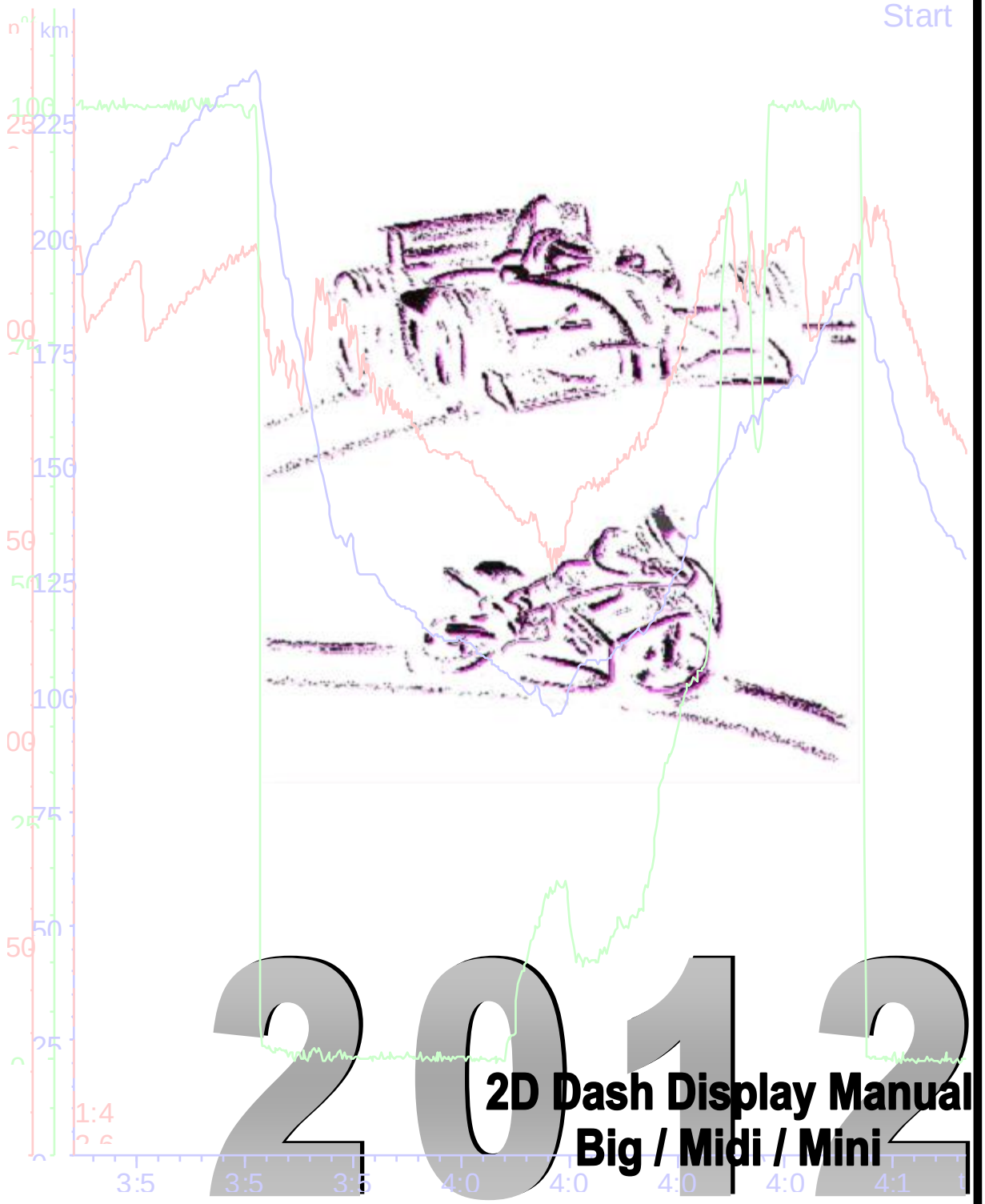
- English -

2d-datarecording.com



2D Dash Display Manual
Big / Midi / Mini

Art.No.: AC-DOC_MultiDash_Docu-000 modified 04.01.2013 MC

Table of contents

Preface

The method for creation of settings for the 2D BigDash, MidiDash and MiniDash displays is quite similar. All three 2D dash displays share many common features, and can therefore be summarised together in one manual. Any additional functionalities that are unique to each Dash are also considered inside "item specific" areas of the manual.

Additional information can be found on the 2D homepage and downloaded as follows:

- Enter 2D homepage (www.2D-datarecording.com)
- Select <2D Products>, then <Displays>
- Inside the <Display Units> page, further information and documentation is available for the 2D range of dashboards
- Product user <manual> and <data sheet> are available for download

Symbols used in the text

In the paragraphs highlighted with this symbol, you will find tips and practical advice to work with the 2D Software.

In the paragraphs highlighted with this symbol, you will find additional information and it is very important that you follow the instructions given.

2D Debus & Diebold
Meßsysteme GmbH
Alte Karlsruher Straße 8
76227 Karlsruhe
Tel.: +49(0)721 94485-0
Fax: +49(0)721 94485-29
EMAIL: mail@2D-datarecording.com
Homepage: http://www.2D-Datarecording.com
Homepage: http://www.2D-Kit-System.com

1 Properties of BigDash, MidiDash and MiniDash

1.1 Specification Summary for the 2D DASH Display

- A Programmable Multi Function Display Unit
- Full CAN-line functionality (2x BigDash, 1x MidiDash, 1x MiniDash)
- Programmable via 1Mbit CAN-bus
- High contrast Dot Matrix Display visualises information with high clarity
- User Configurable Shift Lights and Warning Lights
- Adjustable Screen Contrast and Brightness, Adjustable LED Brightness
- Temperature controlled contrast
- Online calculation channels
- Operating temperature range -20 to 85°C
- Usable as standalone display unit
- Dynamically Adjustable LED and Screen Brightness for day-to-night conditions (BigDash)

1.2 General Properties of the BigDash, MidiDash and MiniDash

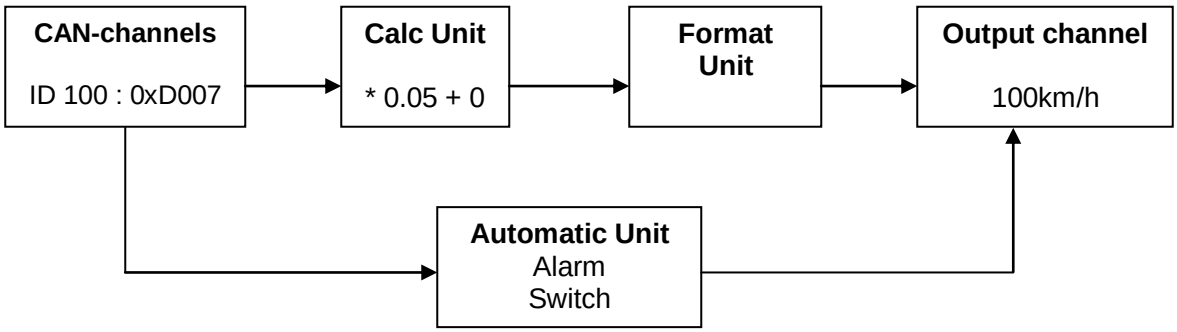
The following explanations describe the system structure and some capabilities of the 2D display modules. For specific information about your Dash (BigDash, MidiDash or MiniDash), see the relevant section later in this Manual.

CAN Bus

The 2D Dash can receive up to 32 CAN-channels. Data sent to the Dash via CAN Bus can be used in many different ways. With the 2D Display you can visualize all data received via the CAN Bus. Every value you want to display can be converted into physical values and displayed in your defined format using a freely programmable conversion formula. CAN channels received by the Dash can be easily programmed to display formatted physical values in various screen positions on each of the 3 Display Pages. All Dash settings are saved permanently by the internal Flash memory.

CAN Data Flowchart

Each CAN Data Channel contains a package of information including the complete CAN Bus Identification (identifier, position and data size (byte, word, double word)), data value, physical value formula, formatting and alarm borders.



EVENT

Event Channels offer special functions like **Alarm**, **Switch Page** features and the **Laptime** function.

The **Alarm** function can be programmed according to a channel value (or a combination of channels), High and Low Thresholds can be predefined.

The **Switch** function enables the display to change page according to the value of a "trigger channel", this can be used to simplify displayed data for when the vehicle exits the pit box.

The **Laptime** function enables internal calculation of Laptime according to an input "trigger channel". The Laptime trigger channel can be a CAN channel, an analog input channel, or a GPS channel from a connected 2D GPS device.

CAN Monitor

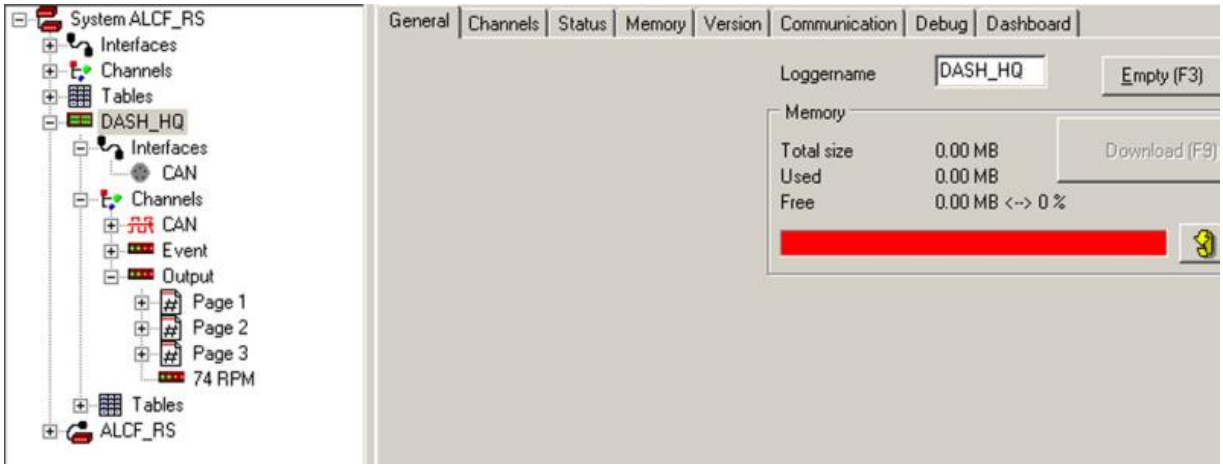
The *CAN Monitor* (optional) enables troubleshooting on the CAN Bus and it displays the following information:

- number of CAN Id's that are sent
- single CAN ID's with corresponding data (in Hex)
- data length for each ID
- counter for data packet

2 Operating the 2D DASH - Fundamentals

The 2D system must be powered on and connected with a PC via USB cable or Serial cable.

After the logger or Dash system is connected to the PC, all commands made to the PC are forwarded to each 2D module that is connected via the CAN Bus. The 2D communication software **WinIt** is used to make changes to the 2D Logger or Dash system. WinIt displays the 2D Dash inside the "System Tree", as shown below.



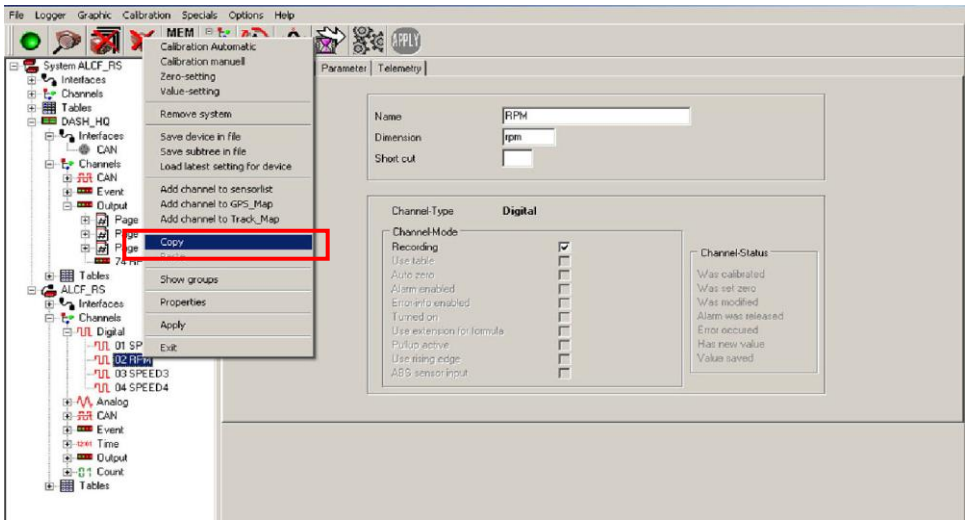
All devices which use the same connection to communicate with the PC are displayed as ***<System>***. In the figure above, the logger (called ***<ALCF_RS>***) is connected to the PC and also the display (called ***<DASH_HQ>***), which communicates with the logger and the PC via the CAN Bus.

2.1 How to copy a reference channel

To copy the reference channel, proceed as follows:

- In the tree view choose the logger <*ALCF_RS*>
- Select a channel for example <*RPM*> from the logger digital group
- Right Mouse Click on channel <RPM>, opening the context menu, then select ***<copy>***

All *Datalogger* channels can be source channels for the CAN–Display.



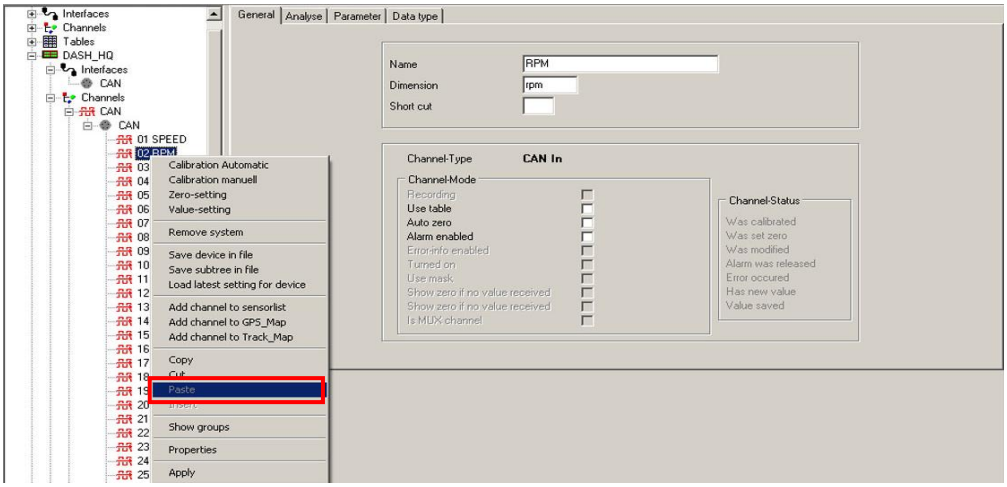
2.2 How to paste a reference channel

To paste the reference channel, proceed as follows:

- Left Mouse Click on the display tree, selecting <CAN>
- Select an unused CAN channel from the display tree, or a channel you want to overwrite
- Right Mouse Click and select ***<Paste>*** from the context menu, click <Apply> to confirm

The channel will be copied to the CAN Bus of the device, and can now use this data.

Instead of the mouse keys you can also use the <CTRL + C> shortcut for copy and <CTRL + V> for paste.

