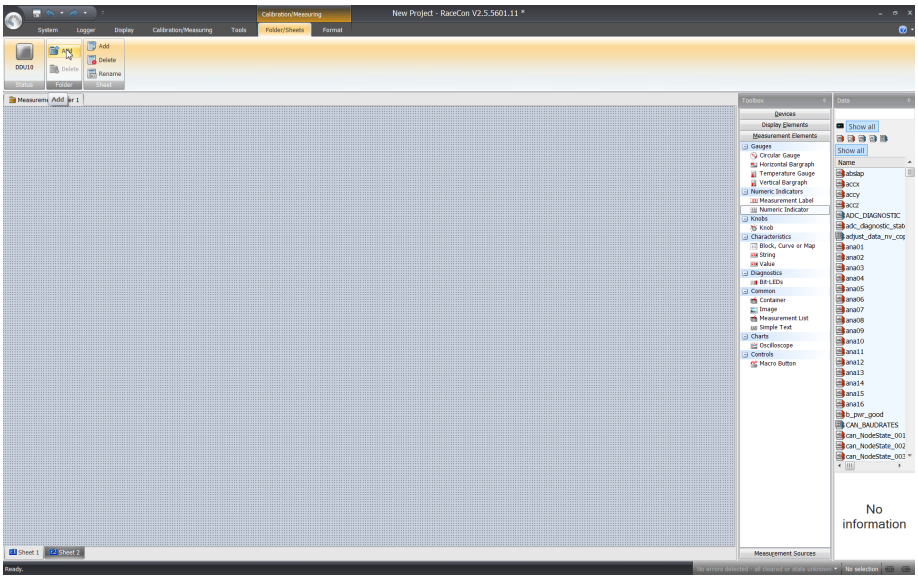


RaceCon changes automatically to "Calibration/ Measuring"



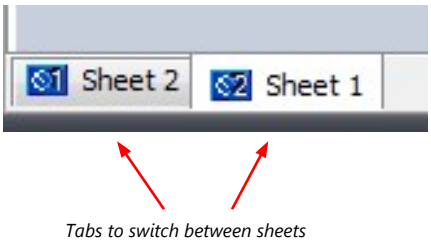
2. Click on the 'Folder/Sheets' tab, which appears when you are in the 'Calibration/ Measurement' window, to create a new measurement folder.

3. Click on the ‘Add‘ button for folders in the upper left corner.



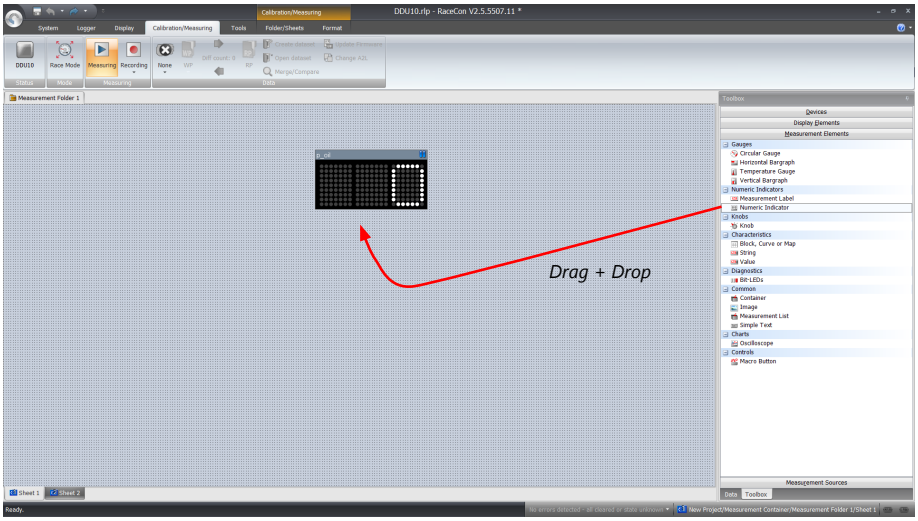
In the menu for sheets, you will find buttons to add, delete and rename new sheets

4. To change between different sheets, click on the tabs on the bottom of the ‘Calibration/Measuring‘ window.



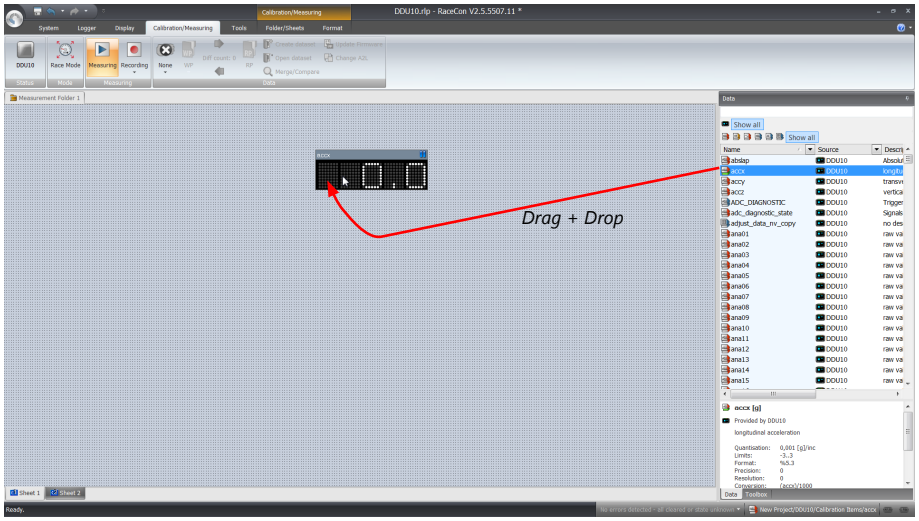
To add an element to a measurement sheet, perform the following steps:

5. Drag a measurement element from the Toolbox and drop it on the measurement sheet.



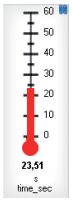
6. Select the desired measurement channel from the ‘Data‘ area and drop it on the measurement element.

   If the DDU 10 shows the green status, the value is displayed.

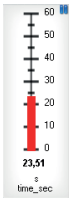


RaceCon offers different types of measurement elements:

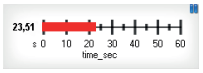
Circular gauge



Temperature gauge



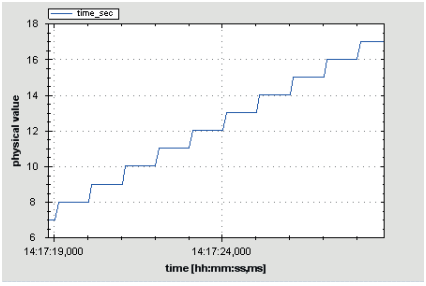
Vertical Bar graph style



Horizontal Bar graph style

Measurement label

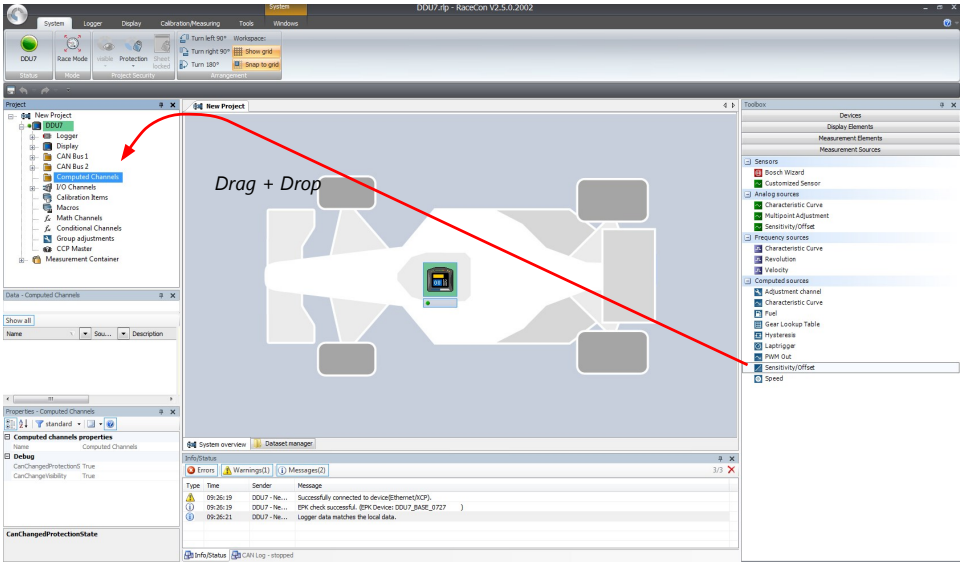
Numeric indicator



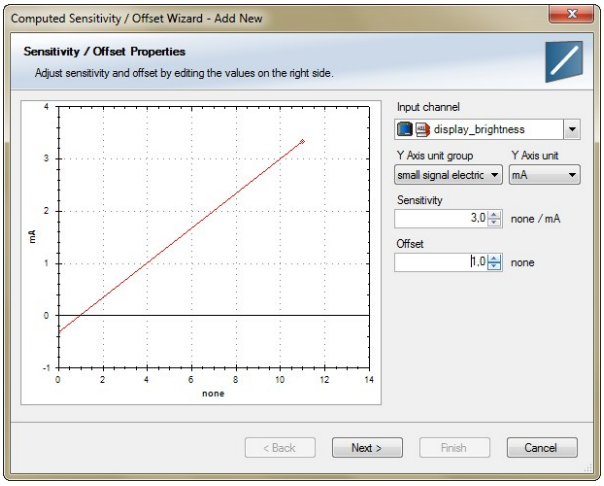
Oscilloscope (Chart)

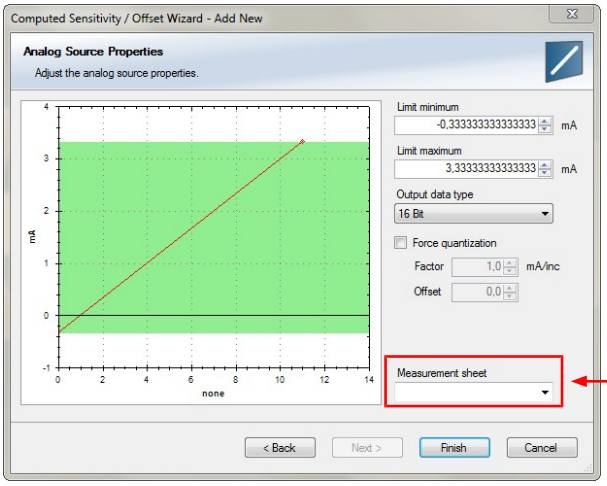
7.1.1 Automatic creation of measurement sheets

RaceCon can create measurement sheets automatically.
You can create and use measurement sheets with the DDU 10 as well as with all other devices connected to RaceCon.

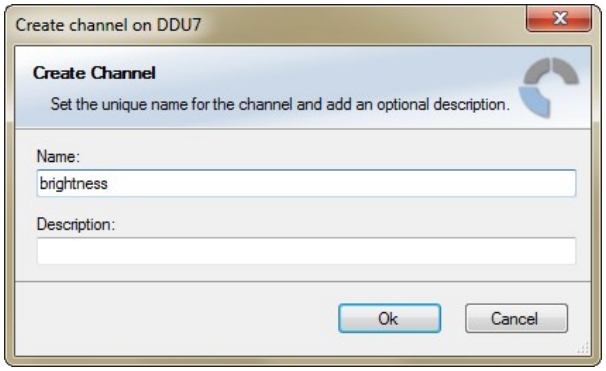


1. During the configuration of a measurement channel, select a measurement sheet from the list box or enter a name for a new measurement sheet.

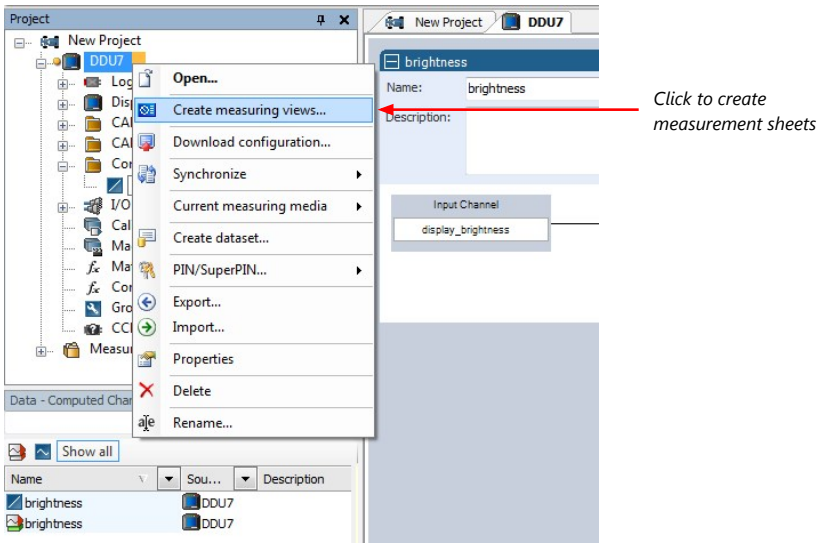




Select existing sheet from list or enter name of new sheet



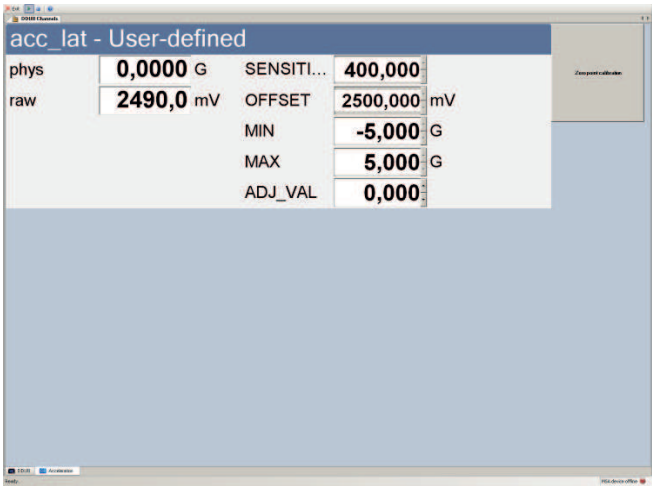
2. To create the sheets, right-click on DDU 10 and select ‘Create measurement views…’ from the DDU 10 context menu.



The automatically created sheet is inserted in the Project Tree under ‘Measurement Container’ and ‘Device Channels’. If the DDU 10 is connected to RaceCon and the status is green, live values of the channels are shown.

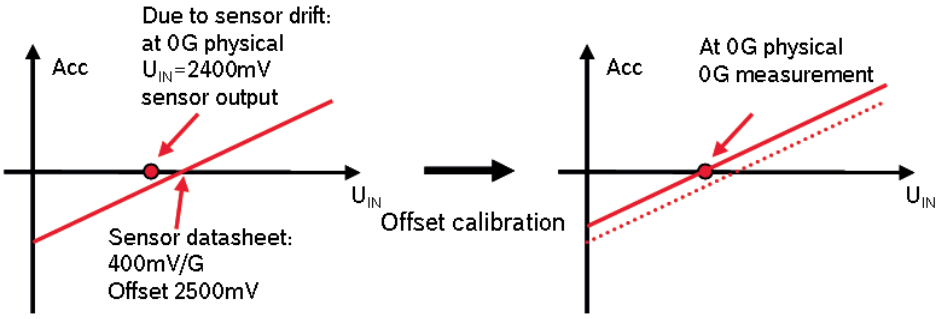
7.1.2 Using the measurement sheets

1. When RaceCon is online, press the ‘F11’ key to switch from ‘Design Mode’ into ‘Race Mode’. The measurement sheet is extended to full screen. The button for offset calibration is active.
2. Switch between different sheets using the tabs at the bottom of the window.
3. Press the ‘Esc’ key to return to ‘Design Mode’.



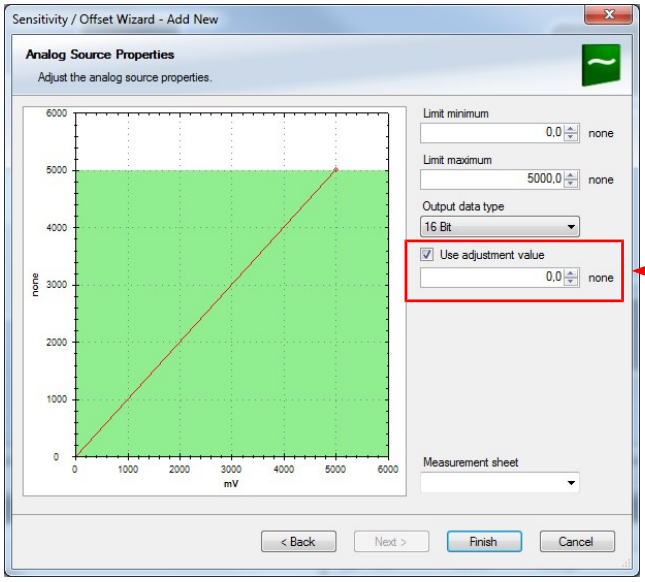
7.2 Online calibration of measurement channels

- Analog sensors drift with age, temperature, etc.
- Manual calibration is necessary
- Solution: online offset calibration
- Example: acceleration sensor



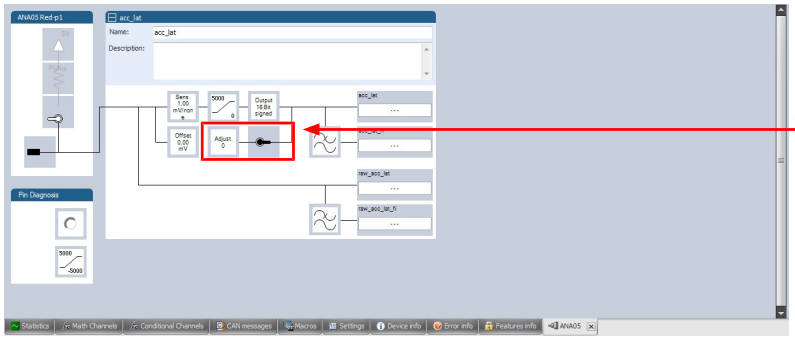
7.2.1 Enable online offset calibration for measurement channel

During creation of the measurement channel



Check box to enable online offset calibration and enter desired physical target value

In the channel view



Activate switch to enable online calibration

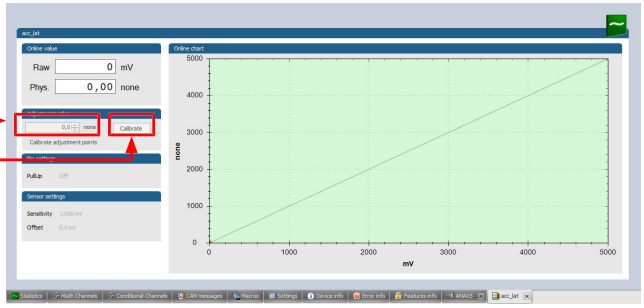
7.2.2 Performing the online offset calibration

DDU 10 has to be connected to RaceCon to calibrate the sensor's offset.

1. Apply the desired physical condition to the sensor (e.g. 1 G to an acceleration sensor).
2. Open the measurement channel's online page by double-clicking on the measurement channel name in the Data Area.
3. Enter the physical target value (e.g. 1 G) and press the 'Calibrate' button.

Calibration target value

Initiate calibration

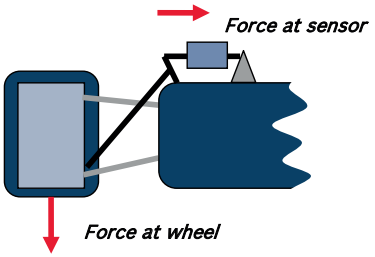


The sensor's offset is now calibrated.

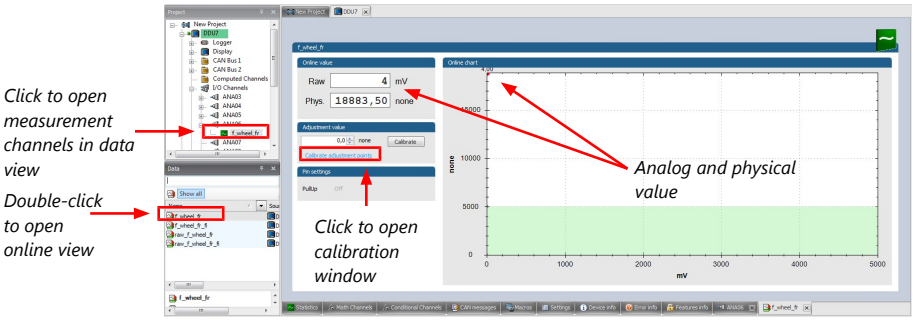
7.3 Online calibration of multipoint adjustment channels

Example: measurement of wheel force

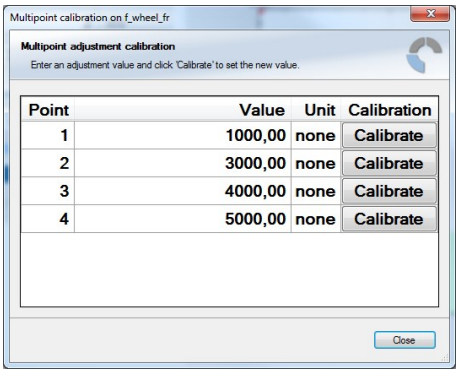
- Physical property 'wheel force' not directly measureable
- Load transfer through suspension kinematics
- Physical value at sensor position defined by vehicle
- Curve definition by online adjustment at vehicle



1. Create a multipoint adjustment measurement channel. To create a multipoint channel, see chapter 'Configuring a multipoint adjustment [▷ 34]'.
2. Download the configuration on the DDU 10. To connect the DDU 10 to RaceCon, see chapter 'Setting up a new RaceCon Project [▷ 9]'.
3. Click on the desired channel in the DDU 10 Project Tree.
4. Double-click on a measurement channel in the Data Area to open the online view.



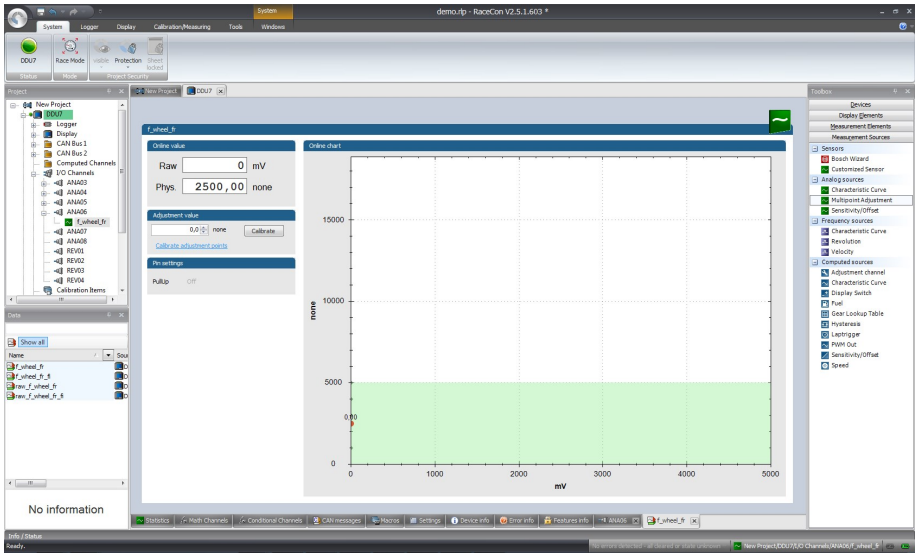
5. Click on 'Calibrate adjustment points' to open calibration window.



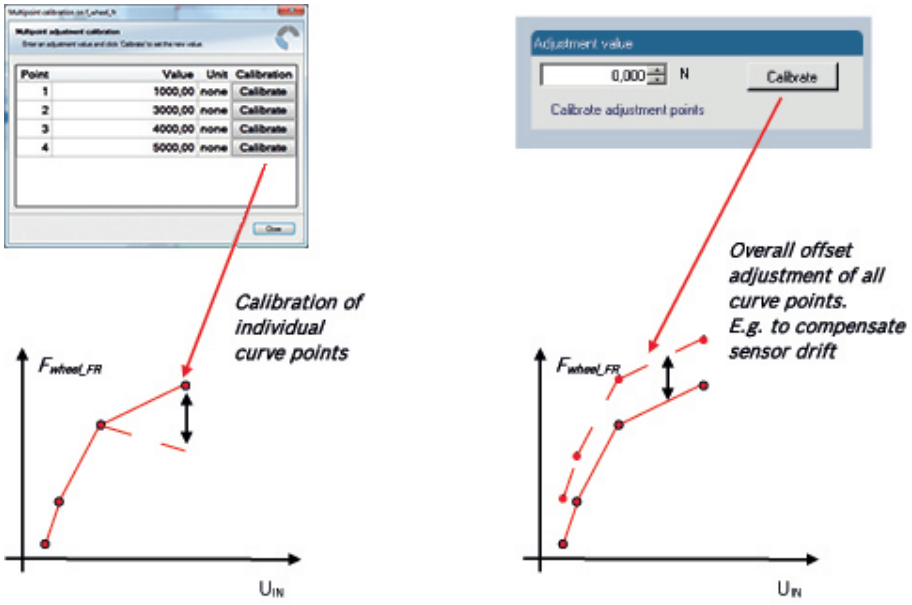
6. Apply the desired physical condition to the sensor (e.g. by applying a force on the wheel).
7. Enter the physical value in the value column of the desired calibration point (e.g. 745 N).

8. Press the ‘Calibrate’ button of the desired calibration point.
9. Repeat for all curve points.
10. Click ‘Close’ when done.

    The calibration curve is displayed in the online view.



Adjustment points vs. offset adjustment

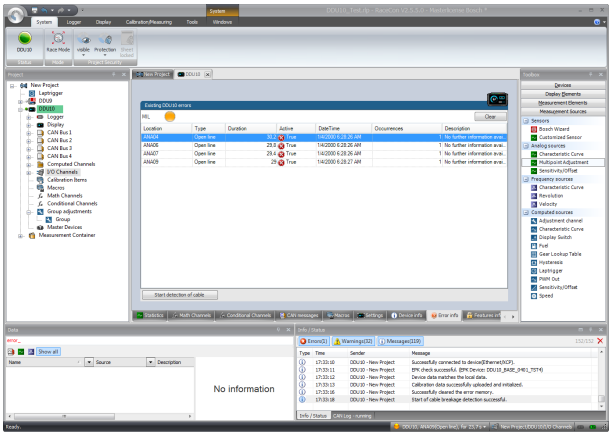


8 Error Memory

In this chapter “Error Memory”, a lot of screenshots are created by way of example for DDU 8. Please consider this and replace the product name ‘DDU 8’ in this case with the name of your product.

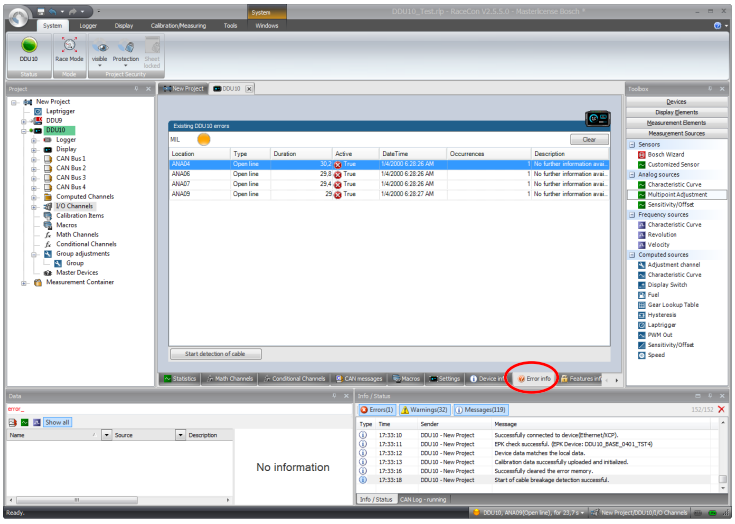
8.1 Error memory representation in RaceCon

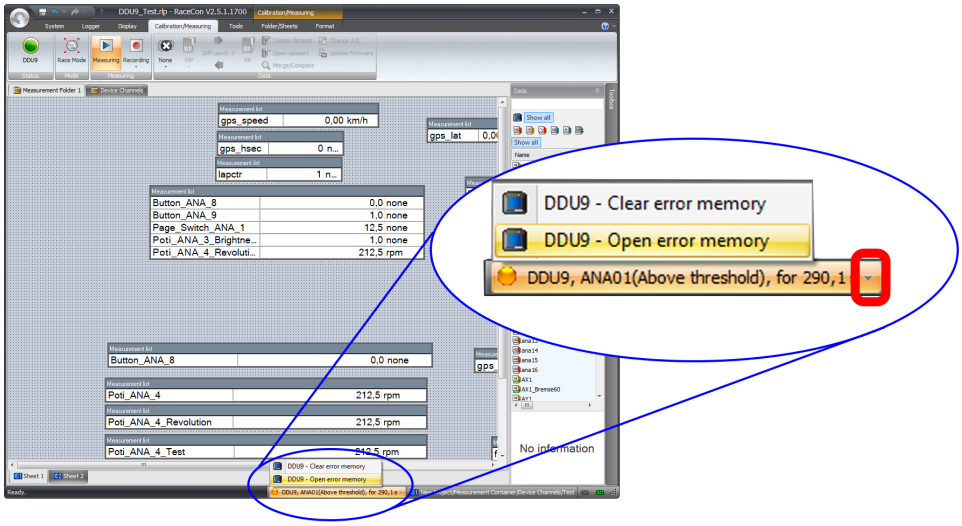
Bosch Motorsport devices feature an error memory. Information on errors can be visualized via RaceCon (online measurement) or can be transmitted via telemetry.



8.1.1 Accessing the memory

The error memory can be accessed as shown in the illustration:





The memory is situated inside the device and is non-volatile. As a consequence, an error which has occurred and has not been cleared by the user will remain in the error memory even after a power cycle. The error state will then reflect if the error is still active or not.

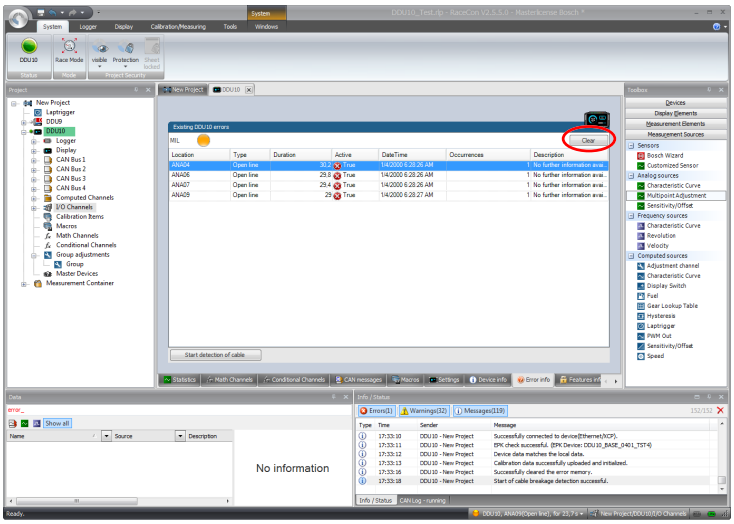
An error is deleted from the list when

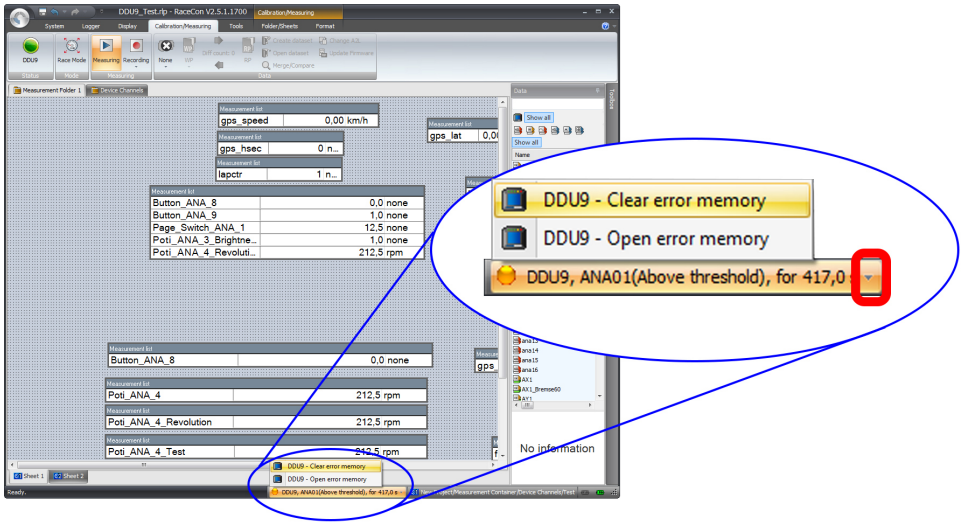
- the user actively clears the error memory
- the user updates the firmware

The error memory is not cleared by a configuration download and is not cleared by a power cycle.

8.1.2 Clearing the error memory

There are two ways of clearing the error memory, both are shown in the following illustration:





8.2 Information on errors available from the error memory

In general, properties of the error memory and properties of an individual error need to be distinguished.

8.2.1 Error Memory Properties

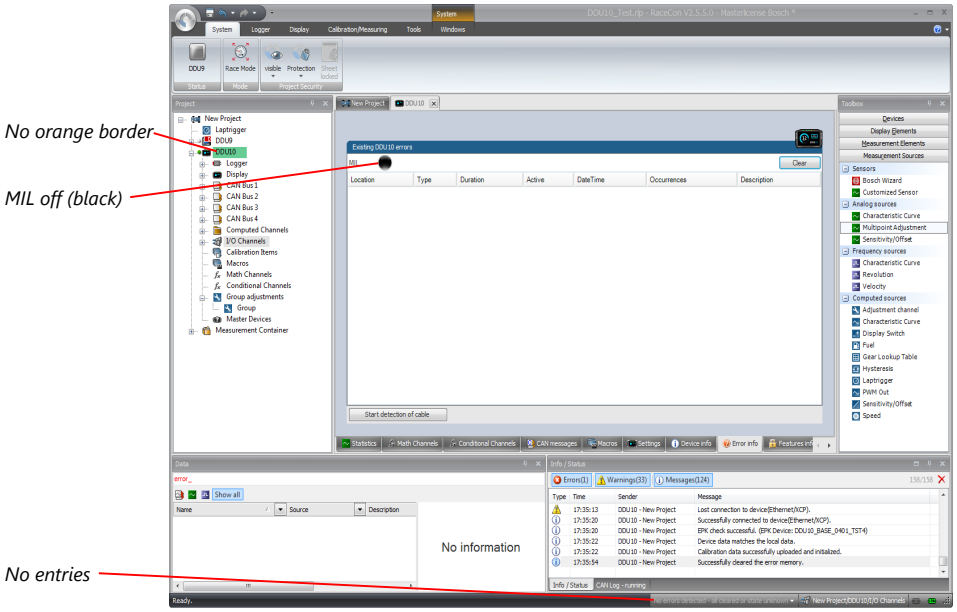
The following property is available for the error memory itself:

- Error Status (device measurement label “error_state”)
  0: no error present in memory
  1: at least one inactive error present in memory, no active errors
  2: at least one active error present in memory

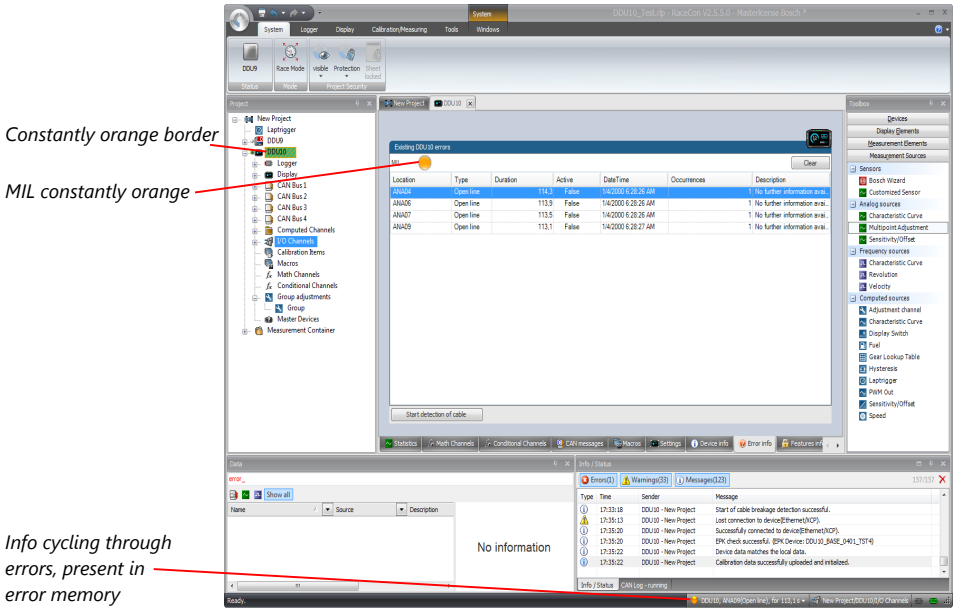
If displayed in a measurement sheet, this property’s value (0, 1 or 2) is translated into a verbal description:

It is also represented by a color scheme within RaceCon (provided RaceCon is online with the system):

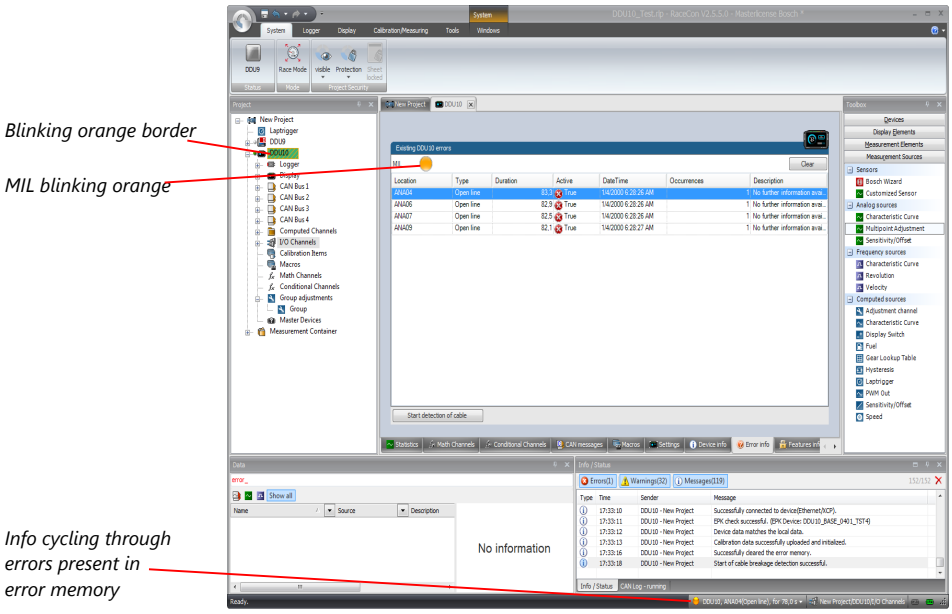
0 (no error present in memory):



1 (at least one inactive error present in memory, no active errors):

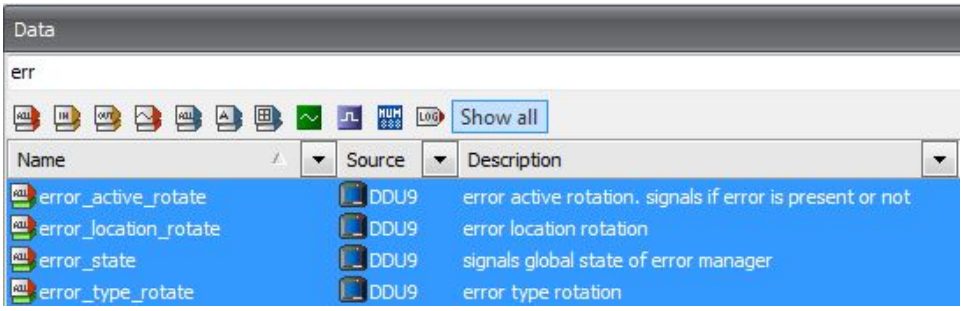


2 (at least one active error present in memory):



8.2.2 Error Properties

The following channels are recognized and memorized inside the devices:

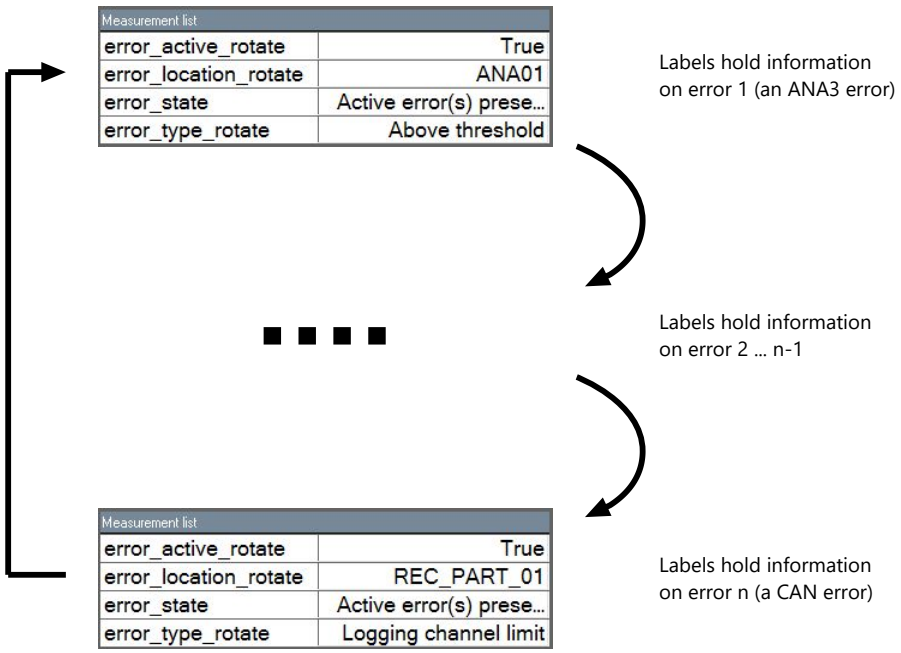


- Error type (device label “error_type_rotate”):
  e.g. “below_threshold” for a violation of the minimum voltage range defined in the configuration, “shortcut_Batt” for a shortcut to battery voltage etc.
- Error locations (device label “error_location_rotate”):
  e.g. “ANA01” for an error concerning the first ANA channel
- Error durations
  How long has the error been active? If an error encounters a non-active period before being cleared from the memory and is then detected again, the error duration keeps on accumulating. The number of active periods can be seen from the “number of occurrences”.
- Number of occurrences
  How many times has the error been detected since the last time the error memory was cleared.
- Error active state (device label “error_active_rotate”)
  All failure modes are continuously diagnosed; any error detected will be written to the error memory. Once an error is detected, it is qualified as “active”.
  - 1 (TRUE) Error was detected in most recent diagnose run (active)

– 0 (FALSE) Error is inactive: error was not detected in most recent diagnostic run, however the error has not been cleared from the memory by the user and remains in the non-volatile memory

The aforementioned channels (error_active_rotate, error_location_rotate, error_type_rotate) are device specific properties (e.g. C 60) and are not related to the complete RaceCon project (e.g. “error no. 3 from the error memory”). Therefore, only one property label is available in each device. The errors from the error memory (possibly more than one error possible per device) share these three labels. The labels cycle through the errors currently present in the memory and represent the respective property of each error periodically.

The following screenshot shows error properties, which can be displayed or logged:

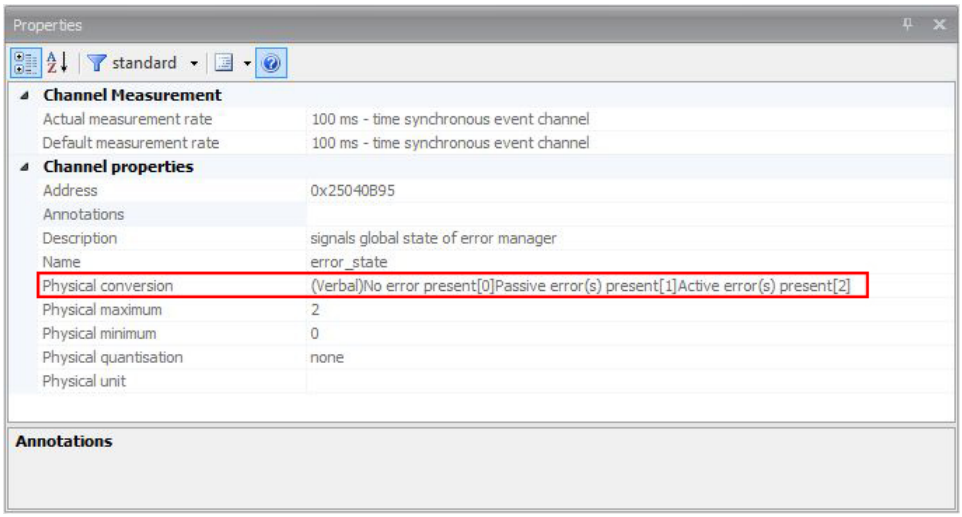


After the last error and its error properties have been displayed, the labels will start again with the first error in the error memory stack and its error properties will be displayed again. Therefore, monitoring these labels over a sufficiently long period provides the information on all individual errors in the error memory.