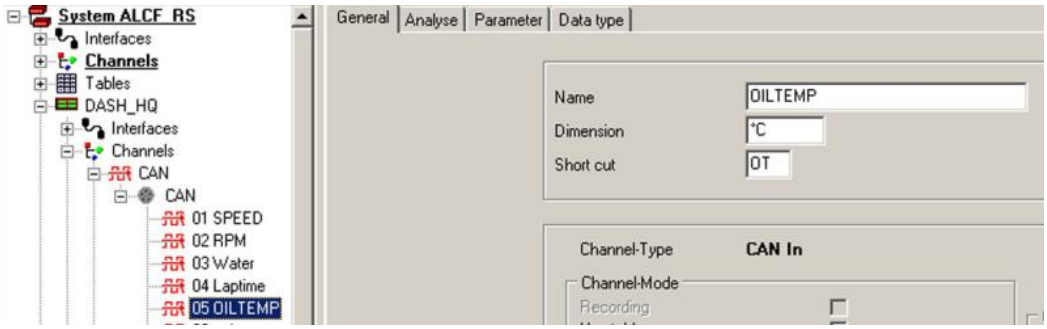


To modify the conversion formula or the formatting, you can begin by clicking on the tab ***<Analyse>*** (see next figure); Here the ***<Multiplicator>*** and the ***<Offset>*** can be changed.
The received values of this channel are processed using this exact formula (generating values to Display and activate Alarms, etc). The formula inside the Dash must match the formula inside the logger! When copying directly from the logger device, the formula will be copied exactly by default.

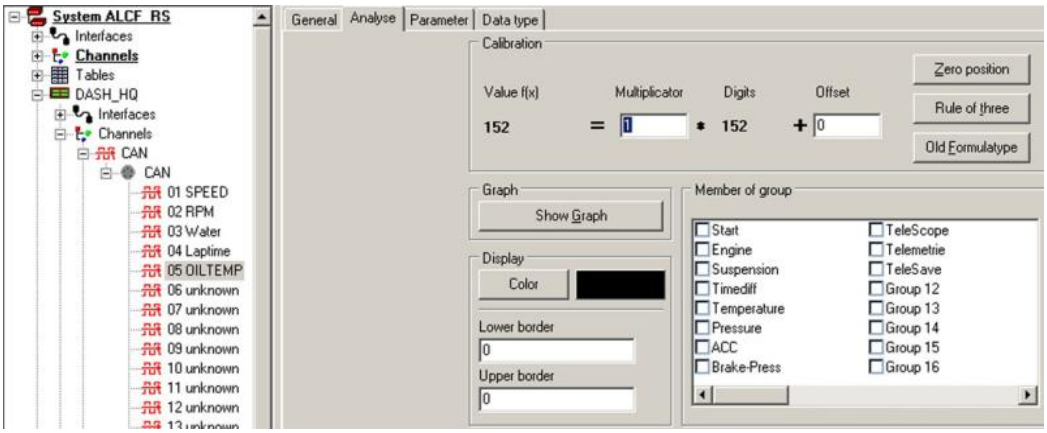
2.3 How to display and format "foreign" CAN-values

In order to display the correct physical value of a "foreign" CAN Bus channel, the **identifier**, **byte position**, **word width** and the **scaling** have to be known. These values are then entered manually into the **channel mask**. After doing this the channel is defined and can be displayed, and also used for alarms, calculations, etc.

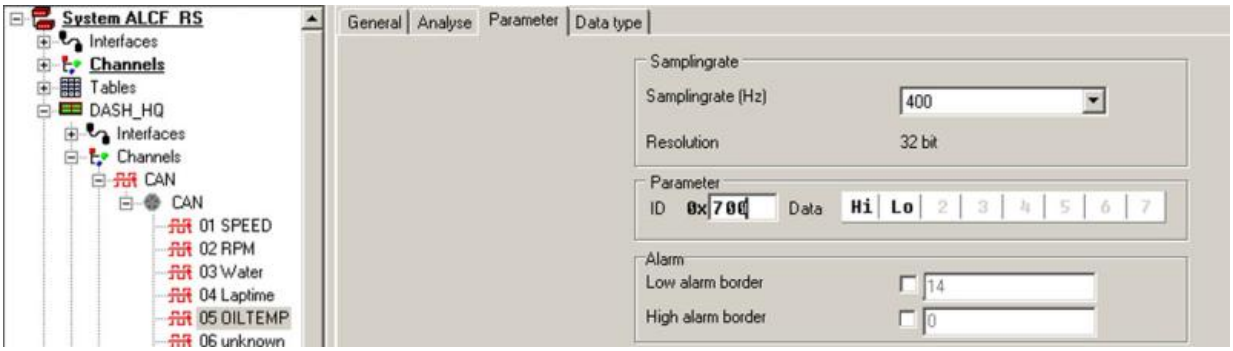
In the tree view, select an available CAN channel that you want to use for the foreign CAN Data.
Click on the tab <*General*> and enter your desired channel name (You can also enter <Dimension> and <Shortcut> if these are needed).



Next, click on the tab <*Analyse*> and enter the conversion formula for the physical value.



Inside the tab <*Parameter*> enter the CAN_ID as a hexadecimal value and choose the received bytes by clicking onto the data grid. Make sure to choose the correct data format
Note: (Motorola = Most significant Byte first or Intel = least significant byte first)



The 2D display supports a data width (word or byte) of 4 bytes for one channel (32 bit). By double clicking the field <*Data*>, the data width is set. If the data that is going to be received is in the intel format, <*Lo*> has to be entered into a higher byte.

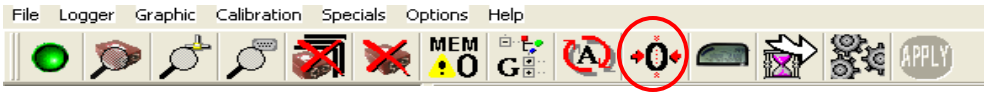
Example: For CAN ID 0X700, having a data width of 2 bytes, and received in the Intel format to bytes 0 and 1. This is configured inside the field <Data>, double click on <1> (enter <Lo>), then double click on <0> (enter <Hi>). These actions will correctly define the data format.

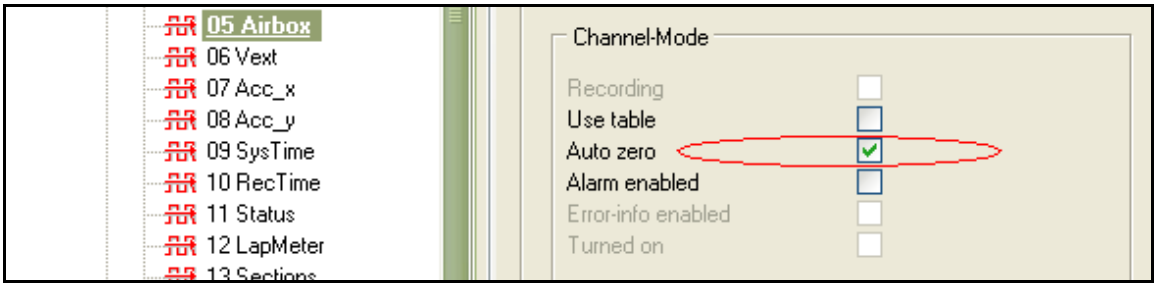
With the CAN In data channel now defined, this can be copied to any display page of the dash, as described by sections 1.3.2 and 1.3.3.

2.4 How to set channels to zero – The Auto zero function

The <*Auto zero*> function enables the user to zero several channels in a logger and/or dasboard at once with a click of a button.

Example: A channel is used for the measurement of Airbox pressure, which is recorded by the datalogger and sent via CAN-bus to the dashboard for display to the rider.
Because the ambient air pressure changes constantly, the measuring channel must be regularly set to zero, making sure the correct air pressure value is recorded by the datalogger and displayed on the dash.
This is a stage of calibration that is necessary for many kinds of sensor in a datalogger system. To make this process quick and easy, the <Auto Zero> function can be enabled inside the Airbox pressure channels of both the logger and dashboard. Both channels will be zeroed by pressing the <Auto Zero> button in the toolbar, as shown below.



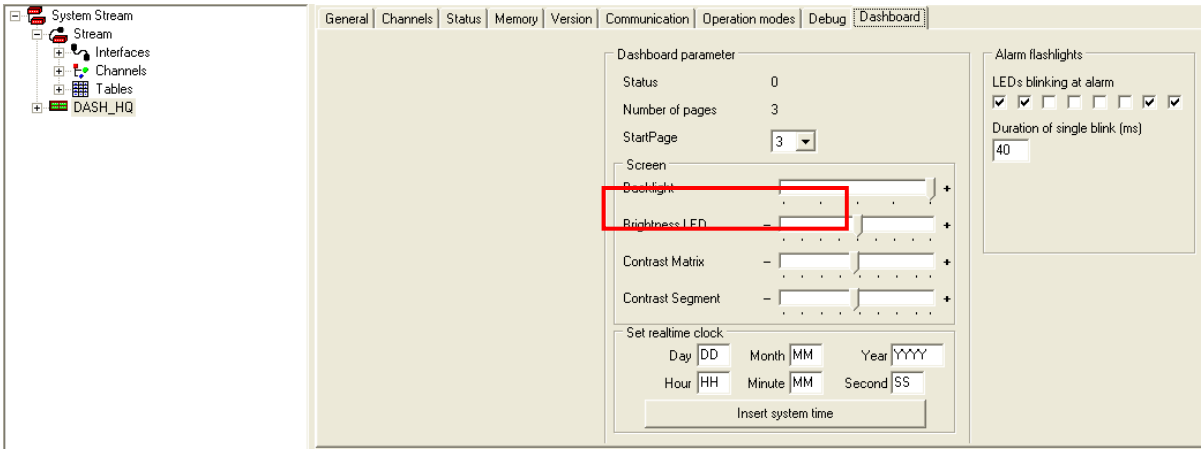


The ***<Auto Zero>*** command will be sent to all connected modules via the CAN Bus. All modules that receive this command, automatically set all channels to zero where the ***<Auto Zero>*** checkbox is activated - the present measuring value is set to equal physical value "0".

2.5 How to change the start page

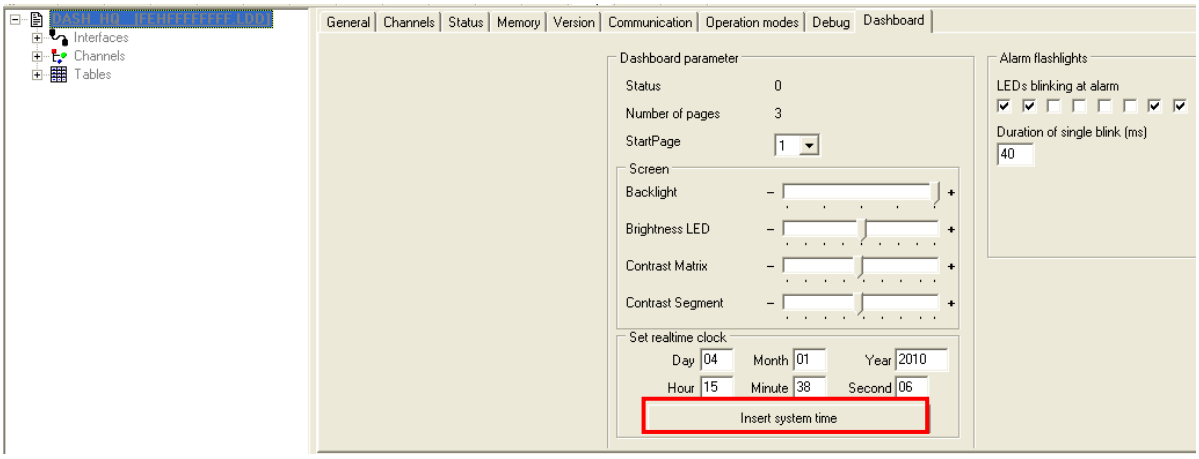
The start page is shown when the display is supplied with power.

To change the start page click on the **Main Node** of the Dash (System Tree), and then select the tab <**Dashboard**> on the right side of the screen. Inside this tab you can change the start page by modifying the dropdown box at <**Startpage**>, then click <**Apply**> to send the modification to the display.



2.6 How to insert current time/date

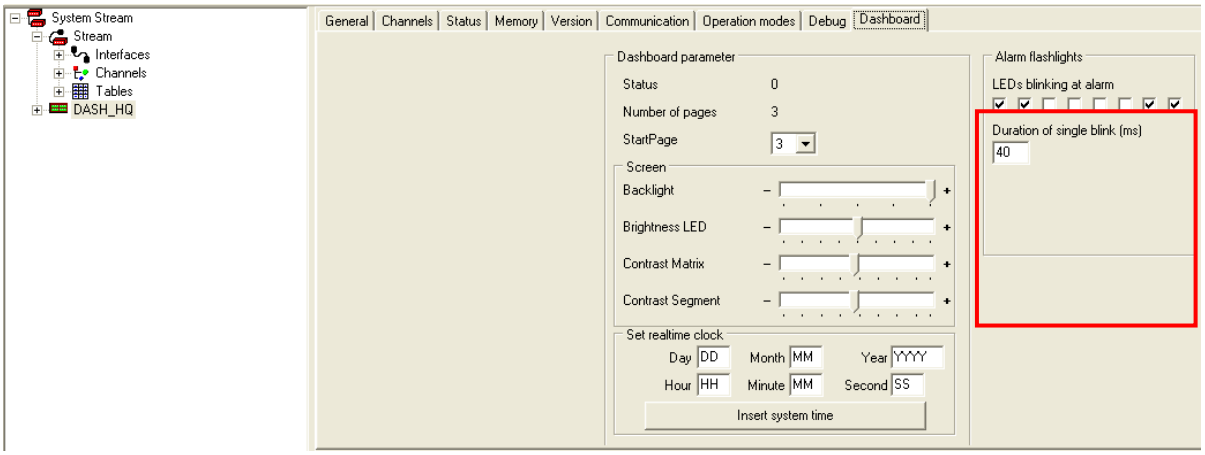
- Click on the Main Node of the Dash in the system tree and Select the Tab <**Dashboard**>
- Click **<Insert system time>** (make sure before that the system time is set correct on the computer)
- Click **<Apply>** to send the changes to the Dash



2.7 How to change the alarm flash light configuration

You can define the number of "Alarm flashlights" that blink when an alarm is activated:

- Click on the Main Node of the Dash in the system tree and Select the Tab <**Dashboard**>
- Select the desired <LED's blinking at alarm> checkboxes. Each box represents an LED on the top side of the Dash
- You can modify the **<Duration of single blink>** to change the speed of LED blinking
- Click **<Apply>** to send the changes to the Dash

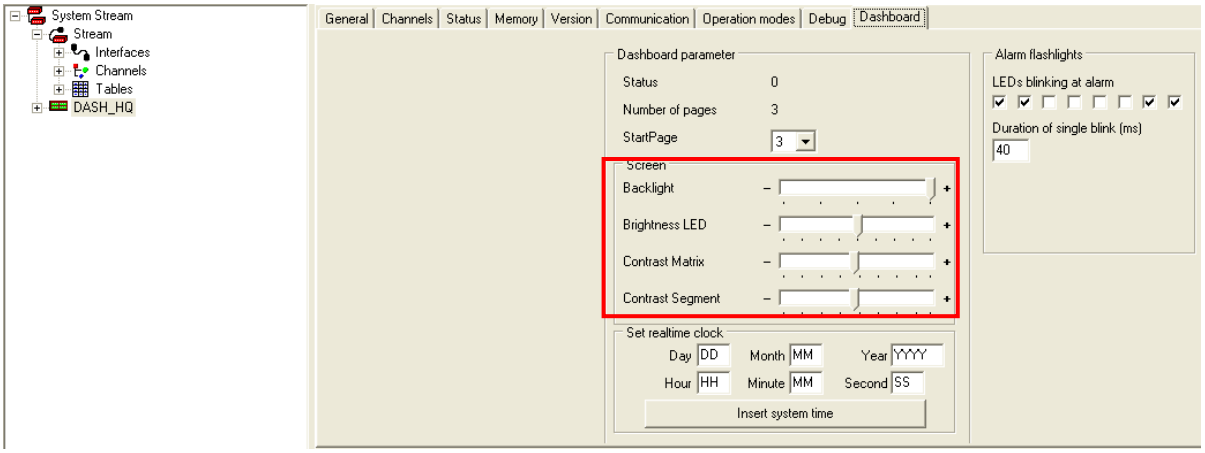


2.8 How to change the illumination of the display and LED's

Inside the Tab <**Dashboard**> (from the Main Node of the Dash in the system tree), the following options exist for defining the illumination of the screen.

The following options exist:

- **Backlight**: Illumination of the main display area (daytime setting)
- **Brightness LED:** Brightness of the top and right-hand-side LED's (daytime setting)
- **Contrast Matrix:** Contrast ratio of the display matrix in the **lower right hand corner**
- **Contrast Segment:** Contrast ratio for the left hand and **upper display area**



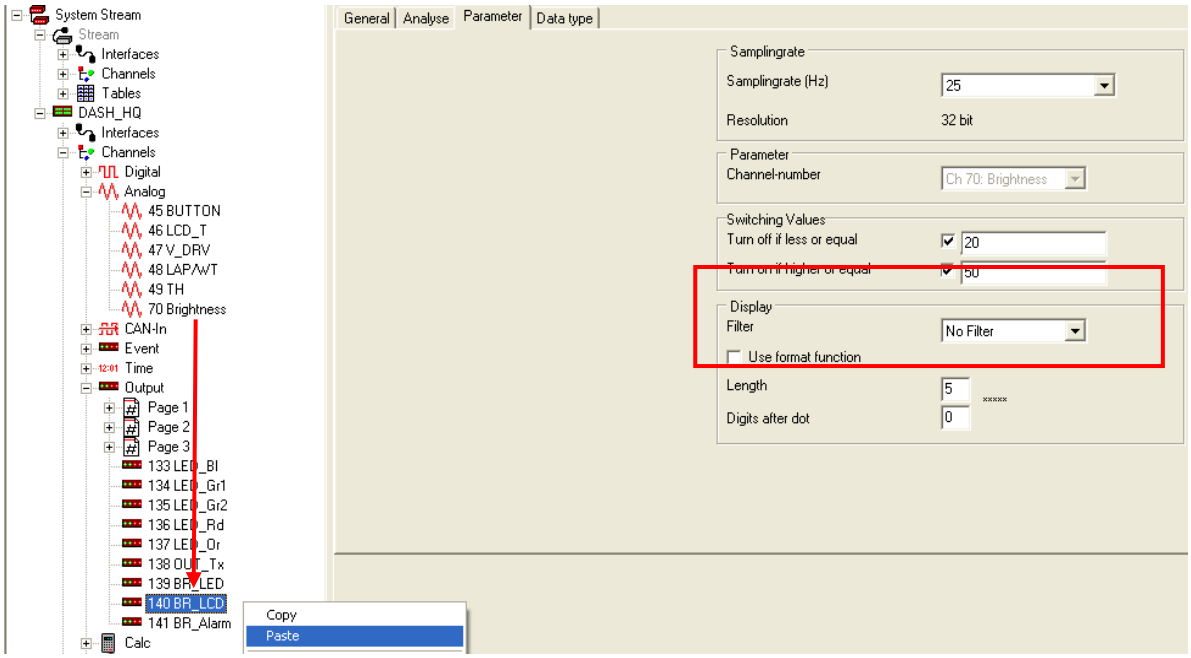
2.9 How to use the automatic dimming of the display and LED's (BigDash)

During night time driving the display and LED brightness can become too intense. Using the BigDash's built in light sensor (or any other data channel e.g. CAN or analogue input) the Display and LED intensity can be dimmed. This function is only available on the 2D BigDash Display.

If dimming is active LCD backlighting is set 1 step lower than the user adjusted daytime setting.

If dimming is active upper and righthandside LED's are dimmed to 25% of their full intensity.

The trigger channel for dimming mode can be set seperately for each type of light source. The Output Channels for the LCD (**<BR_LCD>**), the upper LED's (**<BR_Alarm>**) and the right-hand-side LED's (**<BR_LED>**), can each be defined with a different setting.
To choose the trigger channel for the dimming mode, copy and paste the triggering channel onto the desired output channel. The internal light sensor <Brightness> is the default trigger channel.



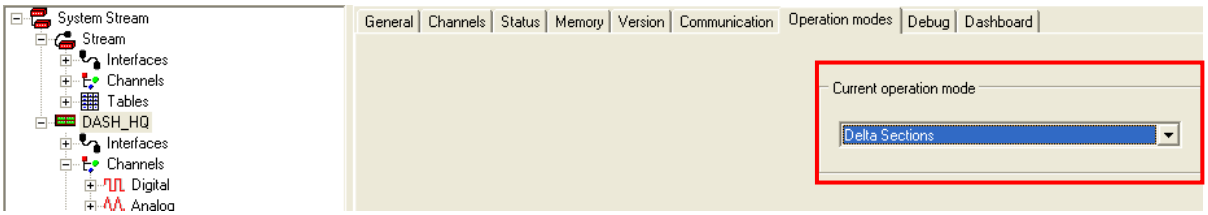
The meaning of the **<Switching values>** dialog are explained below:

- **<Turn off if less or equal> -** threshold value that will start dimming mode if input channel value is below or equal

- **<Turn on if higher or equal> -** threshold value that will start daytime mode if input channel value is higher or equal

2.10 How to change the operation mode

The **Operation Mode** can be changed by selecting the display node from the system tree, then choosing the tab <**Operation modes**>.
4 modes are available: Race, Endurance, Road and Delta Sections.



- **Race:** The default operating mode.
- **Endurance:** Same as Race mode, but a laptrigger with a signal length greater than 75ms is used as a section trigger. This can be useful if for example a minimum pitstop time has to be made. It will be triggered by an extra laptrigger at the pit entry and starts the section time to count down or up to the necessary time.
- **Road:** Same as Race mode, but warnings cannot be accepted by pressing the display buttons. If a warning occurs it will be shown as long as it persists.
- **Delta Sections**: This mode enables a delta time to be viewed on the display when each section of a racetrack or stage is completed. A predefined "Target Time" is set for each section, giving the rider or driver a positive or negative real-time comparison, e.g. -0.25s, indicating the completed section time is faster than the target.

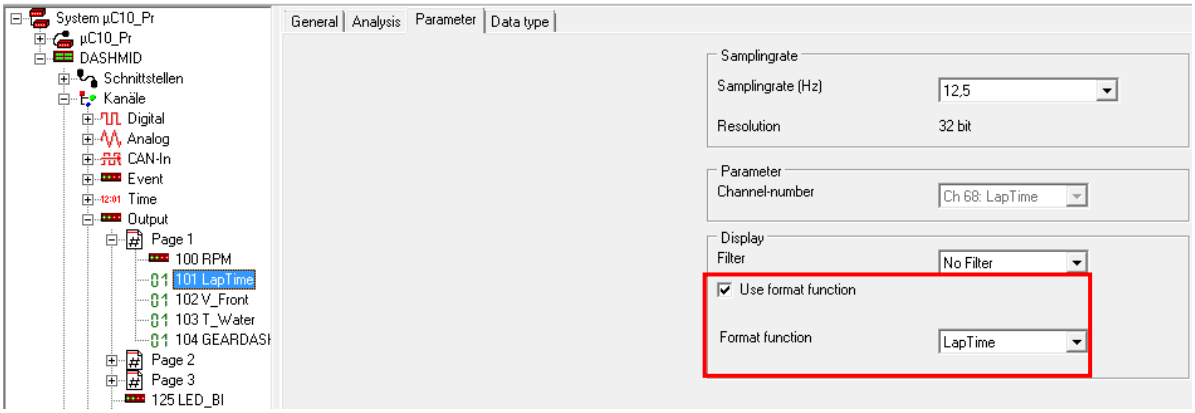
In order to use the endurance mode no 2D Laptrigger receiver of the first generation may be used. The following two figures shows IR-laptrigger receiver of the “first” and the “newest” generation. Also a 2D laptrigger transmitter with the label “Section trigger” at the cable must be used !

IR laptrigger receiver (“first generation”)

IR laptrigger receiver (“newest generation”)

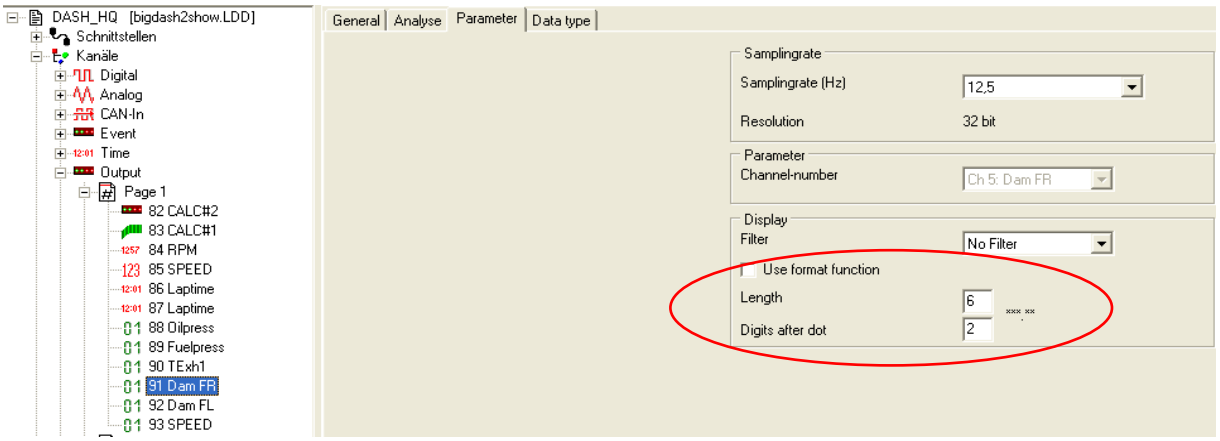
2.11 How to display a laptime

- First make sure to correctly define the Laptime Event channel. For further explanation for setting the Laptime channel see Chapter 3.1.
- Copy the Event channel Laptime onto the desired output page position. Refer back to Chapter 2.1 and 2.2, Copy and Paste a channel.
- Choose the format function Laptime in the tab <**Parameter**> as shown below.

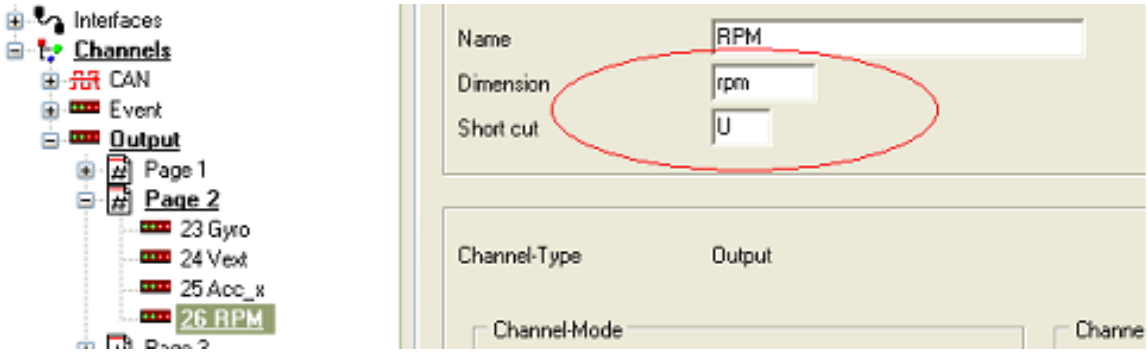


2.12 How to format the measuring values in the display

The displayed value is formatted by opening the tab <Parameter> of the Output channel you want to define (inside the appropriate page number). The displayed value is formatted according to its displayed **length** and the **digits after dot.**



Inside the tab <General>, the <Short Cut> and <Dimension> are initially the same as the copied CAN Channel, but can also be changed.



If the Name, Shortcut or Dimension of a CAN channel (inside CAN-IN node) are changed, the modifications are automatically transferred over to the Output channels.

Example: If you change the name of the CAN Channel <***RPM***> to "RoundsPerMinute" and this channel is already displayed on any page of the Dash, the Display will automatically change the output channel name from <***RPM***> to <***RoundsPerMinute***>.

If instead you only change the channel name of the Output channel, the CAN channel is unchanged and maintains the original name of <***RPM***>. The output channel takes over the new name <***RoundsPerMinute***>, but will display channel values in the same way as before.

3 Event Channels

Event channels include Laptime, Button channels, display Outputs and Alarms. These perform special functions of dashboard, but can also be recorded for analysis.

3.1 The Laptime function

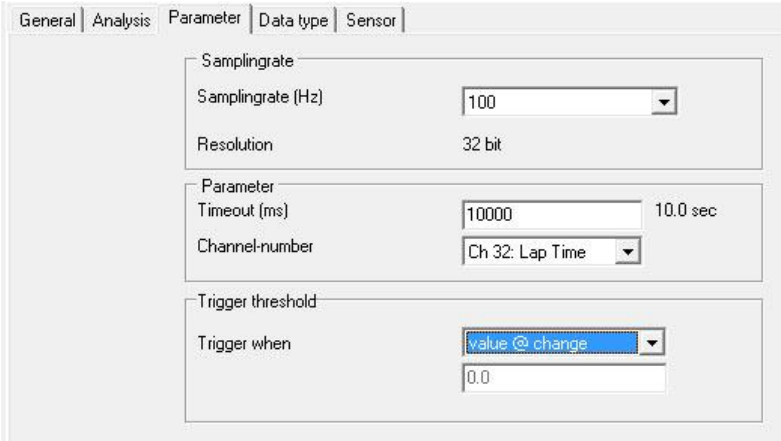
The Laptime channel is used to generate Laptimes inside the dashboard. A trigger channel is required to activate the beginning / end of each lap.

This function can be found inside the system tree via; <***Channels***> ⇒ <***Event***> ⇒ <***Laptime***>. To turn on this function, open the tab <***General***>, then select the <***Turned on***> check-box.
To define the triggering channel, open the tab <***Parameter***>, select the correct trigger channel from the <***Channel-number***> dropdown-box. When the Laptime is activated, its value becomes updated for display to the user, and the Laptime counter is reset to 0.