To understand this behavior, it is recommended to observe the three labels in a measurement sheet (while more than one error is active) and watch the values change periodically:

Measurement list	
error_active_rotate	True
error_location_rotate	REC_PART_01
error_state	Active error(s) prese...
error_type_rotate	Logging channel limit

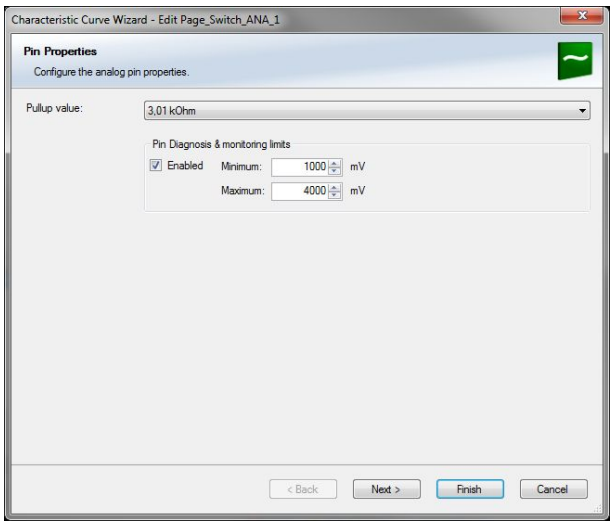
The verbal representation of the numerical codes of these labels can be visualized in the properties window of the measurement page:



8.3 Analog Input Diagnosis

8.3.1 Monitoring limits / Shortcut Detection / Cable Breakage

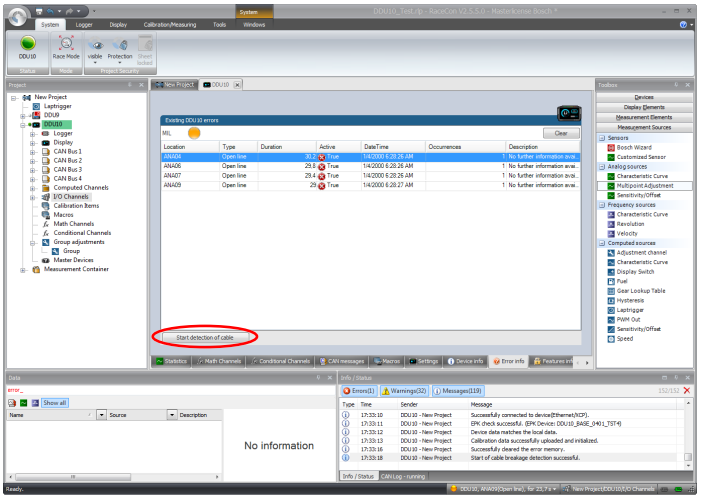
The pin diagnosis functionality (check whether measurement is within the desired range) can be activated in the ANA pin setup wizard; to allow for a diagnosis regarding shortcut to ground, shortcut to battery voltage and cable breakage, a minimum / maximum has to be defined.



8.3.2 Open Line Detection

The implementation of open line detection consists of pull up resistors being activated and deactivated; evaluating the behavior of the measured value detects cable breakage, regardless of the pull up resistor being activated by the user.

1. Open the Error Memory of the Device.
2. Click "start detection of cable".
3. Check the Error Memory for new fault entries, regarding "Open line errors".



9 Technical Data

The display DDU 10 integrates a programmable full color dashboard display with a data logging system for motorsport applications. Additional input devices can be connected via Ethernet, CAN buses and RS 232.

Data Analysis Software WinDarab is available free of charge as “WinDarab V7 free” on our website. A basic logging function of 100 channels with recording of 50 ms (3 GB) is always included. The logger can be upgraded to full logging performance (max. 1 ms). In addition a 2nd logging partition of 1 GB can be activated.

With the DDU 10, a completely new library of graphical elements for the individual design of display pages was implemented and an all-new user interface menu has been developed for the device. A configurable input activates the menu structure and the user can reset for example laptime, fuel consumption and many more, without having to connect a laptop to the DDU. The user can also install own graphics, pictures etc. on the 12 freely configurable display pages. For quick data transfer from the car, e.g. during pit stop, data copy to a USB stick is available as an option.

Mechanical Data

Size	198 x 134 x 35 mm
Weight	875 g
Protection classification	IP67
Operating temperature internal	-20 to 85°C
Max. vibration	Vibration profile 1 (see Appendix or www.bosch-motorsport.com)

Electrical Data

Supply voltage	6 to 18 V
Current consumption (without sensor supply)	2 A (at 12 V)

Inputs

Analog channels	4 standard, plus 12 optional
Input range	0 to 5 V
Resolution	12 bit
Switchable pull up resistor	For all analog inputs
Wheel speed inputs	4 Hall-effect or DF11, switchable

Outputs

Sensor supply 5 V ± 1 % (250 mA)	2
Sensor supply 10 V ± 1 % (250 mA)	1
Sensor supply U_Bat (250 mA)	1
Sensor ground	4

Environment

External switch for page selection, 12 steps	B 261 209 658-01

External switch for brightness adjustment or page selection, 6 steps	B 261 209 659-01

Optional Upgrades

USB Kit	Rugged USB flash drive Bosch File System (BFS) format included, works with Bosch File System (BFS) preformatted USB Flash drive only Adapter cable to USB-Port Adapter for wiring harness SW license USB-Port unlocked
CCP_MASTER	CCP-Master (ASAP2 file from ECU manufacturer required)
FULL_LOG_1	Enable full logging performance of 3 GB partition 1
FULL_LOG_2	Enable full logging performance of 1 GB partition 2
I_O EXTENS	Enable additional 12 analog inputs and 2 CAN channels

Connectors and Wires

Live connector on display AS216-35PN	
Mating connector AS616-35SN	F 02U 000 466-01
Auxiliary connector on display AS212-35PN	
Mating connector AS612-35SN	F 02U 000 443-01

Pin Configuration

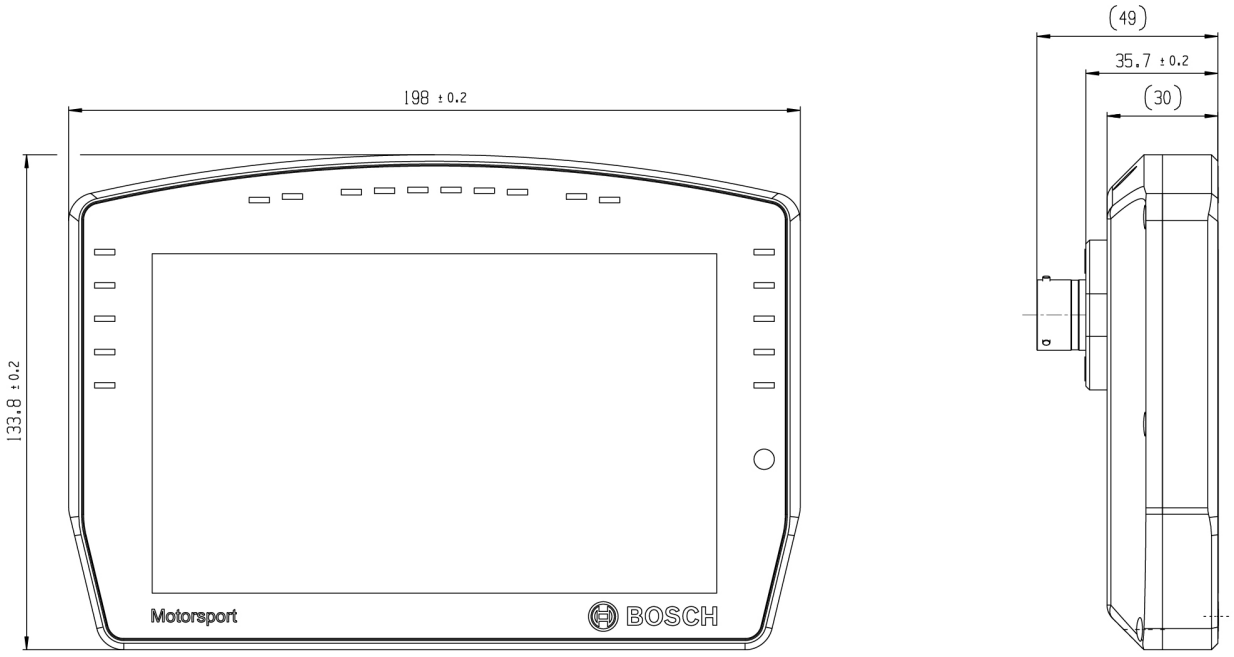
Live connector			
Pin	**Name**	**Comment**	**Status**
1	KL_31		Incl.
2	KL_15		Incl.
3	KL_30		Incl.
4	Rev_In_3	Hall or DF11 switchable	Incl.
5	Rev_In_1	Hall or DF11 switchable	Incl.
6	KL_31		Incl.
7	CAN_2_L	CAN speed selectable	Incl.
8	Ethernet_2_TXP		Incl.
9	Ethernet_2_TXN		Incl.
10	Sens_Power_12V	over current protected	Incl.
11	Rev_In_4	Hall or DF11 switchable	Incl.
12	Rev_In_2	Hall or DF11 switchable	Incl.

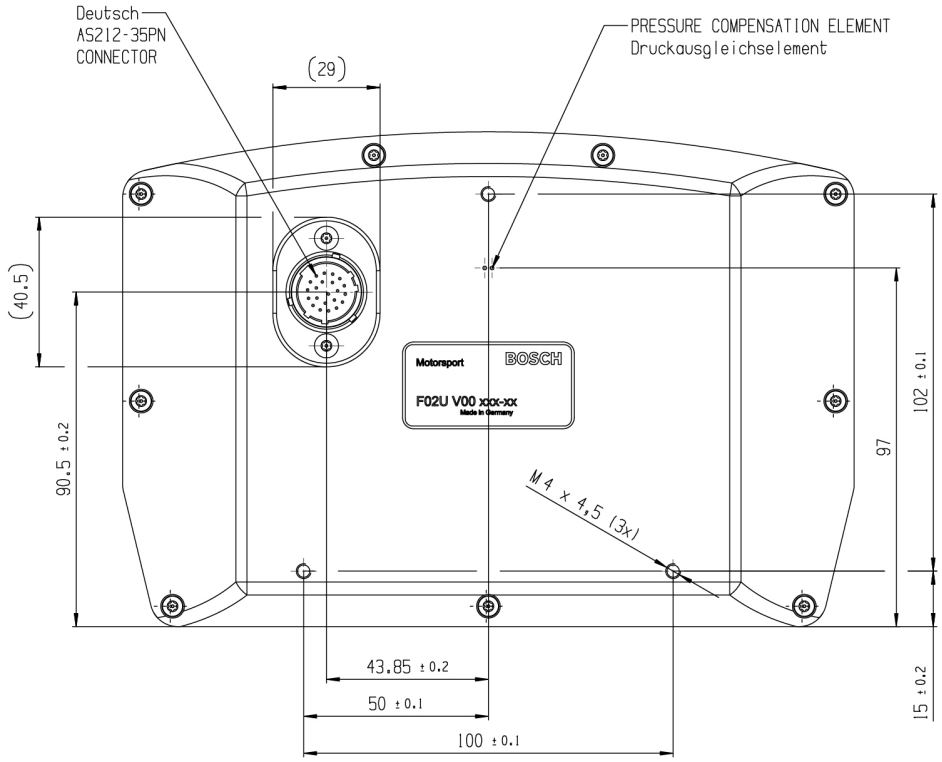
Live connector			
13	Laptrigger_In		Incl.
14	CAN_2_H	CAN speed selectable	Incl.
15	CAN_1_H	CAN speed selectable	Incl.
16	Ethernet_2_RXP		Incl.
17	Sens_Gnd_4	fused	Incl.
18	Sens_Power 5V	over current protected	Incl.
19	ANA_IN_3	3.01 kOhm switchable	Incl.
20	ANA_IN_4	3.01 kOhm switchable	Incl.
21	Time_Sync	connection to Bosch ECU	Incl.
22	CAN_1_L	CAN speed selectable	Incl.
23	Ethernet_screen		Incl.
24	Ethernet_2_RXN		Incl.
25	Sens_Gnd_3	fused	Incl.
26	Sens_Power 5V	over current protected	Incl.
27	ANA_IN_7	3.01 kOhm switchable	Opt.
28	ANA_IN_1	3.01 kOhm switchable	Incl.
29	USB_Device_DP	to Bosch USB stick	Opt.
30	RS232_TX_Telemetry		Incl.
31	Ethernet_1_TXP		Incl.
32	Sens_Gnd_2	fused	Incl.
33	Sens_Power_10V	over current protected	Incl.
34	ANA_IN_8	3.01 kOhm switchable	Opt.
35	ANA_IN_10	3.01 kOhm switchable	Opt.
36	USB_Device_Gnd	to Bosch USB stick	Opt.
37	USB_Device_DN	to Bosch USB stick	Opt.
38	RS232_RX_Telemetry	e.g. GSM telemetry	Incl.
39	Ethernet_1_TXN		Incl.
40	Sens_Gnd_1	fused	Incl.
41	ANA_IN_11	3.01 kOhm switchable	Opt.
42	ANA_IN_9	3.01 kOhm switchable	Opt.
43	RS232_TX_GPS		Incl.
44	ANA_IN_16	3.01 kOhm switchable	Opt.
45	USB_Device_Power	to Bosch USB stick	Opt.
46	Ethernet_1_RXP		Incl.
47	ANA_IN_12	3.01 kOhm switchable	Opt.
48	ANA_IN_6	3.01 kOhm switchable	Opt.
49	ANA_IN_2	3.01 kOhm switchable	Incl.
50	ANA_IN_13	3.01 kOhm switchable	Opt.
51	ANA_IN_15	3.01 kOhm switchable	Opt.
52	Ethernet_1_RXN		Incl.
53	ANA_IN_5	3.01 kOhm switchable	Opt.
54	RS232_RX_GPS	for GPS sensor input	Incl.

Live connector			
55	ANA_IN_14	3.01 kOhm switchable	Opt.

Auxiliary connector			
Pin	**Name**	**Comment**	**Status**
1		Unused	
2		Unused	
3		Unused	
4		Unused	
5		Unused	
6		Unused	
7		Unused	
8		Unused	
9	Ethernet_3_TXP		Incl.
10	Ethernet_3_RXP		Incl.
11	Ethernet_3_RXN		Incl.
12	CAN_4_H		Opt.
13		Unused	
14		Unused	
15		Unused	
16		Unused	
17		Unused	
18	Ethernet_screen		Incl.
19	Ethernet_3_TXN		Incl.
20	CAN_4_L		Opt.
21	CAN_3_H		Opt.
22	CAN_3_L		Opt.

10 Mechanical Drawing

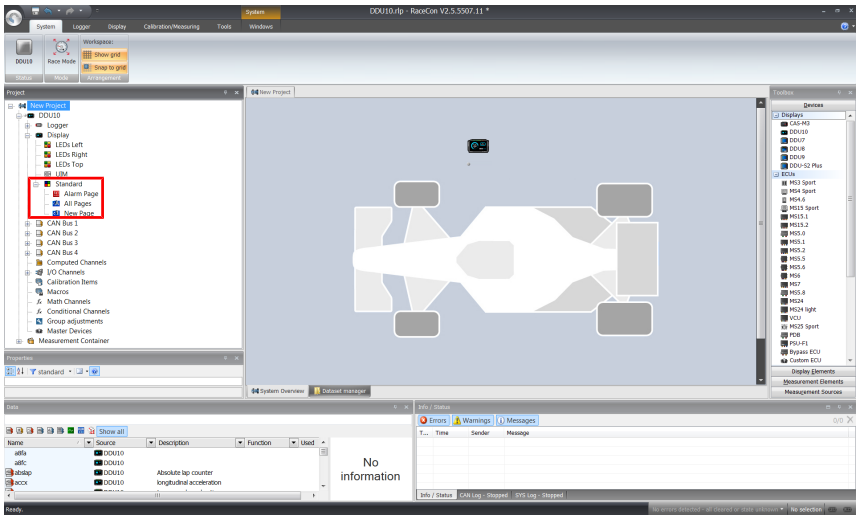




11 Display Configuration

11.1 Display page setup

11.1.1 Organizing display pages



- All Pages: Display elements placed on this page are displayed on all pages.
- Single Page: Display elements placed on this page are displayed only on this page.
- Under a Page set you can create different display configurations, like style or language based. You can choose which page set should be used on the device (via UIM), or via RaceCon, by right-clicking on the Page set in the project tree overview.

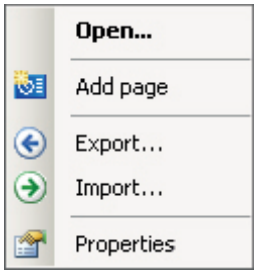
The priority of display elements placed on 'All Pages' is higher than the priority of display elements placed on single pages.

Alarm Elements have the highest priority and are displayed in front of all other elements. You can find alarm pages above "All pages" in the project tree.

Example: A bar graph placed on 'All Pages' is displayed on all display pages and is always in front of other display elements on single pages.

11.1.2 Adding a new display page

Click with the right mouse button on 'Display' in the project tree and click on 'Add page' in the menu or on the 'Add' button at the top.

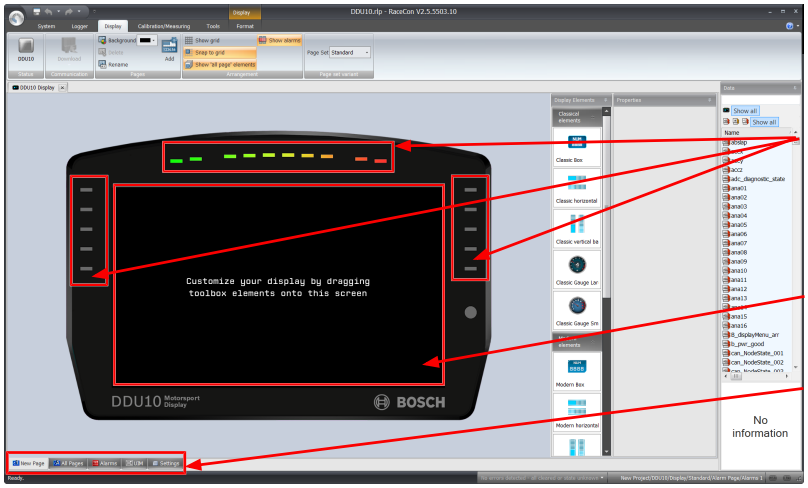


A new empty page opens.

11.1.3 Selecting display pages

Click on the ‘Display’ tab and then on one of the page tabs at the bottom (example: ‘New Page’).

In the Main Area, a representation of the selected page opens and can be configured.



11.2 Element Configuration

You can insert different display elements and configure these in the ‘Properties’ toolbox. For an overview of the display elements, see chapter ‘Types of display elements [▶ 74]’.

1. Drag a display element from the toolbox and drop it on the display page.

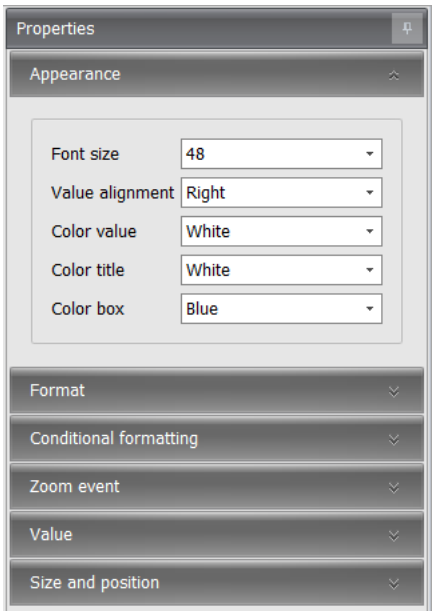
   The ‘Properties’ toolbox opens and you can configure the active element.

2. Use the search bar in the ‘Data’ window, to search for a measurement channel.
3. Drag the desired measurement channel from the ‘Data’ window and drop it on the display element.
4. Alternatively, use ‘Value channel’ in the ‘Properties’ toolbox to define a measurement.
5. To swap the active channel element, click on the needed element.
6. To perform a multi selection, click and hold the mouse button and swipe over the wanted elements. Alternatively, you can add or remove elements from the selection with CTRL + mouse click.

   (Only overlapping properties will be displayed.)

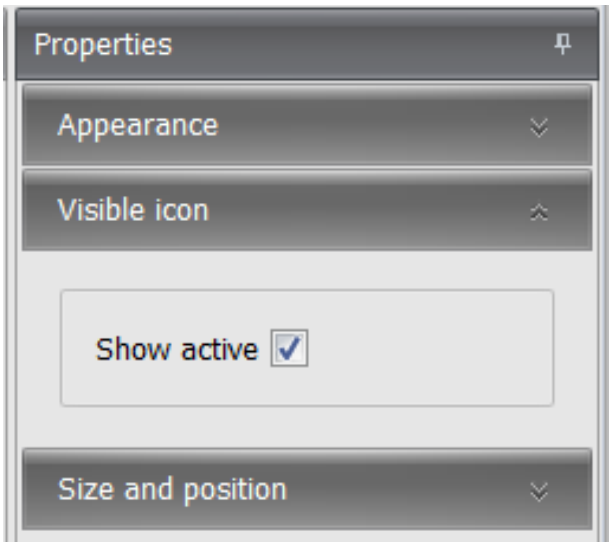
You can find the following configuration options in the ‘Properties’ toolbox. The following screenshots are created by way of example for available options. Which options will be available, depends on which element is active:

Appearance



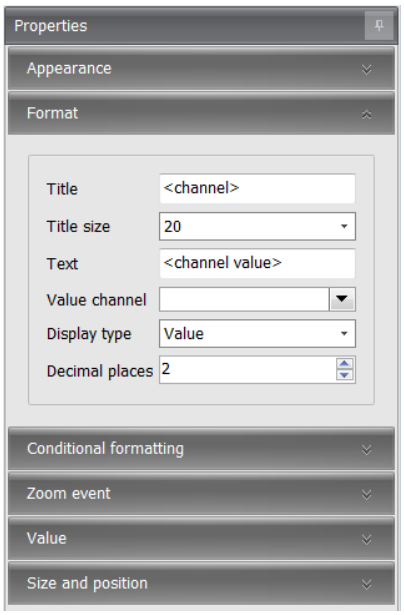
- Font size: Defines the size of the font.
- Alignment: Defines the alignment of the element content.
- Value alignment: Defines the alignment of the value.
- Orientation: Defines the orientation of the element content.
- Color value: Defines the color of the value.
- Color title: Defines the color of the title.
- Color box: Defines the color of the box.
- Color bar: Defines the color of the bar.
- Color gauge: Defines the color of the gauge.
- Color ticks: Defines the color of the ticks.
- Color border: Defines the color of the border.
- Background: Defines the color of the background.
- Image file: Here you can select the image you want to display.
- Scale image: Enables the fitting of the image according to the box.
- Keep aspect ratio: Keeps the original ratio of the image.
- State channel: Defines the channel to which an alarm icon should react.
- Active icon: Defines an icon, which appears when the channel is active.
- Use inactive icon: Enables the usage of an inactive icon.
- Inactive icon: Defines an icon, which appears when the channel is inactive.

Visible icon



– Show active: Enables testing of the active icon (only for alarm icon elements).

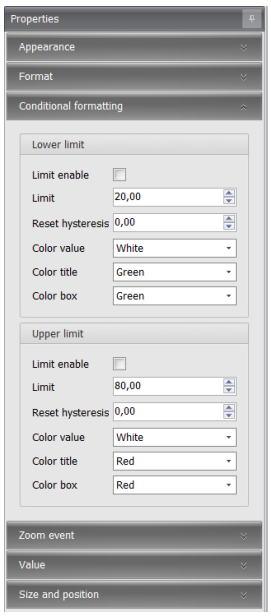
Format



– Title: Defines a specific title.
– Title size: Defines the font size of the title.
– Text: Defines a specific text and/or a value, which will always be displayed in the box.
– Value channel: Defines the value channel, which will be monitored.
– PopUp condition: Defines the condition when the element will be visible on the display.
– Display type: Defines the type of information, which will be displayed. You can differentiate between 'Value', 'Gear' and a type of time display.
– Decimal places: Defines the decimal places, which will be displayed.
– Fill style: Defines the style of the bar filling. You can choose between blocks and solid.
– Tick visibility: Defines the visibility of the ticks.
– Minimum value: Defines the minimum value at which the counting starts.

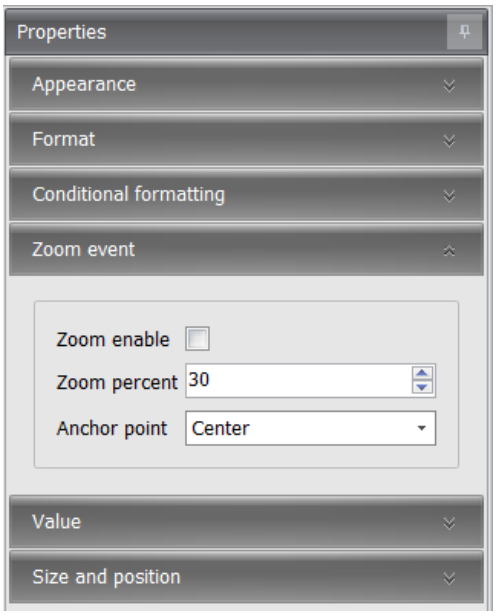
- Maximum value: Defines the maximum value at which the counting ends.
- Start angle: Defines the angle at which the ticks start.
- Angle range +/-: Defines the angle range of the ticks.
- Tick mark list: Here you can edit the tick mark position.
- Sub divisions: Defines the number of sub divisions between the main ticks.
- Conversion table: Here you can edit the conversion table.
- Fallback: Defines the displayed information when the value channel is invalid for the enumeration box.

Conditional formatting (lower and upper limit respectively)



- Limit enable: Enables a lower limit and an upper limit respectively.
- Limit: Defines the threshold limit.
- Reset hysteresis: Defines the value as off when the hysteresis will be reset. You can find more information about the hysteresis function in general in the chapter Hysteresis [▶ 42].
- Color value: Defines the color when the value is above/below the set threshold limit.
- Color title: Defines the new color of the title when the value is above/below the set threshold limit.
- Color box: Defines the new color of the element when the value is above/below the set threshold limit.
- Color gauge: Defines the new color of the gauge when the value is above/below the set threshold limit.
- Color ticks: Defines the color of the ticks when the value is above/below the set threshold limit.
- Color bar: Defines the color of the bar when the value is above/below the set threshold limit.
- Color border: Defines the new color of the border when the value is above/below the set threshold limit.
- Background: Defines the color of the background when the value is above/below the set threshold limit.

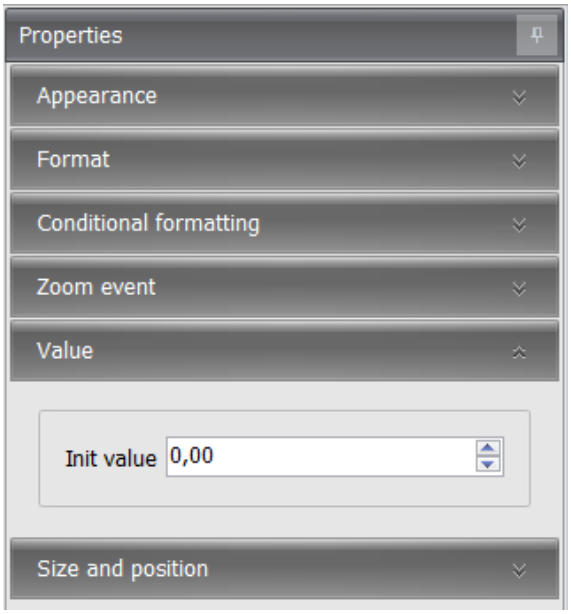
Zoom event



- Zoom enable: Enables the zoom event.

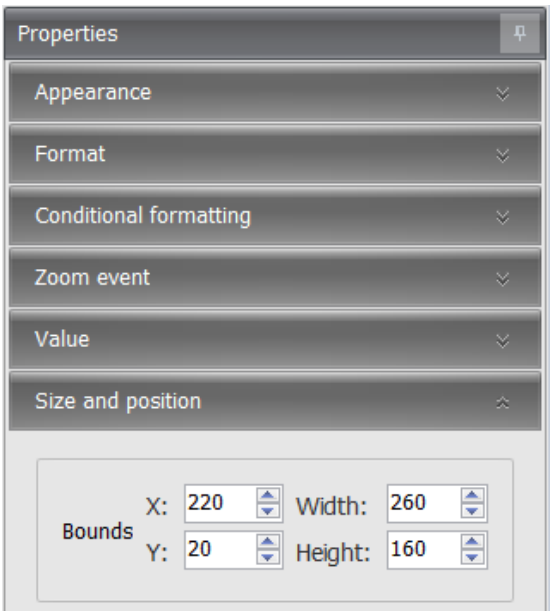
  If checked, the box will increase its size for a short time, after the value changed.
- Zoom percent: Defines the percentage by which the box will zoom out.
- Anchor point: Defines the anchor point, which will be fixed during the zoom event.

Value



- Init value: Defines the value, which will be shown in RaceCon during configuration or on the device if the channel has no link. Allows testing the conditional formatting by entering a representative value.

Size and position



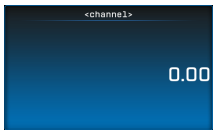
- X-Pos: Defines the x-position of the top left corner of the box within the page.
- Y-Pos: Defines the y-position of the top left corner of the box within the page.
- Width: Defines the width of the element.
- Height: Defines the height of the element.

11.3 Types of display elements

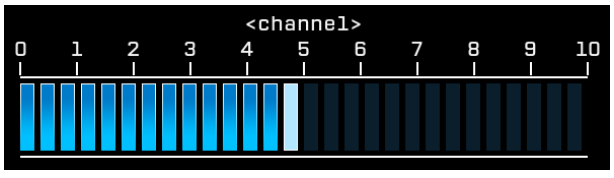
RaceCon has two types of element styles, the ‘Classical elements’ and the ‘Modern elements’. Both styles contain the same type of elements; the difference is solely in appearance. Additionally there is the basic element collection, which consists of elements without any style. Some of these are also transparent and can be used in combination with other elements by layering them.

You can find the following display elements in the ‘Display Elements’ catalog:

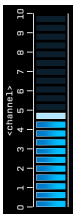
Classic Box



Classic horizontal bar



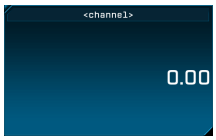
Classic vertical bar



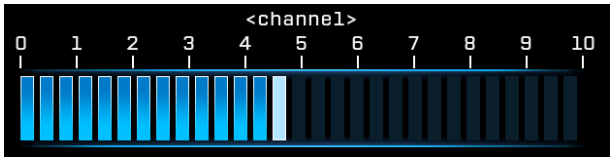
Classic Gauge Large

Classic Gauge Small

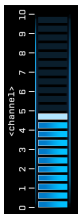
Modern Box



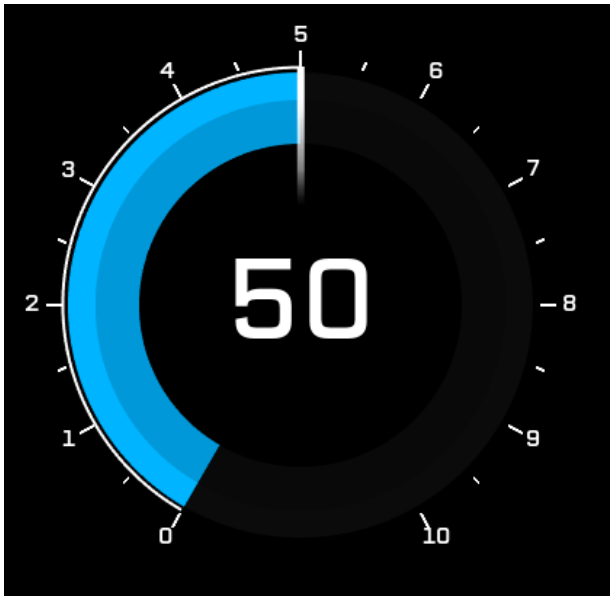
Modern horizontal Bar



Modern vertical Bar



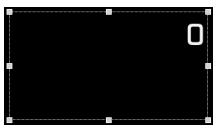
Modern Gauge Large



Modern Gauge Small

Basic Box

Value / Text



Can be used as a simple value or text element, which might be placed on top of another element, like the classic gauge large.

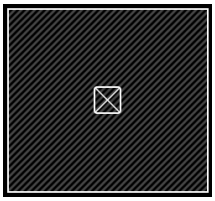
Enumeration

Can be used to translate a value into a text. For example for a wiper setting 0 - Off, 1 - Slow, 2 - fast

Image

Can be used to insert images, for example as background images.

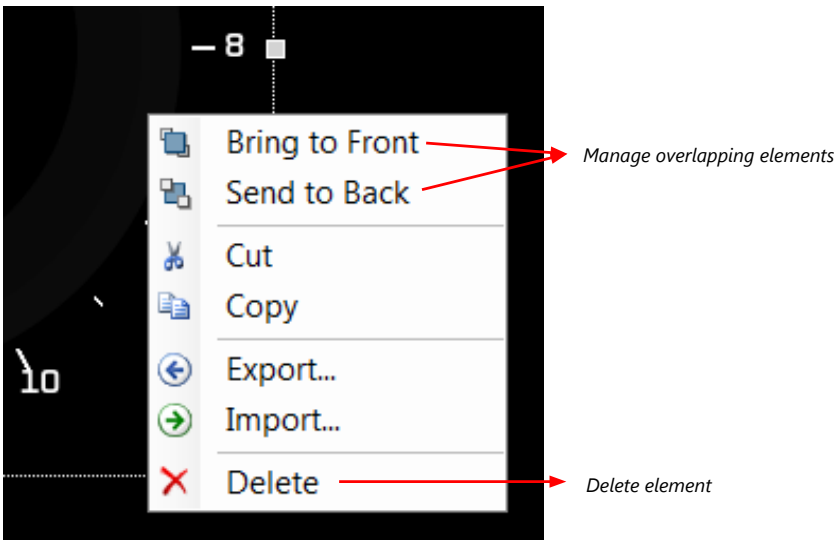
Alarm Icon



Icon or Icons, which can be linked to a conditional channel.

11.4 Context menu

The context menu appears when you click with the right mouse button on a display element.



11.5 LEDs

The LEDs are fully configurable to show for example the optimal shifting point. They can also be configured to flash in case of a customized condition becoming 'true'.

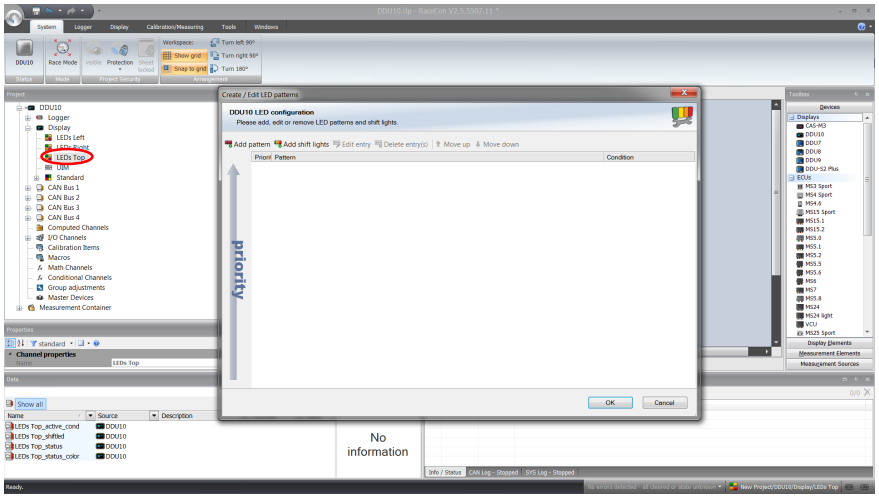
11.5.1 Configuring shift LEDs

To use shift LEDs, you need a channel with the unit RPM. It is also possible to use gear depending shift light.

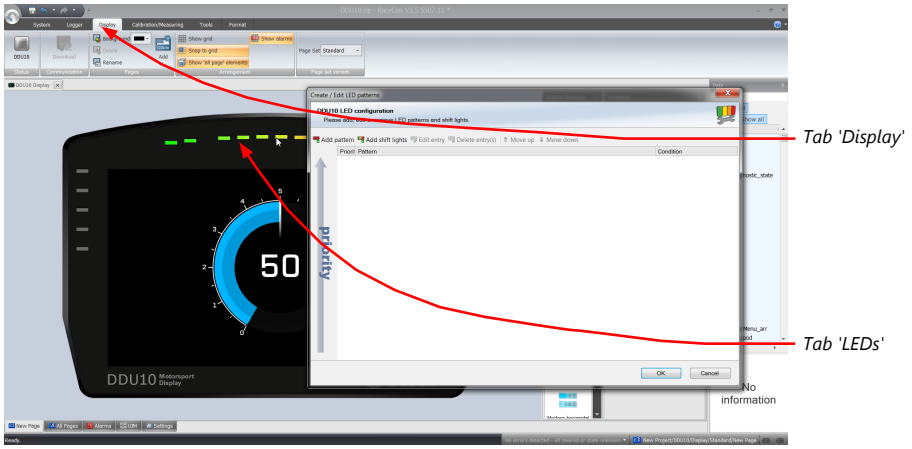
You can configure the shift LEDs only in the 'LEDs Top'.

1. Double-click on 'LEDs Top' in the project tree.

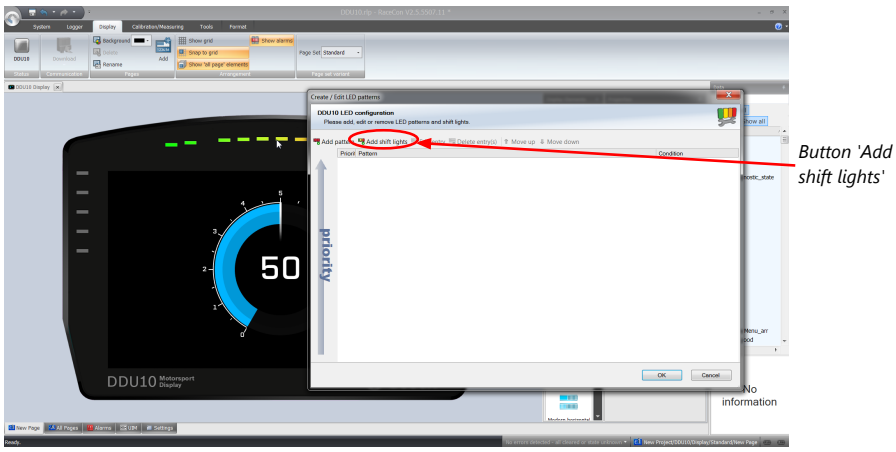
   You will find them under 'Display'.



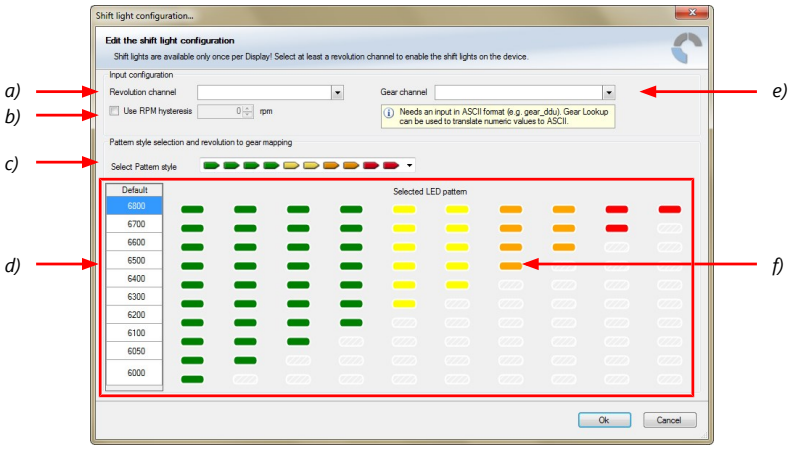
2. Alternatively, click on the 'Display' tab and then click on the colored LEDs at the top of the display image.



3. Click on the button 'Add shift lights'.



4. The ‘shift light configuration’ window appears. Set up the shifting lights using the following configuration possibilities:



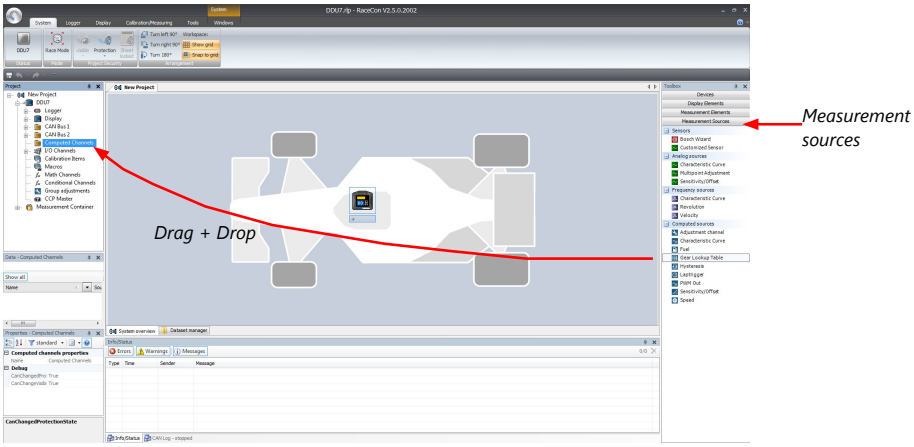
a) Choose the measurement channel for 'Revolution'. Revolution must have 1/min quantization.
b) Enter the limit value when the RPM hysteresis function is active. The RPM hysteresis function avoids the high-frequent switchover of the measurement channel value.
c) Choose a predefined Pattern style.
d) Define the rpm thresholds to show the LED pattern.
e) Choose the measurement channel for 'Gear'. Gear must have an ASCII quantization (1st gear='1' = 49, 2nd gear='2' = 50, ...). (ASCII quantization is standard for the 'gear' channel of Bosch ECUs.
If you get the gear information of a different control unit as the Bosch ECU (e.g. a gearbox control unit), use the Gear Lookup Table to translate numeric values to ASCII format.
For more information see chapter "Converting a gear channel to ASCII representation".)
f) Click with the right mouse button on the LEDs to reconfigure the LED patterns.

5. Click ‘OK’ when done. The configuration is displayed in the DDU 10 LED Configuration window.

11.5.2 Converting a gear channel to ASCII representation

If you get the gear information of a different control unit as the Bosch ECU (e.g. a gearbox control unit), use the Gear Lookup Table to translate numeric values to ASCII format.