Depending on how the behaviour of the trigger channel, four logical options exist for making the Laptime in the correct way. Some of these options require the entry of a threshold value (digits).

- **value smaller than** - Laptime occurs if trigger channel becomes lower than threshold
- **value bigger than** - Laptime occurs if trigger channel becomes higher than threshold
- **value changes** - Laptime occurs if trigger channel value changes (no threshold entry)
- **value @ change -** When the value of the Laptime trigger channel changes, the new value of the trigger channel will replace the old value of 2D Dash Event channel Laptime.

When the vehicle laptime is generated outside the 2D Dash, e.g. Datalogger, then sent to the 2D Dash via Can-Bus, the **<Value @ change>** can be used to make sure the displayed laptime on the Dash is equal to the recorded Laptime of the Datalogger! This requires the CAN channel "Laptime" to be used as the trigger channel!



The <***Timeout***> option can be used to prevent repeated Laptime activation (false lap triggers) by entering a time value (in milliseconds), ensuring no laptimes are generated until the time elapses. The frequency at which the triggering channel is checked for changes is defined by the <***Samplingrate***>.

3.2 The "AlStat" function

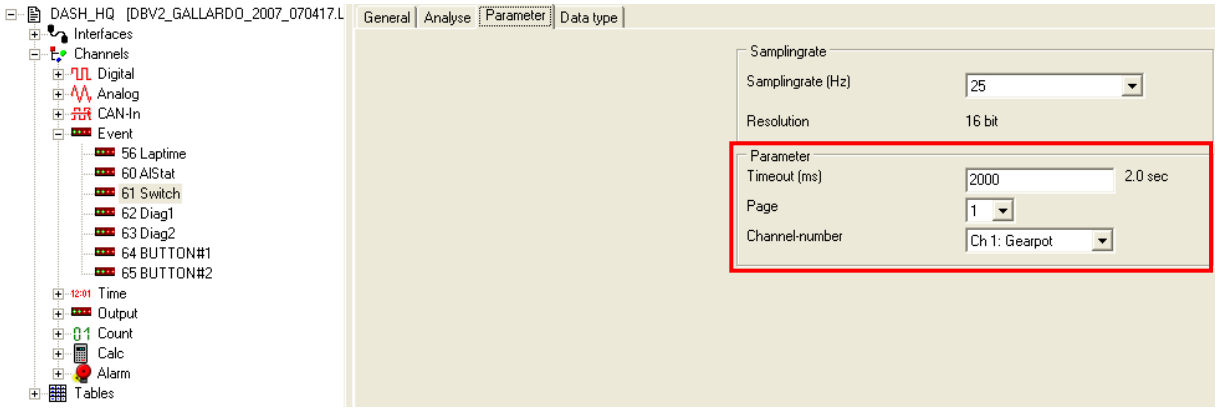
The AlStat channel contains binary information to indicate when each of the 16 alarm channels was triggered. It can be sent on the CAN Bus to monitor the alarm outputs of the dashboard.
If an alarm is triggered the corresponding bit is set to 1 and back to 0 when the alarm is turned off.

For more information on this, refer to Section 8.2 Alarm protocol for the AlStat Event Channel.

3.3 The "Switch" page function

The "switch" page function can be used to change the displayed page to a predefined page when a selected "trigger channel" is greater than 0, the display page will change. Useful trigger channels include Speed or RPM.

This function can be found inside the system tree via; <***Channels***> ⇒ <***Event***> ⇒ <***Switch***>.
To turn on this function, open the tab <***General***>, then select the <***Turned on***> check-box.
To define the triggering channel, open the tab <***Parameter***>, select the correct trigger channel from the <***Channel-number***> dropdown-box. Inside the <***Page***> dropdown box, the page to which the dash will "switch" is assigned. After the trigger channel returns to 0, the "switched page" will remain displayed until the <***Timeout***> (set the time in milliseconds) that has elapsed. The frequency at which the triggering channel is checked for changes is defined by the <***Samplingrate***>.



In the example shown above the 2D Dash will switch from the Start Page to Page 1 as long as the selected gear of the vehicle is higher than 0. A Calculation Channel can also be used as the trigger channel, making possible many logical combinations. For more information on Calculation Channels refer to Chapter 7.

While the trigger channel is greater than 0, the displayed page is fixed to the defined page and cannot be changed by any other method, e.g. Button.

3.4 The "Diag1, Diag2" channels

The diagnostic channels make possible the display of predefined messages on the dashboard when the assigned channel has a certain value, examples include:

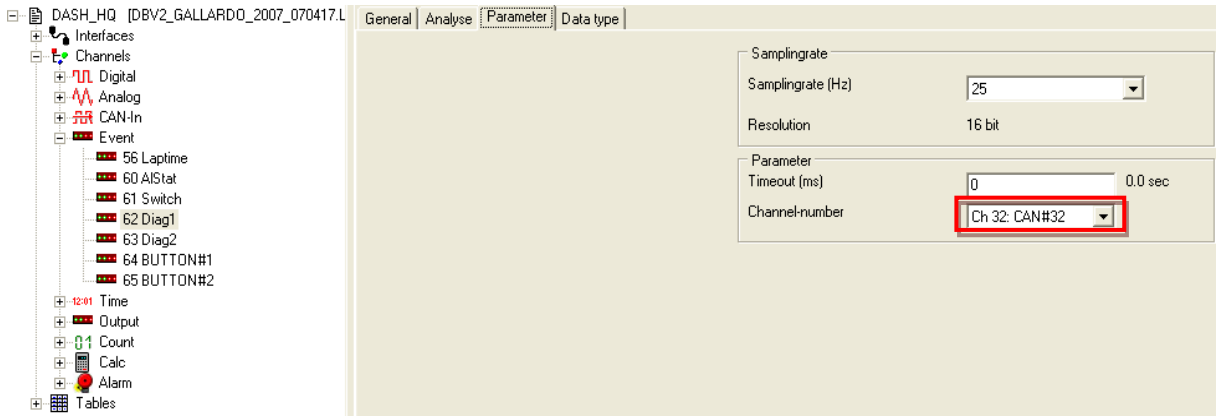
- Display the exact meaning of a "fault channel" signal from your ECU
- Count down the final laps of the race with text on screen
- Give a text-based multi stage alert according to the value of a particular channel, e.g. "rear tyre COLD", "rear tyre OK", "rear tyre HOT"

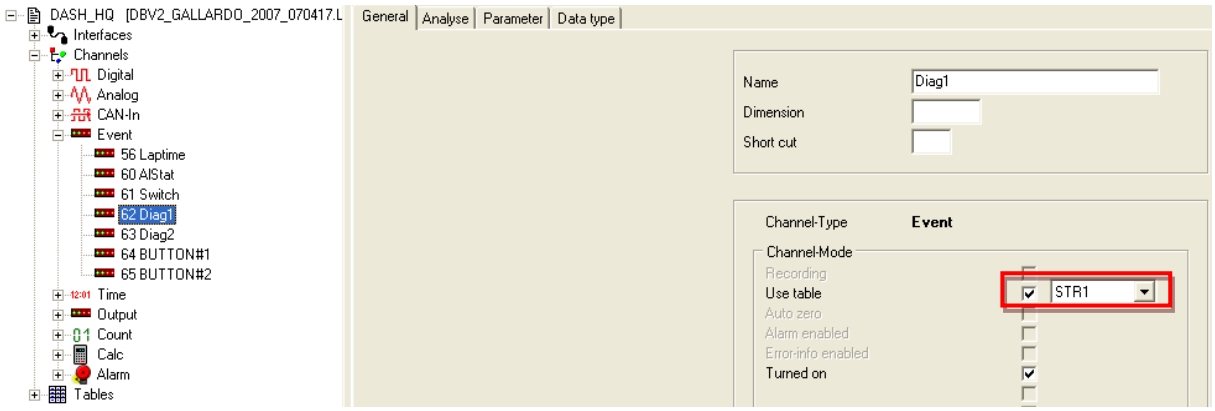
Diag1 – Displays the messages defined by the loaded string table "STR1" **on all 3 pages** in bold
Diag2 – Displays the messages defined by the loaded string table "STR2" but **only on page 3**

The displayed messages are predefined inside a table, which is then uploaded to the dashboard and saved inside. All channel adjustments are equal for both channels.

To activate the "Diag" function you must:

- Open the tab <***General***> for the Diag channel, then select the <***Turned on***> check-box
- Select the driving channel for the Diag function from the <***Channel-number***> dropdown-box. The selected channel will be used in collaboration with the loaded string table to generate the correct display message on the screen.
- Assign an appropriate value to <***Timeout***>, this defines the threshold time before the diagnostic output is displayed
- Define a String (STR) lookup table that defines the messages to be displayed on the screen when the corresponding driving channel value (from step 2) is input to the string table.
- Load the String lookup table to table location <***STR1***> or <***STR2***> of the 2D Dash.
  Note: (STR1 -> Diag1, STR2 -> Diag2)





3.4.1 Using the Diag function - Practical Example

Consider the following example where a multi-stage alert is created to give information on water temperature channel “T_Water” as the vehicle is being warmed up. We only want to display this on Page 3 of the Dash (the engineer page) therefore we must use the Event Channel “***Diag2***” to display the messages on the screen.

In practice, the first step is to define the channel conversion table. At this stage ‘threshold values’ must determined at which the display messages should be activated.
For example you might want to display the following information:

Water temperature (deg C)	Message to display
-40 to 0	Check Motor Freeze!
0 to 50	Motor V Cold!
50 to 70	Motor Cold
70 to 90	Motor OK
90 - 100	Motor Ready
100 - 110	Motor Hot!
110 -	Motor Very Hot!

Step1 - Defining the string lookup table STR2

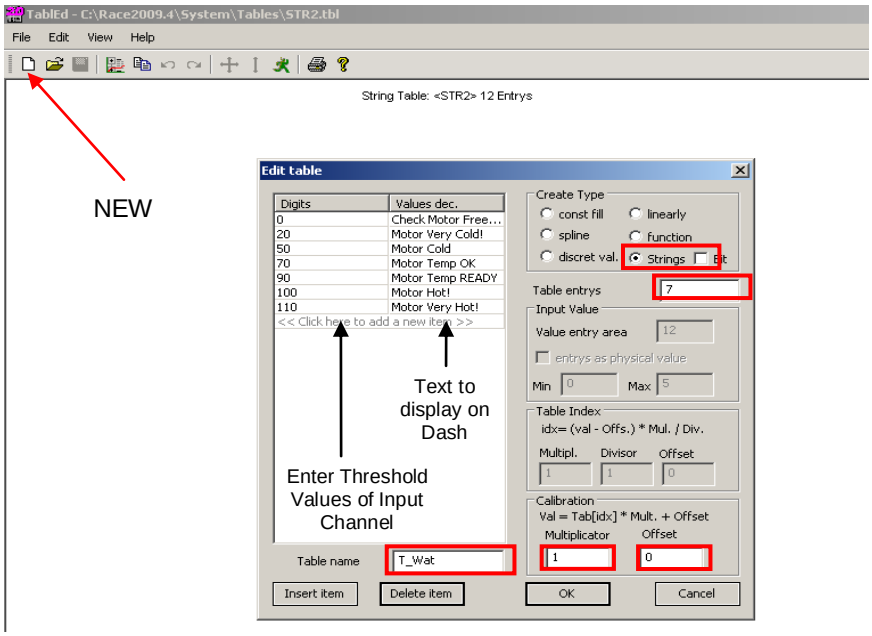
To create the string lookup table:

- Open the 2D WinIt program “TablEd.exe” which can be found in the Race_xx.y (or similar) installation folder
- Click <New> to open the “Create Table” Window
- Set Create Type to “Strings”
- Set Table entrys to 7 (the number of entries you have made)
- Set “Multiplicator” to 1 and “offset” equal to 0
- Name the table in the Table Name field e.g.’T_Wat’, then click <OK>
- In the column “Digits” enter the threshold temperature values above which you want a particular display message to be made

- In the column “Values dec", enter the text you want to display on the dash when the Diag input channel has the value entered to the digits column

To achieve the desired settings that were already introduced, the entries below (in GOLD) should be entered into the table.

Water temperature (deg C)	Value Dec	Digits
0 to 20	Check Motor Freeze!	0
20 to 50	Motor V Cold!	20
50 to 70	Motor Cold	50
70 to 90	Motor OK	70
90 - 100	Motor Ready	90
100 - 110	Motor Hot!	100
110 -	Motor Very Hot!	110



Also ensure that the following settings are made before completing the string lookup table:

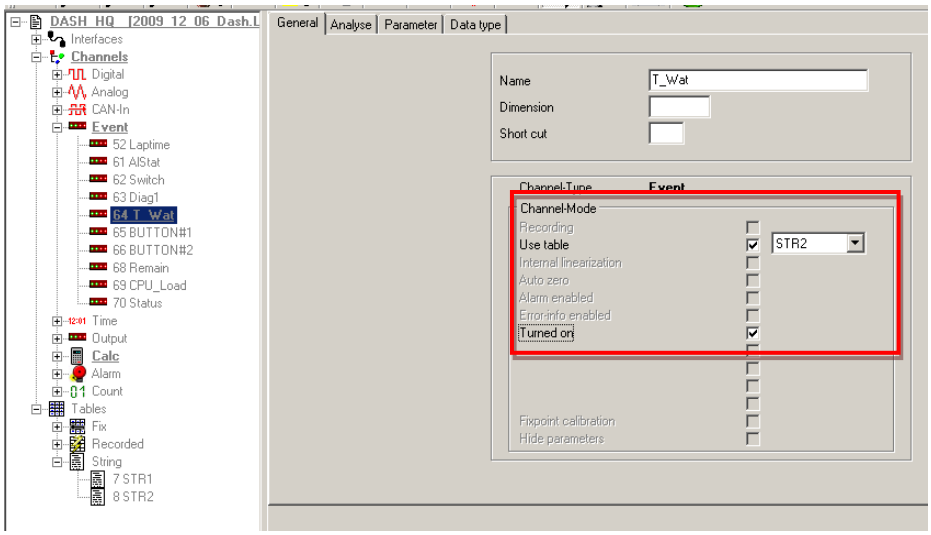
- Go to <***File/Save***> to permanently store this table inside your computer
- Save the file table with the same name e.g.’T_Wat’, **and be sure of the directory into which it is saved**.

Step 2 - To load the defined string table into 2D Dash:

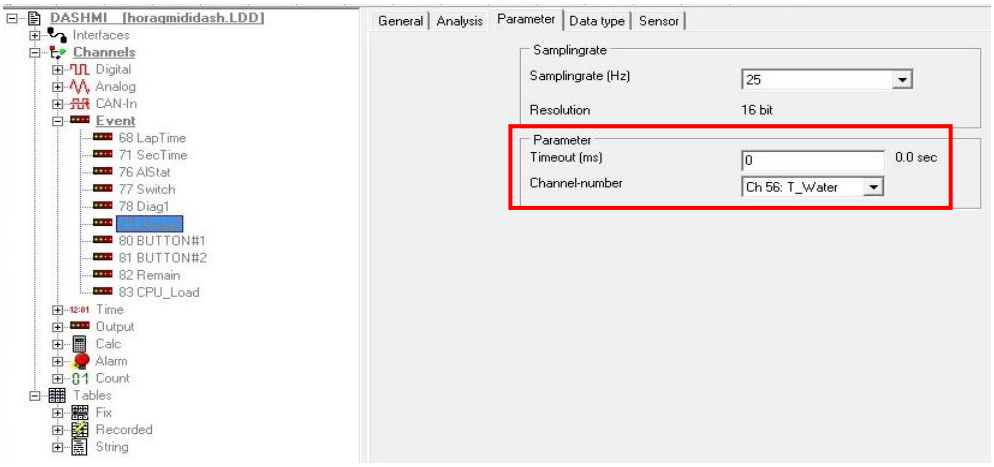
- Open the 2D program WinIt and make communication with the 2D Dash
- Select the node “***Tables***” from the system tree
- Expand the “***String***” section within the “**Tables**” node
- Select the table position named "***STR2***", this is used for the Diag2 function.
- Click <***Load from disk***> on the main window
- Locate the directory into which the “***STR2***” table was saved
- Select the table “***STR2***” and click <***Open***>

Step 3 - To define the settings of the Event Channel Diag2:

- Open the "***Event***" node of the system tree, select "***Diag2***"
- Check the box for "***Use Table***", and choose the previously created table "***STR2***"
- Check the box for "***Turned on***", this will activate the Diag2 function



- Select the tab <***Parameter***> and nominate the data channel (from <***Channel-number***> dropdown box) to use as input for the created string lookup table. In this example the analog input channel "***T_Water***" is selected.



For this example a <***Timeout***> of "0" is used, meaning the message is always displayed while the string table and input channel value persist. If a timeout of "1000" was defined, the displayed message would disappear from Page 3 of the screen after 1 second (provided the displayed message does not require to change in that time!). The message will come back to the screen if or when the data channel value changes enough to make the displayed message also change.

3.5 The “Button#1, Button#2“ channels

The Button channel allows the user to specify an input channel to trigger the switching of pages on the display. This is different from the "Switch" page function as this allows the user to scroll synchronously through the pages of the 2D Dash.

In the tab <***Parameter***> next to <***Channel-number***> the channel to trigger the page switching is assigned. Any Analogue or CAN-channel can be used, and will change the displayed page when its physical value rises above 50% of its maximum (full scale) value. The default trigger channel is the input channel of the external "Button". <***Timeout***> defines the threshold time before the display page is switched.

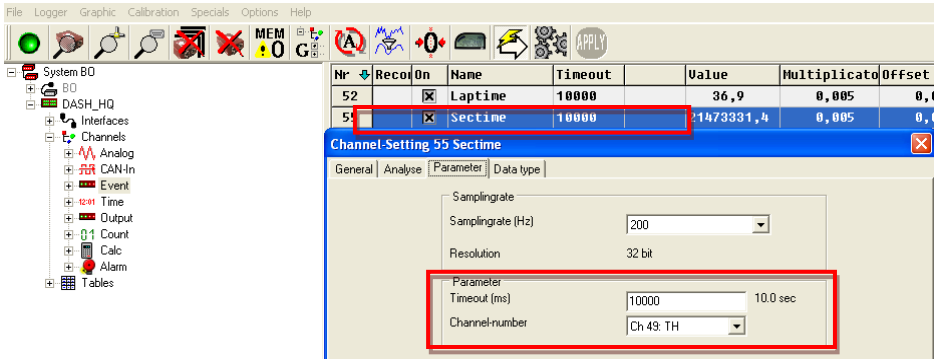
3.6 The Section Time channel (SecTime)

3.6.1 SecTime - Race Mode

The Section time channel allows the user to observe section times on the 2D Dash. With the help of section time triggers on the track, this channel shows the elapsed time between 2 trigger signals

In order to use the section time function, the 2D Dash has to be operated in "***Race mode***" (refer to How to change the operation mode). Also the section time channel must be configured as shown below:

- In the tab <**General**>, select the <***Turned on***> check-box to switch on SecTime
- Select the tab <***Parameter***> and nominate the data channel (from <***Channel-number***> dropdown box) that is used to indicate section time completion e.g. LAP.
- Also in the tab <**Parameter**> the elapsed time before the next section trigger is accepted is specified by the <***Timeout***> value that is assigned.

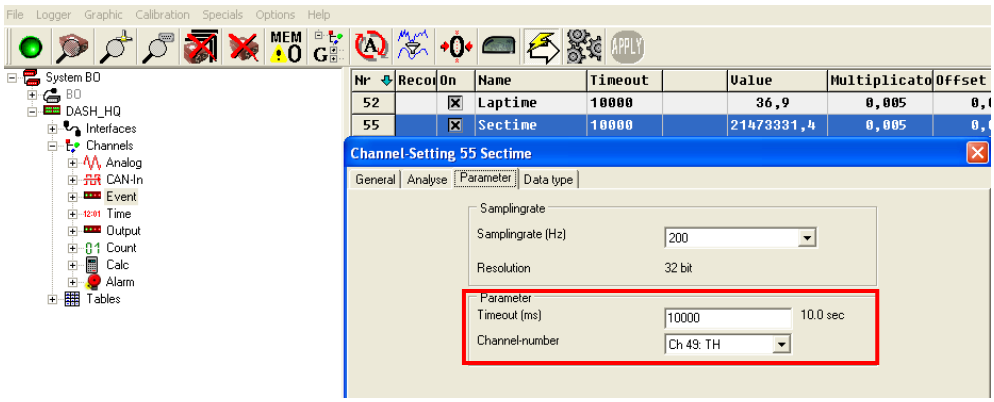


3.6.2 SecTime - Endurance mode

The Section time channel allows the user to specify a countdown time that will be count down to 0 when a section time signal is received. This is useful where a minimum pitlane time is enforced by regulation. GT3 Endurance teams use this function to show the time left before the vehicle can leave the pits after a pitstop.

In order to use the countdown function the 2D Dash has to be operated in "***Endurance Mode***" refer to (How to change the operation mode). Also the section time channel must be configured as shown below:

- In the tab <**General**>, select the <***Turned on***> check-box to switch on SecTime
- Select the tab <***Parameter***> and nominate the data channel (from <***Channel-number***> dropdown box) that is used to indicate section time completion e.g. LAP.
- Also in the tab <**Parameter**> the **Countdown Time** (in ms) is specified by the <***Timeout***> value that is assigned. e.g. for a 90 second countdown, enter 90000 to <***Timeout***>.

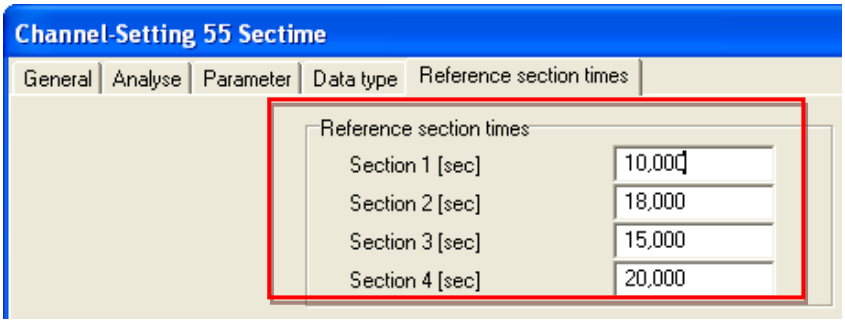


3.6.3 SecTime - Delta Sections Mode

The Section time channel allows to show the time difference between previously defined section times and received section triggers to display gained or lost time in the last completed sector.

In order to use the Delta Section function the 2D Dash has to be operated in "***Delta Section Mode***" (How to change the operation mode ?). Also the section time channel must be configured as shown below:

- In the tab <**General**>, select the <***Turned on***> check-box to switch on SecTime
- Select the tab <***Parameter***> and nominate the data channel (from <***Channel-number***> dropdown box) that is used to indicate section time completion e.g. LAP.
- Also in the tab <**Parameter**> the elapsed time before the next section trigger is accepted is specified by the <***Timeout***> value that is assigned.
- In the tab <***Reference section times***> the target time for completion of each section triggers can be specified as shown below.

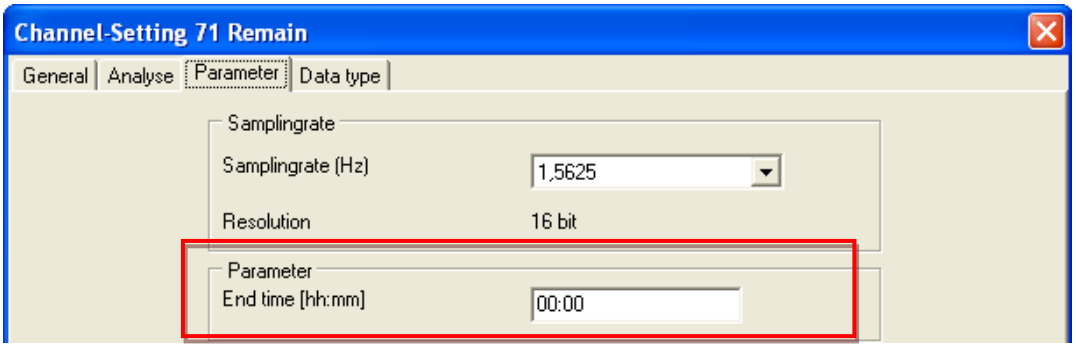


3.7 The "Remain" channel

The channel "***Remain***" shows the difference between the current date and time to a specified time. It can be used as a countdown to show the rider/driver the remaining time to the end of a practice session or any other time related event.

The Remain channel is configured as shown below:

- In the tab <**General**>, select the <***Turned on***> check-box to switch on Remain
- Select the tab <***Parameter***>, enter the required end time in the <***End time [hh:mm]***> field. Unless the end time is modified, the 2D Dash will count down to this exact time every day. After the time has elapsed, the value of "Remain" stays at 0 until the next day begins (time 00:01).
- Make sure to correctly set the system time of the Dash (see How to insert current time/date).



4 Time-channels

4.1 Lap_run

The Lap_run channel shows the elapsed time since the last received laptrigger signal.

This can be displayed on the 2D Dash (Copy and Paste channel to an Output Page), and also sent to the CAN-Bus to be recorded by a 2D datalogger.

4.2 Gap function

The GAP-function shows the driver how much time difference is at the current track position compared to a previously driven fastest lap.

This function enables the driver to monitor time gains and losses through different sections of the track.

- **Positive time gap (e.g. 0.21)** - the current lap is 0.21s ***faster*** than previous fastest lap **at the current lap position**

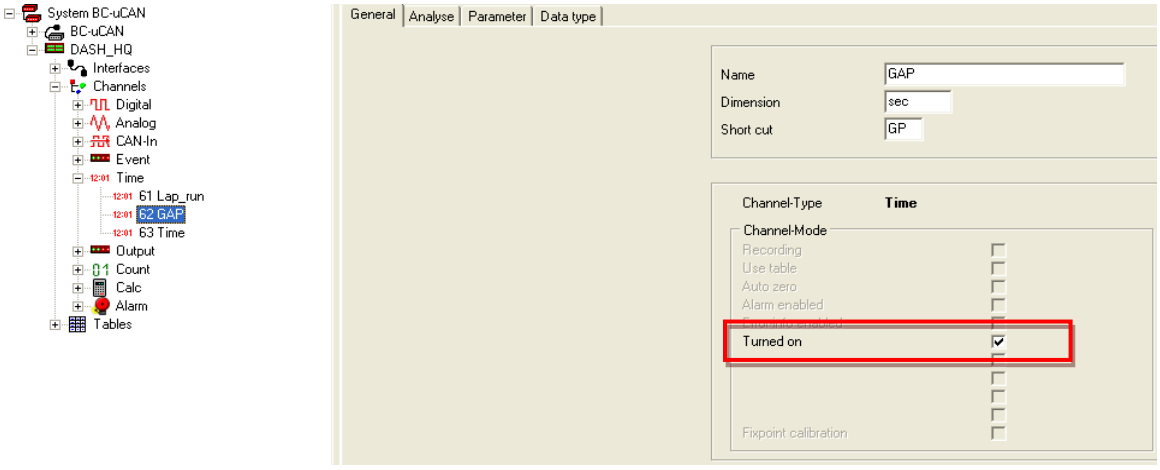
- **Negative time gap e.g.-0.75** - the current lap is 0.75s ***slower*** than previous fastest lap **at the current lap position**

To configure the Gap function channel:

- Select the channel **<*Gap function*>** from the system tree
- In the tab <**General**>, select the <***Turned on***> check-box

In order to use the Gap-function the channel <***Lapmeters***> has to be set correctly. (Refer to Lapmeter section 6.1)

- Select the Count channel **<*LapMeter*>** from the system tree
- In the tab <**General**>, select the <***Turned on***> check-box
- In the tab <***Parameter***>, select the **<*Counted channel*>** dropdown box and choose a valid and accurate speed channel. This is used to calculate lap progression for time comparison.



The Gap function will always compare the current lap to the fastest lap stored inside the 2D Dash. To reset the fastest lap (for a new session or new race event), **Clear the dash memory.**

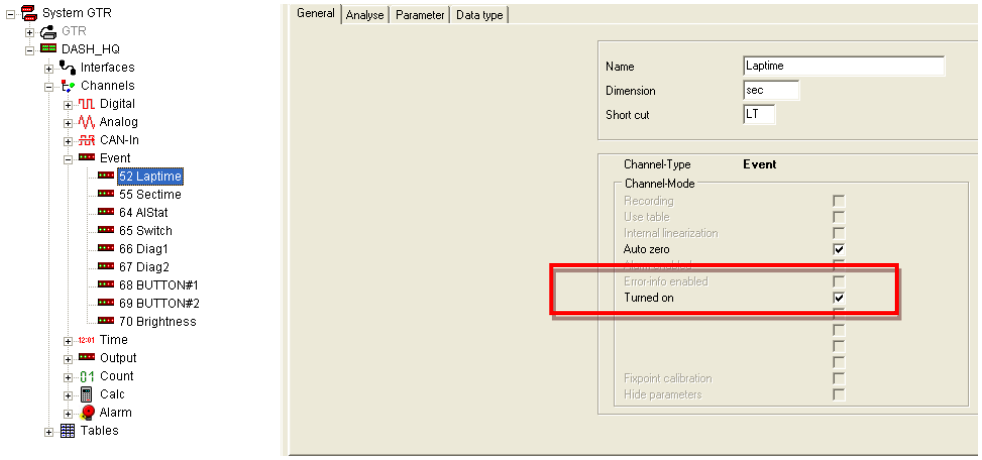
To erase the fastest laptime in the 2D display:

- Select the 2D Dash from the system tree
- Select the tab ***<General>***
- Click the **<Empty>** button, and then click ***<Apply>***.

You can also define the 2D Dash so that the laptime memory is reset every time the Dash is unpowered. This also resets the reference lap for the Gap function.

To automatically reset the Laptime memory when the 2D Dash is powered-off:

- Select the Event channel Laptime via ***2D-Dash->Event->Laptime***
- Select the tab ***<General>***
- Check the ***<Auto zero>*** checkbox and then click ***<Apply>***



5 Output Channels

Output channels are used to display the physical values of data channels to the user while the data system is switched on.