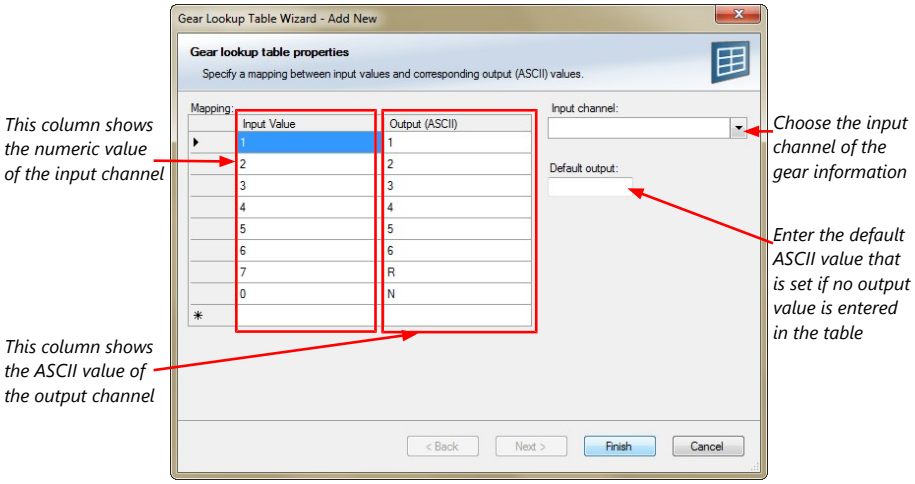
1. Click on the ‘Measurement Sources’ button in the Toolbox.
2. Drag the ‘Gear Lookup Table’ symbol and drop it on the ‘Computed Channels’ folder.



The Gear Lookup Table Wizard appears.

Gear Lookup Table Wizard

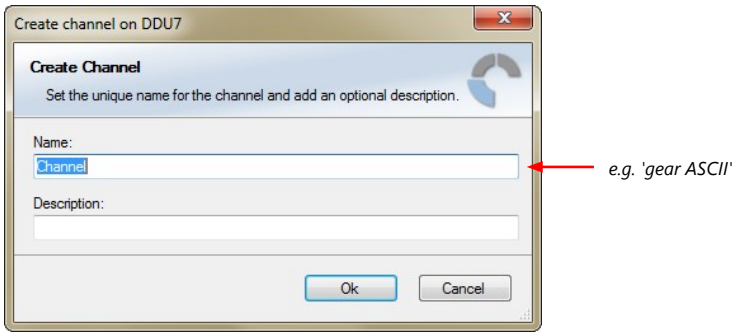
1. Set up the settings as shown in the screenshot.



2. Click ‘OK’ when done.

   The ‘Create channel on DDU 10’ window appears.

3. Enter the name and an optional description of the translated ASCII measurement channel.



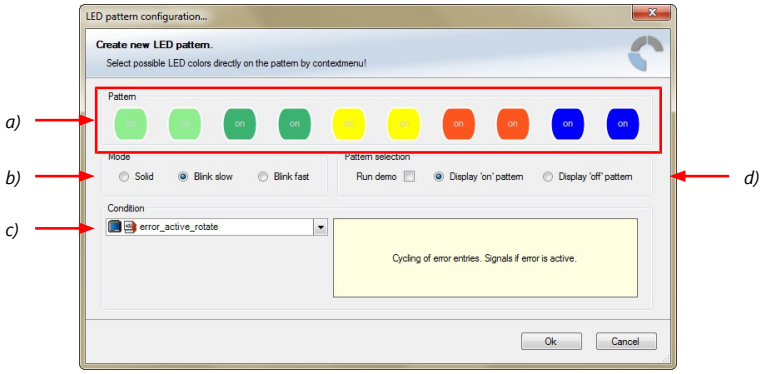
4. Click ‘Ok’ when done.

   A graphic shows the connection between the input and output channels. You can now use the measurement channel in the shift LED configuration.

11.5.3 Creating customized LED pattern

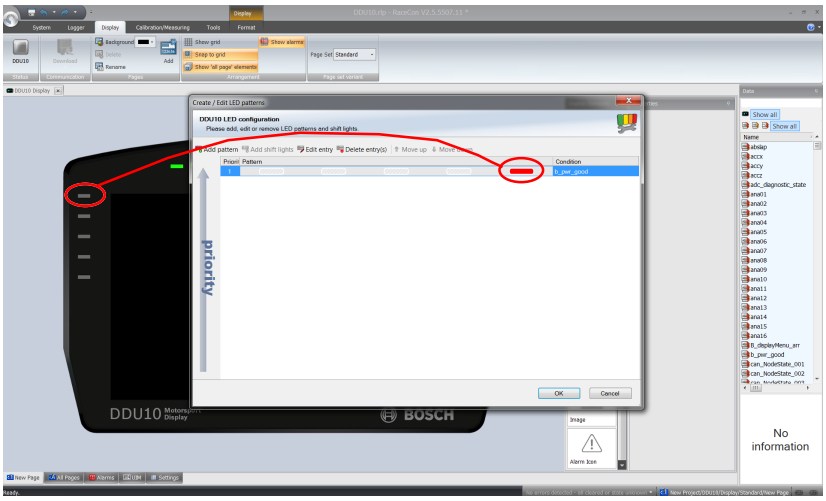
You can create your own LED pattern with an individually created condition, using the top LEDs and the ones on the sides. The LEDs illuminate or flash if the condition becomes true.

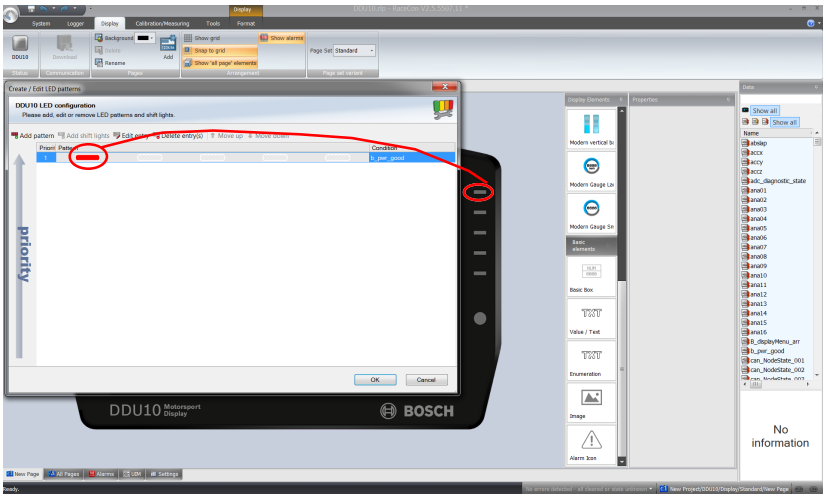
1. Click on the button ‘Add pattern’ in the display view. The LED pattern configuration window appears.



a) Choose the needed LEDs by clicking or via multiselect with ctrl + clicking and define the color of the LEDs by right-clicking on one of the selected LEDs
b) Select if the LEDs blink or do not blink.
c) Choose the condition when the LEDs will flash.
- Create a condition using the Condition Creator. For more information see chapter "Creating a new condition channel".
- Choose an existing condition.
d) Check the box to show a demo of the LEDs. (Important to check blinking)
To create a LED that alternately blinks in two different colors, choose 'Display "on" pattern' and define the LEDs in the first color. Then choose 'Display "off" pattern' and define the LEDs in the second color.

The direction of the pattern changes for each side. For the LEDs on the left side the pattern starts at the bottom LEDs (**right** side of pattern is for the top LEDs), and for the LEDs on the right side the pattern starts at the top LEDs (**left** side of the pattern is for the top LEDs).





2. Click ‘OK’ when done.

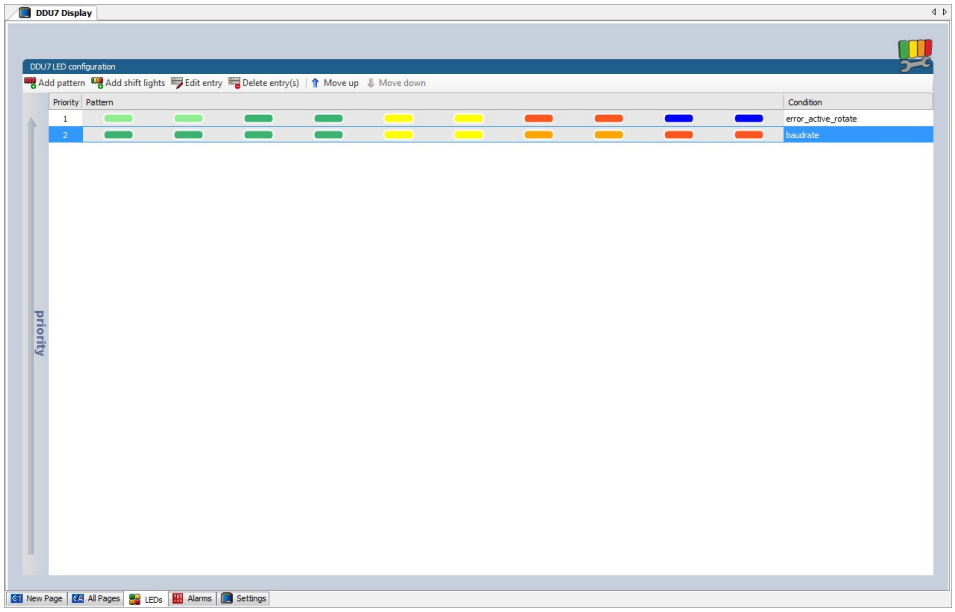
The configuration is displayed in the DDU 10 LED Configuration window.

11.5.4 Assigning display pattern priority

You can assign the priority of the created display pattern and shift lights.

The pattern with a higher priority will always cover patterns with a lower priority, when it becomes active. If a transparent (grey) LED is used in a pattern, the LEDs of lower patterns will be visible. Please ensure that for example shift lights do not cover important warnings.

Click the ‘Move up’ or ‘Move down’ button to change the priority. The pattern with the lowest number will have the highest priority.



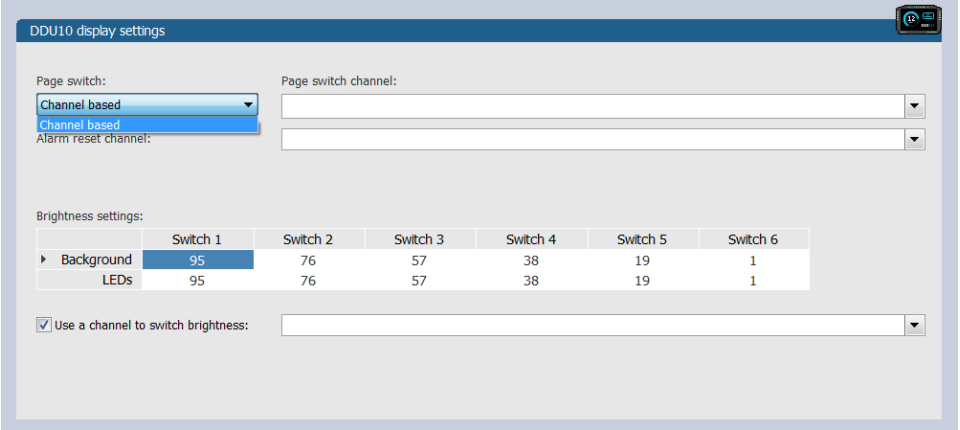
11.6 Page select / Display brightness / LED brightness

Any “event” can be used to change the display and LED brightness or the display page. Those events can be any input channel or an internal calculated channel. In the following chapters, you will find some examples on how to set up such a configuration.

11.6.1 General information

To use a channel as a page switch, select “Channel based” as page switch and select the channel you configure for a switch in the ‘display settings’ dialog (as described in the following chapters).

To use a channel as a brightness switch, check the box “Use a channel to switch brightness” and select the channel you configure for a switch.



Page switch

The pages can be switched from page one too twelve. If not all twelve pages are defined, the device switches up to the highest defined page number, and ignores higher numbers, which are not defined.

If the channel value does not only consist of integers, the pages will be switched as follows:

Page 1 is shown with the value < 1.5

Page 2 is shown with 1.5 <= the value < 2.5

Page 3 is shown with 2.5 <= the value < 3.5

...

Page 12 is shown with 11.5 <= the value

Brightness switch

The brightness can be switched with 6 positions. In the display settings dialog you will find a chart for the 6 switch positions over the display brightness and the LED brightness (the values are in percent).

If the channel value does not only consist of integers, the pages will be switched as follows:

Switch 1 is shown with the value < 1.5

Switch 2 is shown with 1.5 <= the value < 2.5

Switch 3 is shown with 2.5 <= the value < 3.5

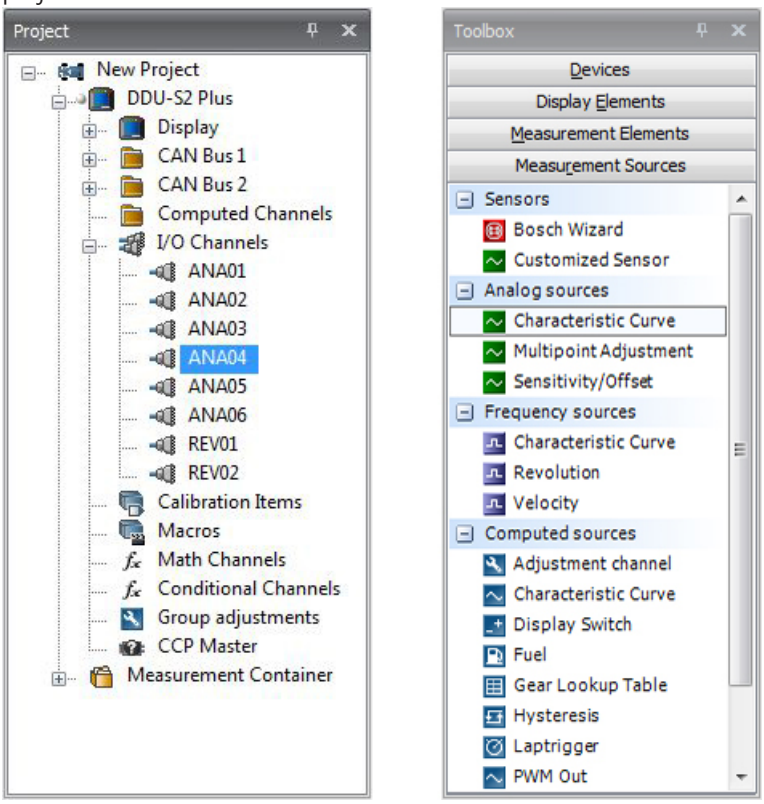
...

Switch 6 is shown with 5.5 <= the value

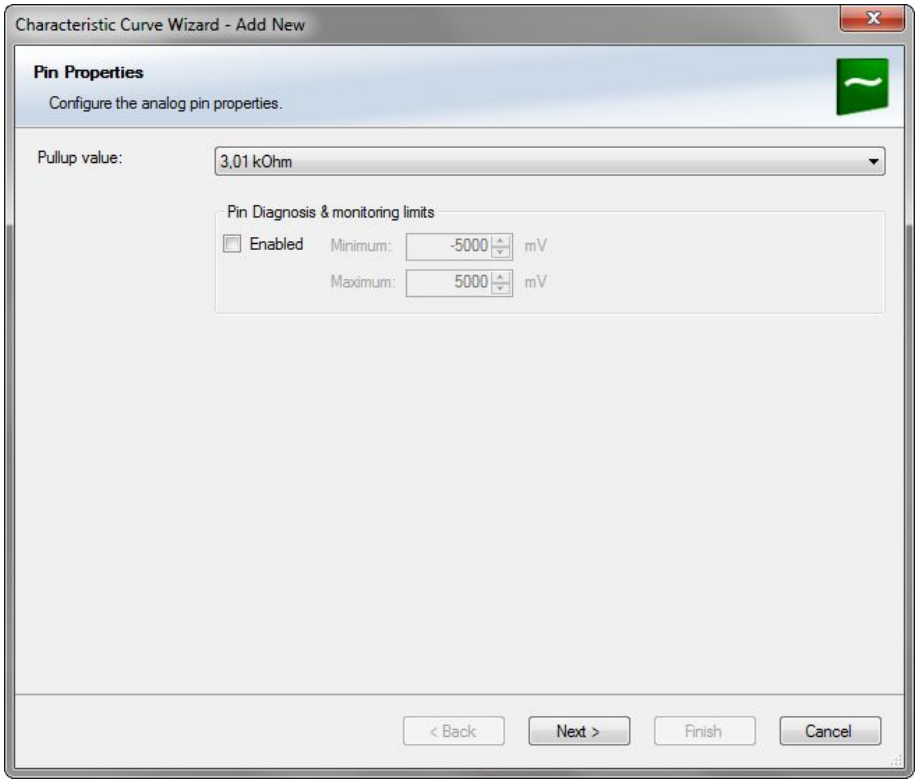
11.6.2 Option 1: 12 Hardwired position switch

1. Connect a 6 or 12 position switch to one of the analog input pins ANAxx and to the sensor ground.
   For recommended position switches, please see the environment section in the chapter “Technical Data.”
2. Select one of the analog inputs in the project tree.

3. Go to ‘Measurement Sources’ in the toolbox and select the ‘Characteristic Curve’ under ‘Analog sources’. Drag and drop it on the selected analog input channel in the project tree.

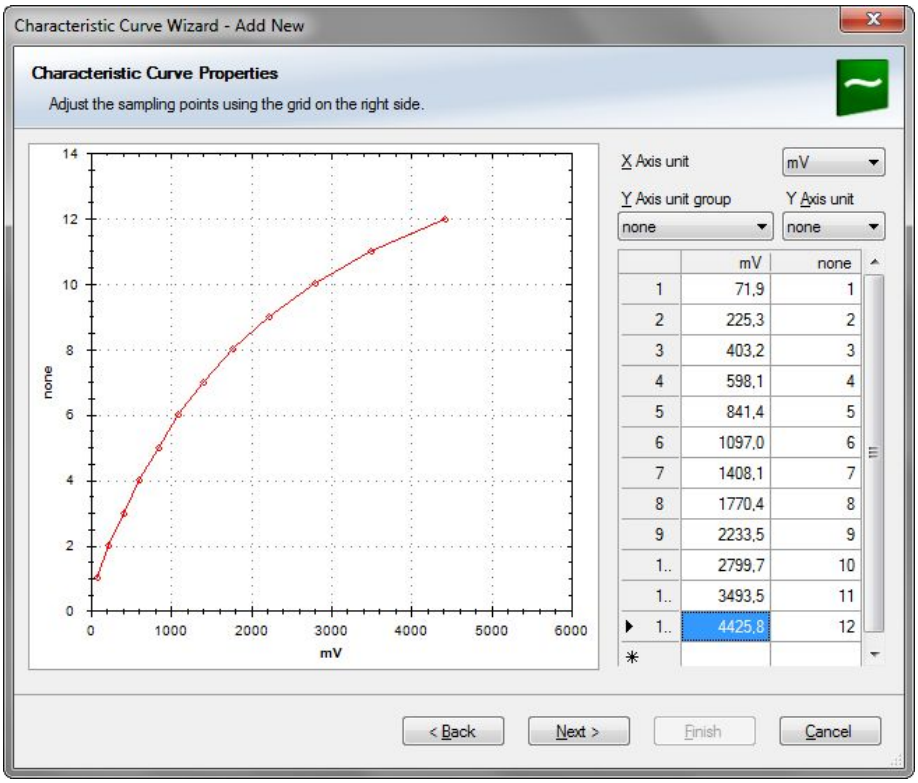


4. Select “Pull-up value:” 3.01 kOhm and click on ‘Next’.

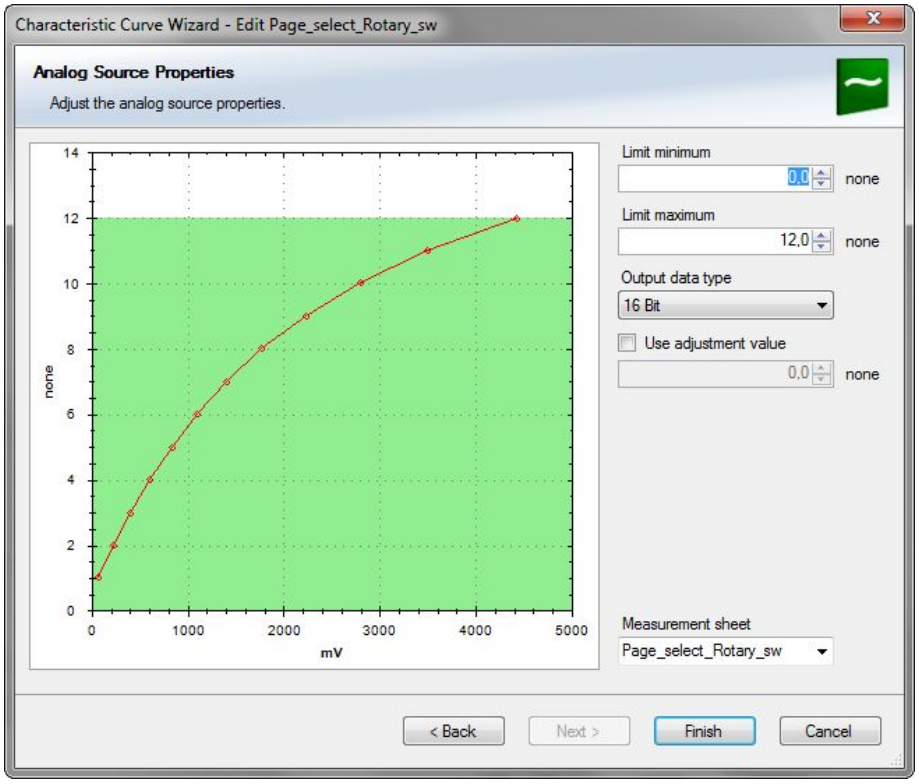


5. Define the relation between voltage and switch position and click on ‘Next’.
   Voltage = 5000 x R/(R + 3010)
   5000: Sensor supply (mV)
   R: Resistor for each Rotary switch position (Ohm)
   3010: Pull-up resistor (Ohm)

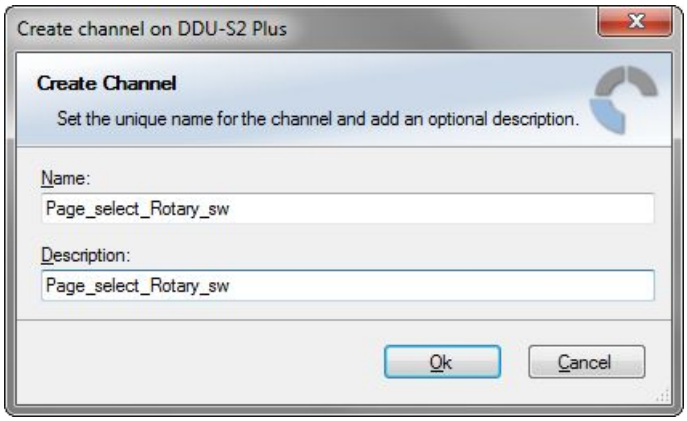
   The following screenshot and the data are an example for a Bosch switch.

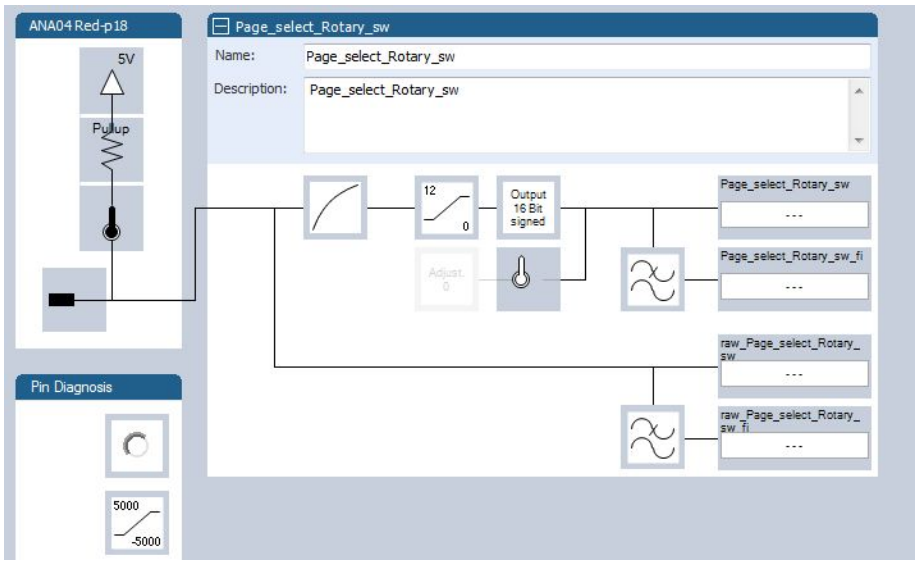


6. Define minimum and maximum Limit.
   Select “Output data type” from 8, 16 or 32 Bit.
   Do not check “Use adjustment value”.
   Choose the Measurement sheet and click on ‘Finish’.



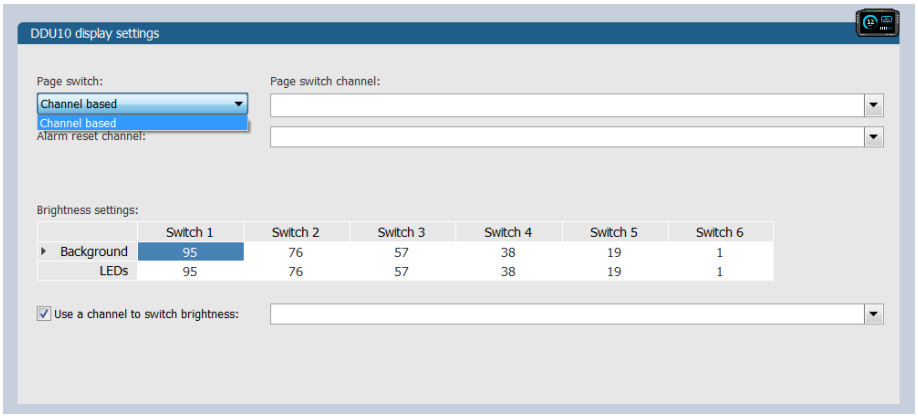
7. Define Name and Description and click on ‘Ok’.



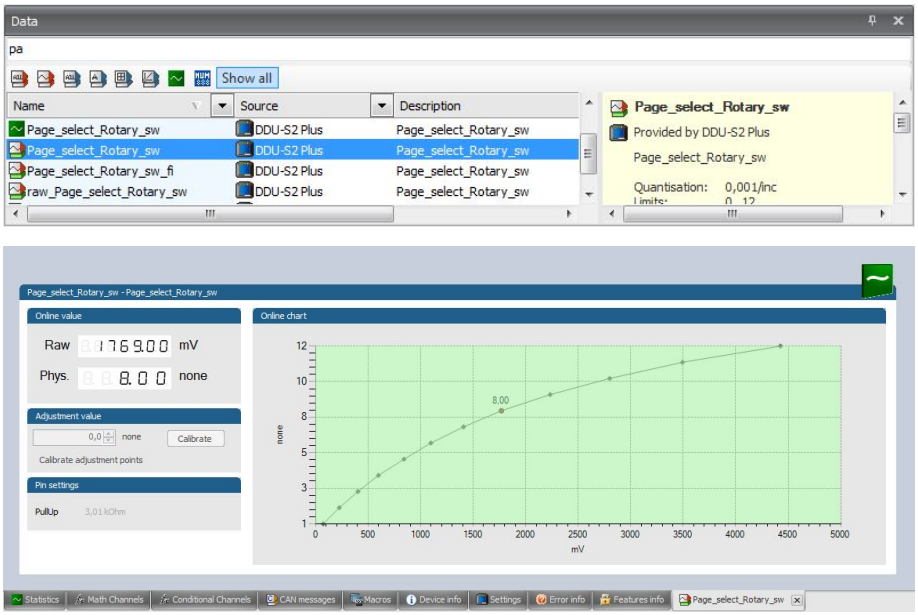


8. Click on the ‘Display’ tab and select the ‘Settings’ tab at the bottom.

9. To use the channel as a Page Switch, Select "Channel based" as page switch and select the channel configured above.
10. To use the channel as a brightness switch, check the box "Use a channel to switch brightness" and select the channel configured above.

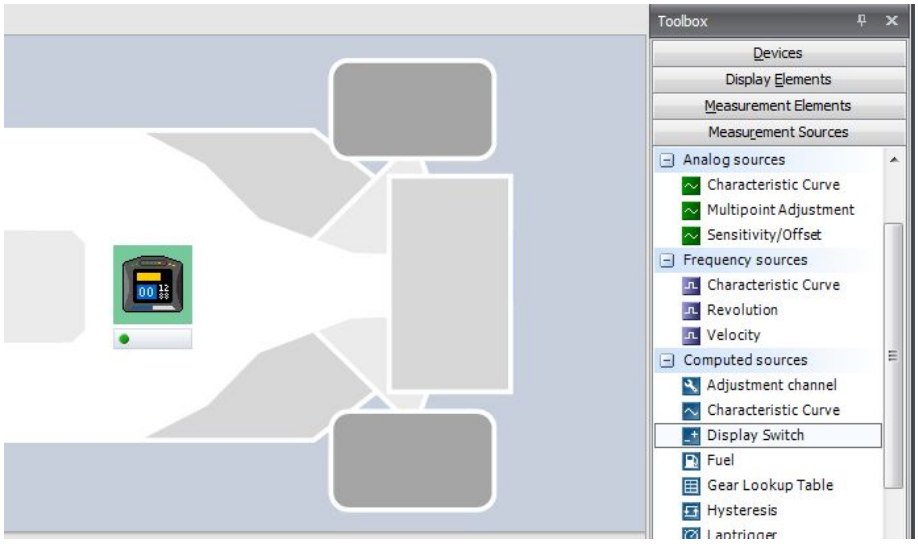


11. Click the ‘Download’ button at the top, to download the configuration.
12. If you want to check your configured channel, ensure that the device status is green, search for the configured channel in the Data window and double-click on it.
You will see a graphical display with the raw and the physical value of the channel.

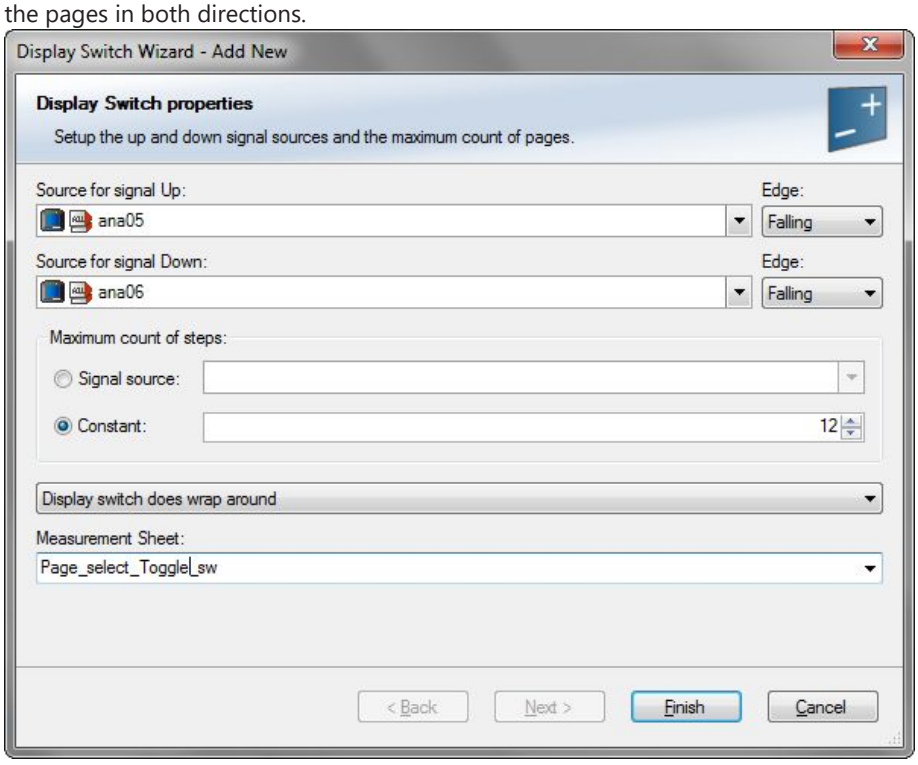


11.6.3 Option 2: Up/Down switches

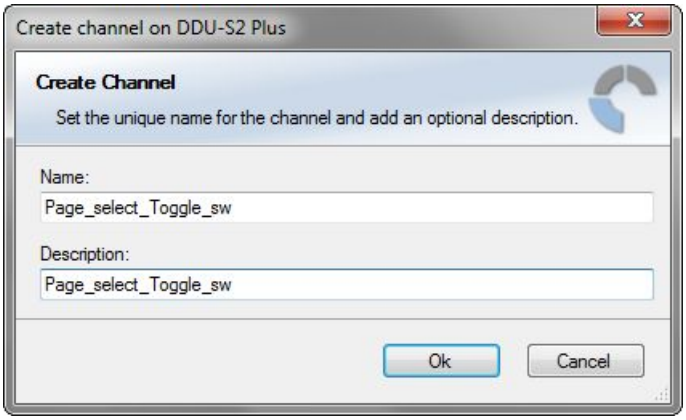
1. Define either one signal for a wrap around switch or two signals for an up/down switch.
2. Select the ‘Display Switch’ and drag it into the DDU 10.

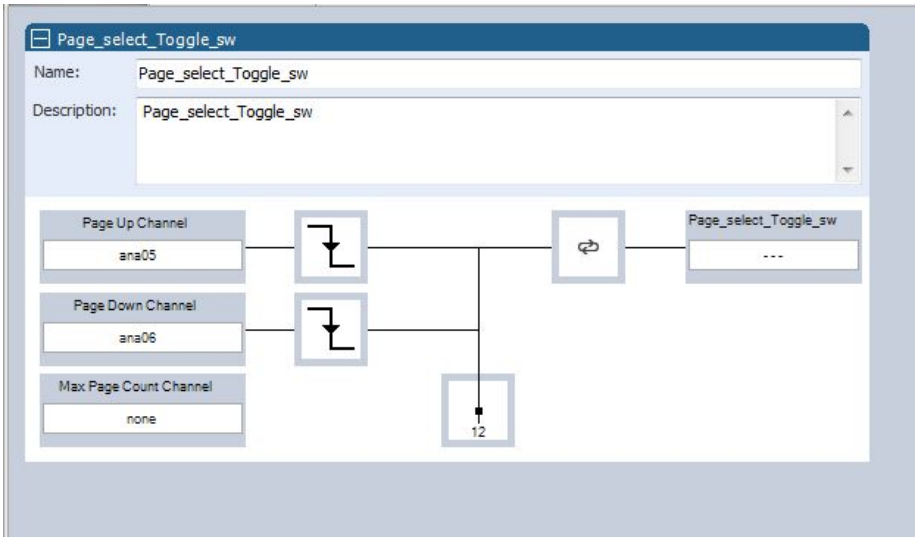


3. Select the Source for signal Up/Down and Edge Falling or Rising.
   Select the Maximum count of steps from signal source or from constant.
   Select “Display switch does wrap around” or “Display switch does not wrap around”.
   “Display switch does wrap around” goes from maximum position to minimum position or the other way around (by switching) in a loop, after the last page it starts again with the first page.
   If you choose “Display switch does not wrap around”, you need two switches to turn the pages in both directions.

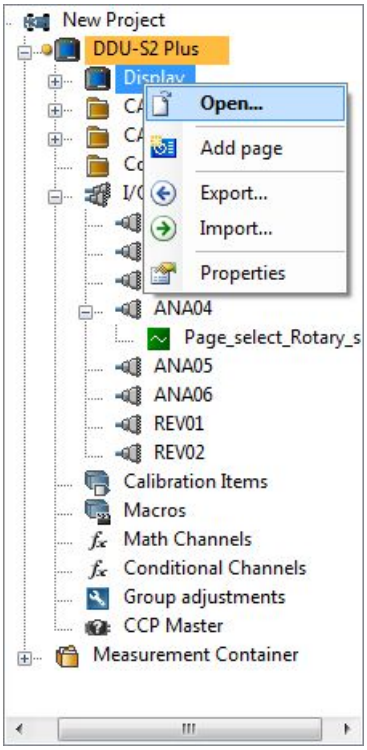


4. Define Name and Description and click on ‘Ok’.



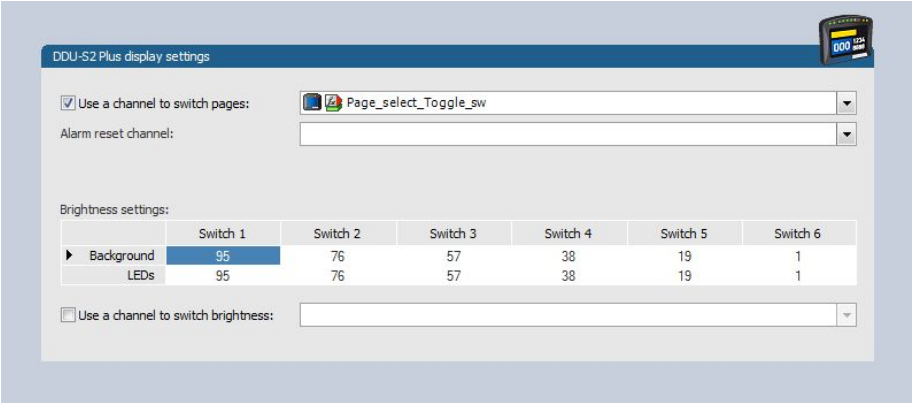


5. In the project tree, select “Display” and then “Open”.

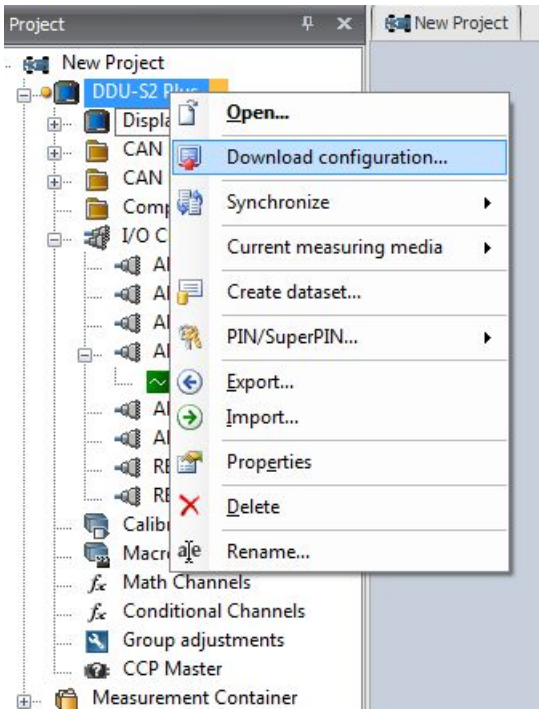


6. To use the channel as a Page Switch, check the box "Use a channel to switch pages" and select the channel configured above.

7. To use the channel as a brightness switch, check the box "Use a channel to switch brightness" and select the channel configured above.

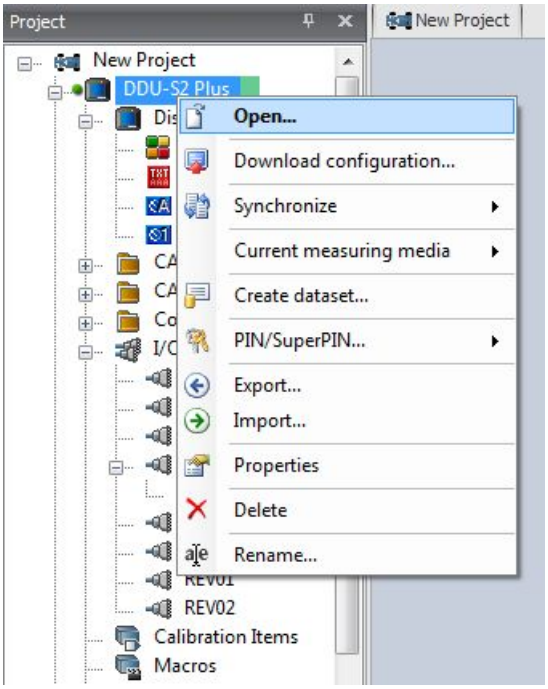


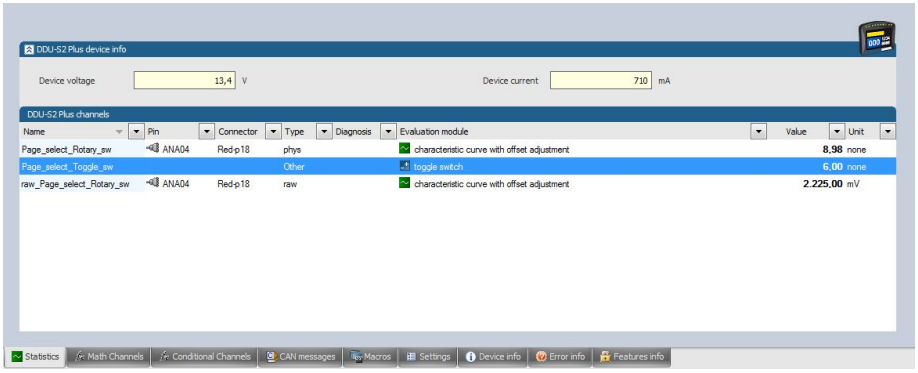
8. In the project tree, select “Download configuration”.



9. If you want to check your configured channel, reassure that the device status is green, double-click on the DDU 10 and select the "Statistics" tab.

   The configured channel position opens.

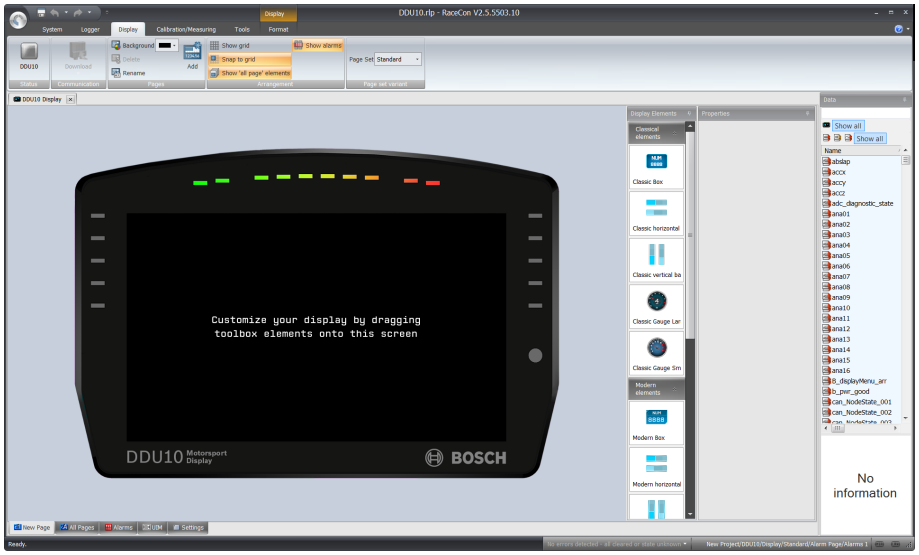




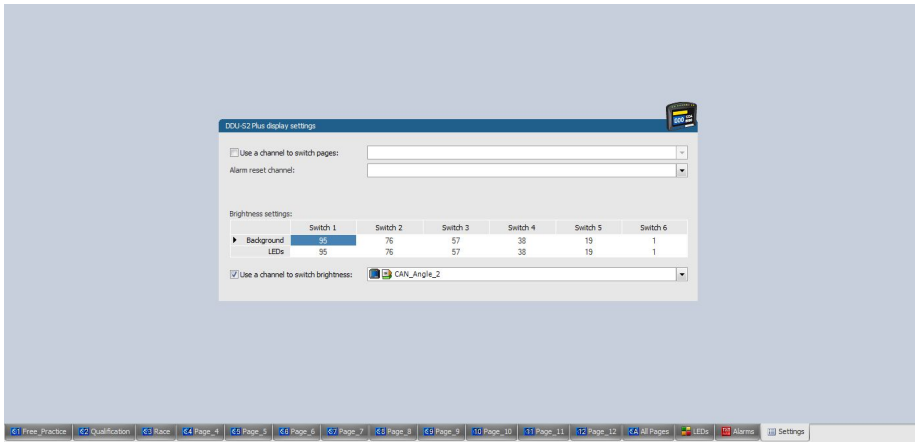
11.6.4 Option 3: CAN input signal, math channel or ANA_IN channel

Define your CAN, math or analog input channel.