The 2D Dash features different types of output channel, including formatted numbers (integer or decimal), LED illumination, text display, or multi-segment bar graph (BigDash).

Any data channel from the entire 2D system can be displayed to the driver (or engineer) on the 2D Dash. To enable the display of a channel, it must be available to the CAN Bus of the 2D Dash. This is achieved by copying each channel from its source device (e.g. Datalogger, Engine Interface Module, Gyro Module) and pasting them to the CAN-In directory of the 2D Dash.

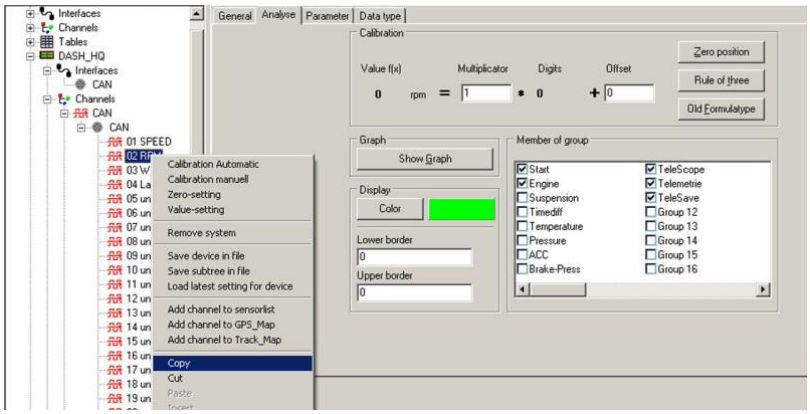
Channels already created inside the 2D Dash by measurement (***<Digital>***, ***<Analog>***), or calculation (***<Event>***, ***<Time>***, ***<Calc>***, ***<Count>***), can also be copied from their respective location and pasted to the required output page location.

Next the required display channel must be copied from the 2D Dash, then pasted to the specified ***<Output>*** page in the desired location. The measured value of the channel is automatically converted into a physical value (using the inherent conversion formula of the channel).

5.1 How to assign output channels ?

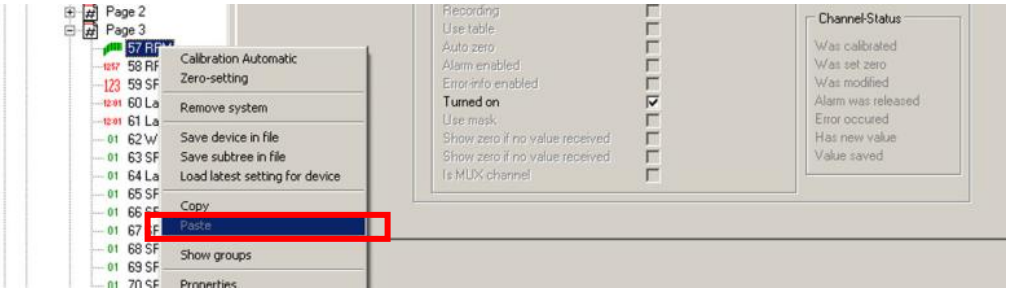
To display a channel on the display page, for example "RPM":

- **Copy** the required channel (#RPM) from the device in which it is located (e.g. ***<Digital>*** channel inside the datalogger)
- Select an available position inside the Can-In directory of the 2D Dash
- Right-Click on the selected position and click **Paste**
- Select and **Copy** the new CAN channel <RPM> from the Can-In directory of the 2D Dash



- Expand the ***<Output>*** node, expand ***<Page 2>***
- Select an available output channel location from Page 2
- Right click the selected location and select **Paste**, click ***<Apply>***

Alternatively the *"DRAG and DROP"* method can be used!



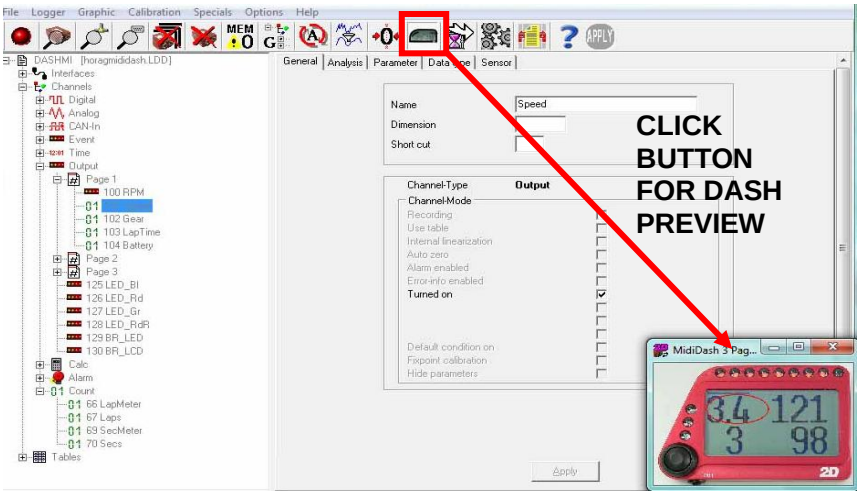
5.2 Output Channel Display Formatting

Many formatting options exist for each type of Output channel. The best way to master this subject is to review some examples, then make some practice!

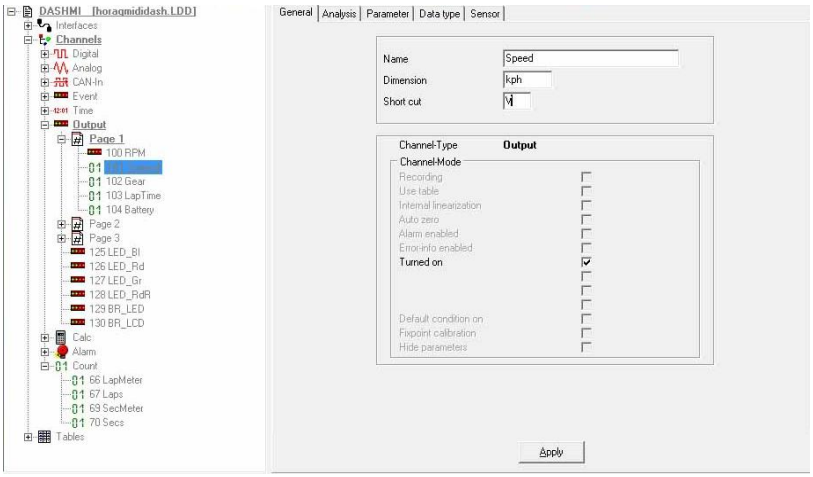
After the data channel is copied to the required Output page of the 2D Dash, the next step is to finalise its formatting. This stage allows the user a high level of customisation, where the appearance of the displayed data can be adjusted to meet specific user requirements.

5.2.1 Practical example - Formatting a Speed channel

The speed channel <Speed> is already copied to Output Page 1 in position 2, as shown below:



Inside the tab ***<General>***, the channel ***<Name>***, ***<Dimension>*** and ***<Short cut>*** can be modified, as shown below.



<Name> - for reference only, modification does not change the displayed appearance
<Dimension> - input characters are displayed **after the numerical value** on the 2D Dash, e.g. "125 kph"
<Short cut> - input characters are displayed before the numerical value on the 2D Dash, e.g. "V 125 kph"
<Turned on> - ensure this is checked to display this value on the 2D Dash!

The **Dimension** can be used to apply physical units to the speed value, indicating that the displayed speed is measured in kilometers-per-hour, and not miles-per-hour.

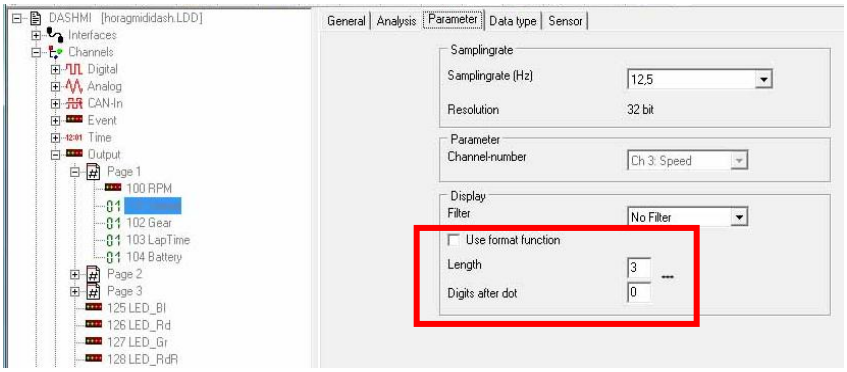
The **Short cut** gives can be used to explain or remind the user what this display field is actually showing values for, V = velocity, or vehicle speed.

When a channel is displayed on Page 1 of the 2D Dash (as for this example), space is quite limited because the display characters are much larger. Because of this it is common that the Dimension, and sometimes the Short cut are not included.

Due to **space requirements**, it is not always possible to display the complete measured value, complete with its assigned Dimension and Short cut. In many cases a compromise must be determined, where the Short cut or Dimension are removed, or the channel moved to an output page location that provides more space!

Inside the tab ***<Analysis>***, no actions or entries are required.

Inside the tab ***<Parameter>***, the ***<Length>*** and ***<Digits after dot>*** can be defined in the area highlighted below.



<Length> - the total number of characters reserved inside the display area for channel values INCLUDING a decimal point (dot). "3" is entered for the channel <Speed>.

<Digits after dot> - this defines how many decimal places of the physical channel value are to be displayed. "0" is entered for the channel <Speed>.

For the speed channel physical values are likely to rise larger than 100 kph, meaning at least 3 digits are required to correctly display its value, therefore the MINIMUM length entry has to be "3".

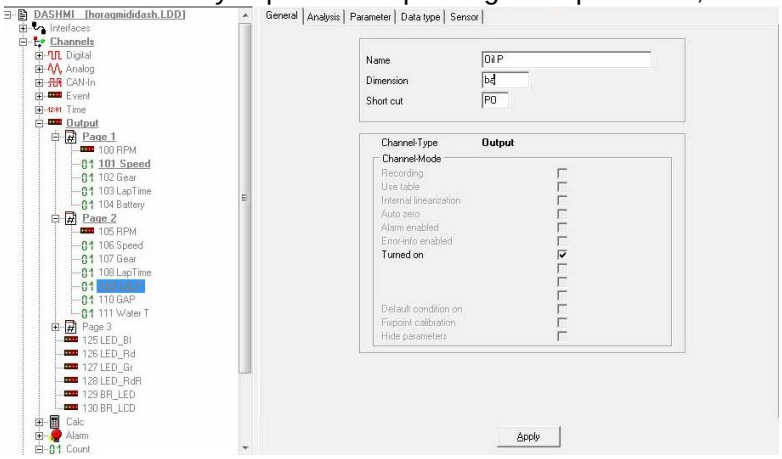
Decimal places or <Digits after dot> are not practical for two reasons:

- speed values are likely to be changing very quickly, decimal values will change too fast to be correctly read by the user!
- this channel is displayed on page 1, so there is limited space for the display of extra decimal places.

No actions or entries are required for the tabs ***<Data type>***, or ***<Sensor>***.

5.2.2 Practical example - Formatting an Oil Pressure channel

The data channel <Oil P> is already copied to Output Page 2 in position 5, as shown below:



Inside the tab ***<General>***, the Oil Pressure channel is configured as follows:
<Name> - default value "**Oil P**" is maintained for easy reference
<Dimension> - "**ba**" is entered to indicate measuring units of "bar", e.g. "2.2 ba"
<Short cut> - "**PO**" is enterered to represent Pressure-of-Oil to the user, e.g. "PO 2.2 ba"
<Turned on> - ensure this is checked to display this value on the 2D Dash!

Because this channel is displayed on Page 2, more characters are available to include the channel <Dimension> and <Short cut>. Also a decimal place can be included to provide more detailed information to the user.

Inside the tab ***<Analysis>***, no actions or entries are required.

Inside the tab ***<Parameter>***, the ***<Length>*** and ***<Digits after dot>*** are defined.
<Length> - "4" is entered for the channel <Oil P>, giving a total of 4 display characters.
<Digits after dot> - "1" is entered for the channel <Speed>.

The resulting configuration of the <General> and <Parameter> tabs means that when the oil pressure value is equal to 2.25 bar, it is displayed as follows -> "**PO 2.3ba**".

Note that using a <Length> of "4" allows a space to be created between the <Short cut> PO and the displayed value "2.3".

If a second decimal place is required, the <Digits after dot> is increased to "2", BUT to maintain the same display appearance (with a gap between "PO" and the displayed value), the <Length> entry must increase to "5"!

If the measured value of the <Oil P> channel increases above 9.99 bar, all 5 assigned characters (from the input <Length>) are required to display it. Because of this, the space between "PO" and the displayed values, e.g. 10.02 disappears!

No actions or entries are required for the tabs ***<Data type>***, or ***<Sensor>***.

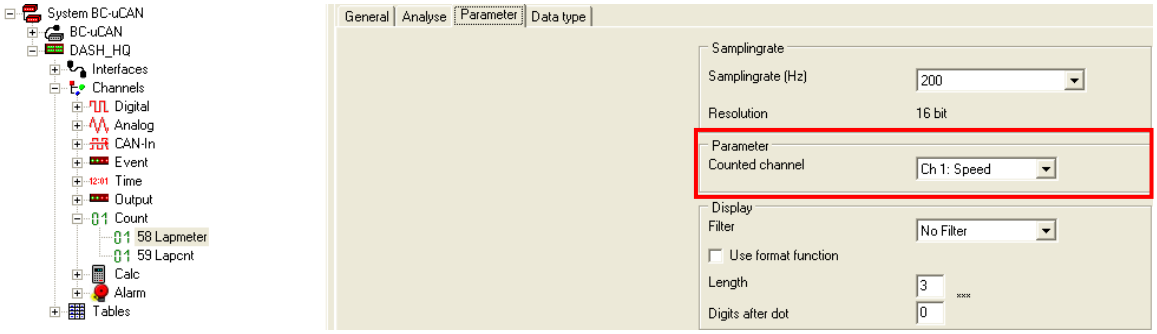
6 Count channels

6.1 Lapmeter

The lapmeter channel integrates the assigned speed channel to show the driven distance in the current lap.

In the tab ***<Parameter>*** next to ***<Counted-channel>***, select a valid speed channel from the dropdown box.

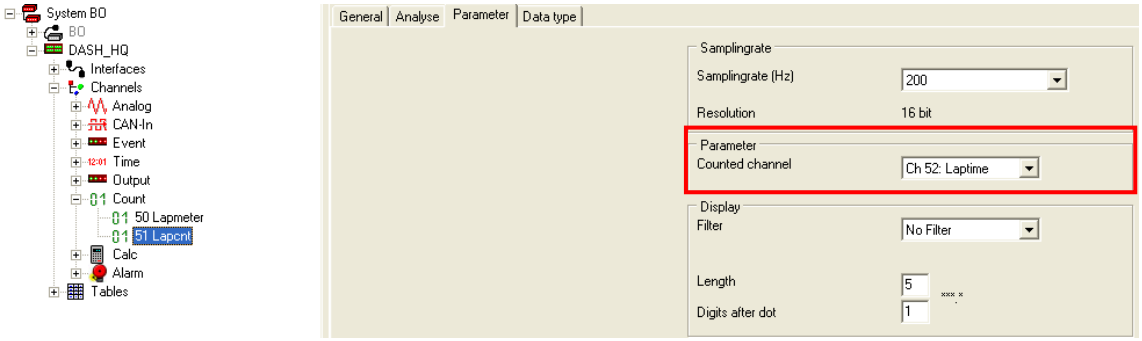
If the 2D DASH has 2 speed channels next to each other on the CAN-In channels with the same calibration the maximum value of both will be used in order to suppress wheel locking.



6.2 Lapcnt

The Lapcnt-channel contains the value of driven laps since system startup (power on), or the last time the Dash Memory was erased.

In the tab ***<Parameter>*** next to ***<Counted-channel>*** a valid laptime channel has to be assigned.



7 Calculation channels

A Calculation or "Calc" channel can perform mathematical operations to existing channels, according to the user-defined formula entered to the channel.

The resulting value of the Calc channel can be copied to the display output, used as a trigger channel for alarm or event channels, or sent to the datalogger for recording.

By using a <Multiplicator> and <Offset>, the calculation result can be displayed on the 2D Dash and transferable to an external CAN-Bus as a 16-bit value.

7.1 Calculation functions - Practical Examples

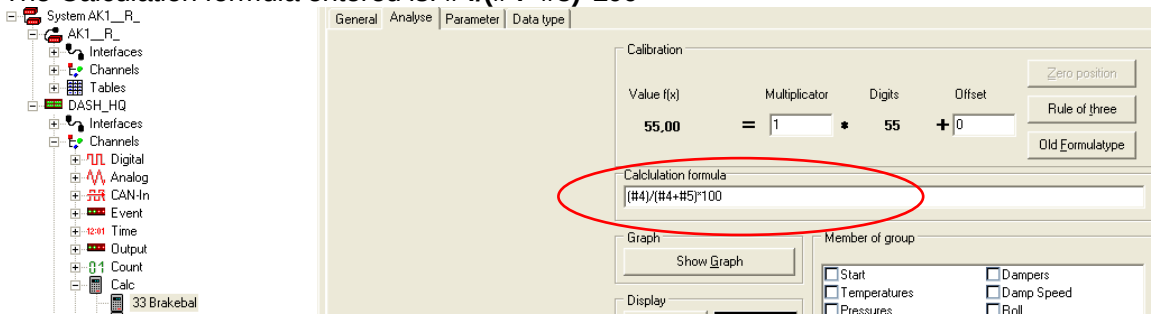
7.1.1 Brake Balance Calculation

The front and rear brake pressure values are measured by a connected data logging system, then sent to the 2D Dash via the CAN bus. The 2D Dash reads the values from the CAN bus on channel #4 (brake pressure front) and channel #5 (brake pressure rear).

Brake balance is defined equal to:

$$\frac{\textbf{100 * Brake pressure front}}{\textbf{(Brake pressure front + Brake pressure rear)}}$$

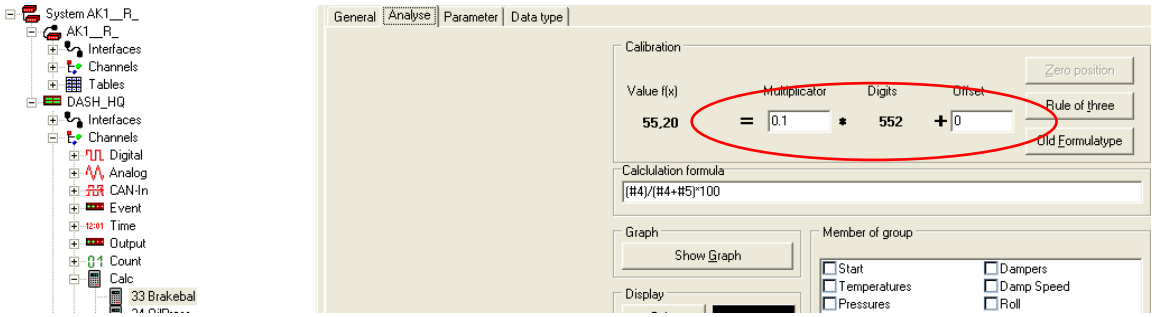
The Calculation formula entered is: **#4/(#4+#5)*100**



If the actual brake bias is **55.26%** (bias to the front), and a result with a resolution of 1% accuracy is acceptable, a <Multiplicator> of 1 can be used, giving a calc channel result of "55". If a higher resolution is needed, e.g. 0.1%, a multiplier of 0.1 would be used to give a channel result of "55.3". The <Offset> value can remain at the default value of "0".

By entering the channel numbers exactly ,e.g. #4, the physical values of this channel are used in the calculation. If the position of the channel changes, you must modify the Calc Channel!

Entering a smaller number inside the <Multiplicator> field will enable a higher level of resolution, with smaller steps between each measurement value change. If the entered value of <Multiplicator> becomes too low, e.g. 0.00001, the device can no longer display the full range of physical values. As a rule, only use a <Multiplicator> value that is practical for the application!

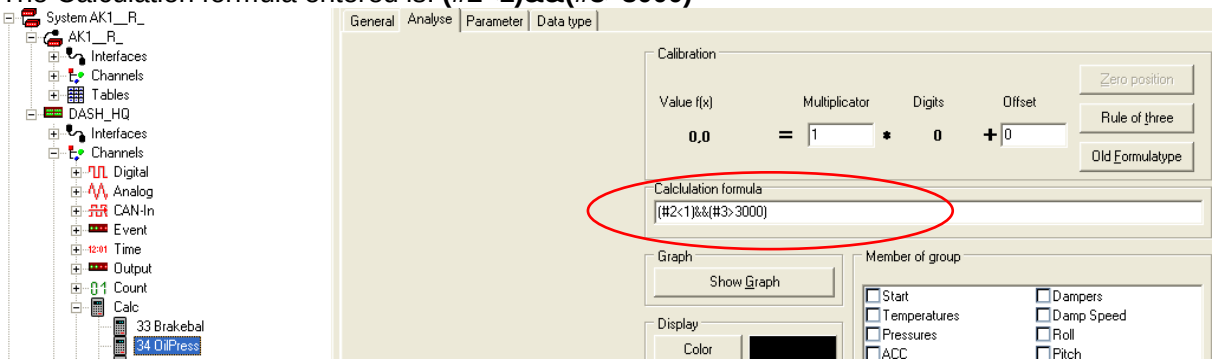


7.1.2 Triggering an alarm channel when Oil pressure is low

Under hard braking or cornering engine Oil pressure can suddenly drop and seriously damage an engine. If this occurs the driver should be warned by a dashboard alarm. With the engine on idle the Oil pressure is usually low but is of no significance as the engine is under no load. A Calc channel can be used to only trigger a low oil pressure warning when a genuinely problematic situation is occurring.

Channel #2 (RPM) and Channel #3 (Oil pressure) are read from the CAN bus. A channel is calculated that has the value of “1” when the engine is above 3000rpm **AND** the Oil pressure is below 1 bar, otherwise the Calc-channel has a value of “0”. An alarm channel will use this Calc channel as its trigger, turning on warning LED’s when a dangerous low oil pressure event happens (Refer to Alarms).

The Calculation formula entered is: **(#2<1)&&(#3>3000)**



A <Multiplicator> of 1 is entered as the channel can only have the values of 0 or 1 and therefore a resolution of 1 is appropriate. The <Offset> value can remain at the default value of "0".

When a calculated channel is created to give a Boolean output (0 or 1), the <Multiplicator> should always be 1!

7.1.3 Calculation of Fuel Consumption (Reset value when display is powererd off)

Using a fuel injector counter signal #Fuelcons, the total fuel injected to the engine can be calculated by the 2D Dash. It is already known that a single injector pulse deposits 0.032768 litres of fuel into the engine.

The Calculation formula entered is: **sum(if(der(#Fuelcons)<0,0.032768,0)))**

In this formula the derivation of the injector counter signal (#Fuelcons) is calculated. If that calculation turns negative, 0.032768 litres of fuel (depending on the injector signal) are added to the result so far. This is an accumulative calculation that will continue to increase in value until the engine is turned off.

For this calculation, removing the power supply from the 2D Dash will cause the calculation channel will start again from zero.

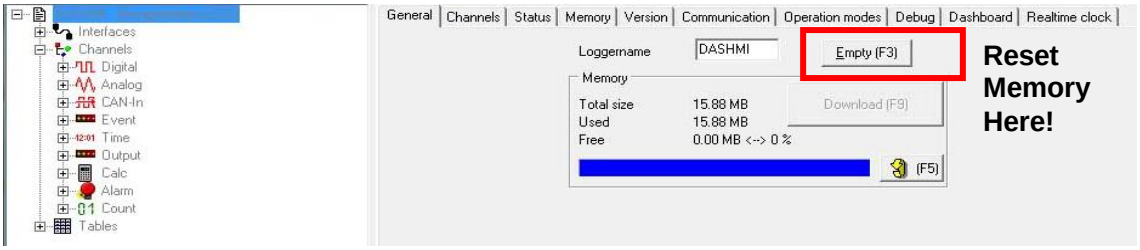
It is also possible to enter the actual channel name, e.g. #Fuelcons, instead of the channel number, e.g. #35. This is a powerful feature as the calculation will continue to work correctly if the channel position changes within the Can-In area of the 2D Dash.

7.1.4 Calculation of Fuel Consumption (Reset value via Button F3 in the program Winit)

Following a similar principle to the previous example, this formula provides a different method of resetting the fuel consumption value. This example formula will maintain the previous session value of Fuel consumption, even after power has been switched off!

The Calculation formula entered is: **m1=m1+(if(der(#Fuelcons)<0,0.032768,0)))**

The calculation occurs the same as before, only in this case the result is stored in the dashboard as long as it is not reset by the program Winit. Reset is performed using the button **<F3>** or **<Empty>**, as shown below. Performing this action will also reset any stored Laptime data from inside the 2D Dash!



7.1.5 Calculate the driven mileage (Reset in Winit via button F3)

Another example of using an accumulative calculation channel is explained below. Here the vehicle speed reference channel is used to determine the distance travelled during a session.
The Calculation formula entered is: **m1=m1+(#Speed*3.6/Samplingrate)**

The speed signal is multiplied by 3.6 and divided by the channel sampling rate to have a signal per second in meters. The accumulated value of distance travelled is **measured in metres**, and stored in the variable **m1**. As in the previous example, Reset is performed using the button **<F3>** or **<Empty>** within the 2D program WinIt.

7.1.6 Calculate the driven mileage (Permanent accumulation)

Alternatively by using the variables **p1** and **p2**, the driven mileage calculation can be permanently accumulated through the life of the vehicle (not resettable).
The Calculation formula entered is: **p1=p1+(#Speed*3.6/Samplingrate)**

7.1.7 Count the number of detonations in a run

The channel #Deto has a value of 1 when detonation occurs and 0 when the engine runs smoothly. The total value of detonation events are counted with the help of the variable m1.
The Calculation formula entered is: **m1=(m1+#Deto)**

7.1.8 Reset variable m1 via Button

Instead of connection to a PC, then using 2D program WinIt to reset the value of **m1**, an external button (Analog channel) can be used to reset the calculation value while on track. For this to work, an additional Calculation channel is created as follows.
The Calculation formula entered is: **if(#ANA1<2500,m1=0,0)**

The analogue signal of channel **#ANA1** is continuously checked, if its value drops below 2500 digits (short circuit to GND), the variable **m1** is reset to 0. In this example an analog signal was used to reset the value of **m1**, however CAN or Digital channels can also be used for this!

7.1.9 Counting the number of samples

If you require to count the number of samples, the Calculation formula entered is: **x+1**
At every sample point, 1 is added to the previous value of **x**. this process continues until power is removed from the 2D Dash.

Variable **x** will be set to 0 after the Dash is unpowered. Variable **x** represents the channel value of the previous sample.

Variables **m1** …**m6** keep the last value after power off and can be erased by pressing Empty(F3) in Winit.
Variables **p1** and **p2** keep the last value after power off and cannot be erased. They should be used for example as distance counter or lifetime counter.

7.2 Calculation functions of the 2D DASH

Calculation channels are generated by the combination of a <Multiplicator>, <Offset>, one or more data channels AND the use of **standard mathematical or logical functions**. The following functions can be used as entries to a Calc channel inside the 2D Dash:

Standard arithmetic functions:
+ (add), **-** (subtract), ***** (multiply), **/** (divide), **%** (modulo)

Extended arithmetic functions:
^ (exponential), **sqrt()** (square root), **ln()** (natural logarithm), **log10()** (base 10 logarithm), **abs()** (absolute value), **der()** (derivate), **sum()** (summation).

Comparative functions:
< (less than), **<=** (less than or equal to), **>** (greater than), **>=** (greater than or equal to),
== (equal to)

Binary functions:
& (binary AND), **|** (binary OR)

Logical functions:
&& (logical AND), **||** (logical OR), **!**(logical not), **~**(binary not)

Basic Trigonometric functions:
rad(), deg(), sig(), sin(), cos(), tan()

Complex Trigonometric functions:
asin(), acos(), atan(), dsin(), dcos(), dtan(), dasin(), dacos(), datan()

Min-max functions:
min(#x,#y) - calculated channel is the minimum of channel x and y
max(#x,#y) - calculated channel is the maximum of channel x and y
hmin(#x,t) - calculated channel holds the minimum of channel x for a time t in seconds
hmax(#x,t) - calculated channel holds the maximum of channel x for a time t in seconds

Conditional function:
if(defined logical condition, value if true, value if false) - a logical function is defined using one or more data channel, if function has a different value depending on the function being TRUE or FALSE.

Non-linear functions:
tab(table number, #channel or formula with channel) - using a customised lookup table loaded to the 2D Dash, a data channel (or formula with channel) is compared to the table to generate the corresponding value. This is used where a non-linear relationship exists between variables.

Filter functions:
flt(#channel, time in sec) - a moving average filter to smooth the data of a channel

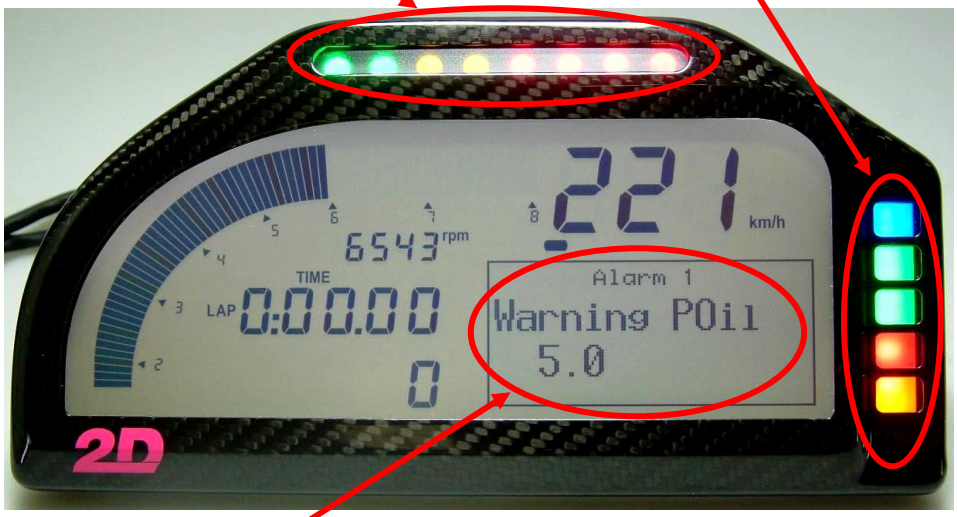
8 The Alarm function

An Alarm-channel is intended to alert the driver or rider about significant or undesirable events as the occur on the vehicle. Many options exist to create user-defined Alarms depending on the value of one or more channels.