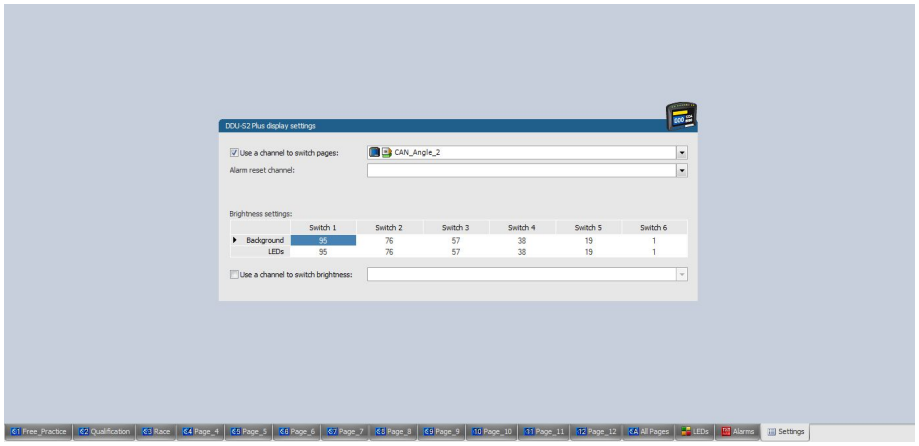
1. Select the 'Display' tab and then the 'Settings' tab at the bottom.



A ‘display settings’ window opens.



2. To use the channel as a brightness switch, check the box "Use a channel to switch brightness" and select the channel configured above.



3. To use a channel as a Page Switch, check the box" Use a channel to switch page" and select the channel configured above.
4. Click on the ‘Download’ button in the upper left corner.

11.7 Alarms

The ‘Alarm’ display element displays a warning message, when a defined condition becomes ‘true’. When a condition becomes ‘false’, the ‘Alarm’ display element disappears.

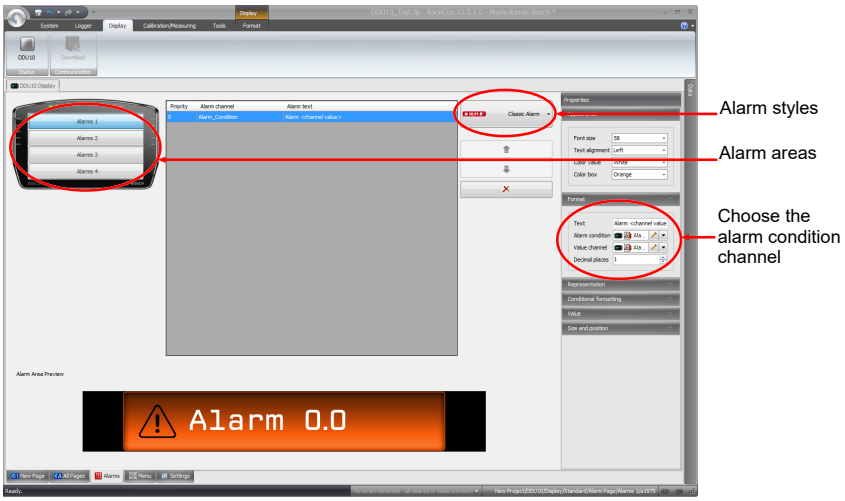
11.7.1 General information on the Alarm Representation

- Since an alarm is generally used for an urgent information, the ‘Alarm’ element has the highest priority and will always be displayed on top of other elements.
- The ‘Alarm’ elements can be defined in four separate areas. In each area, only one alarm message can be visible at the same time. If more than one alarm message is active in one area, each one will be visible for 1.5 seconds.
- The size and position within the alarm area is freely configurable.
- The ‘Alarm’ elements match the design styles of the general display elements and are available in the design styles “Classic” and “Modern”. For better readability, both styles have a non-transparent background.
- An ‘Alarm’ element can display either a text, icon or a value, or any combination of those separately.

11.7.2 Alarm configuration

To create an alarm, perform the following steps:

1. Click on the ‘Display’ tab at the top.
2. Click on the ‘Alarm’ tab at the bottom. The alarm configuration window opens.



3. Choose one of the four alarm areas at the top left.

   This defines the position on the screen, where the alarm will be displayed.

4. Select the design of the ‘Alarm’ element from the drop down list on the top right.

   You can choose between the “Classic” and the “Modern” style. By clicking on the alarm style, an alarm will be added to the alarm list in the middle of the screen.

5. Select the “Alarm condition” channel, which will trigger the alarm when the channel becomes true.

   The “Alarm condition” channel can be any Boolean channel, like any other conditional channel.

6. If desired, select a “Value Channel”.

The value can be displayed or used for a conditional formatting.

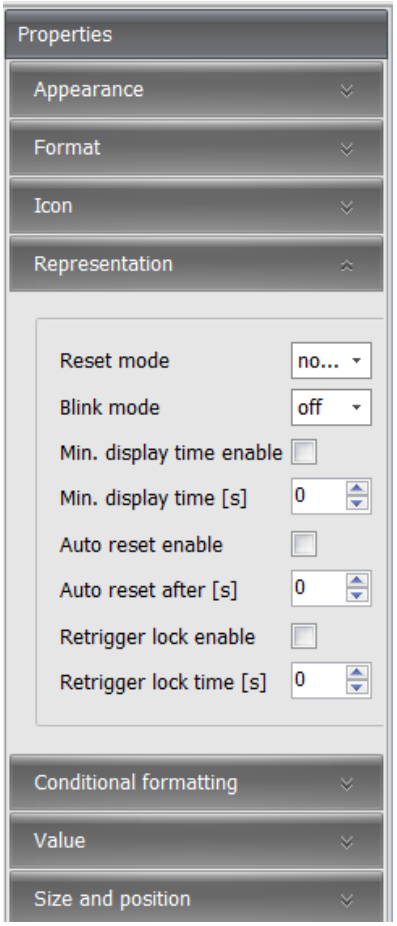
7. Define the text, which will be displayed in the alarm icon.

   If you use the phrase <*channel value*>in the text, it will be substituted with the value of the chosen channel value.

8. Check the size, position and appearance of the alarm in the “Alarm Area Preview”. Change the appearance according to your needs with the properties described below.

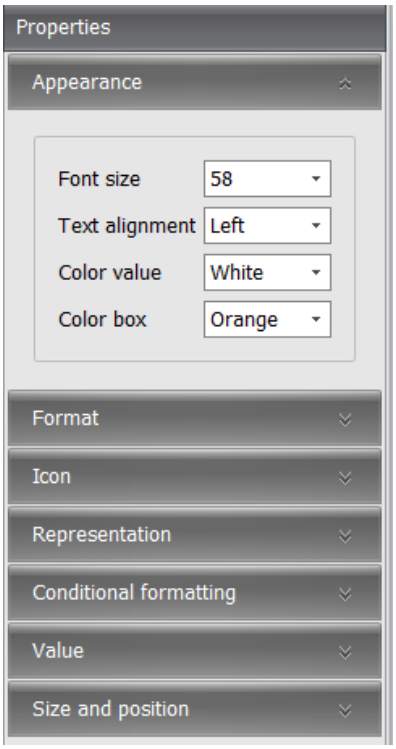
11.7.3 Further Alarm Properties

Representation



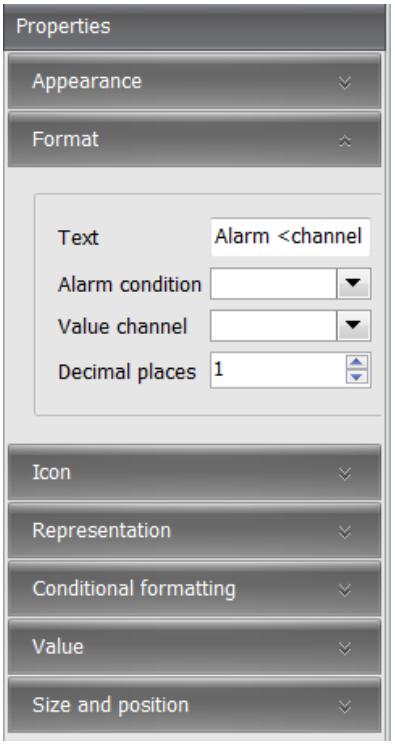
- Reset Mode: Defines if the alarm is resettable. For more details, see chapter Resettable Alarm [▶ 99].
- Blink mode: Defines the blinking frequency of the alarm when displayed.
- Min. display time enable: Enables the function min. display time.
- Min. display time: Defines the minimum time an alarm will be displayed.
- Auto reset enable: Enables the function “Auto reset”.
- Auto reset after: Defines the time after an alarm will be hidden, even when the condition is still true, only active for resettable alarms.
- Retrigger lock enable: Enables the function “Retrigger lock”. For more details, see chapter Resettable Alarm [▶ 99].
- Retrigger lock time: Defines the time in which a new rising edge of the alarm condition will not be accepted anymore, after an alarm was reset.

Appearance



- Font size: Defines the size of the font.
- Text alignment: Defines the alignment of the text.
- Color value: Defines the color of the value inside the alarm box.
- Color box: Defines the color of the alarm box.

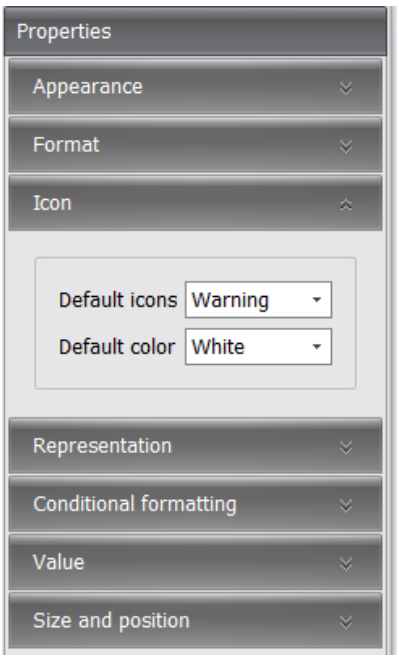
Format



- Text: Defines the Text and/or the value of the alarm.

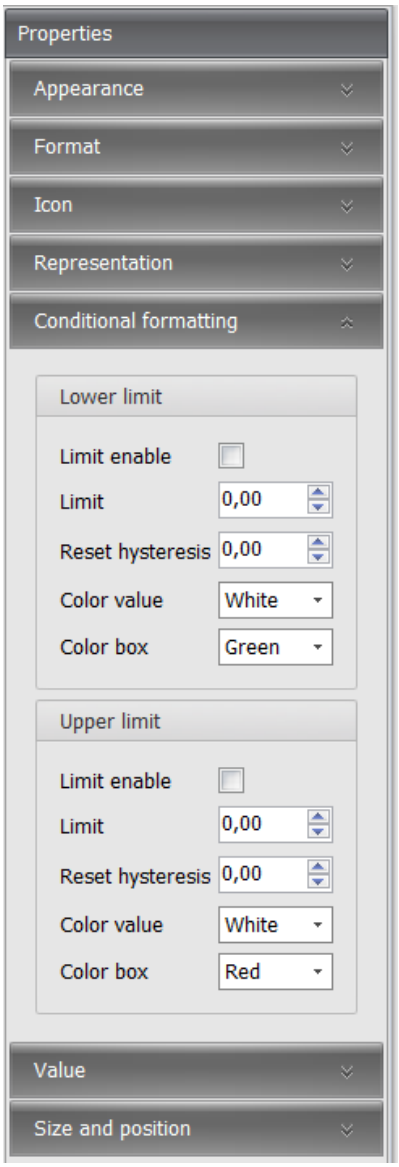
- Alarm condition: Defines the alarm condition.
- Value channel: Defines the value channel, which may be displayed and is relevant for the conditional formatting.
- Decimal places: Defines the decimal places, which should be displayed.

Icon



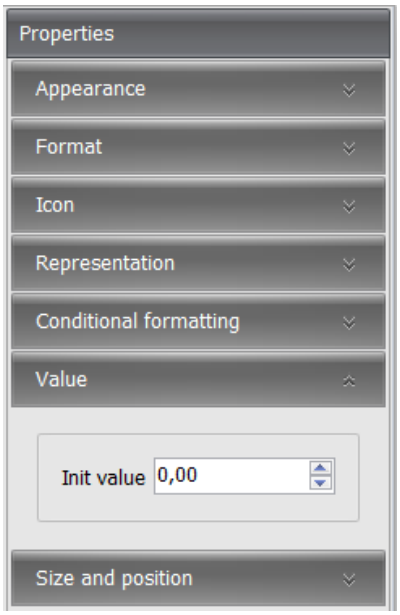
- Default icons: Defines the alarm icon, which should be displayed.
- Default color: Defines the color of the icon.

Conditional formatting (lower and upper limit respectively)



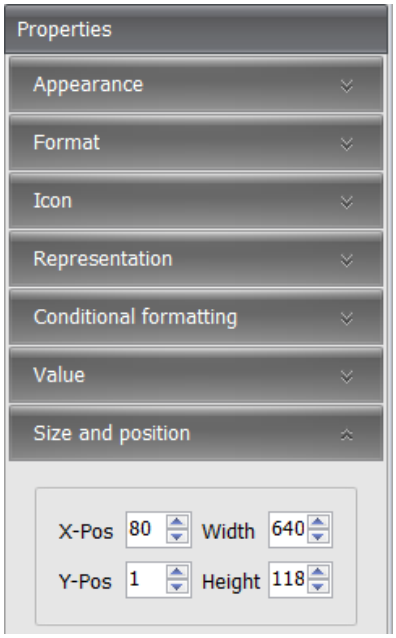
- Limit enable: Enables a lower limit and an upper limit respectively.
- Limit: Defines the threshold limit.
- Reset hysteresis: Defines the value as off when the hysteresis should be reset. You can find more information about the hysteresis function in general in the chapter Hysteresis [▶ 42].
- Color value: Defines the color when the value is above/below the set threshold limit.
- Color box: Defines the new color of the box when the value is above/below the set threshold limit.

Value



– Init value: Defines the value, which will be shown in RaceCon during configuration or on the device if the channel has no link. Allows testing the conditional formatting by entering a representative value.

Size and position



– X-Pos: Defines the x-position of the top left corner of the alarm message within the alarm area.

– Y-Pos: Defines the y-position of the top left corner of the alarm message within the alarm area.

– Width: Defines the width of the alarm message.

– Height: Defines the height of the alarm message.

11.7.4 Edit/Delete/Change priority of an Alarm

Perform the following steps:

1. Click on the ‘Display’ tab and then on the ‘Alarm’ tab to go to the alarm configuration window.
2. Select the alarm area of the desired alarm.

   The center of the alarm configuration window only shows the alarms of the selected alarm area.
3. Click on the alarm, which needs to be edited or deleted.
4. To delete the alarm, click on the red cross.
5. Change the properties of the alarm in the properties window on the right.
6. To change the priority of the alarm, click the up/down arrow to rearrange the priority list.

   If the option “Show alarms” is ticked, the alarm with the highest priority will be visible on the display page configurator. The visibility of the alarms will not be influenced by the priority; the alarms will always rotate, if several conditions become true.

11.7.5 Resettable Alarm

An alarm can be configured as a resettable alarm. This means that the alarm can be linked to a second conditional channel, which will hide the alarm, when the condition becomes true.

To configure and use a resettable alarm, perform the following steps:

1. Click on the ‘Display’ tab and then on the ‘Settings’ tab to go to the “DDU10 display settings”
2. Select your “Alarm reset channel”, which can be any boolean channel, like any other conditional channel.
3. Click on the ‘Alarms’ tab, to change to the alarm configuration window.
4. Create an alarm or edit an existing alarm as described in the previous chapters, to access the properties “Representation”.
5. Change the “Reset mode” to resettable. The alarm will be displayed with a little cross at the top right corner.

Important information on resettable channels:

- Only visible alarms can be reset. If several alarms are active, only the alarm which is visible during the reset condition becoming true, will react to the reset.
- A reset alarm will not be shown again, if the alarm condition becomes true again, until the next power cycle. Exceptions are, a change of the display page or if the retrigger lock is enabled and the retrigger lock time is over.

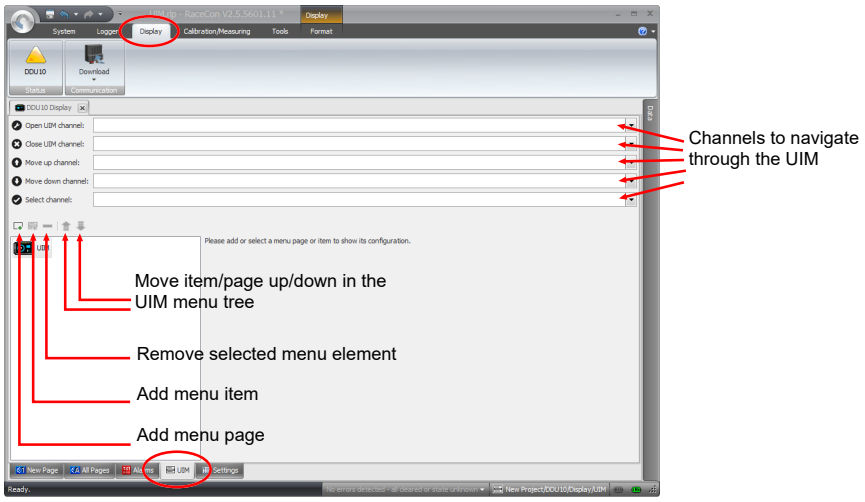
11.8 User Interface Menu UIM

You can use the User Interface Menu (UIM) to conduct minor setting changes on the DDU, like lap time or fuel reset.

How to configure the UIM

Perform the following steps, to set up an UIM on your device:

1. Click on the ‘Display’ tab on the top or double-click on “Display” in the project tree.
2. Click on the ‘UIM’ tab at the bottom. This will open the UIM configurator.



3. Define the channels at the top, to navigate through the UIM. The selected channels can be any Boolean channels, such as conditional channels. You can also use condition combinations to open the UIM, to avoid an unintentional start of the UIM.

NOTICE

When the close condition is true, you cannot open the UIM. This function can be used to avoid an unwanted UIM while driving, i.e with a condition RPM>300

4. Click “Add menu page” to create a new link to a new page on the selected page.
5. Click “Add menu item” to create a new item on the selected page.

   On a new page, you can create new items or pages as needed.
6. Define a name for your items and pages and define the settings for your items.

   With a new item, you can define either labels, special functions or menu triggered events.

Labels

– With a label you can show any channel of the current project or just a text.

Special Functions

– Page set variation

  This menu entry enables you to switch between all available page sets. Activating a different page set automatically reboots the device.

– Outing counter increment

  Increments the outing counter by 1.

– Lap counter increment

  Increments the lap counter by 1.

– Lap distance reset

  Resets the traveled lap distance to zero.

– Lap counter reset

Resets the lap counter to the reset value configured in the Laptrigger options.

– Outing counter reset

  Resets the outing counter to the reset value configured in the Laptrigger options.

– Best laptime reset

  Resets the best laptime.

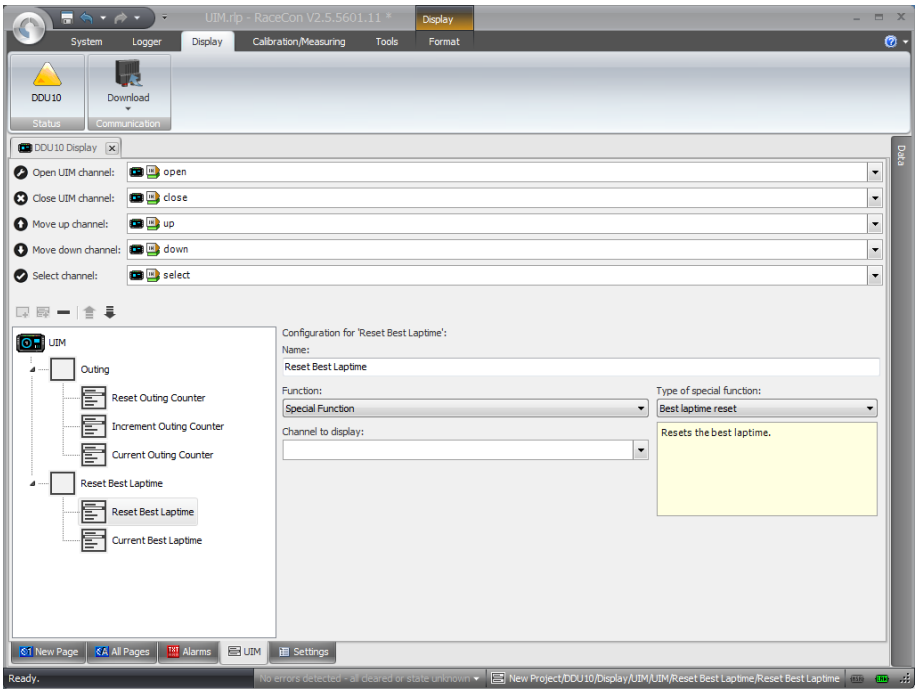
– Segment times reset

  Resets the best laptime segment times.

Menu Triggered Events

– With a menu triggered event you can define a new virtual Boolean channel, which will jump to the condition true for 100 ms, when selected through the UIM. You can use such a channel for sensor calibration or any further custom calculation.

  A UIM configuration to change the current outing counter and to reset the best lap time could look like this:



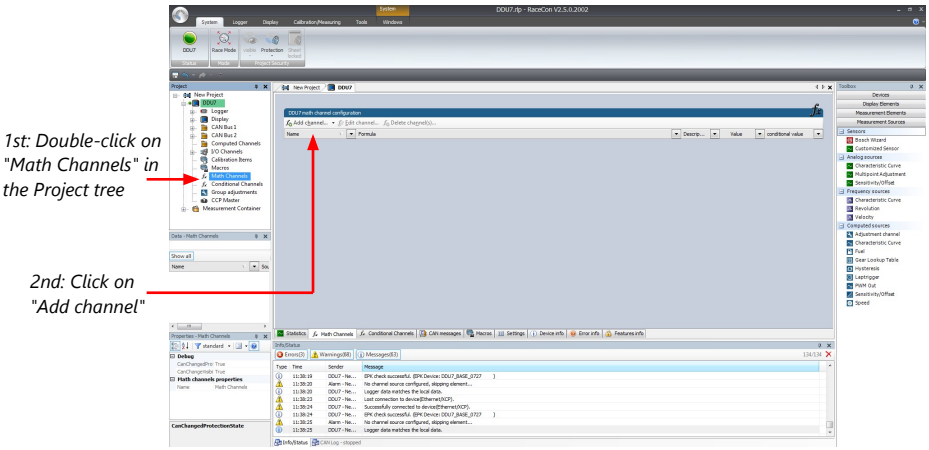
12 Math Channels and Conditional Functions

Math Channel

- Arithmetic and logical operations on up to 4 measurement channel(s)
- Numerical result
- Result can be used as input source for various display elements (numeric elements, alarms, bargraphs) and further calculations in the whole RaceCon project

Creating a new math channel

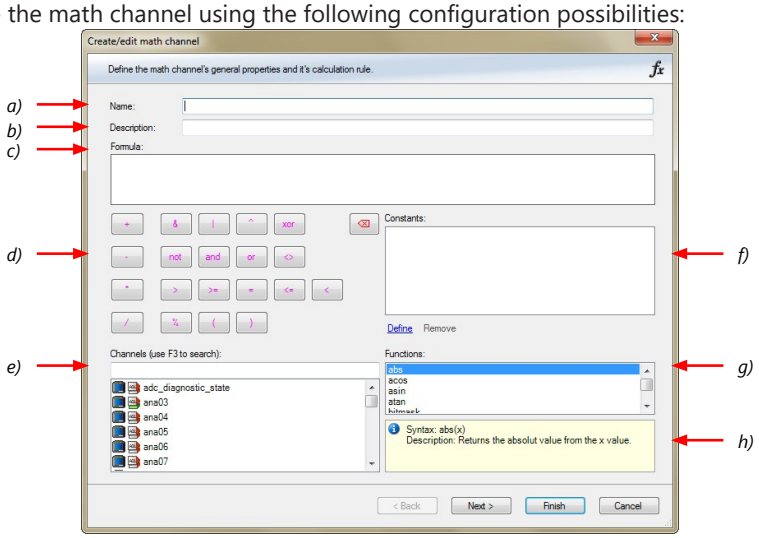
Follow the steps shown in the screenshot.



The “Create/edit math channel” window appears.

“Create/edit math channel” window

1. Define the math channel using the following configuration possibilities:



a) Enter the name of the math channel.
b) Enter a description of the math channel.
c) Enter the formula.
d) Select the logical operator.
e) Choose a measurement channel.
f) Define a value that can be used as a constant in the formula.
g) Choose a function.
h) Describes the function selected above.

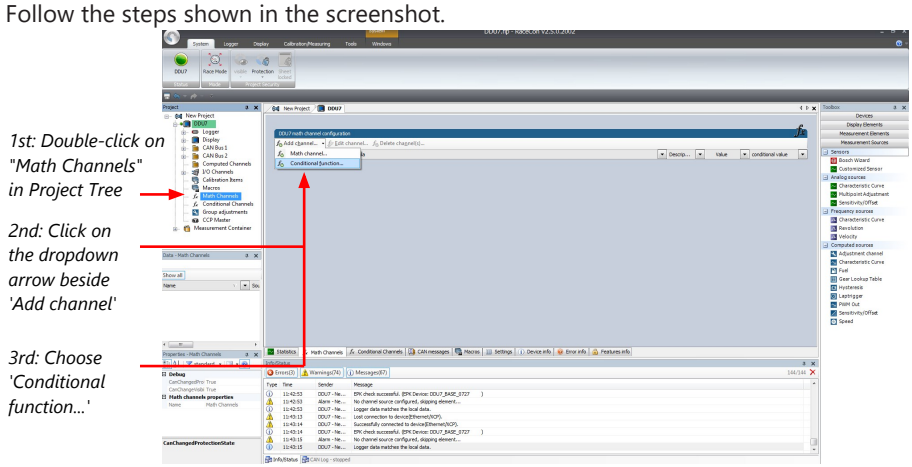
2. Click ‘Finish’ when done.
   The math channel is displayed in the DDU 10 math channel window.

Conditional Function

- Arithmetic and logical operations on one or more measurement channel(s)
- If-Else structure with reset
- Numerical result
- Result can be used as input source for various display elements (numeric elements, alarms, bargraphs) and further calculations in the whole RaceCon project.
  All math and conditional channels can be used globally in the whole RaceCon project.

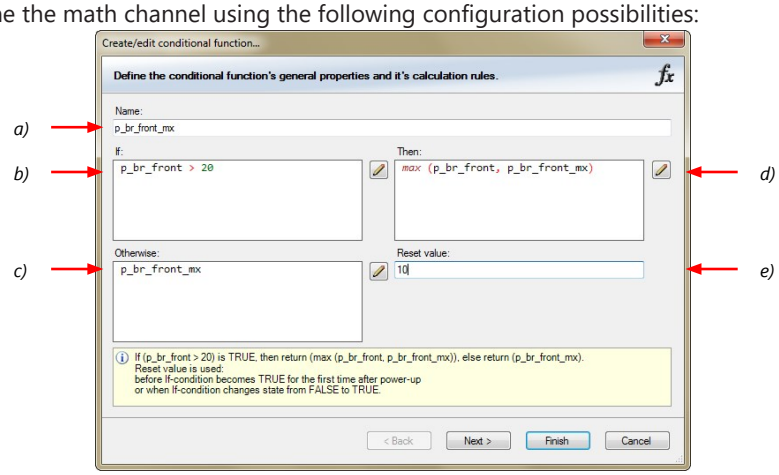
Creating a new Conditional Function

1. Follow the steps shown in the screenshot.



   The “create/edit math channel” window appears.

2. Define the math channel using the following configuration possibilities:



a) Enter the name of the conditional function.
b) Enter the If-condition. Click on the pencil symbol to open an editor to enter expressions.
c) Enter the Then-condition. Click on the pencil symbol to open an editor to enter expressions.
d) Enter the Otherwise-condition.Click on the pencil symbol to open an editor to enter expressions.
e) Enter the reset value (must be a number).

3. Click ‘Finish’ when done.

The conditional function works the following way:

The program always calculates the condition entered in the IF window and checks if the condition is TRUE or FALSE.

If the condition entered in the IF window is TRUE, the program calculates the condition entered in the THEN window. The returned value is the content of the new variable (entered in “Name”).

If the condition entered in the IF window is FALSE, the program calculates the condition entered in the OTHERWISE window. The returned value is the content of the new variable (entered in “Name”).

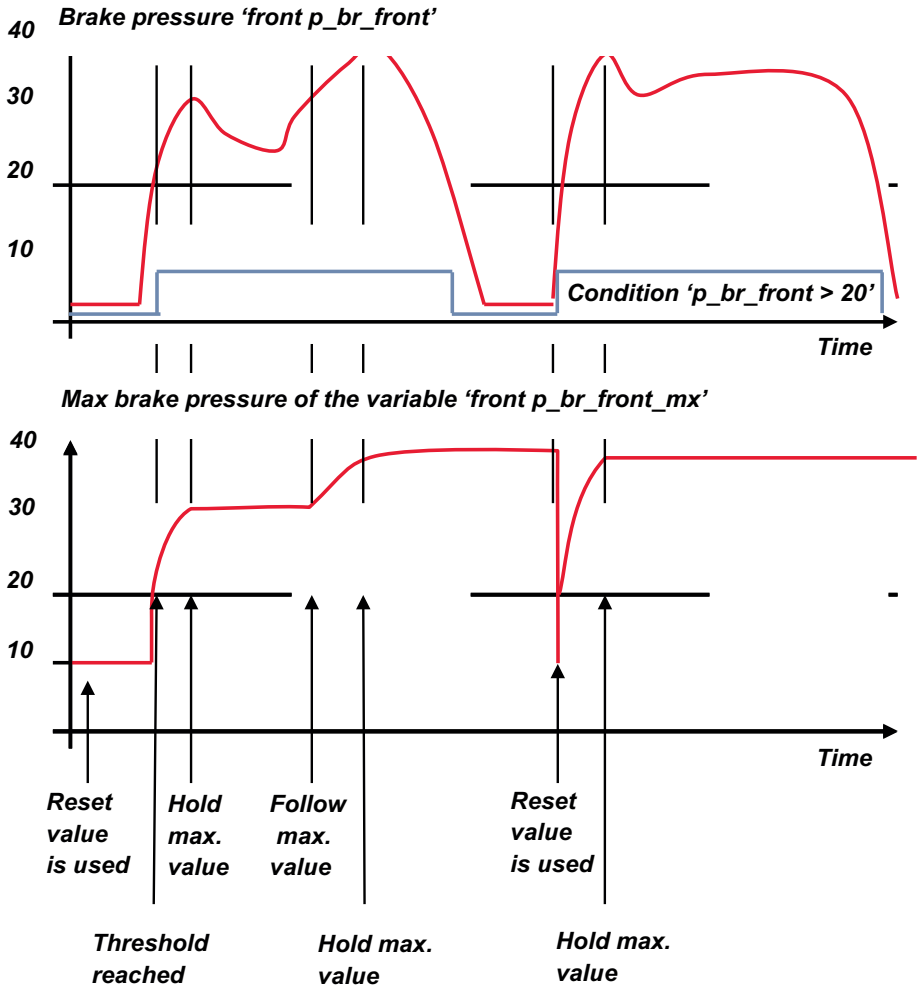
The reset value is always set for the new variable (entered in “Name”):

- before the If-condition becomes TRUE for the first time after power-up
- when the If-condition changes state from FALSE to TRUE.

An example of a condition to set up the maximum front brake pressure is given on the next page.

The conditional function is displayed in the DDU 10 math channel window.

Example: Setting up a condition for maximum front brake pressure



- At power-up, the reset value (10) is used for ‘p_br_front_mx’.
- ‘p_br_front’ rises to 30. As ‘p_br_front’ is > 20 (condition is TRUE), the condition ‘max (p_br_front, p_br_front_mx)’ in the THEN window is triggered. The condition sets the bigger value as new value for ‘p_br_front_mx’. As ‘p_br_front’ (30) is bigger than ‘p_br_front_mx’ (10), the new value for ‘p_br_front_mx’ is set to 30.

- Although ‘p_br_front‘ falls to 25, the value of ‘p_br_front_mx‘ stays 30. This is caused by the THEN-condition, because p_br_front_mx‘ (30) is still bigger than p_br_front‘ (25).
- ‘p_br_front‘ rises to 40. As ‘p_br_front‘ (40) is bigger than ‘p_br_front_mx‘ (30), the new value for ‘p_br_front_mx‘ is set to 40.
- As ‘p_br_front‘ falls below 20, the IF-condition turns to FALSE. Now the OTHERWISE-condition is triggered. Because the condition ‘p_br_front_mx‘ sets the value of ‘p_br_front_mx‘ and the value is already set to 40, nothing changes.
- When ‘p_br_front‘ rises to 40, the IF-condition changes to TRUE again and triggers the THEN-condition. Now the reset value (10) is used for ‘p_br_front_mx‘ in the THEN-condition.
- The new value of ‘p_br_front_mx‘ is 40 because 40 is bigger than 10.

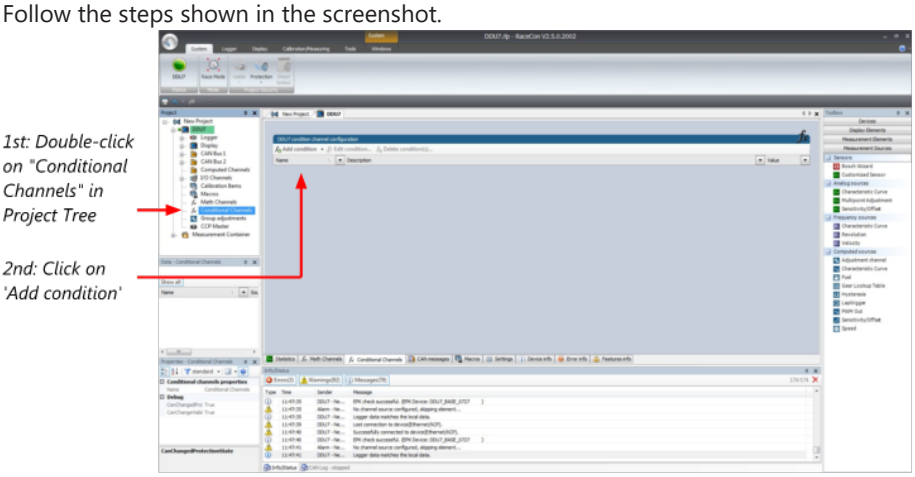
13 Conditional Channels and Condition Combination

Conditional Channel

- Logical operations on measurement channel(s)
- If-Else structure with reset
- Logical result
- Result can be used as input source for alarm display elements and further calculations in the whole RaceCon project.

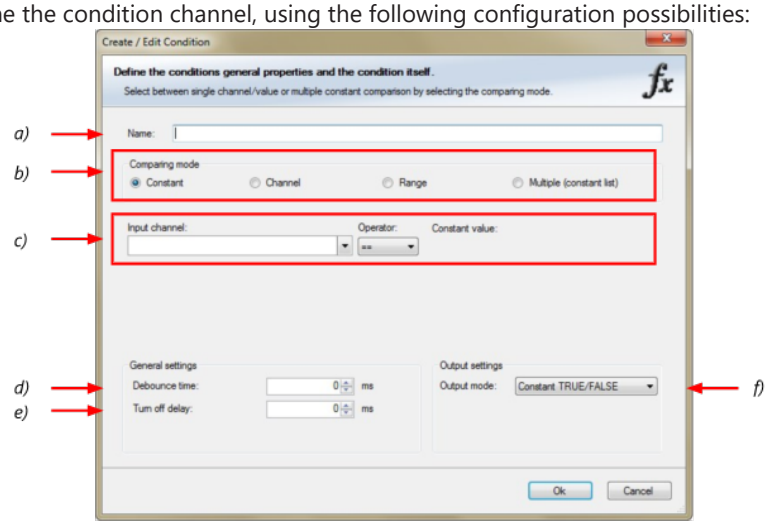
Creating a new Conditional Channel

1. Follow the steps shown in the screenshot.



The “Create/edit condition” window appears.

2. Define the condition channel, using the following configuration possibilities:



a) Enter the name of the conditional channel.
b) Select the comparing mode:
- Constant: Compare a measurement channel with a constant value.
- Channel: Compare a measurement channel with a measurement channel.
- Range: Compare a measurement channel with a defined value range.
- Multiple: Compare a measurement channel with up to 5 constant values.
c) Depending on the chosen comparing mode, you can enter the following values:
- Constant: Choose the measurement channel or condition, the operator and enter the value of the channel.
- Channel: Choose the measurement channel or condition, the operator and the measurement channel or condition to be compared.
- Range: Choose the measurement channel or condition, the operator and define the minium and maximum value.
- Multiple: Choose the measurement channel or condition, the operator and enter the value of up to 5 constants.
d) Enter the minimal time to detect the signal of the measurement channel, to avoid high-frequent switchovers.
e) Enter the time the signal of the measurement channel is delayed after its ending.
f) Choose the output setting of the result.
- Constant TRUE/FALSE: Result is as a constant with the value TRUE or FALSE.
- Blinking: Result is a blinking, if the condition is fulfilled.
- Pulse: Result is a short one-time pulse, if the condition is fulfilled.
- Toggling output: Result is a pulse that lasts until the next condition is fulfilled.

3. Click 'Ok' when done. The conditional channel is displayed in the DDU 10 condition channel window.

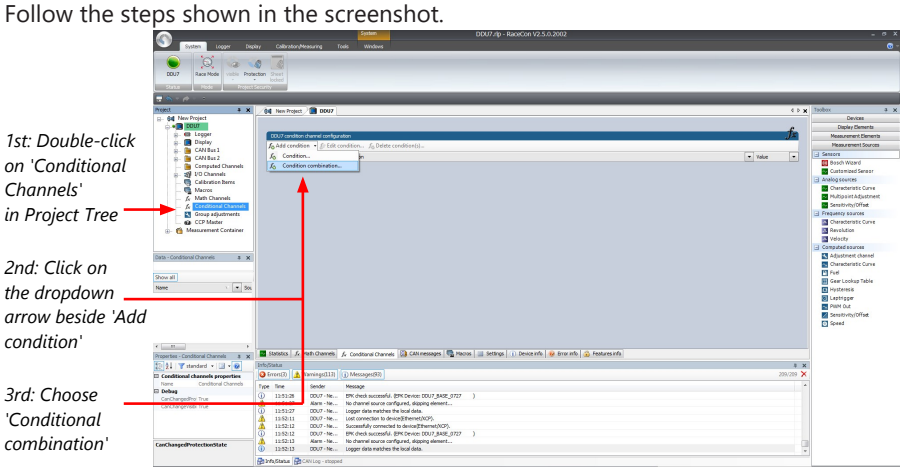
Condition Combination

- Combination of several (up to 16) conditional channels for more complex calculations
- Logical results

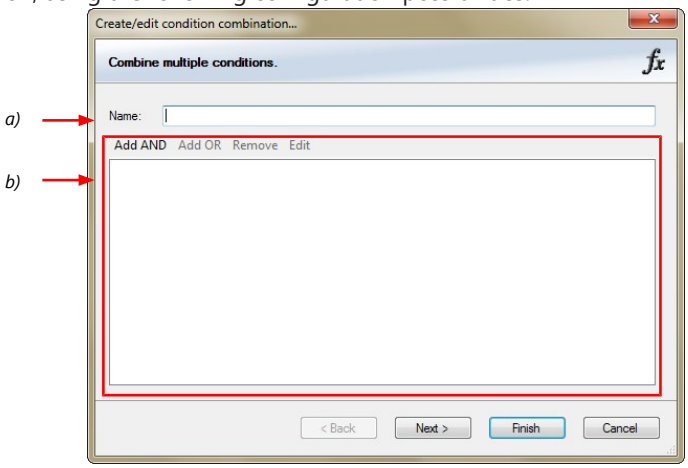
All conditions can be used globally in the whole DDU 10 project.

Creating a new Condition Combination

1. Follow the steps shown in the screenshot.



2. The "Create/edit condition combination" window appears. Define the condition combination, using the following configuration possibilities:



a) Enter the name of the condition combination.
b) Create the condition combination in the window.
- Choose a channel (condition, conditional function, math, measurement channel with binary values) to be compared.
- Combine multiple conditions, by adding 'AND' or 'OR' relations.
- To negate a condition, click with the right mouse-button on the condition and select 'Negation (!)'.
- Combine several (up to 16) conditions.

3. Click 'Next' to go to the next page. Choose the output setting of the result:
   - Constant TRUE/FALSE: Result is as a constant with the value TRUE or FALSE.
   - Blinking: Result is a blinking, if the condition is fulfilled.
   - Pulse: Result is a short one-time pulse, if the condition is fulfilled.
   - Toggling output: Result is a pulse that lasts until the next condition is fulfilled.
4. Click 'Finish' when done. The conditional combination is displayed in the DDU 10 condition channel window.

14 Disposal

Hardware, accessories and packaging should be sorted for recycling in an environment-friendly manner.

Do not dispose of this electronic device in your household waste.

15 CPU Load Limits

As all microprocessors, the two processors of the DDU 10 have limited capacities. The current load of the processors can be monitored using the channel “cpu_load_001” or “cpu_load_002”. When configuring your device, please make sure the used CPU load is in a save range below 100 %.

Bosch recommends a maximum CPU load of 85 % (averaged). Exceeding this limit might result in the DDU 10 not being able to fulfill its required measuring/logging/display tasks or even in crashing and rebooting.

Main factors influencing the CPU load are:

- Number and complexity of math channels
- Number and complexity of conditions
- CAN traffic on both CAN lines
- Display configuration, especially displaying pictures
- Logger configuration (total logging rate [kB/s], conditional measurement rates)

To help respecting the limit of 85 % CPU load, the DDU 10 creates an error memory entry. To trigger this error entry, the CPU load must exceed the limit for 5 minutes without interruption.

When being confronted with this error memory entry (see ‘Error info’ in RaceCon) or when being confronted with DDU 10 resets due to complex configuration setups, please consider reducing the demands on the DDU 10 adapting the influencing factors mentioned above.

16 CAN Bus

The DDU 10 has four fully configurable CAN buses. Two of these are available as an upgrade.

- Baudrate (250 kbaud , 500 kbaud, 1 Mbaud)
- 11 Bit or 29 Bit identifiers
- Input configuration: Read messages from CAN bus and convert to DDU 10 measurement/display variables. CAN bus supports row counter configuration.
- Output configuration: Write RaceCon measurement variables to CAN messages, output frequency and row counter are configurable, CAN gateway functionality (transfer from one bus to another).

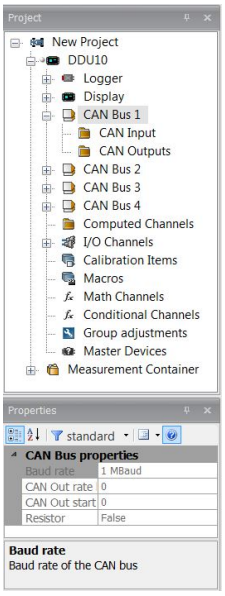
16.1 CAN bus trivia

CAN message

- 11 Bit (standard) or 29 Bit (extended) identifier
- Up to 8 bytes of data payload

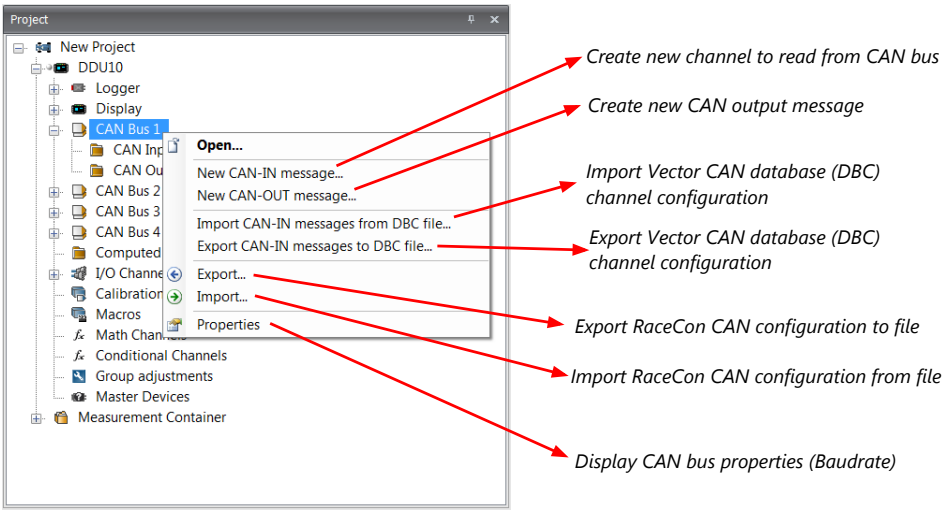
CAN bus

- Needs termination resistors (60 Ohm) in wiring harness
- All devices connected to the bus, must use identical data rate
- Configuration of bus data rate in the 'CAN messages overview' menu. To access the menu, double-click on one of the CAN bus items of the project tree (1 MBaud, 500 kBaud, 250 kBaud).



CAN item drop-down menu

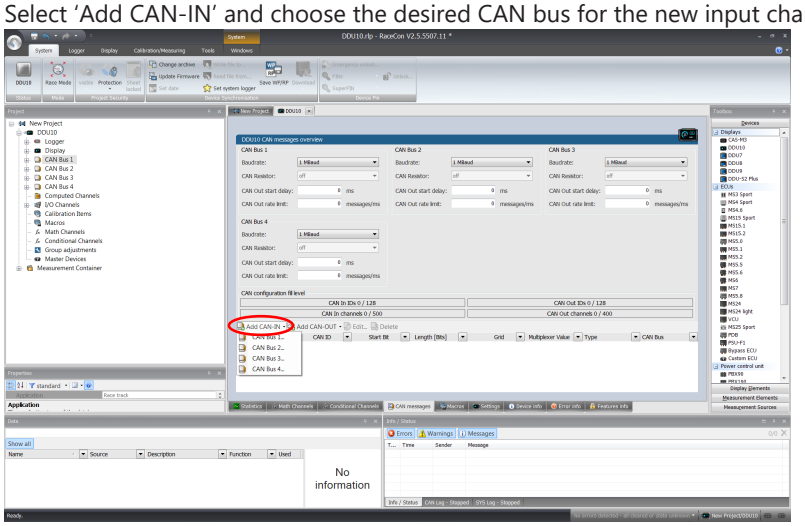
Click with the right mouse button on the desired CAN bus to open the CAN bus drop-down menu.



16.2 CAN input

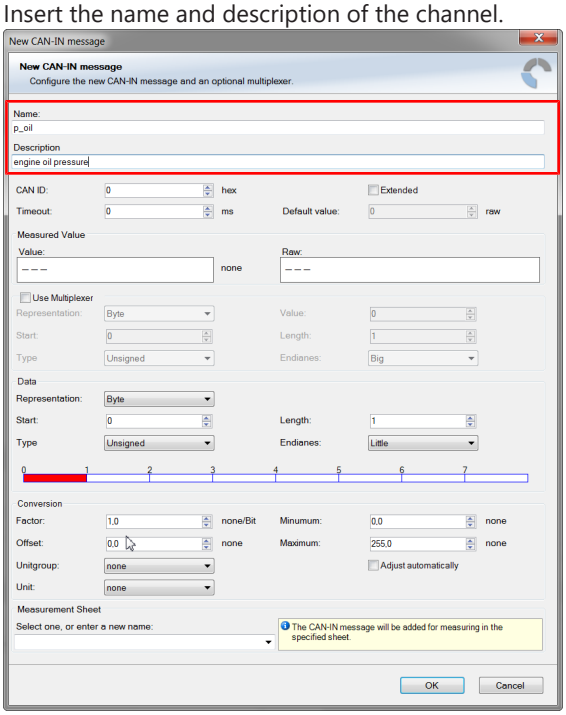
16.2.1 Create new CAN Input channel

1. Double-click on any CAN bus item, to open the "CAN messages overview".
2. Select 'Add CAN-IN' and choose the desired CAN bus for the new input channel.

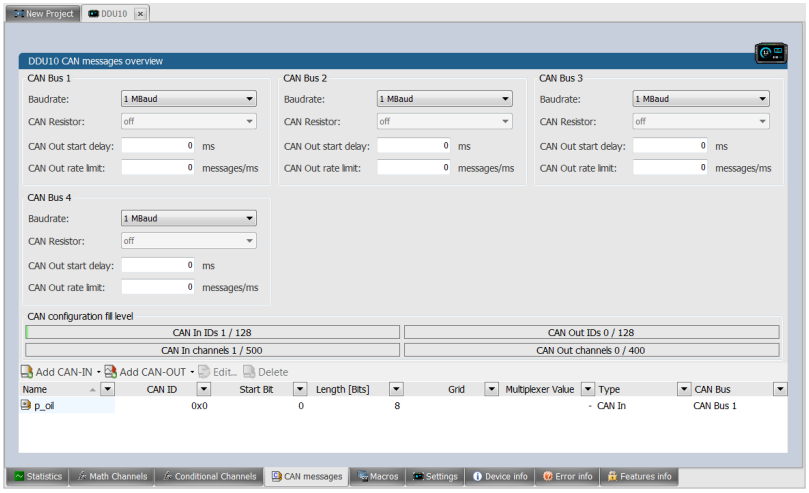


A CAN channel configuration window opens.

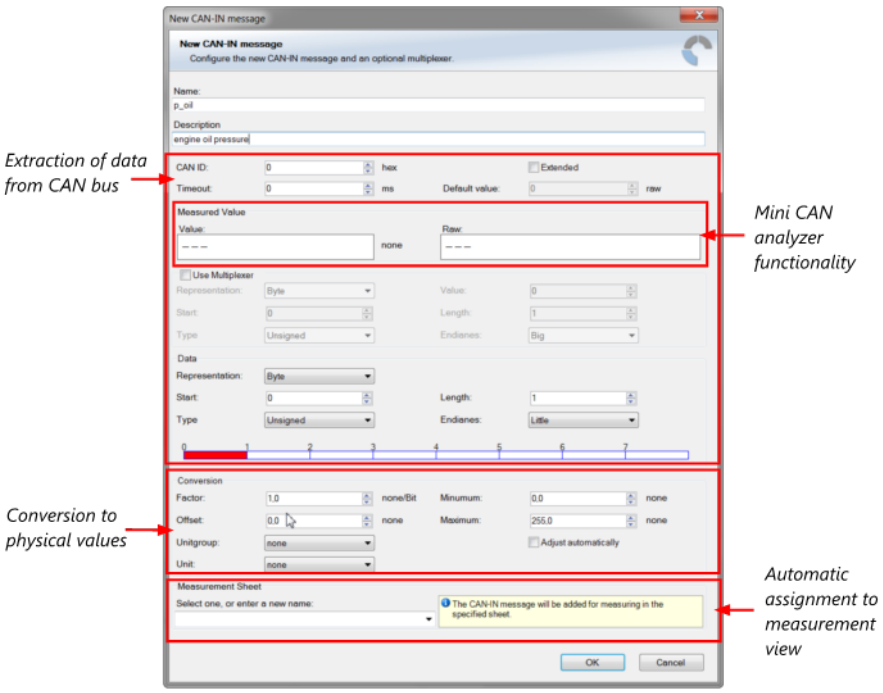
3. Insert the name and description of the channel.



4. Click ‘OK’ when done.
   The channel is listed in the Data window.



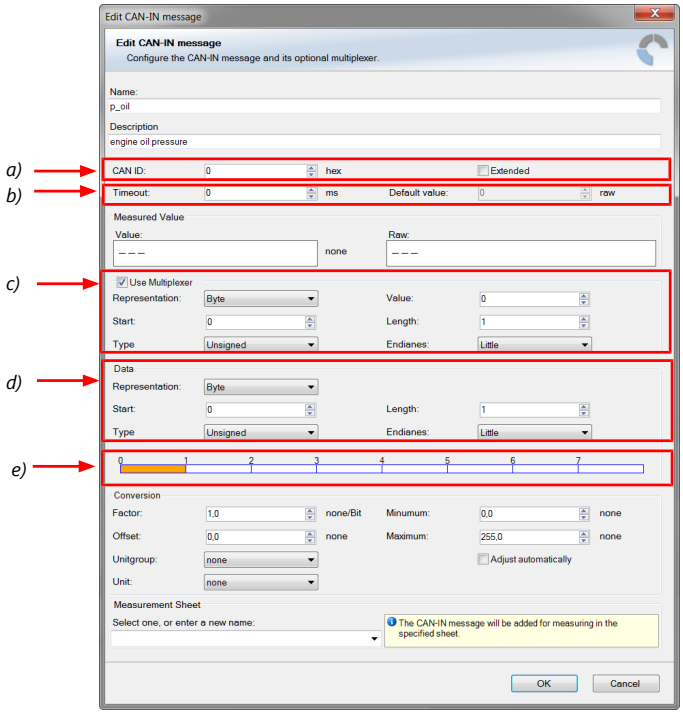
CAN channel configuration



Extracting data from CAN bus

Representation: Byte

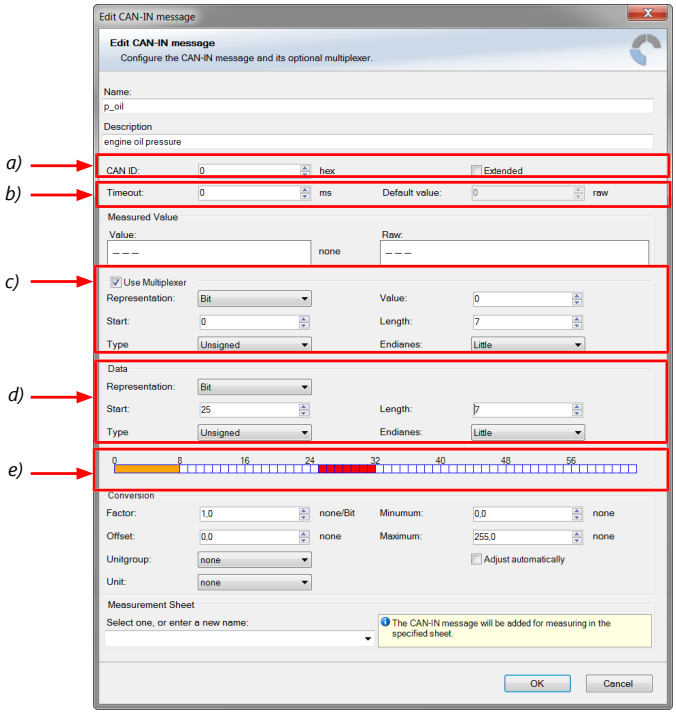
Some CAN devices need to be addressed by a byte represented CAN channel. The address can be assigned in this window and is illustrated by a bar graph.



a) Enter CAN message ID. If extended IDs (29 bit) are used, check the box.
b) If replacement values are used, specify timeout period and raw value.
c) If a multiplexer (row counter) is used, check the box.
d) Enter data position, length and format.
e) The bargraph shows assignment of the bytes.
- Red colored fields show the assignment of the data bytes.
- Orange colored fields show the assignment of the multiplexer bytes.

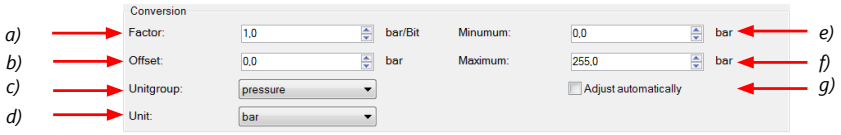
Representation: Bit

Some CAN devices need to be addressed by a bit represented CAN channel. The address can be assigned in this window and is illustrated by a matrix table.



a) Enter CAN message ID. If extended IDs (29 bit) are used, check the box.
b) If replacement values are used, specify time-out period and raw value.
c) If a multiplexer (row counter) is used, check the box.
d) Enter data position, length and format.
e) The bargraph shows the assignment of the bits.
- Red colored fields show the assignment of the data bits.
- Orange colored fields show the assignment of the multiplexer bits.

Conversion to physical values



a) Enter factor (gain) for conversion to physical value.
b) Enter offset for conversion to physical value.
c) Select type of physical value.
d) Select unit of physical value.
e) Enter minimum physical limit of the channel. (for manual setup)
f) Enter maximum physical limit of the channel. (for manual setup)
g) Check the box to automatically adjust the limits of the channel.

Special features

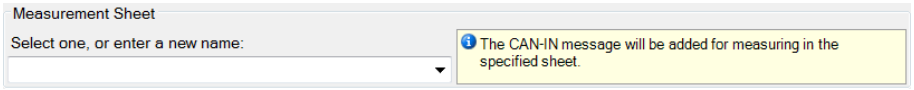
CAN analyzer functionality

This functionality is only available, if a MSA-Box (I or II) is used to connect the DDU 10 to the PC. Choose the CAN bus that is connected to the MSA-Box to display the raw value and the converted physical value here.

Automatic creation of online measurement sheets

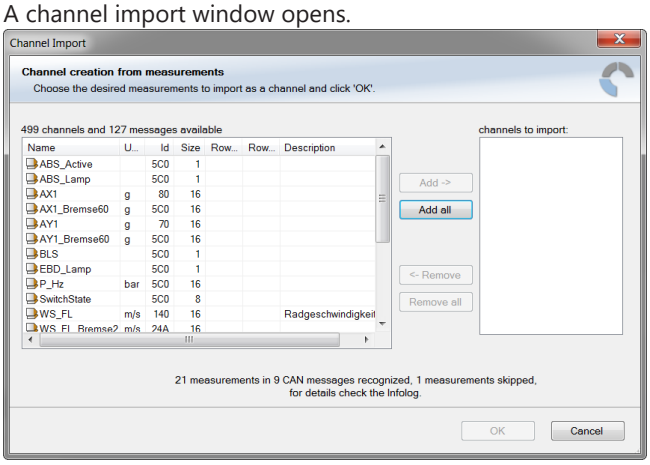
The CAN channel can be automatically inserted into a measurement sheet. Insert a name for a new sheet or select an existing sheet from the list box.

For an online view of the value measured by the DDU 10 , insert the channel in an online measurement sheet which is described in the chapter Setting up an online measurement [▷ 46].



16.2.2 Import a CAN database (DBC) file

1. Click with the right mouse button on any CAN bus item.
2. Select 'Import CAN-IN messages from DBC file...' from menu.
3. A file browser opens.
4. Select the DBC file to import and click 'Open' when done.
5. A channel import window opens.



6. Select the desired channels on the left and use the 'Add' button to add them to the import list.
7. Click 'OK' when done.

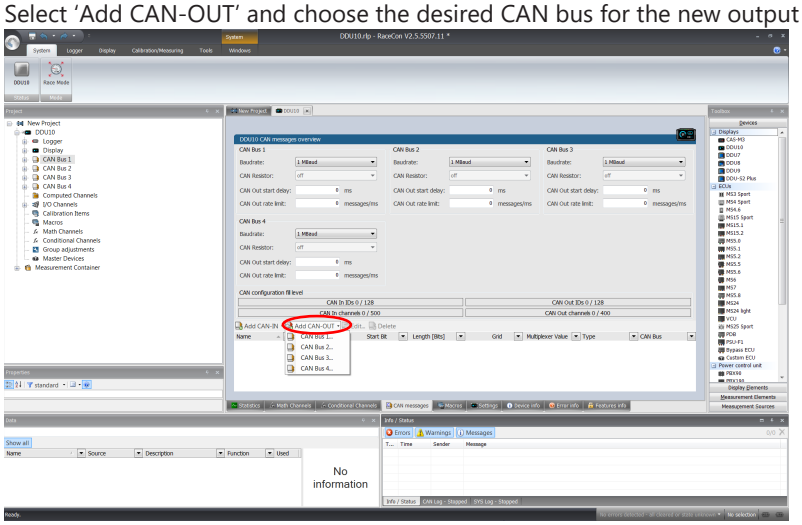
   The channels are inserted in the Data window.

16.3 CAN output

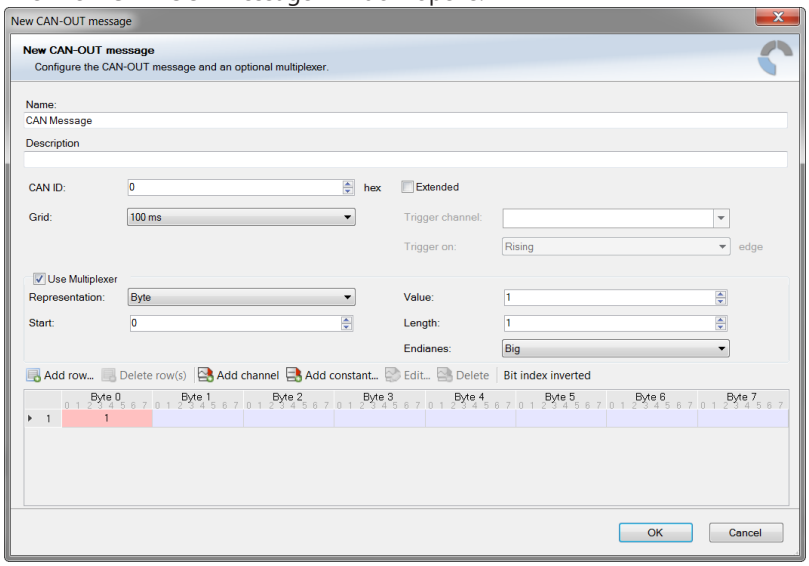
16.3.1 Create a new CAN output message and channel

Double-click on any CAN bus item, to open the "CAN messages overview".

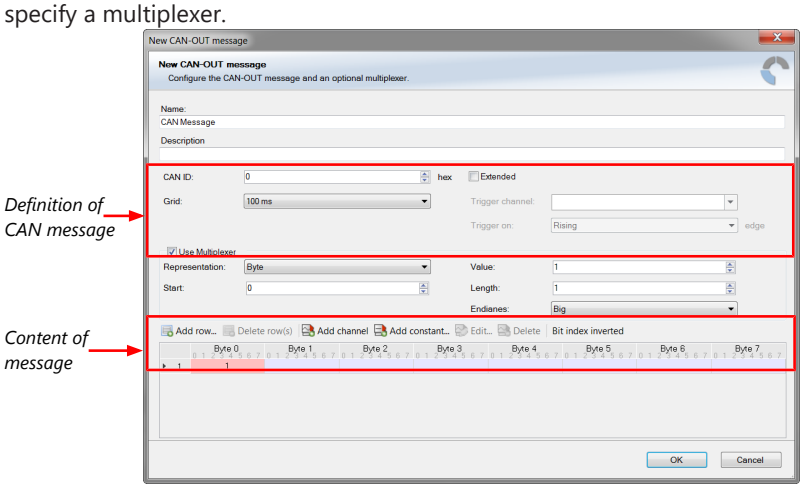
1. Select ‘Add CAN-OUT’ and choose the desired CAN bus for the new output channel.



The ‘New CAN-OUT message’ window opens.

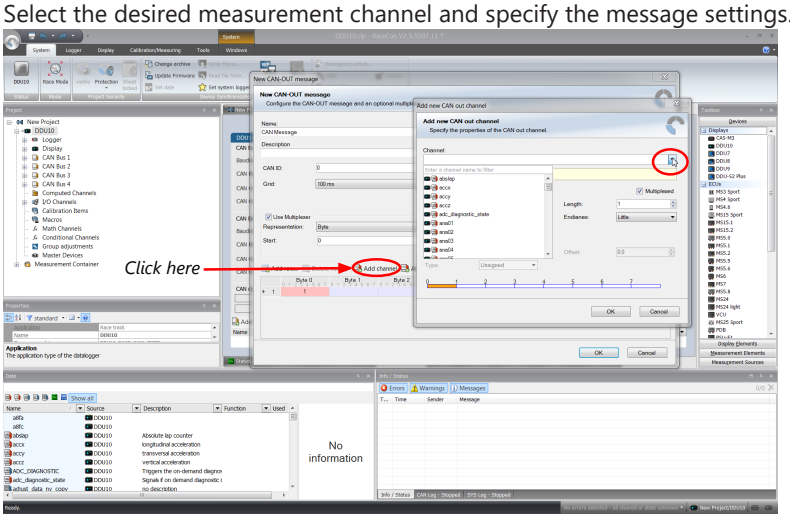


2. Enter name of message, description, CAN-Id and Grid (output interval). Optionally, specify a multiplexer.



3. Click on ‘Add channel’ or ‘Add constant’, this opens the ‘Add new CAN out channel’ window.

4. Select the desired measurement channel and specify the message settings.

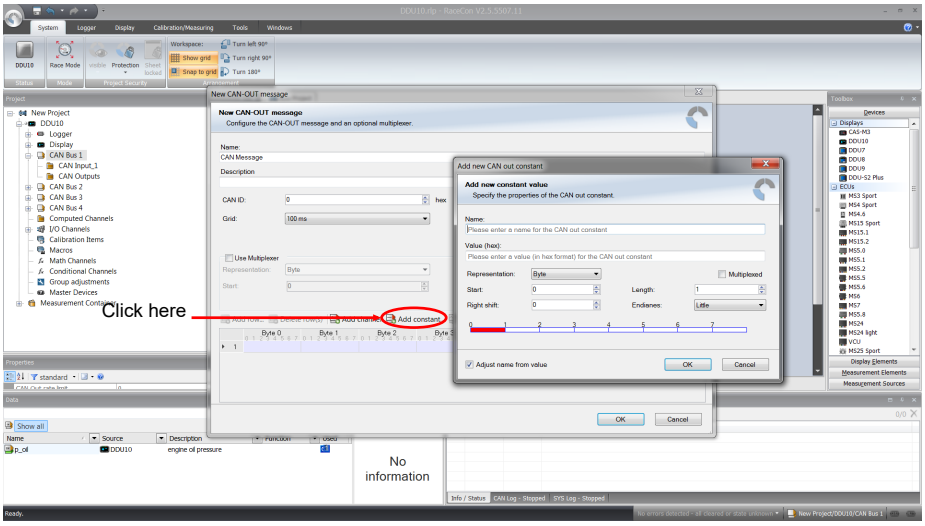


The measurement channel is now assigned to the CAN message.

16.3.2 Add CAN out constant

To send a constant value on the CAN, perform the following steps:

1. Create a new CAN output message or edit an existing message.
2. Click ‘Add constant’. The ‘Add new CAN Out constant’ window appears.
3. Define the name of the constant, the required value in hex and define the CAN channel settings.
4. Click ‘OK’ when done.



16.4 Multiplexer

Row counter concept

If certain channel messages are not time-critical and can be imported or exported slowly, you can use a multiplexer to put several channel messages on one message identifier.

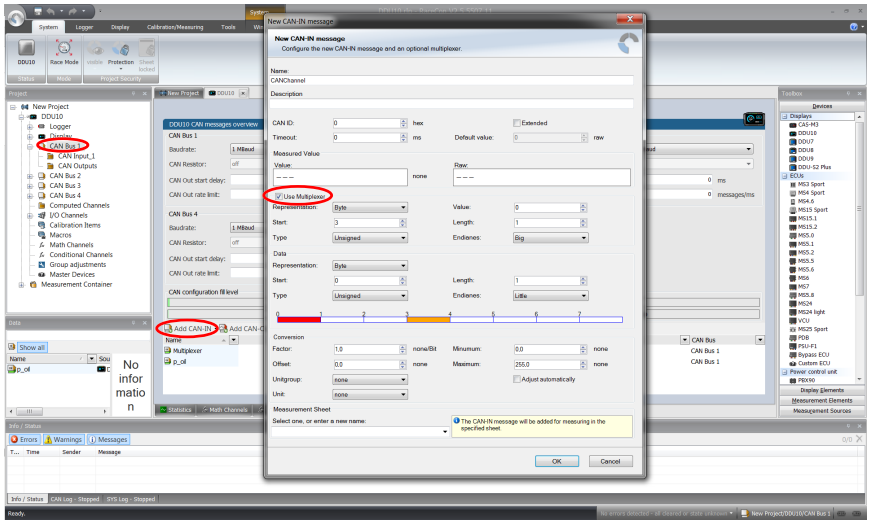
- Re-use (multiplex) of message identifiers by splitting it into several rows.
- Every row is assigned to a unique value of the multiplexer.
- One byte of message contains row counter.

– 7 bytes payload remaining. A multiplexer does not have to consist of one byte only, it can consist of several bytes as well as single bits.

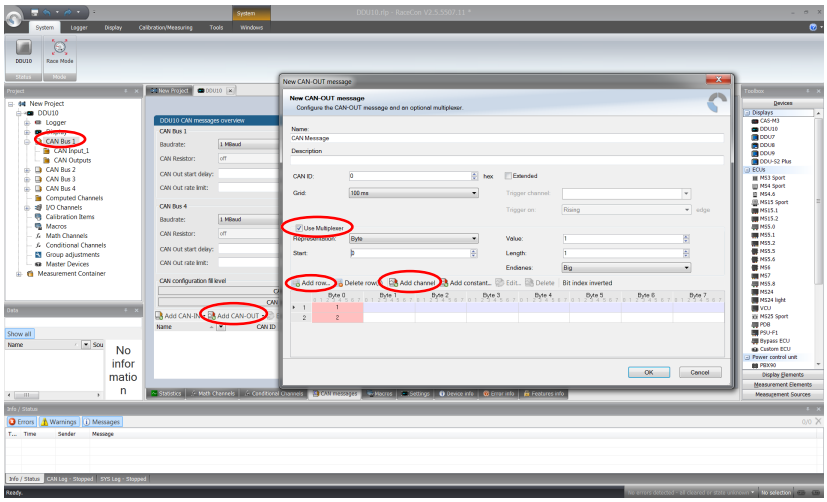
– Position of row counter is configurable.

To use a multiplexer perform the following steps:

1. Double-click on any CAN bus item to open the “CAN messages overview”.
2. Select ‘Add CAN-IN’ and choose the desired CAN bus for the new input channel.
3. Check the box ‘Use Multiplexer’ and configure the multiplexer for the new CAN-IN channel.

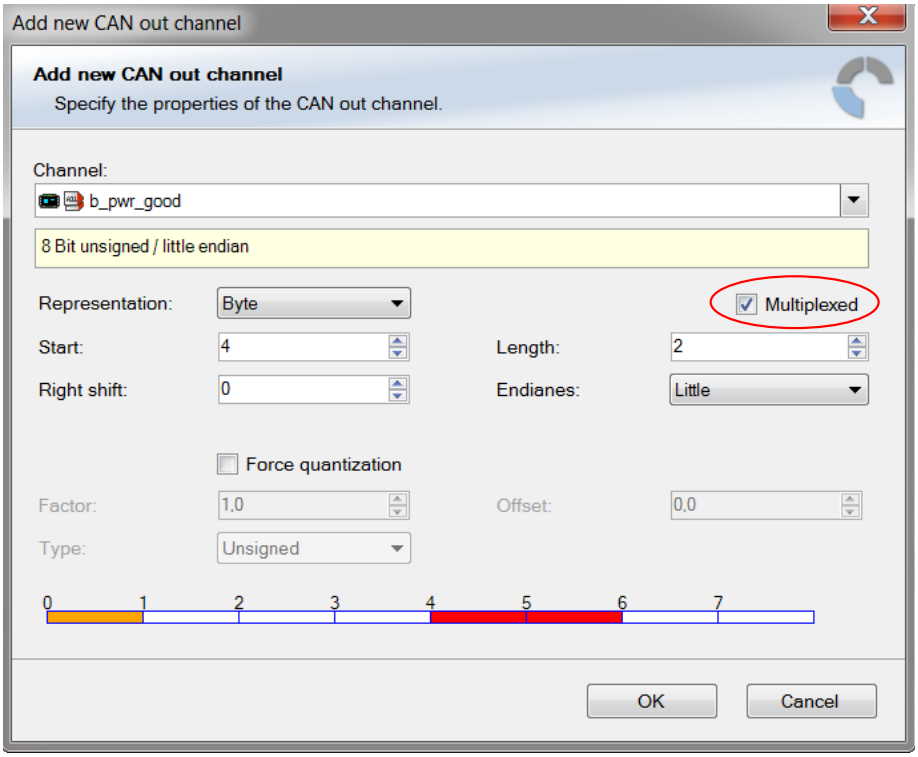


4. To configure the multiplexer for a CAN-OUT channel, select ‘Add CAN-OUT’.
5. Check the box ‘Use Multiplexer’ and click on the button ‘Add row…’ to split the message identifiers into several rows.
6. Click on one row and select ‘Add channel’ to assign a channel to the row.

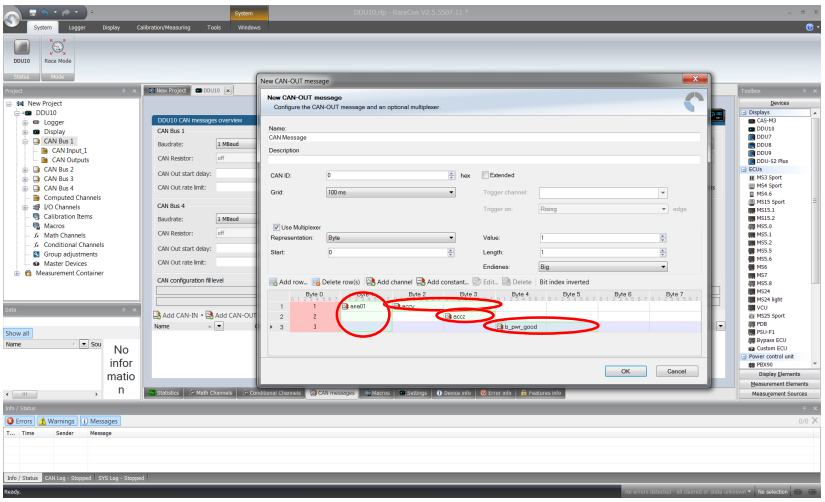


The ‘Add new CAN out channel’ dialog opens.

7. Select a channel and configure it. To assign it to the row selected before, check the box ‘Multiplexed’.
8. To move the channel message, change the “Start” value or click and hold the green field in the “Add new CAN out message” window.
9. Click ‘OK’ when done.



10. The channel message is assigned to the selected fields.
11. Click 'OK' when done.



	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
0x100	0	p_oil		t_oil				
0x100	1	s_dam_fl		s_dam_fr				
0x100	2	s_dam_rl		s_dam_rr				

Message Id ***Row Counter*** ***Payload Area***

17 Recording and Telemetry

Some of the functions described below may require an upgrade. Please refer to the data sheet of your device.

17.1 Features

Recording

- Synchronized recording of DDU 10 analog and digital input channels, DDU 10 internal measurement channels, ECU data, Data from external sensor interfaces
- Up to two independent recordings
- Measurement rate 1 ms to 1 s
- Two global start conditions (thresholds)
- Up to 16 measurement conditions (fast-slow-switches)

Telemetry

- Support for long-range online telemetry
- Individual programmable team code
- Fast block – slow block mechanism
- Programmable data rate

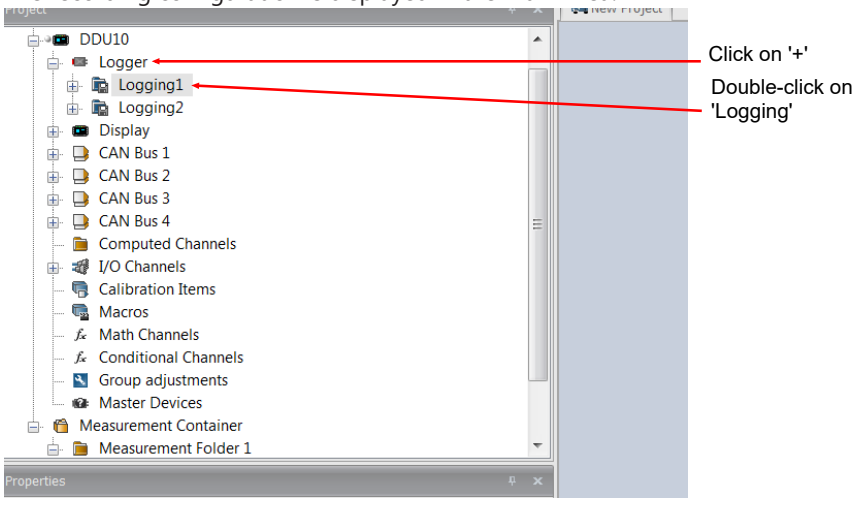
Burst telemetry

- Support for burst telemetry (BT 60)
- Programmable IP configuration
- BT 60 diagnosis via DDU 10

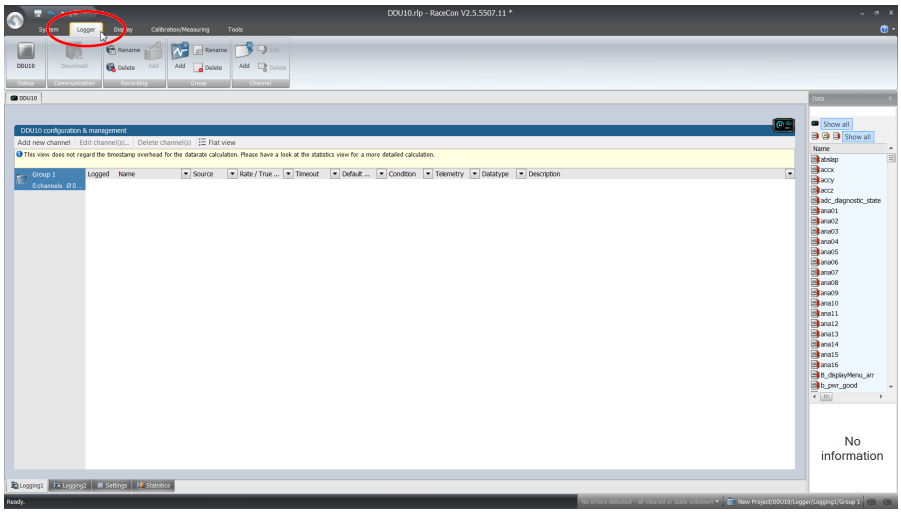
17.2 Configuration of Recordings

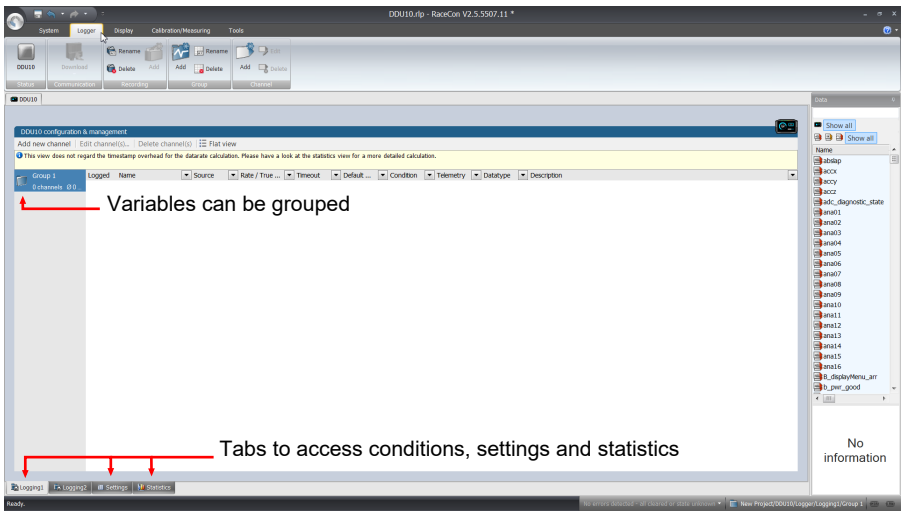
1. Expand the 'Logger' list by clicking on '+' in the DDU 10 Project Tree.
2. Double-click on 'Logging' in the DDU 10 Project Tree.

   The recording configuration is displayed in the Main Area.

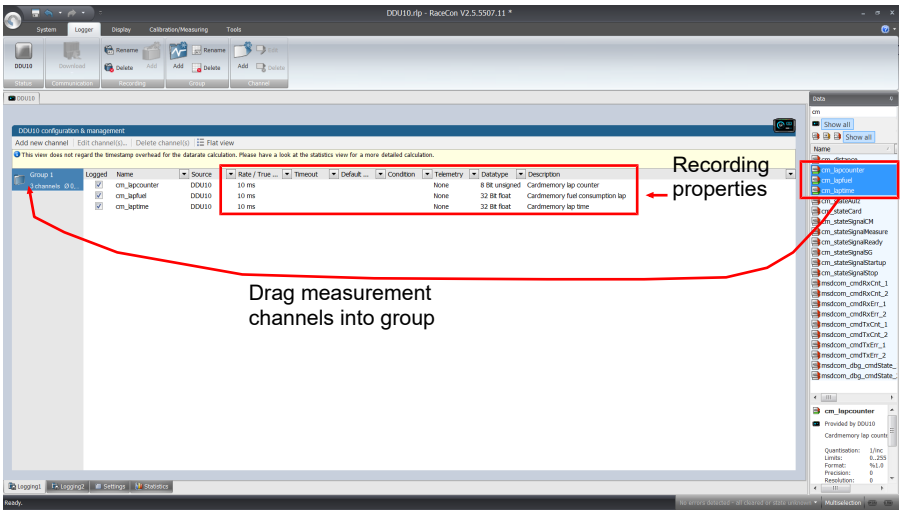


3. Alternatively, click on the 'Logger' tab to open the configuration directly.



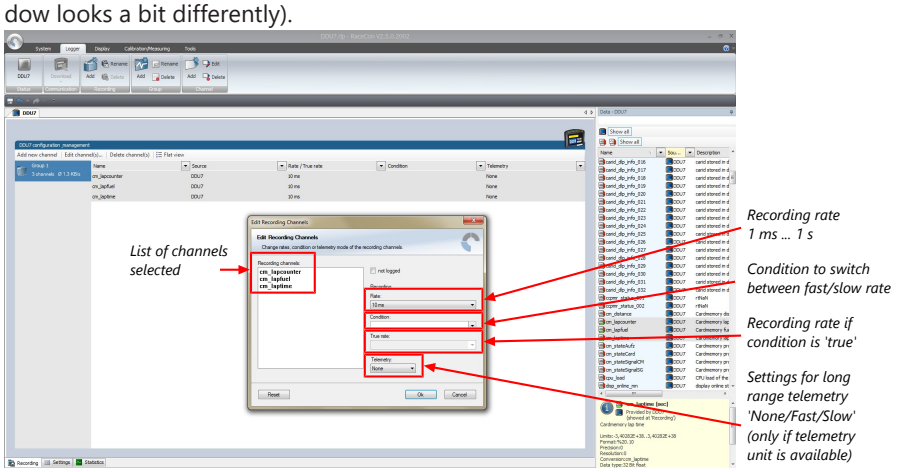


1. To add a measurement channel to a recording, select the wanted channel, drag and drop it onto the measurement group.



1. To edit channel settings, mark the channel(s) and click ‘Edit Recording Channel(s)’.

An ‘Edit Recording Channels’ window opens (if you choose only one channel, the window looks a bit differently).



NOTICE

If no recording condition is defined or the recording condition is ‘false’, measurement channels are recorded at the value chosen in ‘Rate’.

If the condition is ‘true’, measurement channels are recorded at the value chosen in ‘True rate’.

2. Click ‘OK’ when done.

17.2.1 Adding a recording group

Recording channels can be grouped. These groups will also be visible in Darab and will help to get a better overview during the data analysis

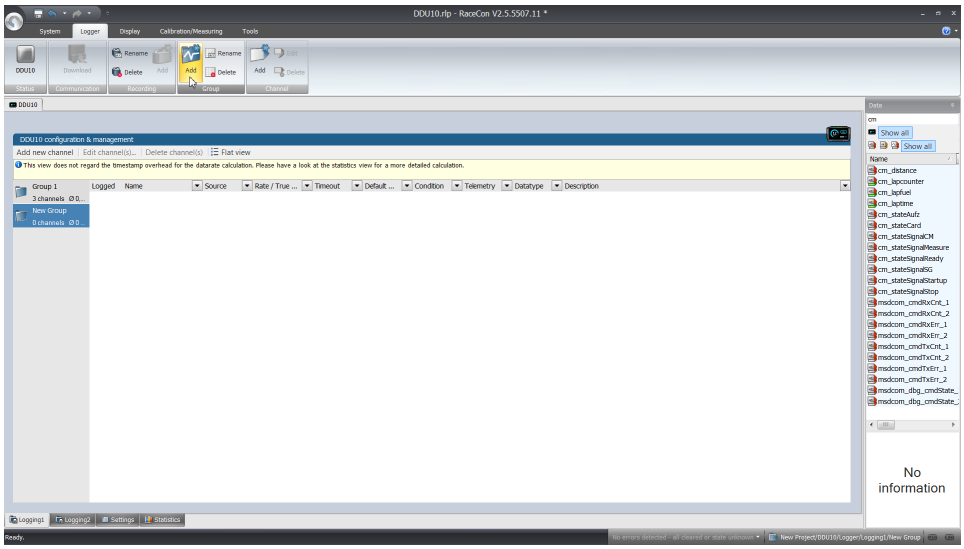
1. To add a new group, select the ‘Logger’ tab.

   The ‘Logger’ window opens.

2. Click on the ‘Add’ button.

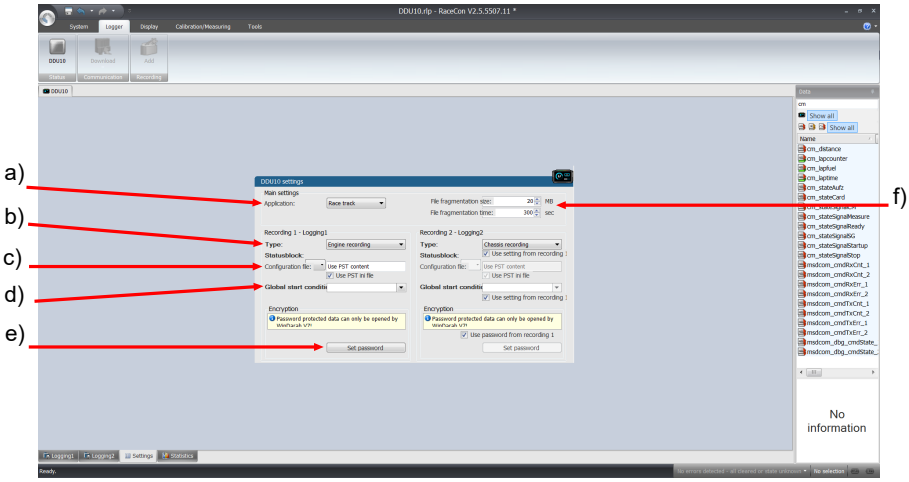
   A new group will be added and can be renamed ‘Gearbox’, ‘Aero’, ‘Engine’, etc..

3. To rename a group, click on the ‘Rename’ button next to the ‘Add’ button.



17.2.2 Logger settings

To display the global DDU 10 settings, open the ‘Logger’ window and click on the ‘Settings’ tab at the bottom.



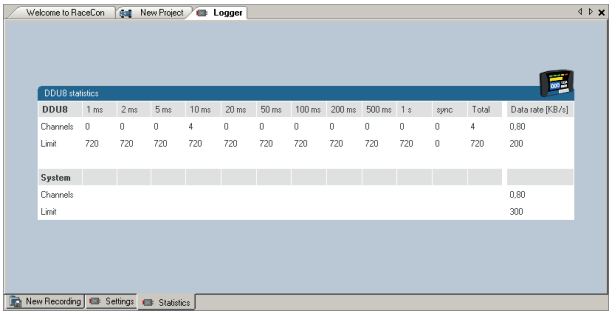
a) Choose setting for outing counter mode:
• For testbench (without lap trigger) select ‘Testbench’.
• For racetrack (with lap trigger) select ‘Racetrack’.
b) Choose wether the logging partition shall be named Engine recording or Chassis recording.
c) Advanced setting: Select your logging configuration file, if provided by your Bosch Support Engineer.
d) Choose or create the condition to start recording
e) Enter a password hint and a password (optional).
f) Setting for automatic fragmentation. Do not change!

17.2.3 Recording statistics

The tab ‘Statistics’ shows the channels’ allocation and their current data rate related to the transmission frequency of the DDU 10 and the whole transmission system.

The overview helps to detect bandwidth bottlenecks of channels. Bandwidth bottlenecks can be solved by changing the logging rate setting for each channel.

The data rate of the whole system is often less than the data rate of the DDU 10 and limits the overall transmission speed.



17.2.4 Recording diagnosis

The channel ‘statectrl_ok’ of the DDU 10 can be used for online monitoring of recording status.

Bit	Value	Name
0	1	RECORD

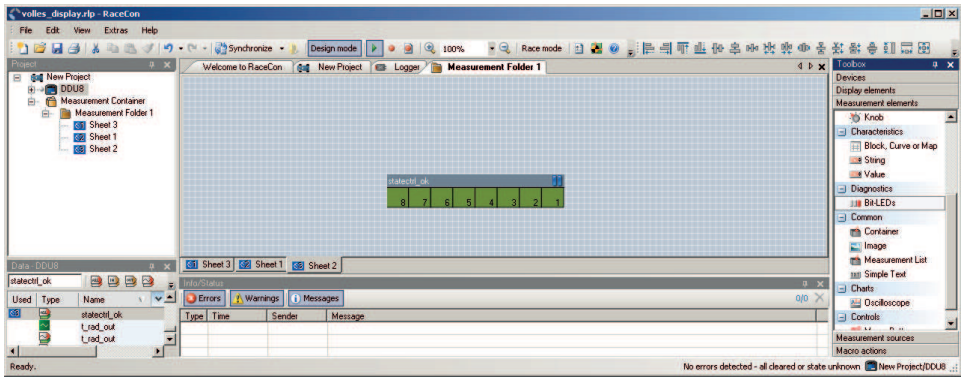
Bit	Value	Name
1	2	DATAOK
2	4	BLKOK
3	8	-
4	16	-
5	32	-
6	64	STARTED
7	128	-

Content of status bits

Name	Bitset	Bit cleared
RECORD	Measurement data is recorded.	No measurement data will be stored because measurement thresholds are not reached.
DATAOK	Received data without error.	Discarding received data because of wrong timestamps. Check wiring of SYNC signal.
BLKOK	All measurement blocks have been set up correctly.	Some measurement blocks have not been set up correctly.
STARTED	A measurement has been set up.	A measurement is not set up. Either no recording configuration has been found or logger software upgrade is not activated.

17.2.5 Displaying online recording diagnosis ("statectrl_ok")

1. To add a Recording Diagnosis element to a measurement sheet, drag a 'Bit-LED' element from the Toolbox and drop it on measurement sheet.
2. Drag channel 'statectrl_ok' from the Data Area and drop it on the 'Bit-LED' element.

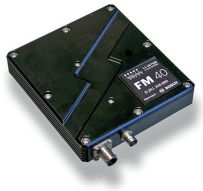


The 'Bit-LED' element shows the state of received channel data in bit-representation. A green highlighted channel means 0, a red highlighted channel means 1.

- Measurement correctly initialized, but recording threshold(s) not reached: 254
- Measurement correctly initialized, DDU 10 is recording data: 255
- Values less than 254 indicate an error state

17.3 Configuration of online Telemetry

17.3.1 Long range telemetry system FM 40



440 MHz band

25 KHz bandwidth

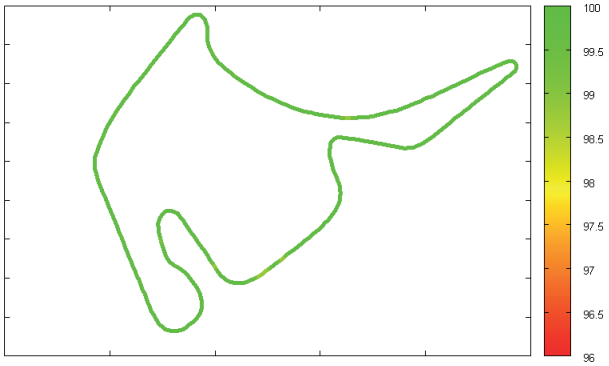
10 W max. RF output

19.2 kBit/s data rate - unidirectional

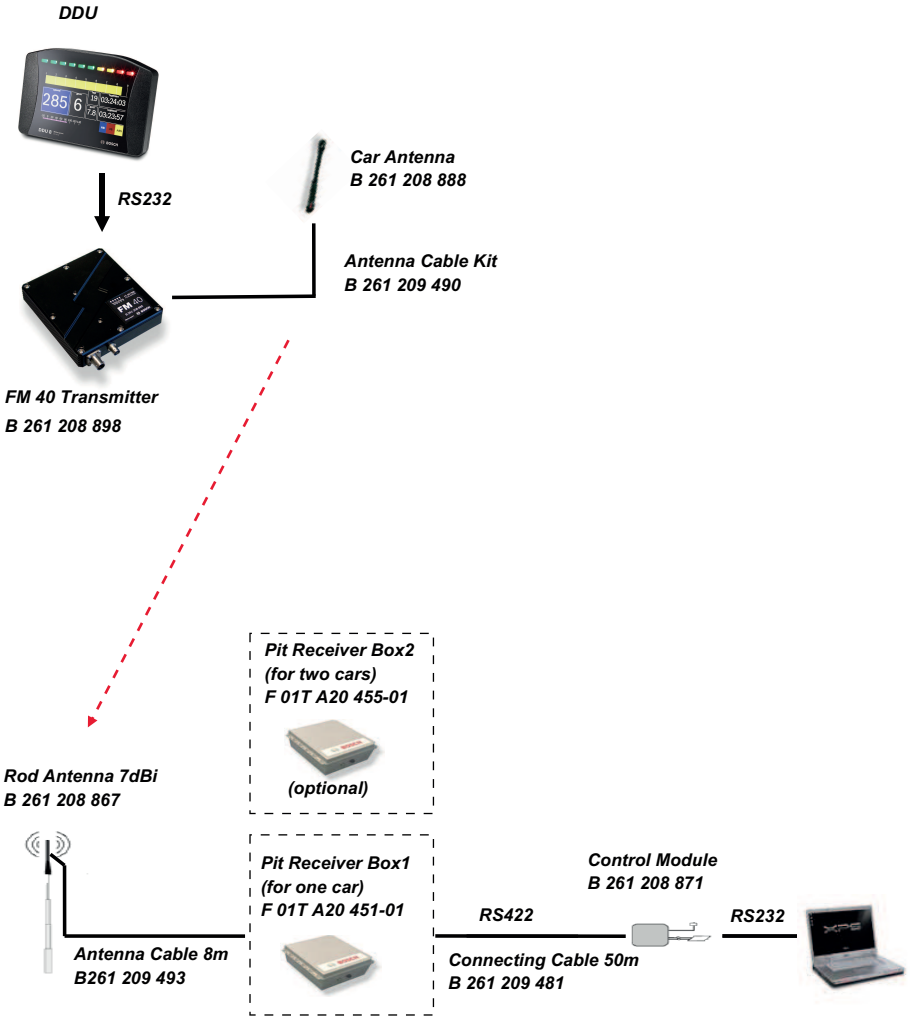
RS232 interface

Full online track coverage on almost all tracks

Link quality at Hockenheim

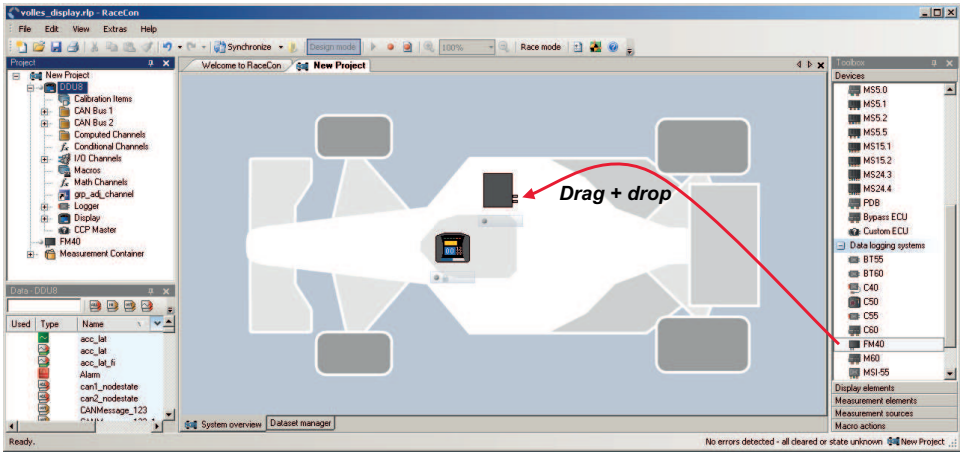


17.3.2 Hardware setup

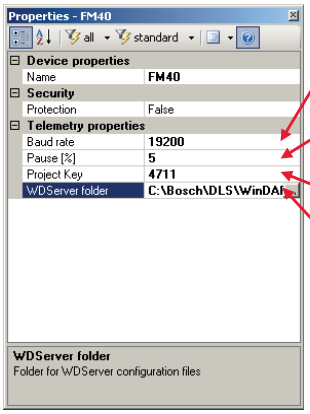


17.3.3 Software setup

1. Drop FM 40 from Toolbox into system overview.



2. Click on FM 40 in Project Tree to display the Properties Menu.



Baudrate of DDU (must match baudrate of FM 40)

Transmission pause (5% recommended for improved re-synchronisation)

Project Key (1111…9999)

Path to configuration folder of WDServer on receiver PC (usually ….\WDServer\DCP) to store telemetry configuration file

Adding channels to telemetry

1. Expand the list of 'Loggers' by clicking on '+' in the DDU 10 Project Tree.
2. Double-click on 'Recording' in DDU 10 Project Tree.

   The recording configuration is displayed in the Main Area.
3. Click 'Edit channel(s)'.

   The 'Edit Recording Channels' window appears.
4. Choose between 'Fast/Slow block' transmission.

Using fast block/slow block transmission

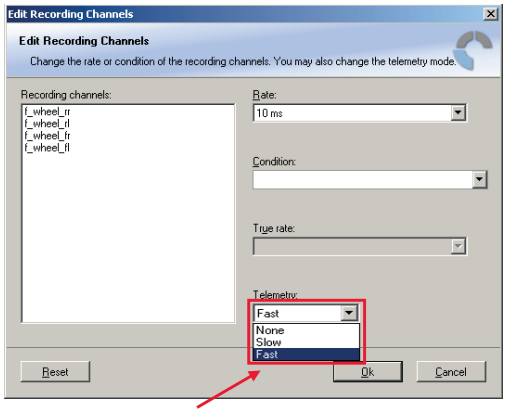
DDU 10 telemetry uses available bandwidth of Telemetry Unit FM 40 (19,200 baud -> approx. 1,700 bytes/s). The bandwidth has to be divided into channel information to be transmitted high-frequently and low-frequently using the 'fast/ slow block' setting.

Channels are grouped into 8 blocks which are transferred each cycle:

- Fast block (Block 1) is transferred every cycle and used for a high-frequent transmission of channel information (e.g. speed, rpm).
- Slow blocks (Block 2…n) are transferred every n-th cycle and used for a low-frequent transmission of channel information (e.g. tire pressure, oil temperature).

Transmission Scheme

If the maximum bandwidth of a block is reached, a warning will be displayed. To fix this problem you can view the allocation of the channels and data rate in the 'Statistics' tab of the Main Area.



None – channel(s) are not transferred
Slow – channel(s) are transferred in the slow telemetry block
Fast – channel(s) are transferred in the fast telemetry block

See also

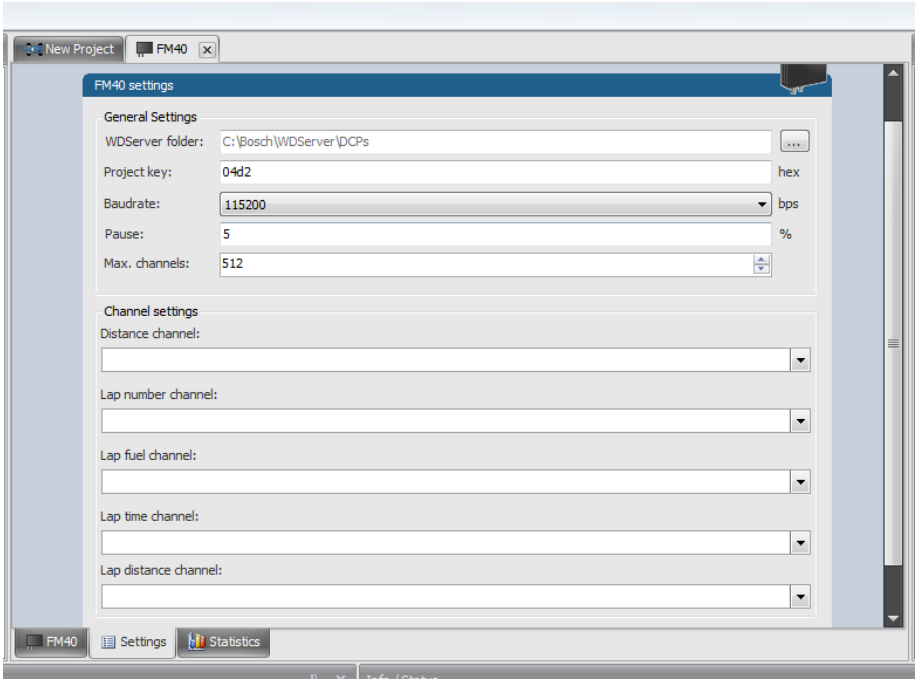
- *Adding a recording group [▶ 124]*

17.3.4 Telemetry channels with special functionality

The FM40 allows the transmission of special information such as running distance of current lap, lap number of current lap and lap time, fuel consumption of last lap completed. You have to assign the channel type to the telemetry channel so that it is recognized accurately by RaceCon.

Channel's names are e.g.: distlap, fuelcons, lapctr, laptime. Different channel names are possible between different devices (e.g. ECU MS 3 Sport, ECU MS 5.1).

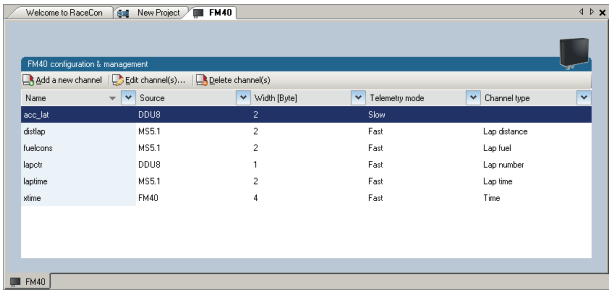
1. Double-click on FM40 in the project tree. An overview of all available telemetry channels is displayed.
2. Click on the 'Settings' tab at the bottom, to edit the channels.



3. Assign the desired channels to the channel types. The table below shows the function of the available channel types.
4. Click 'Ok' when done.

Measurement channel	Function
Distance	Running distance of current lap
Lap number	Lap number of current lap
Lap fuel	Fuel consumption of last lap completed
Lap time	Exact lap time of lap completed

The telemetry channels and their assigned channel types are displayed in the overview list.



17.4 Configuration of Ethernet Telemetry

The usage of ethernet telemetry requires the software upgrade ETHER_TELE (Enable ethernet telemetry).

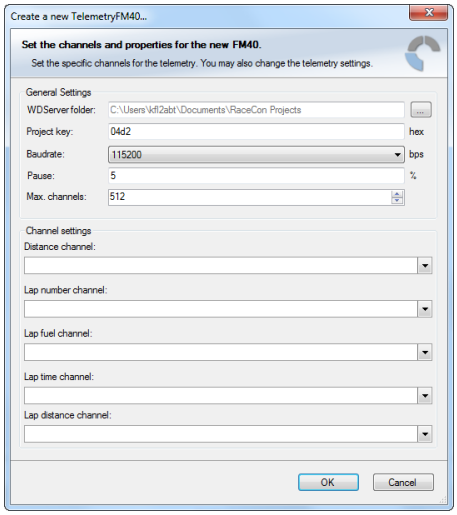
17.4.1 Hardware Setup

You can use ethernet telemetry to connect a transmitter on an ethernet line of your device, to send the data to a pit unit and from there to your computer. It is also possible to connect the ethernet line directly to your computer, for example when you use the device on a dyno.

17.4.2 Software Setup

The following steps will show you how to insert an FM40 telemetry device in the RaceCon project for your vehicle. Please note that, in order for this action to be available, first you have to configure a data logger in RaceCon.

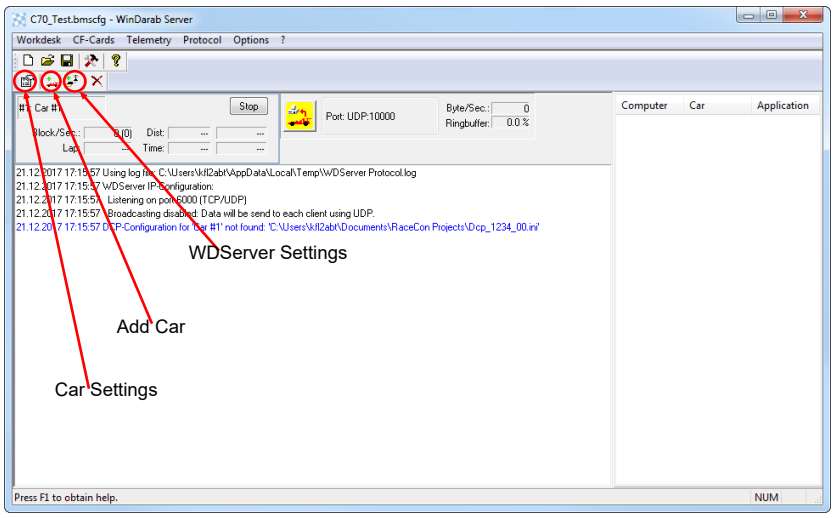
1. In the toolbox of the device, click on an FM40 telemetry device and drag it into the project window.
   Please note that at this point, an ethernet module has not been integrated into the RaceCon software, yet. However, ethernet telemetry setup within RaceCon is identical to that of the FM40.
2. Adding a FM40 should open up the FM40 settings window. If this window does not show up, select FM40 - Settings.



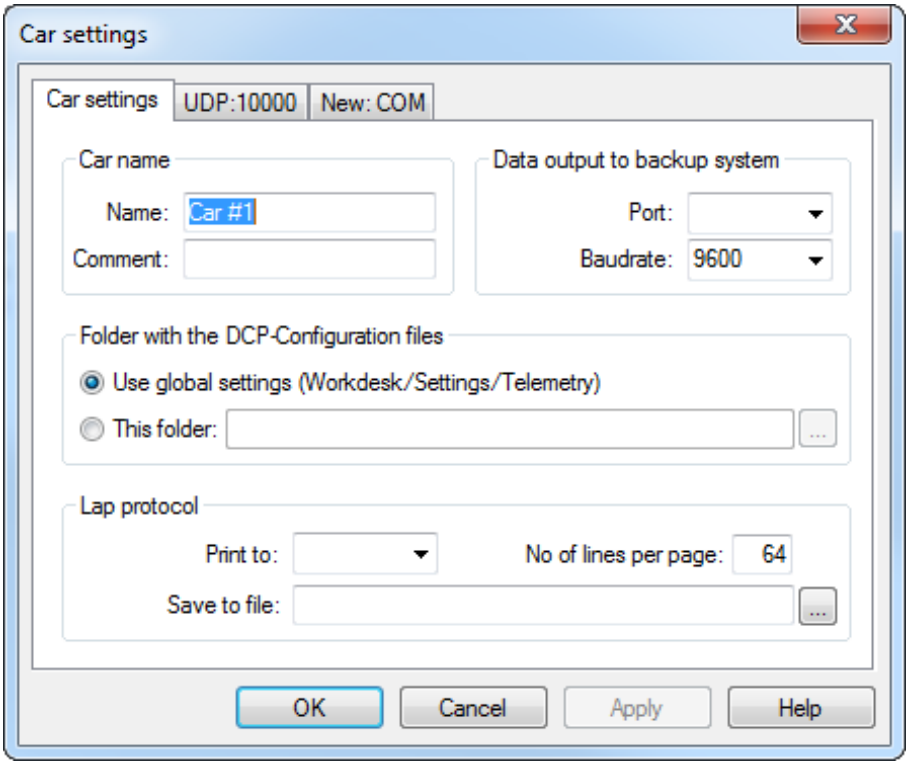
3. Define the settings as described below:
   - WDServer folder: This is the folder where the ".INI" files for the WDServer are stored. Please make note of this folder path, as this path will need to be referenced by WDServer in order to enable telemetry live viewing and logging.
   - Project Key: The project key is a data security function, that is used by Darab to differentiate data from different users. This key should always be 4 digits long.
   - Baudrate: Set to 115,200 bps.
   - Channel settings: Select the needed channels for the FM40 configuration.
4. Change to a calibration/measuring page and drag and drop the following channels to the measurement sheet:
   - TELE_UDP_IP,
   - TELE_UDP_PORT,
   - TELEMETRY_MODE.
5. Change the TELEMETRY_MODE to UDP.
6. Set the TELE_UDP_IP to the IP address of the receiving device.
7. Set the TELE_UDP_PORT to a unique value for each unit. This will be relevant when connecting to WDServer.
8. Define your telemetry channels.

17.4.3 Setting up car in WDServer

WDServer is a program used to capture data streaming from a transmitter and convert to WinDarab; WDServer also creates a log of the data received over telemetry.

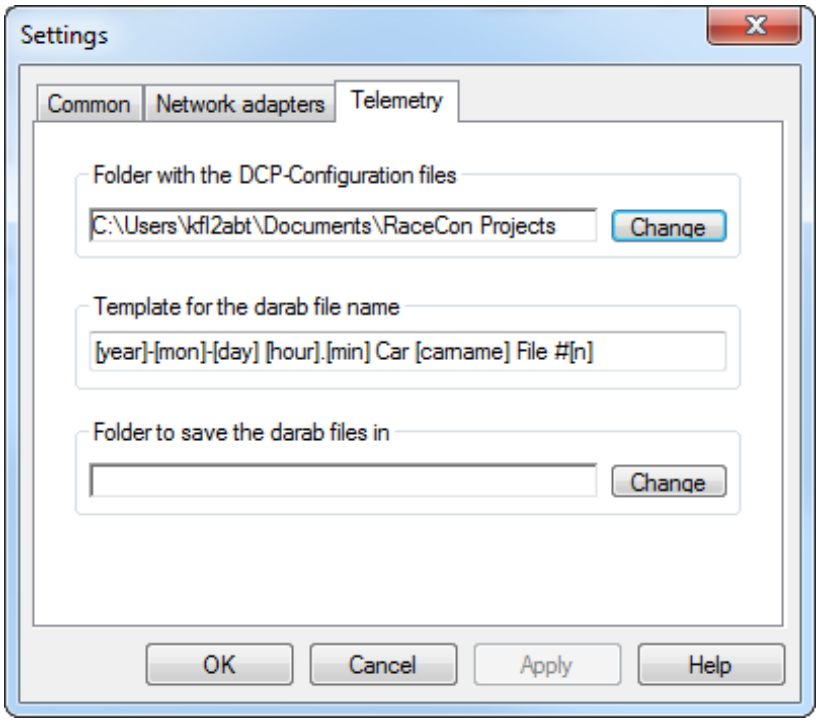


1. To set up a new car, select 'Add Car'.
2. In the Car settings tab, enter a name for the new vehicle.
   This name will be used as a part of the file name for WDServer's log of received telemetry data and will show up in WinDarab, when searching for the telemetry stream in the Network folder.

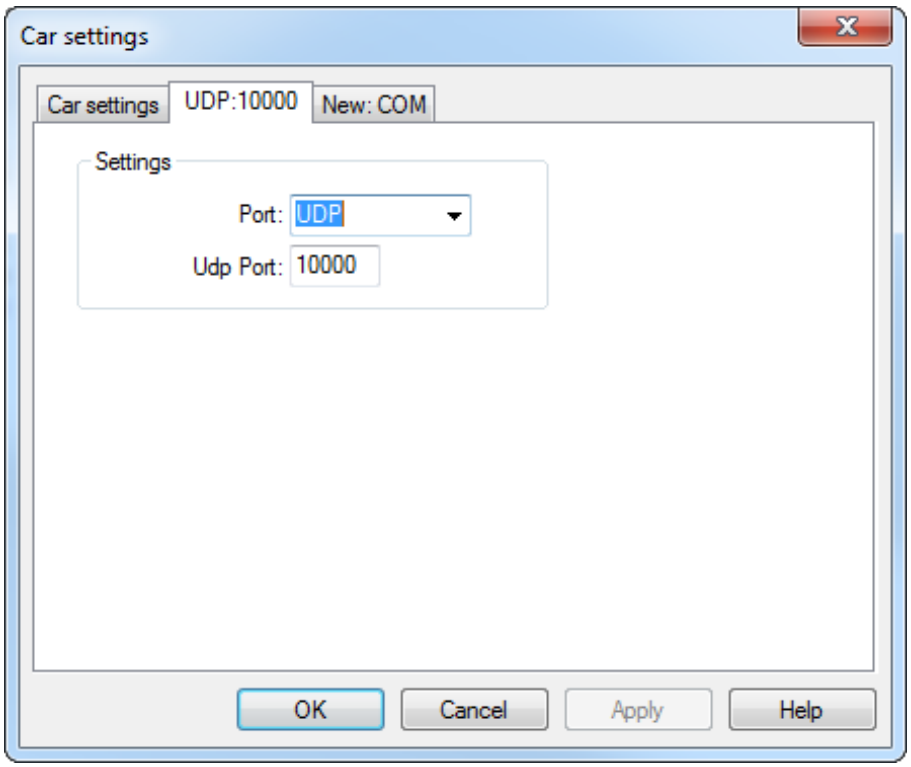


You are now at the final step of configuring the telemetry stream. In order for the data to be decrypted by WDServer, two *.ini files must be referenced by WDServer. After the configuration is sent to the logger, these two different *.ini files will be created in the base folder. You can find the base WDServer folder, if you right-click the FM40 and select 'Properties'. You can change this folder location for easier access if desired.

3. Define the link to the folder of the *.ini files for each car or define it in the general WDServer settings, under the ‘Telemetry’ tab.

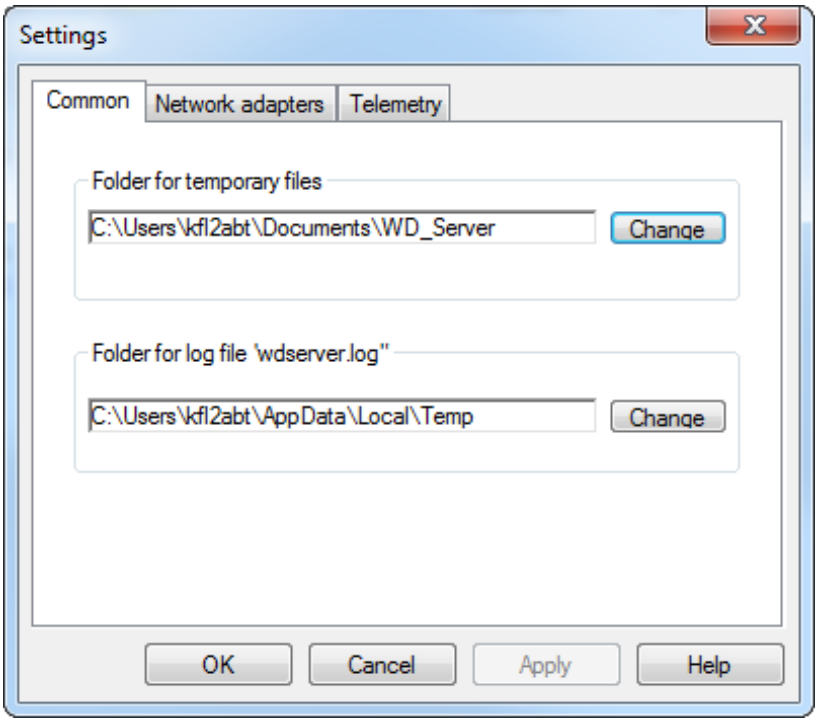


4. Under the ‘UDP’ tab, select the drop-down menu and type in “UDP”.
5. For the UDP Port, type in the port number assigned to the device in RaceCon.
Each vehicle being read by a single receiver device must have a unique port number.
This information will be provided by Bosch upon delivery of the devices.



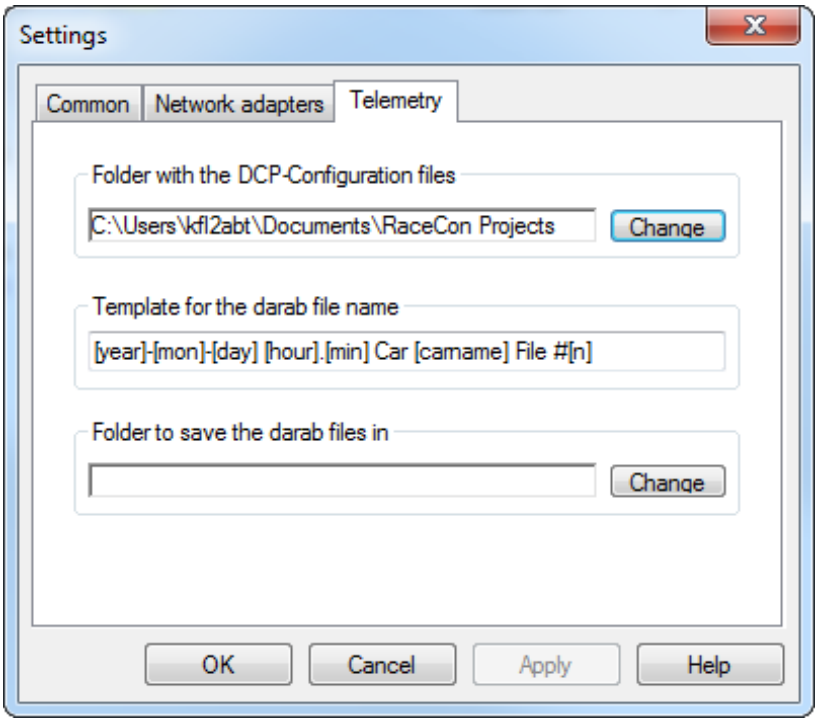
6. Click ‘OK’, to close the window.
7. Select the button ‘WDServer Settings’.

8. Under the ‘Common’ tab, choose directories where WDServer can store its temporary files and log files. These are created during telemetry reception and can be used to help diagnose issues.



Under the ‘Telemetry’ tab, the first section requests a folder path for the DCP- Configuration files. This is the folder path where RaceCon stored the *.ini files required by WDServer.

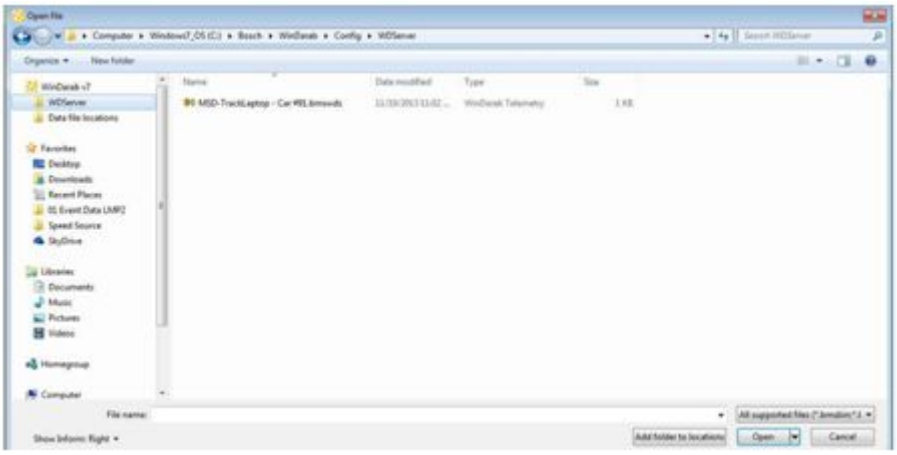
9. Click on the “Change” button next to this section and navigate to this folder.
A template can also be specified for the file nomenclature for logged telemetry as well as a save location.



- To insure proper communication between WDServer and the receiver, do not delete any old *.ini files from this folder path. As mentioned in section 5, RaceCon will generate a new *.ini file each time a project is synchronized; each new *.ini file instance has an incremented file name. Retaining all of these *.ini file iterations will insure that WDServer always has a reference to whichever configuration is programmed into your vehicle's logger system.
- If the RaceCon project for the vehicle resides on a different computer, than that which is used for telemetry, then all *.ini files for a given project should be transferred to the telemetry computer after every data synchronization in RaceCon. WDServer may have trouble recognizing *.ini files stored on removable media, so best practice is to copy these files to the telemetry computer's hard drive.

17.4.4 Loading the telemetry data

The following is an example of a file name and data format for Car #91. File is typically located in WinDarab/Config/WDServer:



In the File Explorer, click 'Open' and navigate to the data set. Under 'Network', search for the car or cars that are required for viewing.

17.5 Setup for USB recording

This function requires the installation of Software Upgrades. Look into the datasheet of your device, to see which upgrades are available for your device.

Software Upgrade USB_DATA enables USB recording. To activate Software Upgrade USB_DATA, enter the license key as described in the chapter 'Feature activation' [▶ 19].

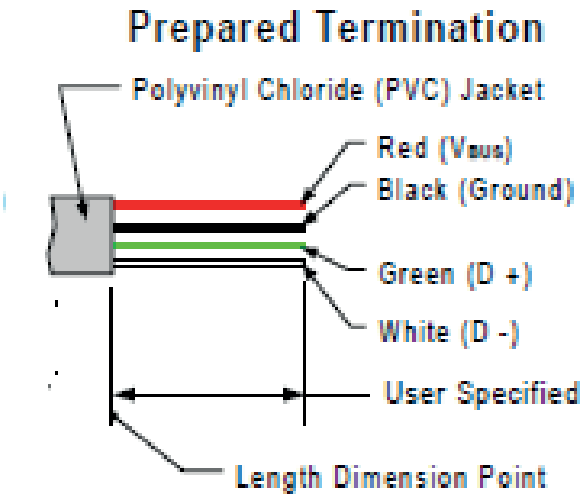
For USB recording, Software Upgrade FULL_LOG_1 should also be enabled.

Wiring harness

Bit	Value
USB_Device_Power	Power (red)
USB_Device_DP	D+ (green)
USB_Device_DN	D- (white)
USB_Device_Gnd	GND (black)

For further information, see the pinlayout of the device.

Colors matching a standard USB cable



Storage device

The recording function can be used with a dedicated Bosch Motorsport USB device. The USB device has to be preformatted with the Bosch File System (BFS) in RaceCon before first use.

To format the USB device with the Bosch File System (BFS), do the following steps:

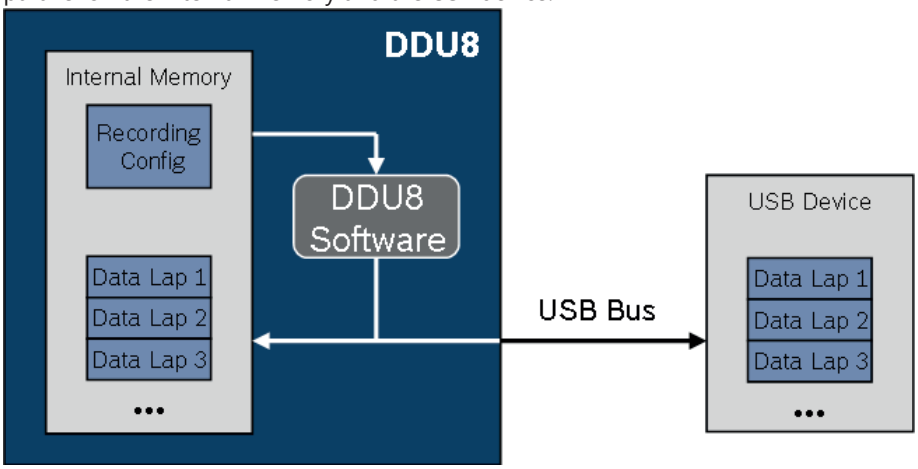
In RaceCon, select 'Tools' - 'Extras' and choose 'Format USB stick'.

Press 'Format'.

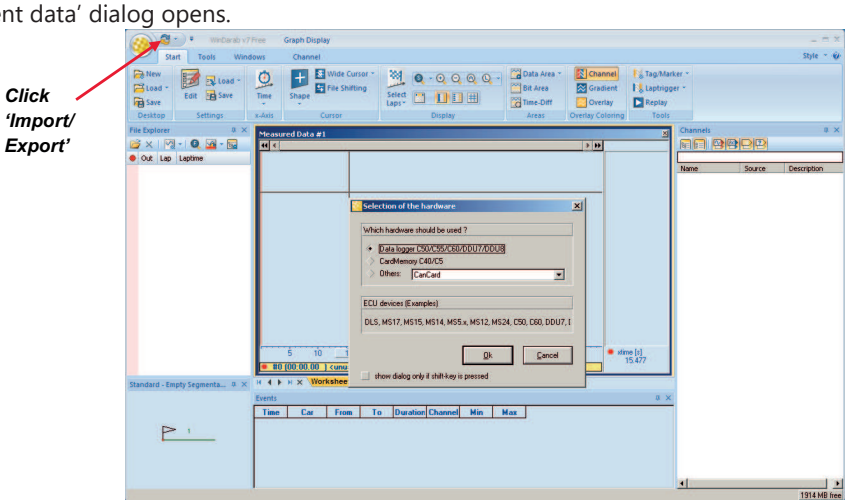
An USB device is recognized by Windows as a 'storage medium', but it can only be initialized with RaceCon and read with WinDarab.

17.5.1 Recording data on a USB device

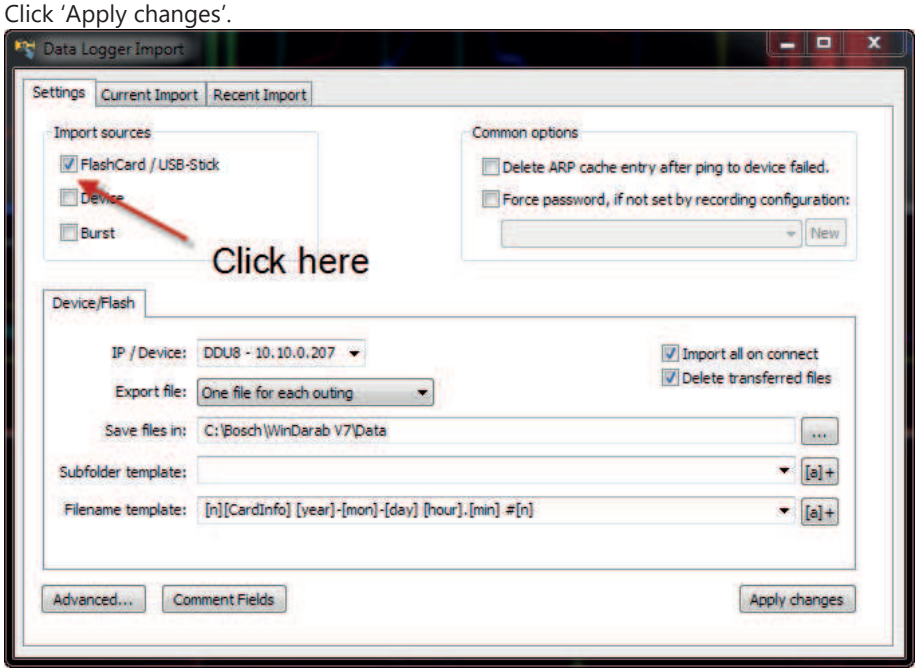
1. Plug an USB device to DDU 10.
2. Prepare a recording configuration in RaceCon.
3. Power on the system and connect with RaceCon to the vehicle.
4. Download the configuration to the DDU 10.
5. Record measurement data. If an USB device is present, the DDU 10 stores the data in parallel on the internal memory and the USB device.



6. Power off the system.
7. Remove USB device from the vehicle.
8. Start the WinDarab software.
9. Click on the ‘Import/Export’ icon.
10. Select ‘Data logger CXX/DDUX/MSX’ and click ‘OK’ when done. The ‘Read measurement data’ dialog opens.



11. Click on ‘Settings’ tab and select the option ‘Flash Card/USB Stick’.
12. Click ‘Apply changes’.

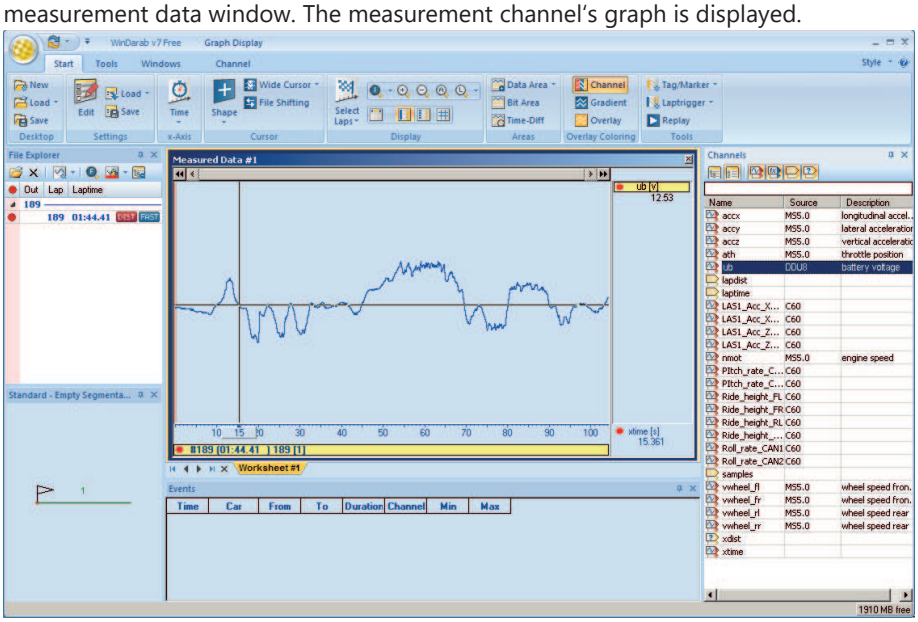


13. Change to the ‘Current import’ tab.
14. Insert the USB device. Data transmission from the device starts automatically, if "import all on connect" is selected in the settings, otherwise start the import manually. Measurement files are stored automatically in the base folder.
15. Click ‘Close’ when transmission has finished.

NOTICE

Refer to the WinDarab V7 manual for instructions on how to use the 'Data Logger Import' dialog and for more detailed descriptions and instructions.

16. Click on the Start button and choose 'Open measurement file'.
17. Select the measurement files from the storage folder.
18. Click on 'Open'.
19. Click 'New Desktop' to open a new measurement data window.
20. Drag the desired measurement channel from the Channel list and drop it into the measurement data window. The measurement channel's graph is displayed.



17.5.2 USB device handling hints

Using the USB device

Always plug the USB device into vehicle before power up to ensure that all measurement data is stored on the USB device.

If the USB device is plugged in after recording has started, only the current data is saved.

Data recorded on the DDU 10 before the USB device is plugged in will not be saved.

Removing the USB device

Always power off the system before unplugging the USB device!

If the USB device is unplugged while recording is active, parts of the measurement data may be missing.

If the USB device is unplugged and re-inserted for < 4 s while the DDU 10 is powered up, the DDU 10 still records data.

If the USB device is unplugged and re-inserted for > 4 s while the DDU 10 is powered up or a different USB device is plugged in, the DDU 10 restarts. In this case, the DDU 10 is not operational for 1.5 s.

17.5.3 Troubleshooting

When no data on the USB device is recorded:

Configure the measurement label **usb_mediastate** on a RaceCon measurement view or on a DDU 10 display page.

The value of **usb_mediastate** reflects the operating condition of the USB bus:

State	Description
0: Wait: Device not found	The USB device is not found (also: waiting for re-plug stick). No USB device inserted. USB device is defect. No electrical connection or wiring harness problem. USB software upgrade not activated (Purchase of unlock code needed).
1: Wait: Device detected	An USB device is found, but not yet installed.
2: Ok: Media installed	The USB device is found and is operational (idle). This does not imply that recording data is written!
3: Stop: Device unplugged	The USB device has been removed. The DDU 10 performs a restart when an USB device is re-plugged in.
4: Ok: Media access	Data is currently read from/written to the USB device.
5: Error: Media error	The communication to the USB device broke down. The USB device is defect. The USB device is not supported by DDU 10.
6: Error: Media corrupt	The USB device is not in valid BFS format. (Hint: Re-format the USB device in RaceCon.)

18 Ambient Light Sensor

The DDU 10 is equipped with an ambient light sensor. The sensor is located at the front of the DDU 10 below the LEDs on the right side. The sensor is linked to the channel “eamb” with the unit lux.

The ambient light sensor can be used to calculate an automatic brightness switch. To use this function, please follow the steps below.

1. Measure the range of light intensity in your vehicle during all relevant conditions.

   The easiest option to evaluate the needed range is to log the “eamb” channel and expose the vehicle to the brightest and darkest conditions it will face during usage.

   In this example, the expected value will be 10 lx as the lowest and 300 lx as the highest measured light intensity in the cockpit.

2. Drag and drop a “Characteristic Curve” from the “Computed Sources” onto the “Computed Channels”.

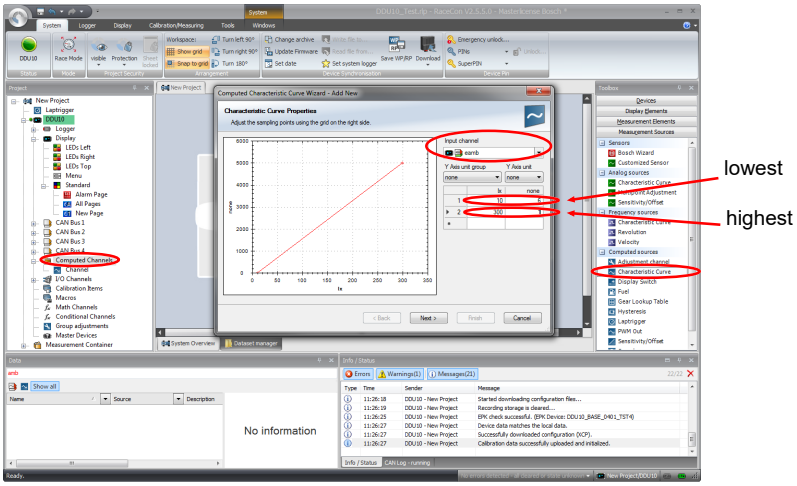
   This will open the configuration wizard.

3. Select “eamb” as input channel.

4. Assign the lowest lux value to the highest possible brightness switch setting (6) and the highest lux value to the lowest switch position (1).

   This will generate a linear change of the switch. If needed you can include additional points to change the linear behavior.

5. Click “Next” when all points are defined as needed.



   The “Analog Source Properties” window opens in the wizard. The maximum and minimum value of the channel is automatically assigned according to the previous settings. In general, it should be from 1 to 6.

6. Click “Finish”.

7. Assign a suiting name to the new channel and click “OK”.

   In this example, the channel will be named “Ambient_Light_Switch_Computed”

   To avoid a toggling brightness setting, a math channel with a kind of hysteresis has to be created.

8. Double click on “Math Channels” and click on “Add channel”.

9. Copy the Formula below into the new math channel and replace the example name with your name of the computed channel.

latch(#DDU10#Ambient_Light_Switch_Computed+0.5, *abs*(#DDU10#Ambient_Light_Switch_Computed-#DDU10#display_brightness) > 0.6)

The formula compares the brightness setting of the device with the computed channel. If the deviation is within a certain range, it allows sending of the new value to the device. This formula only works correctly if it is used as the brightness switch channel.

10. Assign a suiting name to the new channel and click “OK”.
11. Go to “Display” - “Settings” and define the created math channel as the brightness switch channel.
12. Define the percentagewise brightness of the display and the LEDs according to each switch position.

19 Open Source Software (OSS) Declaration for the Display

These third party software components are used within the graphic rendering engine

19.1 EZXML License

The ezxml library (ezxml.sourceforge.net) provides XML model parsing support for the Graphic Engine.

Applies To:

libgre.dll, libgre.a

The ezxml xml parsing library is used by all Graphic Engine runtime configurations

Copyright 2004-2006 Aaron Voisine aaron@voisine.org

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

19.2 Simple OpenGL Library License (SOIL)

The SOIL library (http://www.lonesock.net/soil.html) is used to load images in a Graphic application.

Applies To:

libgre.dll, libgre.a

The SOIL image library is used by all Graphic Engine runtime configurations

Jonathan Dummer

2007-07-26-10.36

Simple OpenGL Image Library

Public Domain

using Sean Barret's stb_image as a base

Thanks to:

* Sean Barret - for the awesome stb_image

* Dan Venkitachalam - for finding some non-compliant DDS files, and patching some explicit casts

* everybody at gamedev.net

19.3 FreeType License

The FreeType Project's www.freetype.org) library is used by the Graphic Engine for non-bitmap font rendering.

Applies To:

All Graphic Runtime Engines using sbfreetype libraries

Applies To:

librender-plugin-*.dll, librender-plugin-*.a

Unless explicitly configured, all Graphic Engine render plugins link against the FreeType libraries.

Portions of this software are copyright © <2011> The FreeType Project (www.freetype.org). All rights reserved.

Legal Terms

0. Definitions

Throughout this license, the terms `package', `FreeType Project', and `FreeType archive' refer to the set of files originally distributed by the authors (David Turner, Robert Wilhelm, and Werner Lemberg) as the `FreeType Project', be they named as alpha, beta or final release.

`You' refers to the licensee, or person using the project, where `using' is a generic term including compiling the project's source code as well as linking it to form a `program' or `executable'. This program is referred to as `a program using the FreeType engine'.

This license applies to all files distributed in the original FreeType Project, including all source code, binaries and documentation, unless otherwise stated in the file in its original, unmodified form as distributed in the original archive. If you are unsure whether or not a particular file is covered by this license, you must contact us to verify this.

The FreeType Project is copyright (C) 1996-2000 by David Turner, Robert Wilhelm, and Werner Lemberg. All rights reserved except as specified below.

1. No Warranty

   THE FREETYPE PROJECT IS PROVIDED `AS IS' WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT WILL ANY OF THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY DAMAGES CAUSED BY THE USE OR THE INABILITY TO USE, OF THE FREETYPE PROJECT.

2. Redistribution

   This license grants a worldwide, royalty-free, perpetual and irrevocable right and license to use, execute, perform, compile, display, copy, create derivative works of, distribute and sublicense the FreeType Project (in both source and object code forms) and derivative works thereof for any purpose; and to authorize others to exercise some or all of the rights granted herein, subject to the following conditions:

   – Redistribution of source code must retain this license file (`FTL.TXT') unaltered; any additions, deletions or changes to the original files must be clearly indicated in accompanying documentation. The copyright notices of the unaltered, original files must be preserved in all copies of source files.

– Redistribution in binary form must provide a disclaimer that states that the software is based in part of the work of the FreeType Team, in the distribution documentation. We also encourage you to put an URL to the FreeType web page in your documentation, though this isn't mandatory.

These conditions apply to any software derived from or based on the FreeType Project, not just the unmodified files. If you use our work, you must acknowledge us. However, no fee need be paid to us.

3. Advertising

Neither the FreeType authors and contributors nor you shall use the name of the other for commercial, advertising, or promotional purposes without specific prior written permission.

We suggest, but do not require, that you use one or more of the following phrases to refer to this software in your documentation or advertising materials: `FreeType Project', `FreeType Engine', `FreeType library', or `FreeType Distribution'.

As you have not signed this license, you are not required to accept it. However, as the FreeType Project is copyrighted material, only this license, or another one contracted with the authors, grants you the right to use, distribute, and modify it. Therefore, by using, distributing, or modifying the FreeType Project, you indicate that you understand and accept all the terms of this license.

19.4 Lua License

The Lua engine (http://www.lua.org) provides a script interface for manipulating content in a Graphic application.

Applies To:

libgre-plugin-lua.dll, libgre-plugin-lua.a

The Lua Graphic Engine plugin

License for Lua 5.0 and later versions

Copyright © 1994-2008 Lua.org, PUC-Rio. Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions: The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

19.5 BSD 4-Clause License (getopt)

The NetBSD getopt argument parsing function is used to parse command line arguments.

The Graphic Runtime engines used on all Windows platforms (win32, wince, wincompact7, wec2013) includes the getopt source from the NetBSD operating system distribution.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgement:

   This product includes software developed by the University of California, Berkeley and its contributors.
4. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

19.6 BSD 3-Clause License (getsubopt)

The NetBSD getsubopt argument parsing function is used by Graphic Engine plugins to parse plugin options.

Applies To:

libgre.dll, libgre.a

The Graphic Runtime engines used on most operating systems, including all Windows platforms (win32).

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

19.7 Imagination OpenGL License

The libEGL.dll and libEGLv2.dll libraries from Imagination Technologies to provide OpenGL rendering capabilities

Applies To:

libEGL.dll, libGLESv2.dll

The Graphic Runtime engines used on all Windows platforms (win32) that use the OpenGL bindings include the Imagination libraries.

The MIT License (MIT)

Copyright (c) Imagination Technologies Ltd.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

19.8 GNU LESSER GENERAL PUBLIC LICENSE (pthread-win32)

Applies To:

All Graphic Runtime Engines using Windows operating systems

Applies To:

pthreadVC2.dll, pthreadVC2.lib

This provides POSIX pthread API functionality in a Windows environment (win32, wince, wincompact7, wec2013).

Project Page: https://www.sourceware.org/pthreads-win32/

pthreads-win32 - a POSIX threads library for Microsoft Windows

This file is Copyrighted

This file is covered under the following Copyright:

Copyright (C) 2001,2006 Ross P. Johnson

All rights reserved.

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Pthreads-win32 is covered by the GNU Lesser General Public License

Pthreads-win32 is open software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation version 2.1 of the License.

Pthreads-win32 is several binary link libraries, several modules, associated interface definition files and scripts used to control its compilation and installation.

Pthreads-win32 is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.

See the GNU Lesser General Public License for more details.

A copy of the GNU Lesser General Public License is distributed with pthreads-win32 under the filename:

COPYING.LIB

You should have received a copy of the version 2.1 GNU Lesser General Public License with pthreads-win32; if not, write to:

Free Software Foundation, Inc.

59 Temple Place

Suite 330

Boston, MA 02111-1307

USA

The contact addresses for pthreads-win32 is as follows:

Web: http://sources.redhat.com/pthreads-win32

Email: Ross Johnson

Please use: Firstname.Lastname@homemail.com.au

Pthreads-win32 copyrights and exception files

With the exception of the files listed below, Pthreads-win32 is covered under the following GNU Lesser General Public License Copyrights:

Pthreads-win32 - POSIX Threads Library for Win32

Copyright(C) 1998 John E. Bossom

Copyright(C) 1999,2006 Pthreads-win32 contributors

The current list of contributors is contained in the file CONTRIBUTORS included with the source code distribution. The current list of CONTRIBUTORS can also be seen at the following WWW location:

http://sources.redhat.com/pthreads-win32/contributors.html

Contact Email: Ross Johnson

Please use: Firstname.Lastname@homemail.com.au

These files are not covered under one of the Copyrights listed above:

COPYING

COPYING.LIB

tests/rwlock7.c

This file, COPYING, is distributed under the Copyright found at the top of this file. It is important to note that you may distribute verbatim copies of this file but you may not modify this file.

The file COPYING.LIB, which contains a copy of the version 2.1 GNU Lesser General Public License, is itself copyrighted by the Free Software Foundation, Inc. Please note that the Free Software Foundation, Inc. does NOT have a copyright over Pthreads-win32, only the COPYING.LIB that is supplied with pthreads-win32.

The file tests/rwlock7.c is derived from code written by Dave Butenhof for his book 'Programming With POSIX(R) Threads'. The original code was obtained by free download from his website http://home.earthlink.net/~anneart/family/Threads/source.html and did not contain a copyright or author notice. It is assumed to be freely distributable.

In all cases one may use and distribute these exception files freely. And because one may freely distribute the LGPL covered files, the entire pthreads-win32 source may be freely used and distributed.

General Copyleft and License info

For general information on Copylefts, see:

http://www.gnu.org/copyleft/

For information on GNU Lesser General Public Licenses, see:

http://www.gnu.org/copyleft/lesser.html

http://www.gnu.org/copyleft/lesser.txt

Why pthreads-win32 did not use the GNU General Public License

The goal of the pthreads-win32 project has been to provide a quality and complete implementation of the POSIX threads API for Microsoft Windows within the limits imposed by virtue of it being a stand-alone library and not linked directly to other POSIX compliant libraries. For example, some functions and features, such as those based on POSIX signals, are missing.

Pthreads-win32 is a library, available in several different versions depending on supported compilers, and may be used as a dynamically linked module or a statically linked set of binary modules. It is not an application on it's own.

It was fully intended that pthreads-win32 be usable with commercial software not covered by either the GPL or the LGPL licenses. Pthreads-win32 has many contributors to it's code base, many of whom have done so because they have used the library in commercial or proprietry software projects.

Releasing pthreads-win32 under the LGPL ensures that the library can be used widely, while at the same time ensures that bug fixes and improvements to the pthreads-win32 code itself is returned to benefit all current and future users of the library.

Although pthreads-win32 makes it possible for applications that use POSIX threads to be ported to Win32 platforms, the broader goal of the project is to encourage the use of open standards, and in particular, to make it just a little easier for developers writing Win32 applications to consider widening the potential market for their products.

19.9 GNU LESSER GENERAL PUBLIC LICENSE

Version 2.1, February 1999

59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies

of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts

as the successor of the GNU Library Public License, version 2, hence

the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.
   You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.
2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:
   - a) The modified work must itself be a software library.
   - b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.
   - c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.
   - d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

   Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

   This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

   If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

   However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

   You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

   - a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)
   - b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.
   - c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.
   - d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.
   - e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

   For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything

that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:
   - a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.
   - b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.
8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.
9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.
10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein.

    You are not responsible for enforcing compliance by third parties with this License.
11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

    If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.
13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time.

    Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

    Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.
14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

    NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.
16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR

LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

zlib

Compression library, required by FreeType library http://www.zlib.net/

/* zlib.h -- interface of the 'zlib' general purpose compression library version 1.2.8, April 28th, 2013

Copyright (C) 1995-2013 Jean-loup Gailly and Mark Adler

This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

Jean-loup Gailly Mark Adler

jloup@gzip.org madler@alumni.caltech.edu

*/

Fonts

CPMono OpenTypeFont:

Copyright (c) 2009 by Tino Meinert. All rights reserved.

You are free:

* to copy, distribute, display, and perform the work

* to make derivative works

Under the following conditions:

Attribution. You must give the original author credit.

* For any reuse or distribution, you must make clear to others the licence terms of this work.

* Any of the above conditions can be waived if you get permission from the copyright holder.

* Nothing in this license impairs or restricts the author's moral rights.

http://creativecommons.org/licenses/by/3.0/de/deed.en_GB

20 Lap Trigger

20.1 Lap trigger (timing beacon)

Why do we need a lap trigger (timing beacon)?

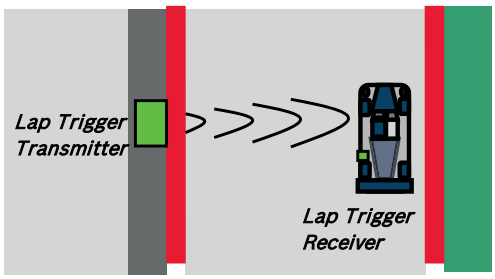
- Vehicle lap time measurement
- Calculation of lap-dependent functions (lap fuel consumption, min/max values)
- Calculation of lap distance dependent functions
- Control of data logging system

Types of Systems

- GPS based (low cost, low precision)
- IR based (low cost, high precision, limited reliability)
- RF (microwave) based (high precision, high reliability)

IR and RF based Systems consists of

- Transmitter (trackside unit)
- Receiver (in-vehicle unit)



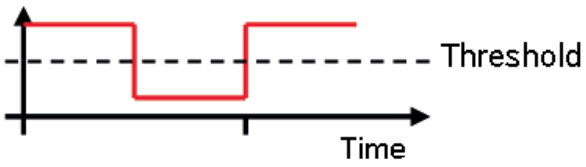
20.1.1 Electrical trigger signal

In DDU 10 all sources of measurement channels can be used as trigger signal.

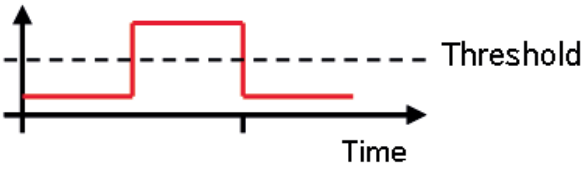
- Analog input
- Digital input
- CAN input

Signal (measurement channel) properties

Low active signal (Bosch triggers): Trigger releases if signal is below the threshold.



High active signal (other manufacturer's triggers): Trigger releases if signal is above the threshold.



Two types of trigger signal:

- Main trigger (end-of-lap at start/finish line)
- Sub-trigger (segment time, optional, not applicable with GPS lap trigger)

Bosch standard:

- Main trigger 20 ms, low active (Recommendation for RaceCon “Detecion Time” setting: 15 ms, Setting must be a slightly shorter period than the signal length of the trigger to avoid a missed trigger due to the update rate)
- Sub trigger 40 ms, low active (Recommendation for RaceCon “Detecion Time” setting: 30 ms)

20.1.2 GPS Lap trigger

The GPS lap trigger uses a GPS signal to trigger the lap timer. To function this timer an external GPS sensor (see GPS Sensor [▶ 171]) has to be connected to the device and a detection point with a detection range has to be defined in RaceCon.

The GPS detection point is defined by the latitude and longitude. The easiest way to get the latitude and longitude of a finishing line is due to a web mapping program such as google maps. With google maps, simply left-klick on the spot where you want to set the detection point. The information about the latitude and longitude will show up, in general the latitude is given at first. You should insert at least five decimal places for sufficient precision.

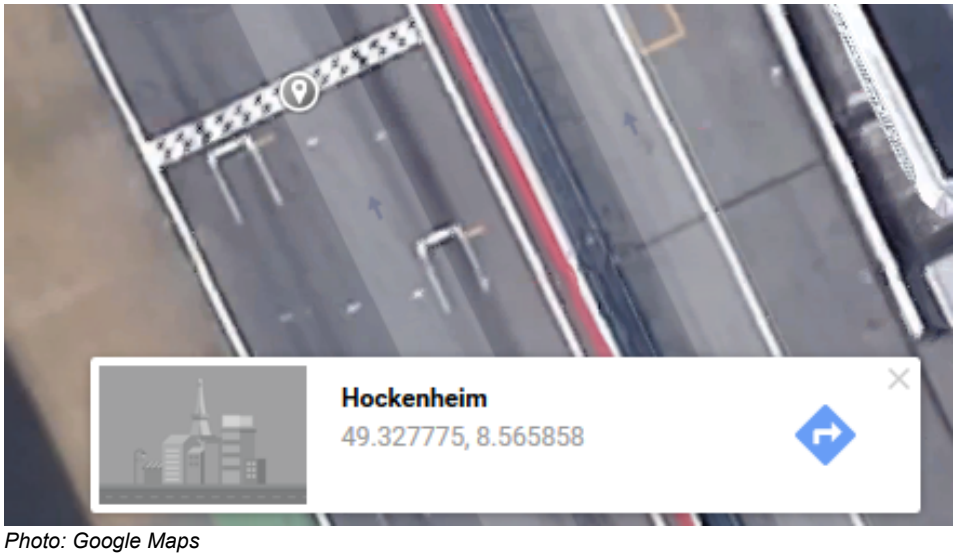


Photo: Google Maps

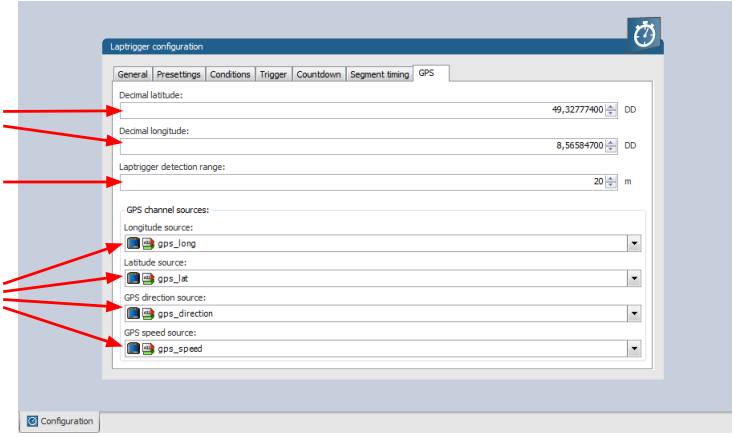
The detection range defines the radius of a circle around the detection point in which the lap trigger can be set. The lap trigger will be set as soon as the distance between the car and the detection point has reached its smallest peak. By this function an imaginary finishing line is calculated inside of the detection circle.

The imaginary finishing line can only be calculated if all channel sources are defined correctly. The latitude and longitude channel sources are mandatory for the functionality. Missing direction or speed source lowers the precision of the system.

Define the latitude and longitude of the GPS detection point.

Define the detection range around the detection point.

Define the channel sources for Longitude, Latitude, Direction and Speed.



NOTICE

The configuration of the sensor update rate and the detection range must insure to receive a valid GPS point in the detection range, despite the occurring vehicle speed near the detection point.

20.1.3 Prevention of false triggers

- Race track topology and transmitter location frequently cause false triggers.
- Software functionality prevents acceptance of false triggers.
- Minimum vehicle speed for acceptance of trigger prevents false triggers while vehicle is stationary in the pits.
- Time based re-trigger protection prevents false triggers due to signal reflections on main straight.
- Lap distance based retrigger protection prevents false triggers due to track topology.

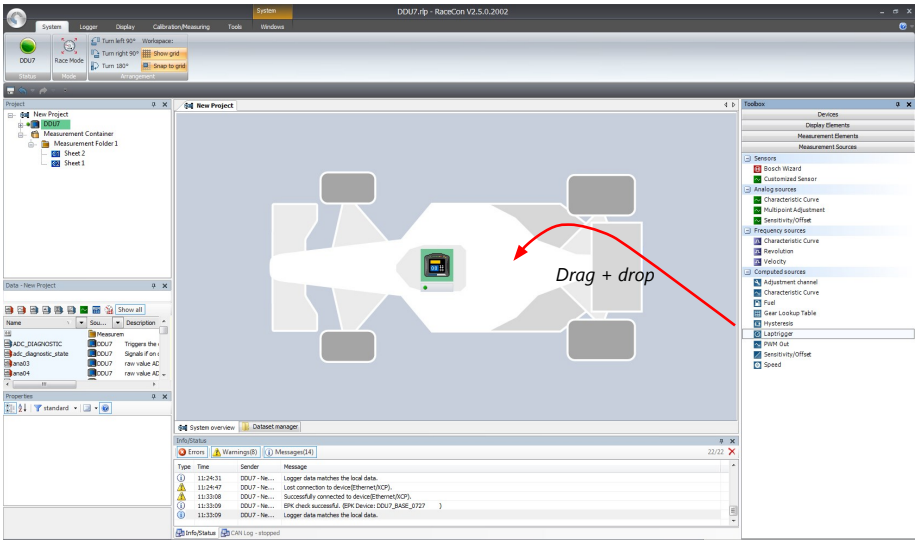
20.1.4 Forced triggers

Lap distance based insertion of ‘forced trigger’.

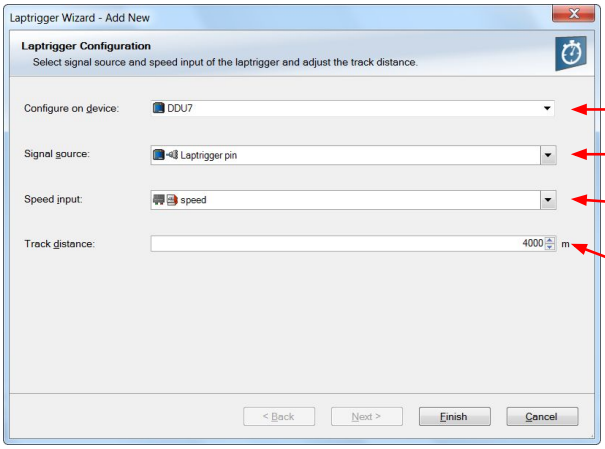
Under race conditions, trigger signals are sometimes missed. Software functionality introduces ‘forced trigger’.

20.1.5 Setting up a lap trigger

1. Click ‘Measurement Sources’ in Toolbox.
2. Drag ‘Laptrigger’ into ‘System Overview’. Do not drop it on ‘DDU 10’!



A ‘Laptrigger Wizard’ window opens.



Choose the device which receives the trigger signal

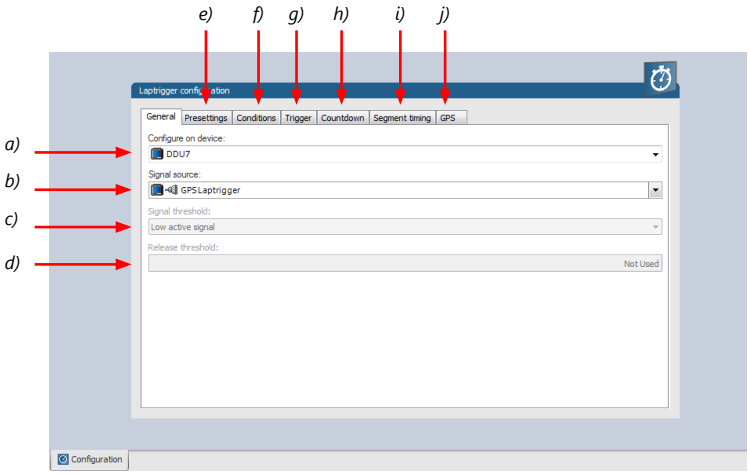
Choose the signal channel for the trigger signal

Choose the source for the vehicle speed

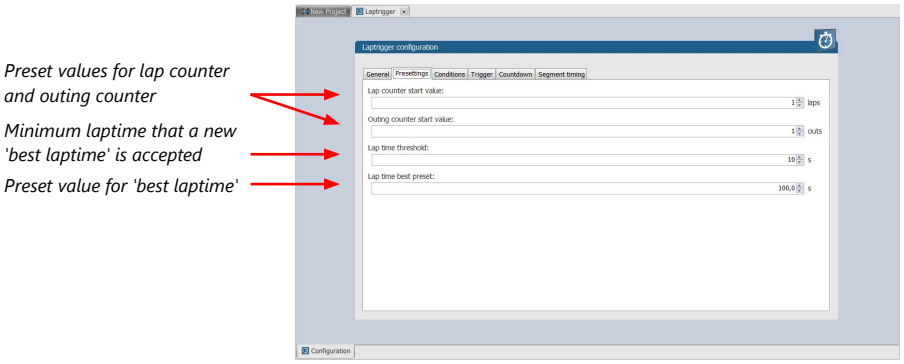
Enter the distance of the racetrack

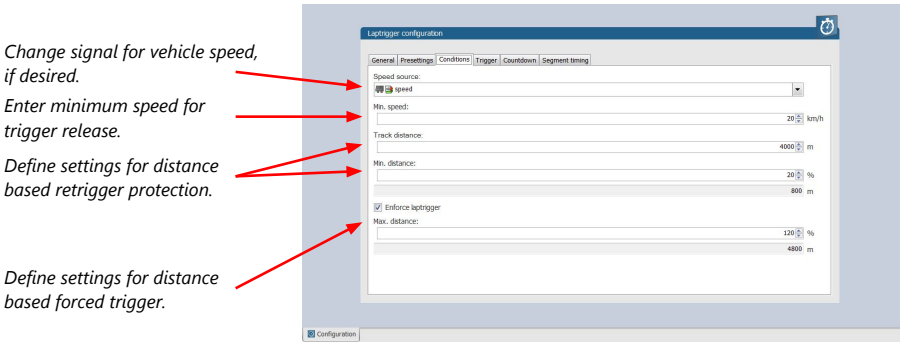
3. Click ‘Finish’ to complete the operation.

   A pre-configured lap trigger window opens.



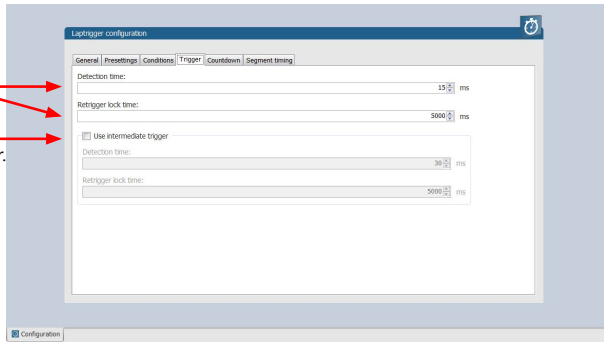
a) Change signal device, if desired.
b) Change signal channel, if desired.
c) Choose signal threshold. See chapter 'Electrical trigger signal' for details.
d) Define threshold of input channel signal when trigger is released.
Only possible, if no digital source is selected as signal source.
e) Define presettings for trigger. See chapter 'Lap trigger presettings' for details.
f) Define condition settings; change signal for vehicle speed, define speed settings.
See chapter 'Distance based retrigger protection' and 'Distance based forced trigger' for details.
g) Define settings for main trigger. See chapter 'Lap timing' for details.
h) Define settings for counddown timer. See chapter 'Countdown timer' for details.
i) Define settings for sub trigger. See chapter 'Segment timing' for details.
j) Define settings for a GPS lap trigger. See chapter 'GPS lap trigger' for details. Only applicable if the signal source is set to 'GPS lap trigger'.



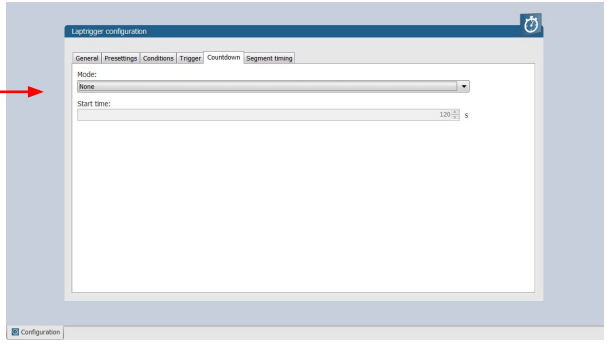


Define settings for lap timing (main trigger).

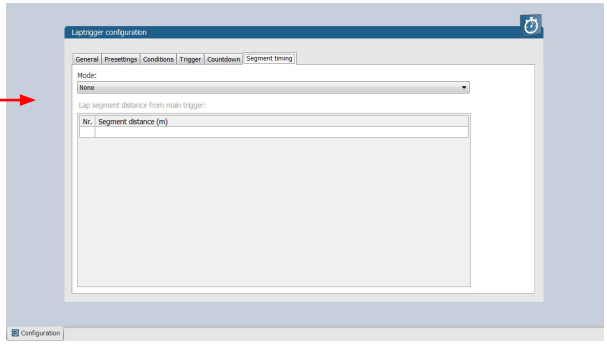
Define settings for sub trigger. Not applicable with a GPS lap trigger.



Define settings for countdown timer.



Define settings for segment timing.

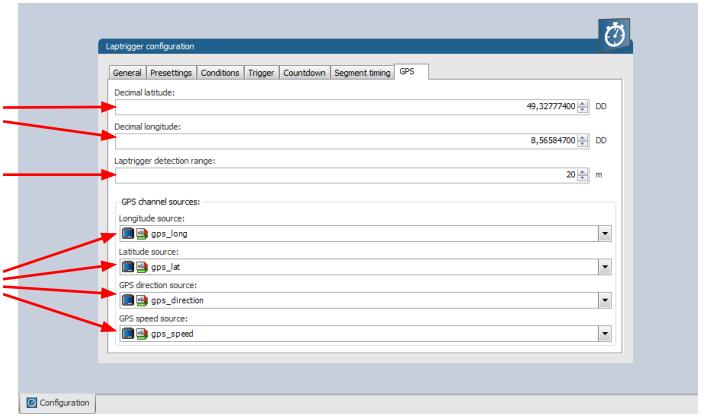


Only applicable for a GPS Laptrigger

Define the latitude and longitude of the GPS detection point.

Define the detection range around the detection point.

Define the channel sources for Longitude, Latitude, Direction and Speed.

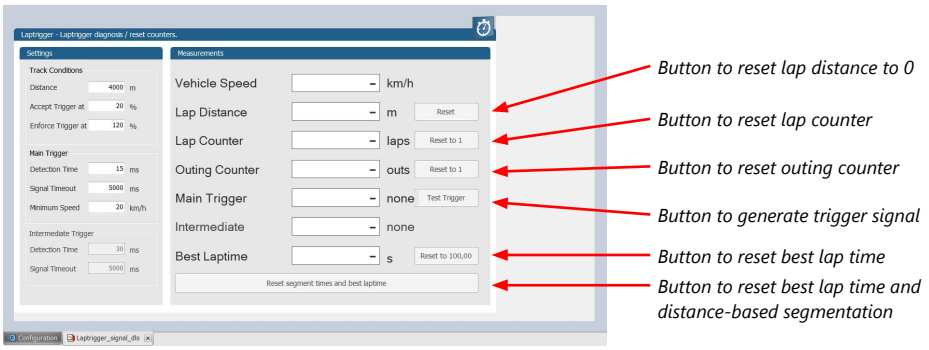


20.1.6 Lap trigger channel diagnosis/counter reset

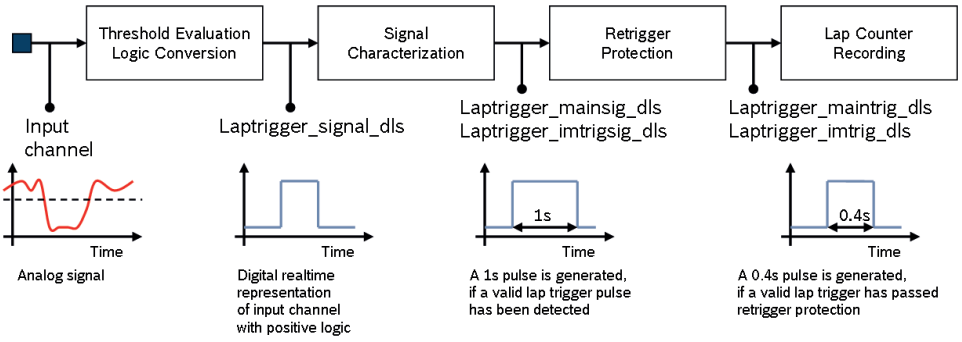
To display a quick lap trigger channel diagnosis and to reset counters use the diagnosis page in RaceCon. Any 'Laptrigger_xxx' channel can be displayed.

Double-click on any 'Laptrigger_xxx' channel in the Data Area. Example: 'laptrigger_lap-dist_dls'

A diagnosis window opens in Main Area.

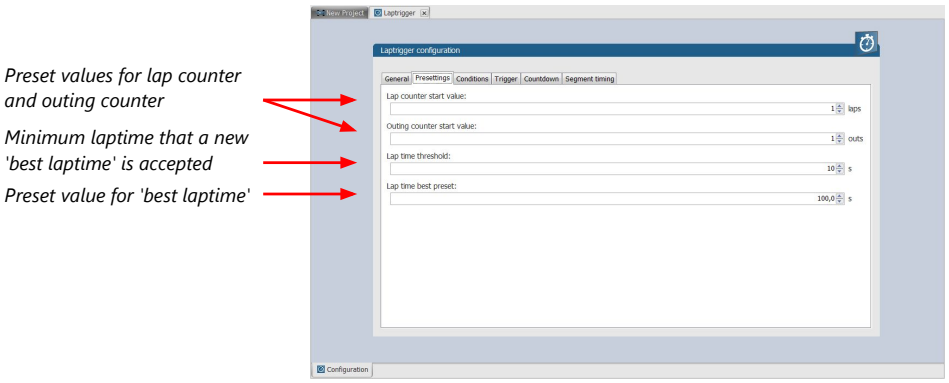


Lap trigger diagnosis scheme

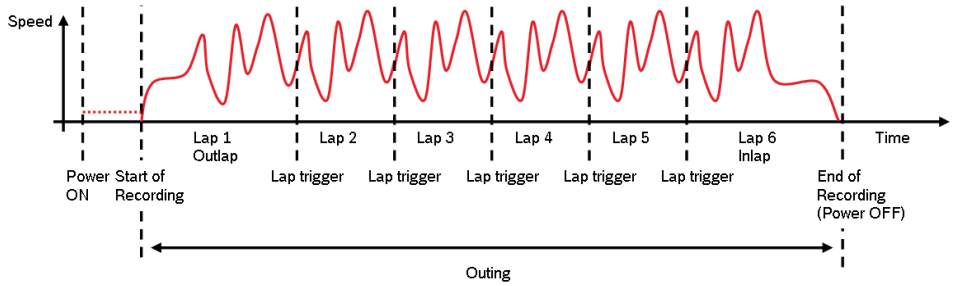


20.1.7 Lap trigger presettings

When the reset buttons on the diagnosis page are activated, these values are used.



20.2 Counting outing/laps/fragments



Functionality

- Power ON: system + measurement is initialized but not yet started
- Global start condition fulfilled: recording starts
- Reception of valid lap trigger: recording of lap completed, new lap starts
- Power OFF or Global start condition not fulfilled: recording of lap completed, system shutdown

The system is counting:

Outing:

- The outing counter is incremented with each power cycle when at least one valid lap (not by forced lap trigger) was recorded

Lap:

- Leaving the pits to lap trigger
- Lap trigger to lap trigger
- Enforced lap trigger (see Distance based forced trigger [▶ 167])

Fragment:

- Test bench operation
- Power cycle on track or box (e.g. engine stalled)
- File fragmentation size [MB], time [sec]

Channels for display

To display counters use the following channels:

Channel	Function
Laptrigger_outcnt_dls	Outing counter
Laptrigger_lapctr_dls	Lap counter
Fractr	Fragment counter

Counting in WinDarab

To automatically name recorded files use filename templates in WinDarab dialog:

Filename template	Function
[outing]	Value of outing counter
[lap]	Value of lap counter
[fragment]	Value of fragment counter

[###03] indicates: ‘always use 3 digits with leading zeros’.

20.3 Lap timing

There are different possibilities to adjust the lap trigger to the timing situation.

The detection time defines the minimum time the input signal changes its state. E.g. a low active signal needs to be below the threshold for min. 15 ms to release the trigger.

Channels for display

To display lap times use the following channels:

Channel	Function
Laptrigger_lapctr_dls	Number of completed laps
Laptrigger_laptime_dls	Running laptime
Laptrigger_laptime_best_dls	Laptime of best lap
Laptrigger_laptimeold_dls	Laptime of last lap completed
Laptrigger_laptimeseg_dls	Segment time of last segment
Laptrigger_lapctr_dls	Number of completed laps

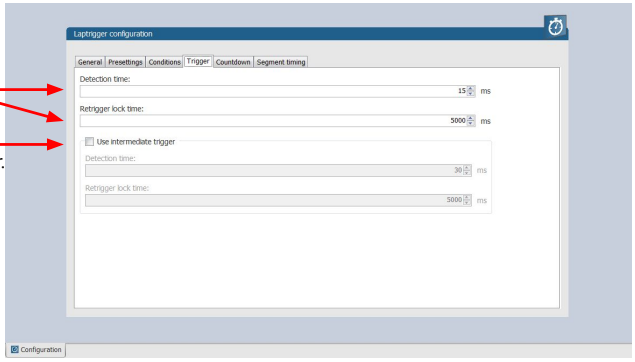
20.3.1 Time based retrigger protection

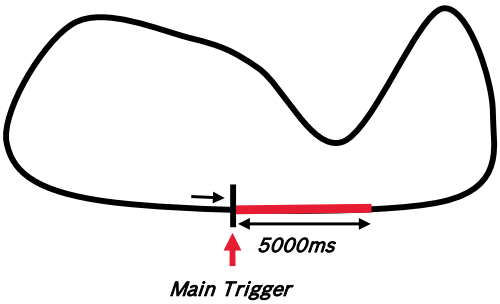
Trigger is locked for 5 s after main trigger was received.

To deactivate time based retrigger protection, set ‘Retrigger lock time’ to 0 ms.

Define settings for lap timing (main trigger).

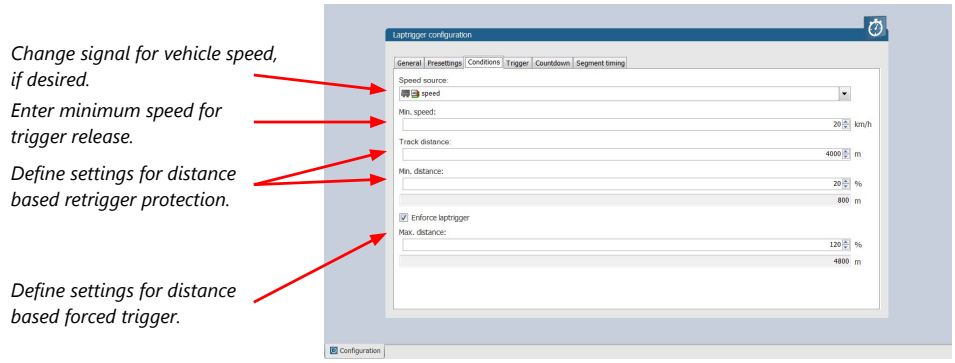
Define settings for sub trigger. Not applicable with a GPS lap trigger.

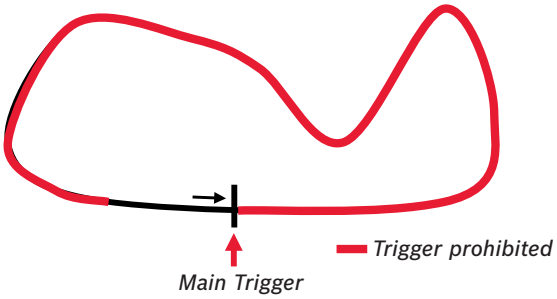




20.3.2 Distance based retrigger protection

Trigger is locked until configured min distance (i.e. 80 % → 3200 m) of track distance (i.e. 4000 m) has been covered. To deactivate distance based retrigger protection, set min distance to 0 %.

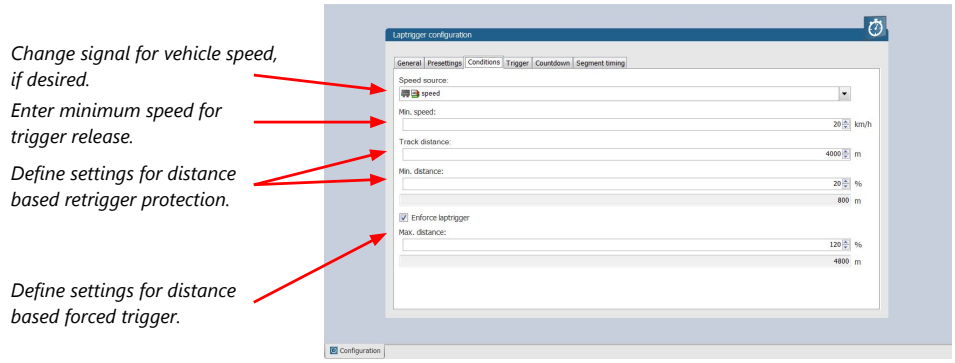




20.3.3 Distance based forced trigger

After a missed main trigger, a forced trigger is inserted, if the configured max. distance (i.e. 120 % → 4800 m) of the track distance (i.e. 4000 m) has been reached. In this case, the channel 'Laptrigger_distlap_dls' starts at the delta between the max. distance and the track distance (i.e. 800 m).

To deactivate distance based forced triggers, uncheck box.



20.4 Segment timing

Segment timing is the calculation of elapsed time for parts of laps (segments).

Segments are defined:

- based on sub-trigger signals (additional transmitters)
- based on distance travelled

Times for segments are compared to:

- Last lap completed
- Fastest lap

Channels for display

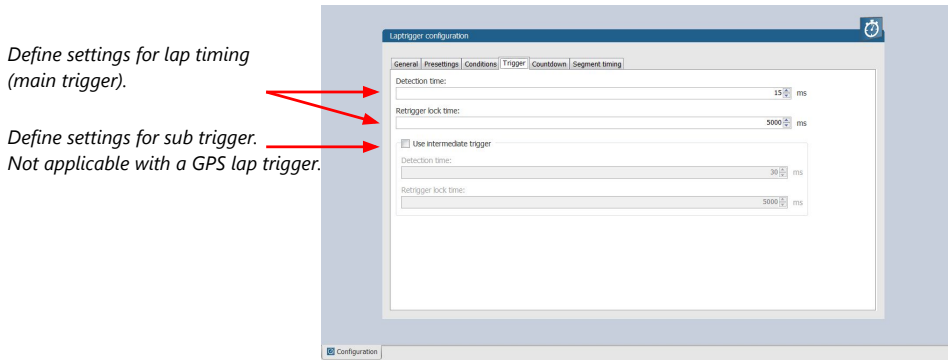
To display segment times use the following channels:

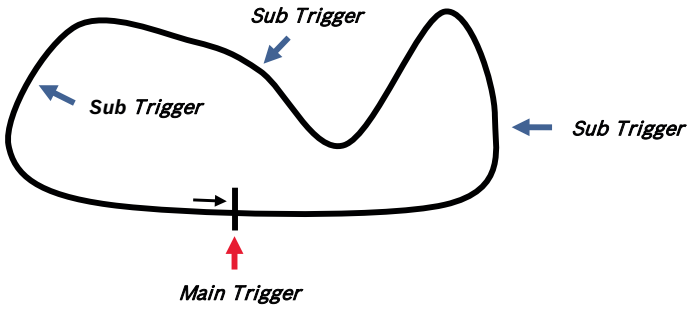
Channel	Function
Laptrigger_lapdiff	Time difference between finished lap and last lap
Laptrigger_lapdiffb	Time difference between finished lap and best lap
Laptrigger_lapseg_dlast	Difference of lap segment time compared to last lap
Laptrigger_lapseg_dbest	Difference of lap segment time compared to best lap

20.4.1 Sub trigger mode

Using main trigger (20 ms pulse) at Start-Finish-Line. 3 sub triggers (40 ms pulse) positioned at 1,000 m, 2,000 m and 3,000 m.

To deactivate sub trigger mode uncheck box.





The sub trigger mode cannot be used with the GPS lap trigger.

20.4.2 Distance mode

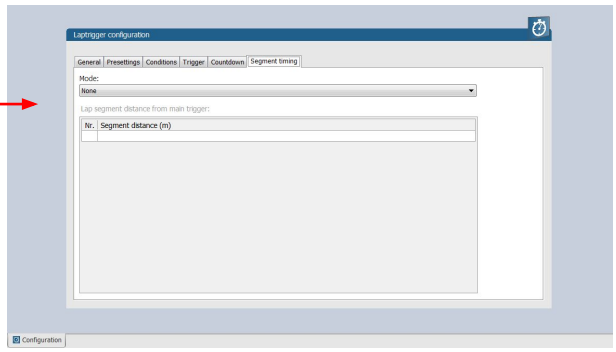
Using main trigger (20 ms pulse) at Start-Finish-Line.

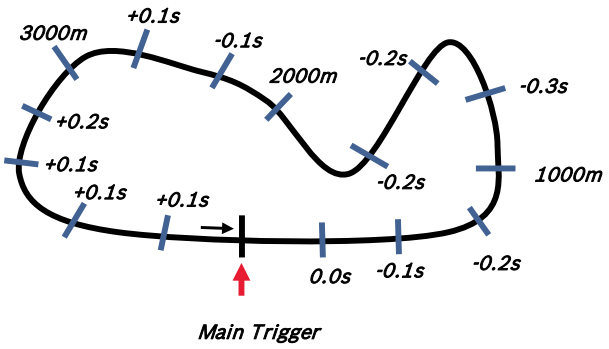
Set ‘Mode’ to ‘Distance’ and enter desired segment distances.

Segment time is automatically calculated at each segment. Time difference to last lap and fastest lap is automatically calculated at each segment.

To deactivate distance mode set ‘Mode’ to ‘None’.

Define settings for segment timing.



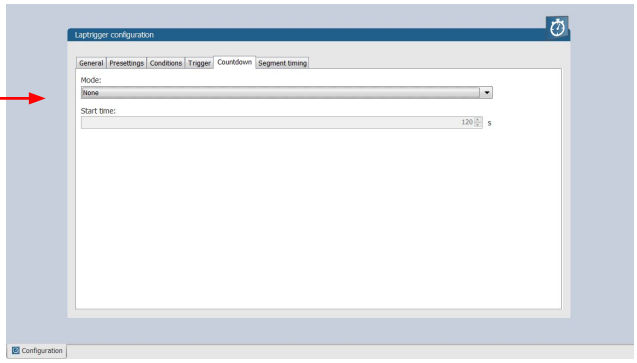


20.5 Countdown timer

Some race classes require a minimum time spent in the pits. An additional lap trigger Tx is configured as a segment trigger positioned at pit entry. The trigger signal starts a timer countdown.

The current value of the timer is stored in the variable **Laptrigger_cntdown_dls** which can be displayed.

Define settings for countdown timer.

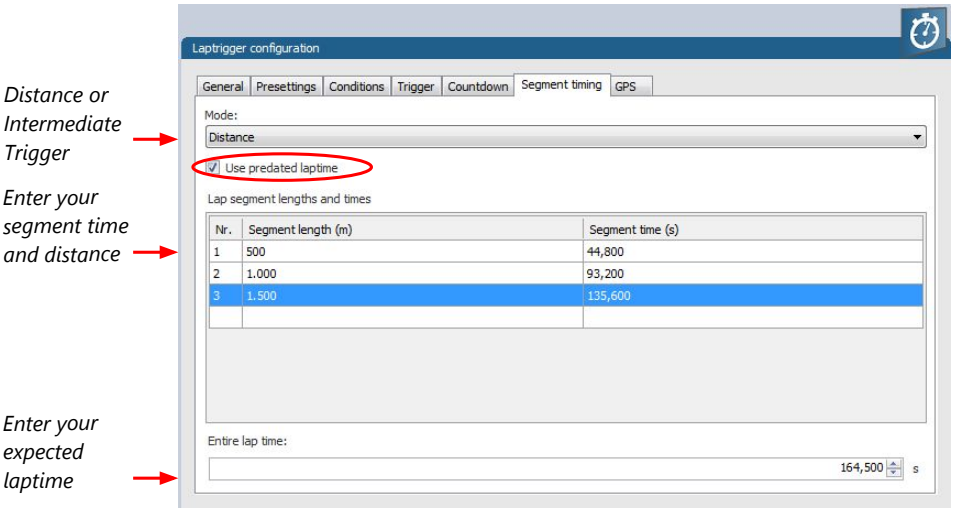


21 Predated Laptime

The predated laptime function allows to compare the current lap- and segment time with the predated time of an expected lap. Additional the function can estimate the laptime of the current lap. This functionality is integrated in the laptrigger module in RaceCon.

21.1 Setting up the predated laptime

To use the predated laptime function you need to set up a laptrigger as described in the chapter Lap Trigger [▶ 158]. Under the ribbon “Segment timing” you need to choose your segmentation mode which can either be distance or intermediate trigger based.



For the distance mode you need to check on an old lap or estimate how long it takes to travel the segment distance. Please enter those values into input field. The values can also be copied and pasted to the input field from an excel sheet as a normal text. In the intermediate trigger you just need to set the expected time the driver takes to reach the segment trigger.

NOTICE

Please note that the segment time and length is always measured from the start line or where the main lap trigger is set.

21.2 Functionality and channel outputs

Following output channels are generated by the predated laptime function.

Laptrigger_lapdiff_pred_dls	Laptime difference between the predated and the last laptime
Laptrigger_lapsegdiffpred_dls	lagseg difference between the last segment and the predated segment.
Laptrigger_Lapcurrpred_dls	Estimated laptime of the current lap, based on the predated laptime and the predated segment deviations

The channel Laptrigger_lapdiff_pred_dls is updated as soon as the main lap trigger is received. Both other channels are updated as soon as the next segment distance is travelled or the next intermediate trigger is received.

22 GPS Sensor

22.1 GPS (Global Positioning System)

Space-based global navigation satellite system.

GPS provides positioning, navigation, and timing services to worldwide users.

GPS receiver (sensor) gives digital information about position (longitude, latitude, height), ground speed, course, and status.

Two types of GPS receivers:

CAN output -> Read in messages via CAN Input of DDU 10 (not covered here)

Serial output -> Read in messages via RS232 Interface of DDU 10 (serial interface 2)

22.1.1 Serial interface characterization

Serial Interface is characterized by:

Voltage levels: RS232 is standard (+/-12 V), UART (0 V/ 5 V) needs level shifter

Baud rate: 9,600 is standard for GPS, DDU 10 supports 1,200 to 115,200 baud. GPS Rx interface baud rate must match DDU 10 interface baud rate. DDU 10 Baud rate can be set with the 'GPS_BAUDRATE' characteristic Data format: DDU 10 expects 8 data bits, no parity bit, 1 stop bit (8N1)

22.2 Protocol

DDU 10 expects NMEA Protocol (ASCII).

The following messages are decoded:

Message	Function
GGA	GPS fix information
GSA	Overall satellite data
GSV	Detailed satellite data
RMC	Recommended minimum data for GPS
VTG	Vector track and speed over the ground

On most GPS sensors, these messages are activated in the default configuration.

22.3 Sensor recommendation

The system has been tested with the Navilock NL-8004P MD6 Serial PPS Multi GNSS Receiver. This sensor is based on a U-Blox 8 chipset and is fully configurable with the Navilock "U-Center" software. To use this sensor with Bosch Motorsport components the transfer rate, the satellite system and the update rate need to be reconfigured. More information about the configuration can be found in the Appendix.

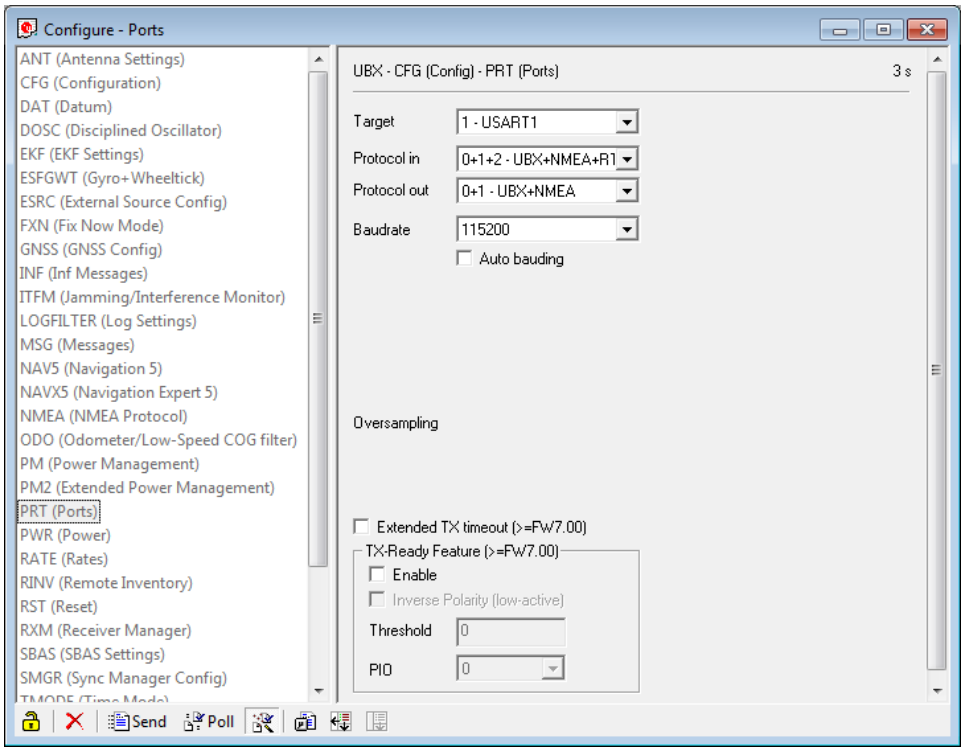
22.3.1 Configuration of the recommended Navilock NL-8004P MD6 Serial PPS Multi GNSS Receiver

For the sensor configuration, the sensor needs to be connected to the Navilock software “U-Center” which is available from Navilock free of charge. Navilock offers a USB connection cable for the sensor.

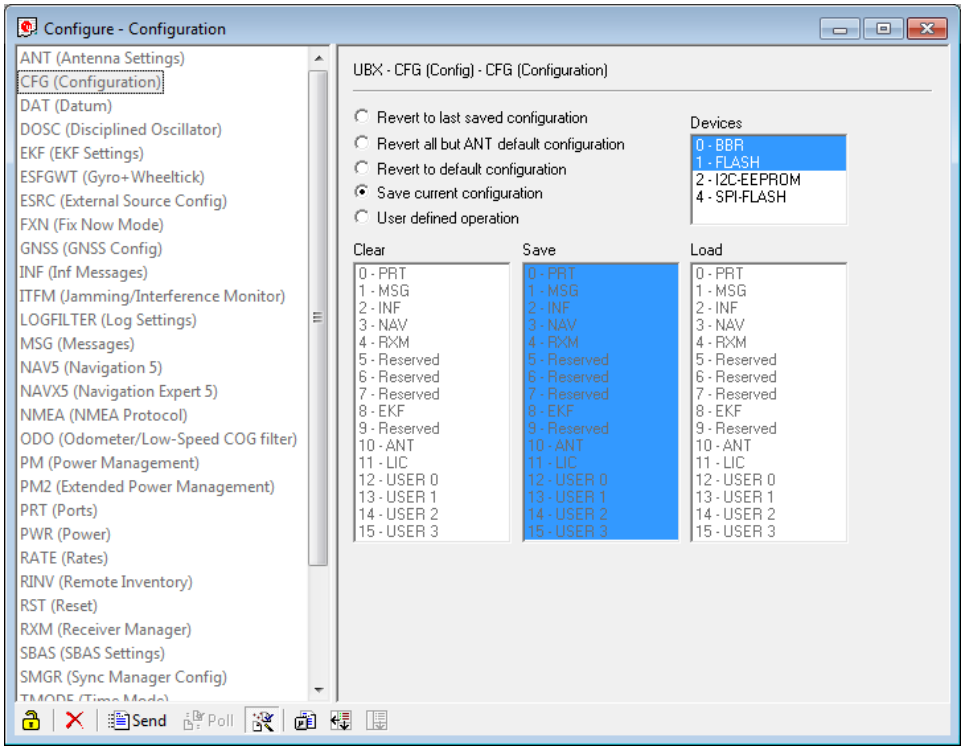
In “U-Center” click **“View”** – **“Configuration View”** to start the configuration. The following 3 points have to be changed:

Transfer Rate

- Click on “**PRT (Ports)”.**
- Change the baud rate to a fixed value, this value needs to meet the setting of RaceCon. For a good signal quality we recommend 115,200 baud.
- Click on **“Send”** to store the new setting in “U-Center”.

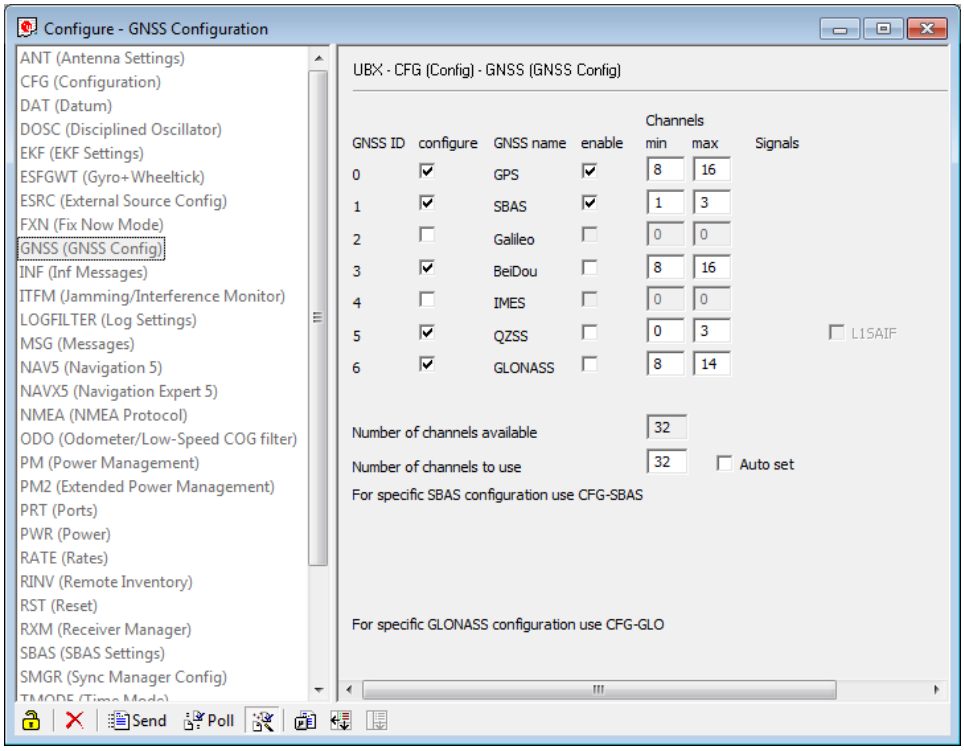


- Click on **“CFG (Configuration)”.**
- Click on **“Send”** to save the new setting on the sensor.



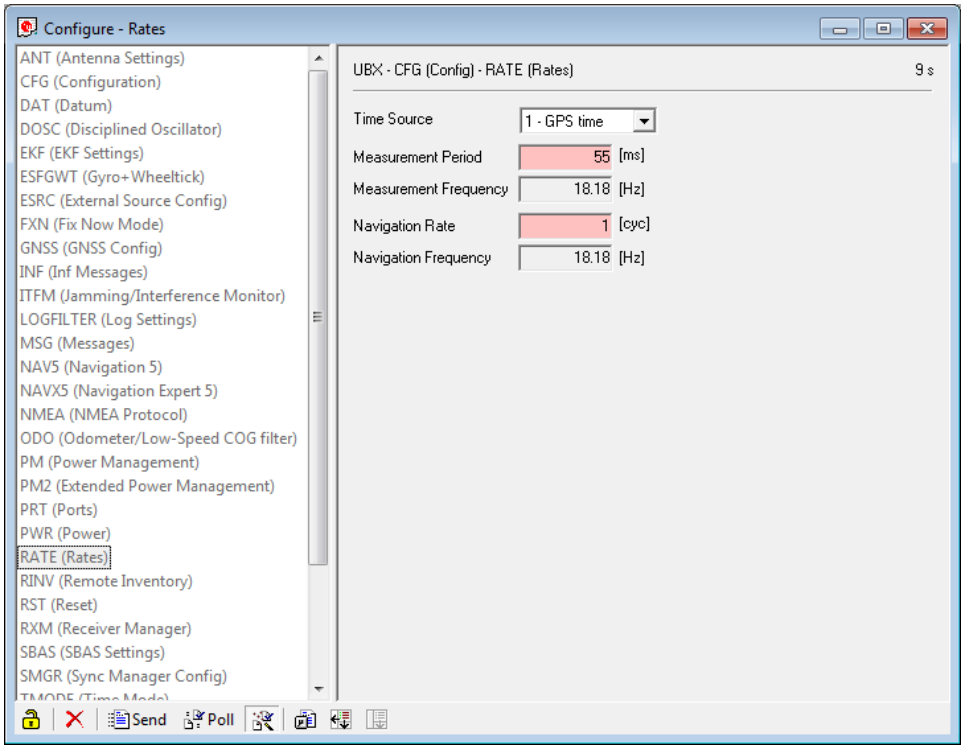
Satellite System

- Click on **“GNSS (GNSS Config)”.**
- Set the ticks as shown in the following picture.
- Click on “**Send**” to store the new setting in “U-Center”.
- As during configuration step 1, click on “**CFG (Configuration)**”.
- Click on“**Send**” to save the new setting on the sensor.



Update Rate

- Click on **“RATE (Rates)”.**
- Change the “Measurement Period” to 55 ms.
- Change the “Navigation Rate” to 1 cyc.
- Values which lead to a lower frequency will lower the precision of the sensor, we recommend the mentioned values.
- Click on “**Send**” to store the new setting in “U-Center”.
- As during configuration step 1, click on “**CFG (Configuration)**”.
- Click on“**Send**” to save the new setting on the sensor.



NOTICE

Sensor needs reception for visible signal.
It takes time to start the sensor.

22.4 Measurement labels

The decoded NMEA messages are copied to these DDU 10 measurement labels.

Measurement label	Function
gps_PDOP	Position Dilution Of Precision
gps_HDOP	Horizontal Dilution Of Precision
gps_VDOP	Vertical Dilution Of Precision
gps_lat	Latitude +/- [degree]
gps_long	Longitude +/- [degree]
gps_elv	Antenna altitude above/below mean sea level (geoid) in meters

Measurement label	Function
gps_speed	Speed over the ground in kilometers/hour
gps_direction	Track angle in degrees
gps_declination	Magnetic variation degrees (Easterly var. subtracts from true course)
gps_year	Years since 1900
gps_mon	Months since January - [0,11]
gps_day	Day of the month - [1,31]
gps_hour	Hours since midnight - [0,23]
gps_min	Minutes after the hour - [0,59]
gps_sec	Seconds after the minute - [0,59]
gps_hsec	Hundredth part of second - [0,99]
gps_smask	Bit mask over received NMEA sentences (Bit 0 = GGA, Bit 1 = GSA, Bit 2 =GSV, Bit 3 = RMC, Bit 4 =VTG) within last second.
gps_sig	GPS quality indicator (0 = Invalid; 1 = Fix; 2 = Differential, 3 = Sensitive)
gps_fix	Operating mode, used for navigation (1 = Fix not available; 2 = 2D; 3 = 3D)

These measurement labels are arrays, where the indexed element points to the same satellite.

(E.g. gps_info_satsigstrength[3] tells the receiving signal strength of satellite 3. Satellite 3 has the SAT-ID given in gps_info_satid[3])

Measurement label	Function
gps_info_satid[ ]	Satellite PRN number
gps_info_satinuse[ ]	Used in position fix
gps_info_satelevation[ ]	Elevation in degrees, 90 maximum
gps_info_satazimuth[ ]	Azimuth, degrees from true north, 000 to 359
gps_info_satsigstrength[ ]	Signal, 00-99 dB

22.5 GPS troubleshooting

Electrical

Is the transmitter signal of the GPS sensor connected to the receiver pin of serial interface 2 of the DDU 10?

Is the GPS sensor powered up?

Does the GPS sensor deliver RS232 signal levels?

Is the sensor connected to the „sensor ground“ of the device?

Interface

Do the baud rates of the GPS sensor and the DDU 10 match?

Is the GPS sensor set up for 8N1 transmission parameters?

Is the GPS sensor set up for NMEA messages?

Are the GGA, VTG, RMC messages activated?

GPS sensor start-up

Does the GPS sensor ‘view‘ the sky?

Did the GPS sensor complete its initial start-up procedure? This may take up to 20 min.

A correct reception is indicated when ‘gps_fix‘ is showing ‘3D Fix‘.

GPS sensor values are frozen

Does the sensor has lost its reception? The old values will be kept if the reception is lost. The gps_smask channel shows which NMEA sentence is received.

23 Firmware

23.1 Firmware and configuration

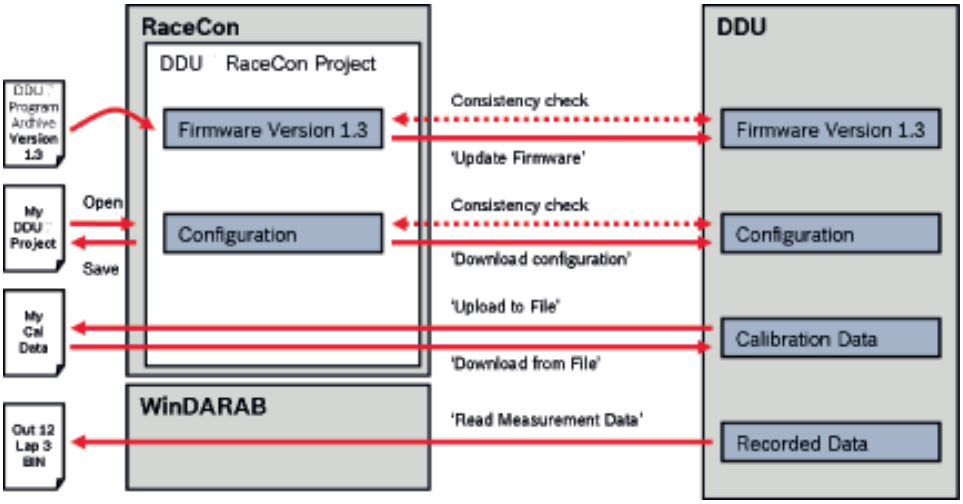
DDU 10 holds 4 types of data:

Firmware: the software (PST program file) of the DDU 10.

Configuration: the configuration of Input channels, CAN I/O, PWM, display configuration, recording + telemetry configuration.

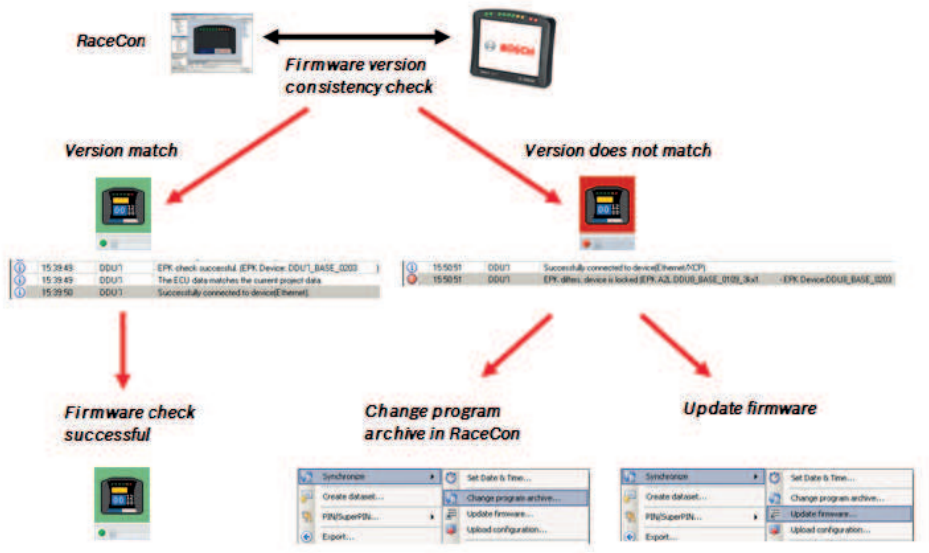
Calibration data: Characteristic curves and offsets created by online calibration at the vehicle.

Recorded data: Measurement data recorded during vehicle operation.



23.2 Firmware update

The scheme shows the process during each connection between RaceCon and DDU 10.

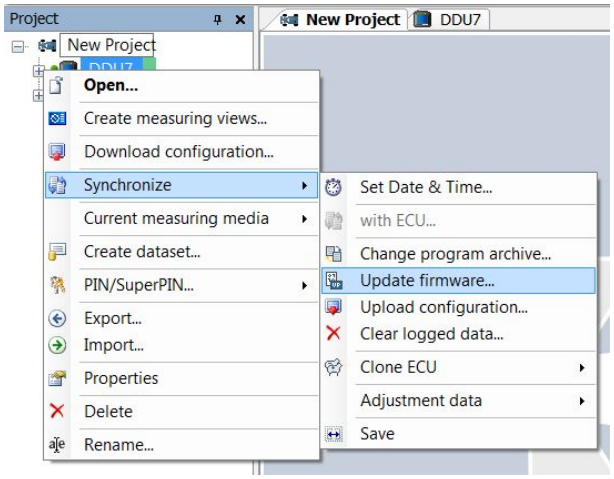


23.2.1 Performing the firmware update

Firmware update is only possible if the DDU 10 is connected to RaceCon.

The configuration of Input channels, CAN I/O, display, recording + telemetry will not be changed.

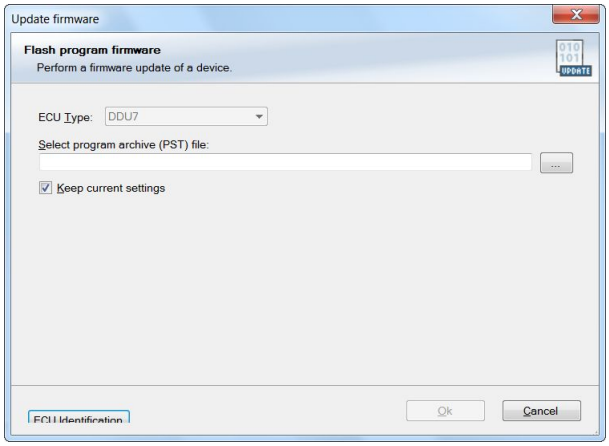
1. In the DDU 10 Project Tree, right-click on ‘DDU 10’ and choose ‘Synchronize’ then ‘Update firmware’.



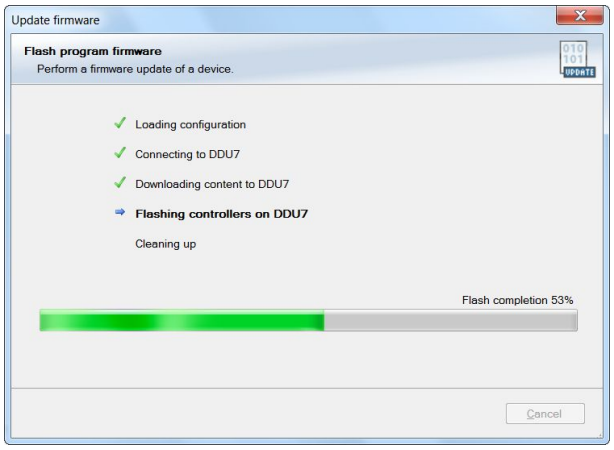
A pop-up menu opens.

2. Select the destination of the firmware archive (PST).

You can find the latest firmware for the device at the Bosch Motorsport homepage.

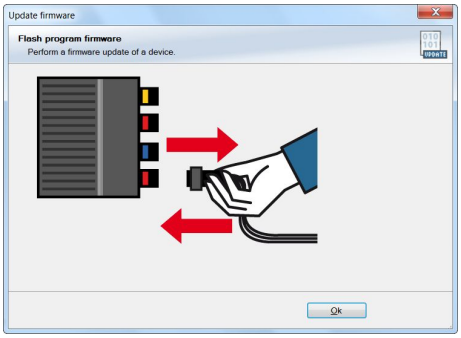


3. Click ‘OK’ when done.
The firmware update starts. The DDU 10 displays the message ‘Updating firmware’. Do not switch off the car’s ignition or interrupt the power supply of the DDU 10!



When the firmware update is complete, the DDU 10 displays the message ‘Updating firmware finished. Do a powercycle.’

Switch the car’s ignition off and on again to cycle the power of the DDU 10.

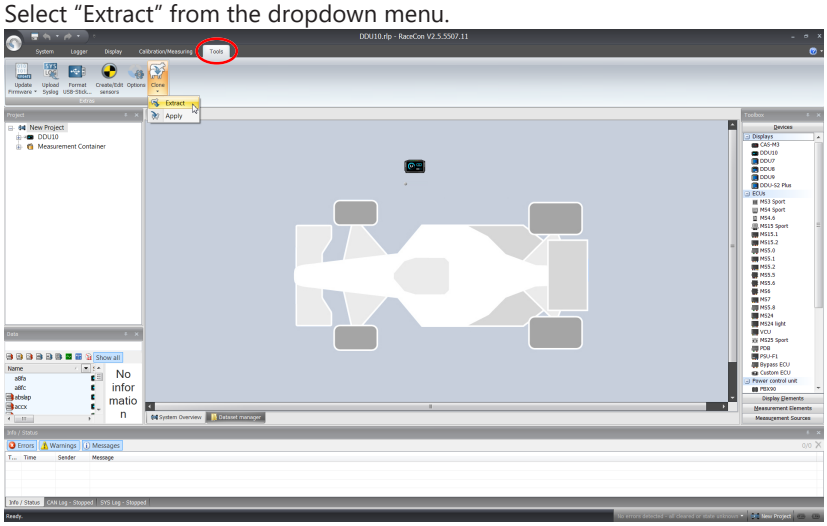


24 Cloning the Unit

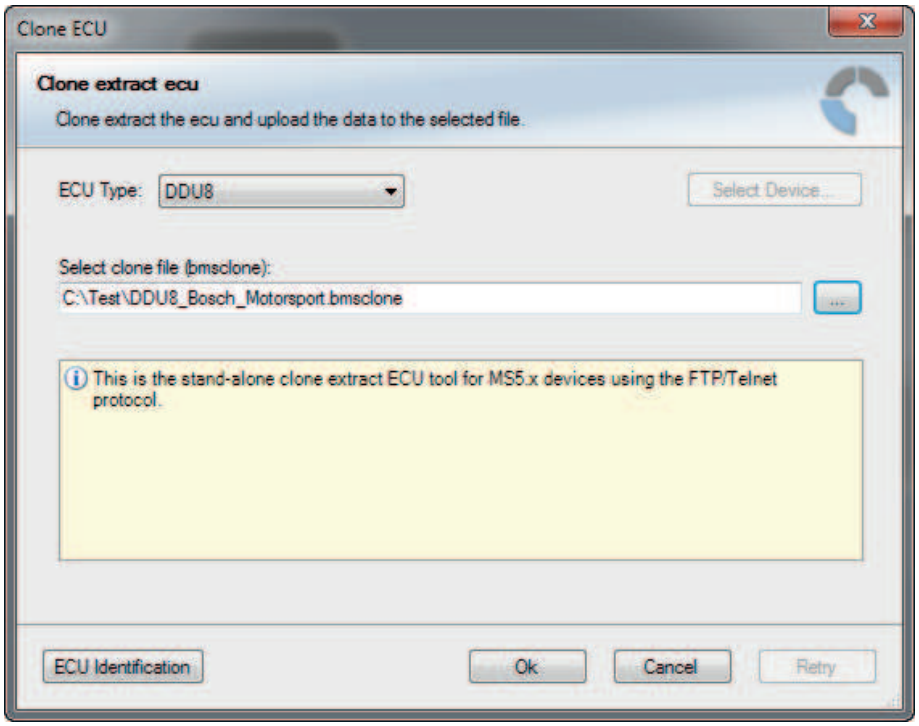
To replace a DDU 10 by another device, it is possible to clone it. A clone is a 1:1 copy of a device. This can be useful for copying specific data, like sensor-offset calibration to a spare unit for a specific car.

Creating a clone file

1. Open the 'Tools' window and click on the 'Clone' button in the 'Extras' menu.
2. Select "Extract" from the dropdown menu.



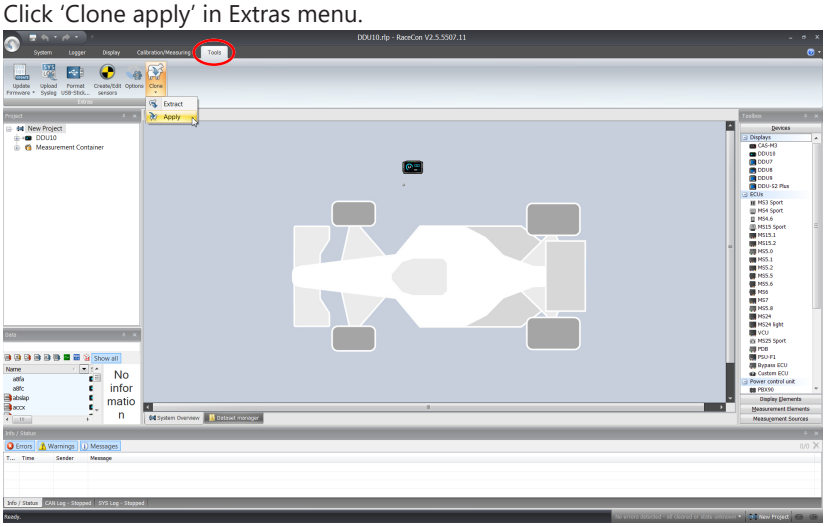
3. Choose the hardware device, which should be cloned.
4. Define destination and filename.



5. Click 'OK' to start procedure.

Applying a clone file to a device

1. Click ‘Clone apply’ in Extras menu.



2. Choose clone file.
3. Click ‘Ok’.

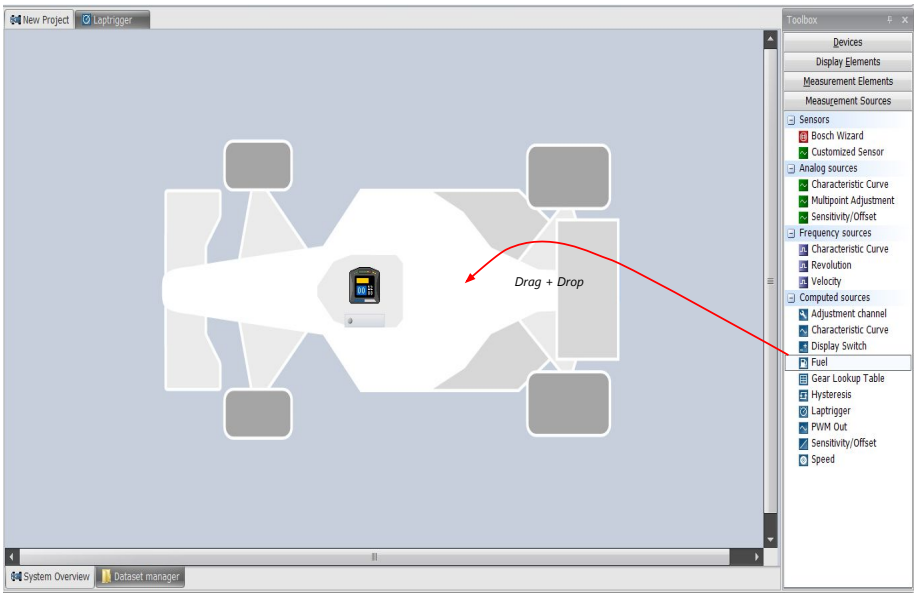
Please remember that following properties are not stored into the clone:

- Lifetime of device
- Serial number
- Upgrade features

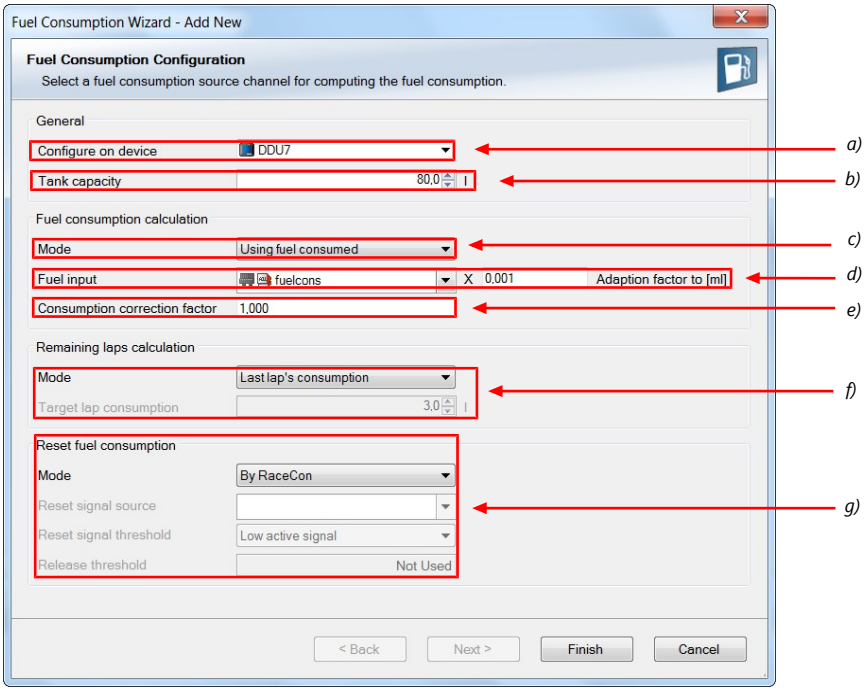
25 Fuel Consumption Calculation

25.1 Setting up fuel consumption calculation and tank management

1. Select ‘Measurement Sources’ in Toolbox.
2. Drag ‘Fuel’ element and drop it on the vehicle in System Overview. Do not drop it on the DDU 10!



A ‘fuel consumption wizard‘ opens.



a) Change device for fuel calculation, if desired.
b) Enter tank capacity of vehicle.
c) Choose calculation mode:
• using fuel consumed (summed-up fuel consumption)
• using fuel flow rate (momentary fuel consumption)
d) Choose input channel and enter adaption factor. Use adaption factor to adapt value of input channel to:
• 1ml per inc for summed-up fuel consumption
• 1ml/s per inc for momentary consumption
e) Enter factor to correct calculated consumption in device vs. 'real' consumption of vehicle, if required.
f) Choose method to calculate remaining laps with fuel in tank, if desired:
• using fuel consumption of last lap completed
• using target lap consumption (entered in the field 'Target lap consumption')
g) Choose values to initiate a reset of fuel consumption, if desired:
• Manually using RaceCon
• On 'power down' (assuming that the tank is filled each time the ignition is turned off)
• By signal source as input channel (e.g. a switch connected to input pin)

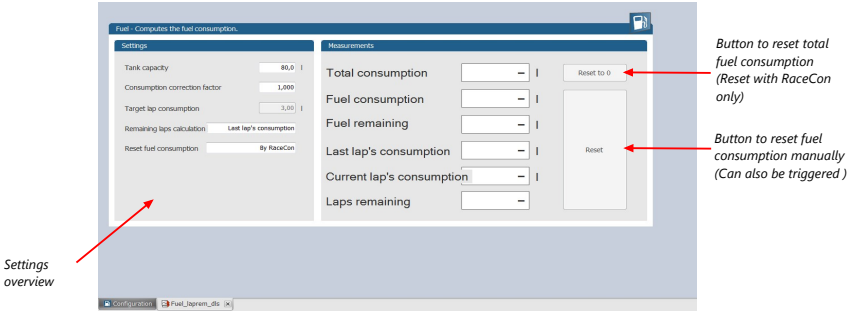
3. Press 'Finish' when done.

25.2 Fuel consumption diagnosis/counter reset

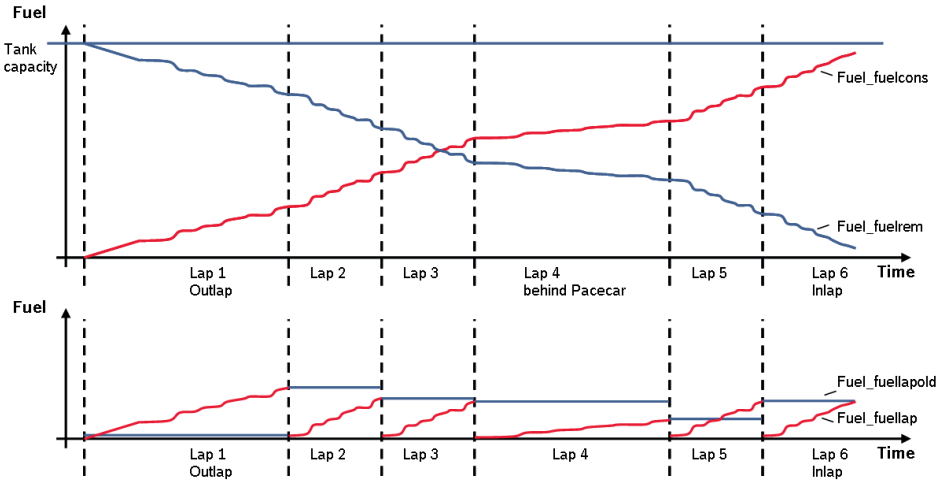
To display a fuel consumption diagnosis and to reset counters, use the diagnosis page in RaceCon.

Double-click on any 'fuel_xxx' channel in channel list.

A diagnosis window opens in Main Area.



25.3 Example



Measurement label	Function
Fuel_fuelcons_dls	Running fuel consumption, starting at '0'
Fuel_fuelrem_dls	Remaining fuel in tank, starting at tank capacity
Fuel_fuellap_dls	Fuel consumption for current lap, starting at '0'
Fuel_fuellapold_dls	Fuel consumption of last lap completed
Fuel_laprem_dls	Remaining laps with fuel in tank

26 RaceCon Shortcuts

The table shows important shortcuts simplify controlling the DDU 10 in RaceCon.

Shortcut	Function
General navigation	
F1	Open RaceCon help
F2	Rename selected object
F3	Select Data Area
F4	Select Project Tree
F5	-
F6	Start the data comparison
F7	Start dataset manager
F8	Toggle WP/RP
F9	Start measurement
CTRL + F9	Start recording
F10 or Alt	Go to menu bar
F11	Toggle display to fullscreen ‘Race Mode’
F12	Enlarge main screen
CTRL + Tab	Switch between opened windows
Project Tree	
Plus (+) at numeric pad or right cursor	Expand selected node
Minus (-) at numeric pad or left cursor	Close selected node
Star (*) at numeric pad	Open all nodes
DEL	Delete seleted object
Display page, measurement page	
Cursor	Move selected display element one grid unit in chosen direction
SHIFT + cursor	Enlarge/reduce selected display element one grid unit
Tab	Switch between display elements

Bosch Engineering GmbH
Motorsport
Robert-Bosch-Allee 1
74232 Abstatt
Germany
www.bosch-motorsport.com