Three types of alarm are available as shown below.

1 <Flashing LED's>
for shift lights and
emergency alarm

2 <Output Channel to Activate>
lamp illuminates to make alert



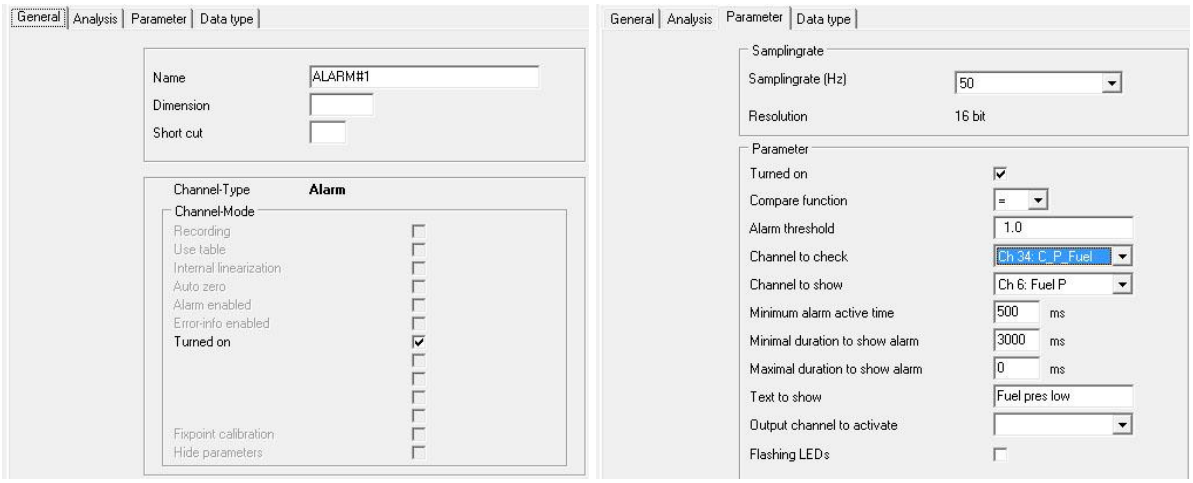
<LED_Bl>
<LED_Gr1>
<LED_Gr2>
<LED_Rd>
<LED_Or>

3 <Text to Show>
Channel name and
user-defined text is displayed

8.1 Defining an Alarm Channel

Select an available alarm channel, the number of alarm channels will be depending on what type of 2D Dash you are using.
After choosing an available channel, the following actions are required:

- Select the tab **<General>**, and tick the <Channel switched on> box as shown below.
- No actions are required inside the tab **<Analysis>**.
- Select the tab **<Parameter>**, choose from the many options that exist inside this tab. Each option is explained below.
- No actions are required inside the tab **<Data type>**.



The following parameters must be defined according to the requirements of the alarm channel you are creating. The image above illustrates the appearance of the Alarm options menu.

<Samplingrate> Sets the amount of checks of the alarm condition per second

<Compare function> Sets how the input channel for the alarm will be checked with the constant

<Alarm threshold> Is the threshold value used to trigger the alarm

<Channel to check> The channel to be checked for the alarm condition(s)

<Channel to show> The channel value that will be shown when warning is displayed

<Minimum alarm active time> Defines a threshold time in milliseconds that the alarm condition has to be active before warnings are shown.

<Minimal duration to show alarm> Is the duration the warning will be shown even if the alarm condition is not valid anymore.

<Maximal duration to show alarm> Is the maximum duration the warning will be shown even if the alarm condition is still valid. The alarm will be retriggered again if the alarm condition becomes valid again.

<Text to show> The entered text will be displayed on screen while the alarm is active.

<Output channel to activate> From the dropdown box, many available output channels (in the form of LED's)

<Flashing LED's> Top LED's of the Dash (normally used for shift lights) will flash while the alarm is active.

To show warnings for as long as the alarm condition is active, the duration time has to be set to 0.

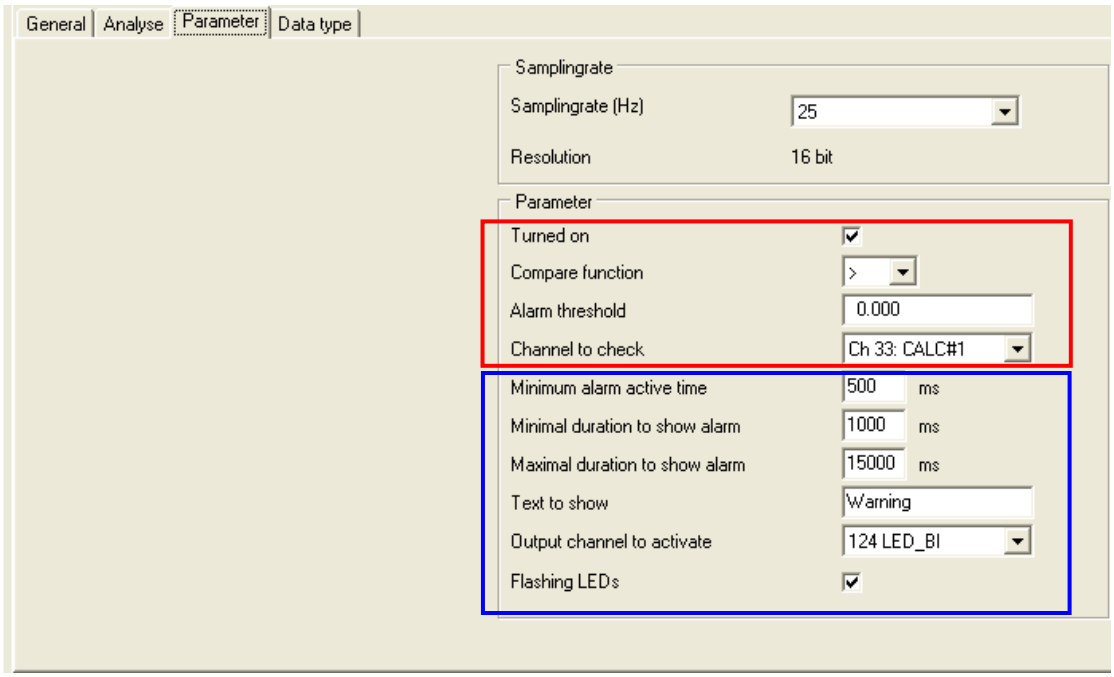
8.1.1 Example - Displaying an alarm when Oil pressure is low

If Oil pressure drops below 1 bar for more than 0.5 seconds with the engine RPM above 3000rpm, an alarm is to be displayed via a blue LED and a warning message shown for at least 1 second, on permanent error it is to be released after 15 seconds.

To trigger the alarm channel only when the dangerous low oil pressure event is happening, a Calc-channel is used to trigger the alarm function (Refer to the Calc channel example in Section 7.1.2 Triggering an alarm channel when Oil pressure is low).

For this example, the following actions were taken:
<Turned on> - ticked, make sure this is ALWAYS TICKED!
<Compare function> - ">" is selected so alarm will activate if the <Channel to check> has a value ABOVE the <Alarm threshold> value
<Alarm Threshold> - "0.000" entered, alarm will activate when checked channel rises above this
<Channel to check> - Calc channel for low oil pressure is selected, the value of this channel is compared to the <Alarm threshold> according to the <Compare function>

When the low oil pressure event occurs, the value generated by the Low Oil Pressure Calc channel is compared to the <Alarm threshold> according to the selected criteria (>, >=, etc). In this example, these options are selected so the alarm will activate when the value of the Calc channel rises above 0 (true).



As shown by the image above, additional options exist for the alarm channel definition. These options define how (and for how long) the alarm function will be displayed to the driver.

<Minimum alarm active time> - 500ms, alarm event must occur for 0.5 seconds before the alarm will become visible on the display
<Minimal duration to show alarm> - 1000ms, an activated alarm will be displayed for a minimum of 1.0 seconds, even if the alarm is no longer valid!
<Text to show> - "Warning" is displayed on the Dash while the alarm is active
<Output channel to activate> - "124 LED_Bl", blue LED at side-of-dash will become illuminated
<Flashing LED's> - this alarm is very important so LED's will flash to provide extra alert to driver

8.2 Alarm protocol for the AlStat Event Channel

Every alarm channel can be sent outside the 2D Dash to an external logging device via the CAN bus. This enables the system to record every different alarm states experienced at the 2D Dash. Using this function you can monitor the alarm messages given to the driver after a download.

Every channel has a 8 bit message where different Bits are turned on if the alarm is triggered. Each **<Bit>** of the AlStat channel have the following meanings.

Bit 1	Alarm channel is turned on
Bit 2	Alarm has a text output to show on the matrix if an alarm is active
Bit 3	Upper LED's will blink if an alarm is active
Bit 4	Right hand side LED's will be turned on if an alarm is active
Bit 5	Alarm is confirmed by a pressed button, alarm is oppressed until next triggering
Bit 6	Alarm is oppressed due to timeout
Bit 7	Alarm is given and alarm is showing on display if not suppressed by timeout or button
Bit 8	Alarm is triggered and shown

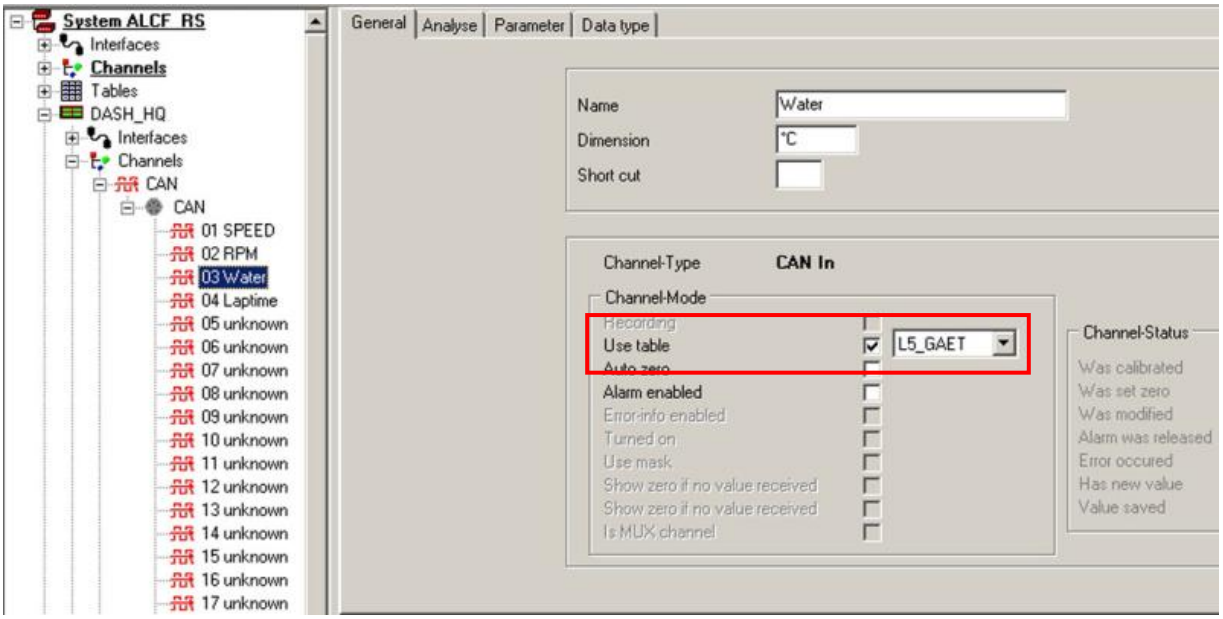
9 The Table function

A table enables the transform of a linear input into a nonlinear output. To each input value an output value is assigned according to the entries of the predefined table.

9.1 How to choose a table via WinIt

To explain the Table function, the value of a CAN channel will be transformed by a selected table.

- A CAN channel is selected (see Figure below)
- Select the tab ***<General>***, choose ***<Use Table>*** and select the table you want to use from the drop down list on the right (in the figure below the table is called ***<L5_GAET>***)
- To confirm the modifications, click the button ***<Apply>***



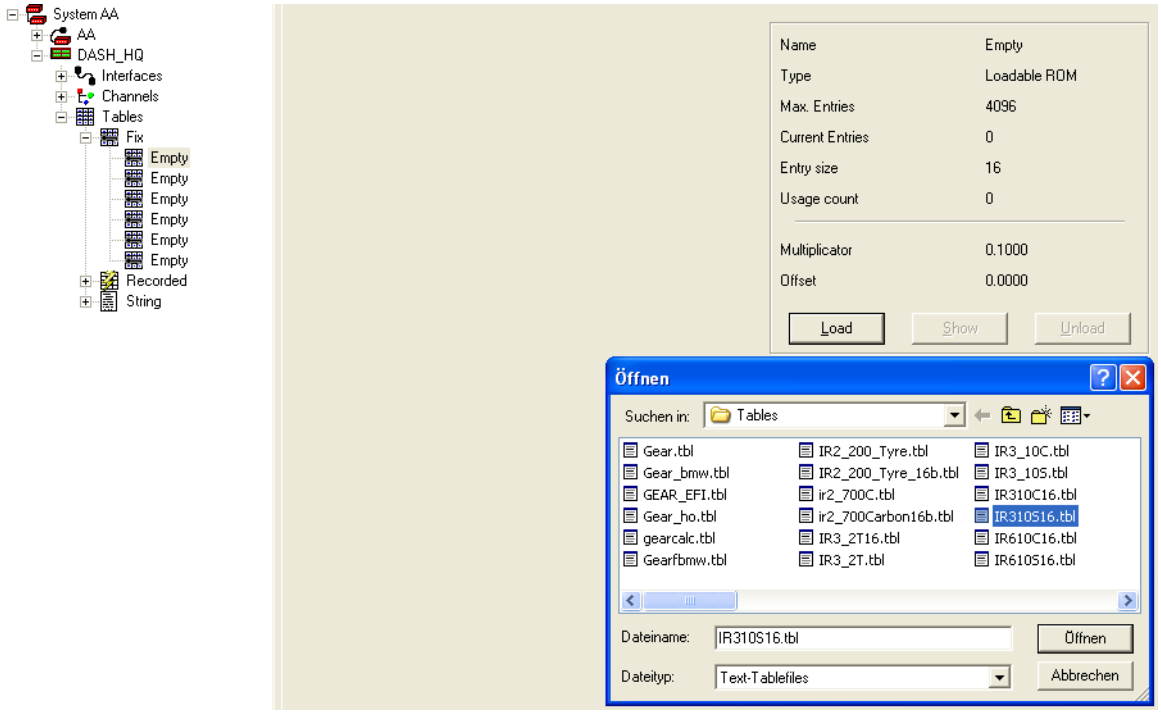
9.2 Types of tables

9.2.1 Fix

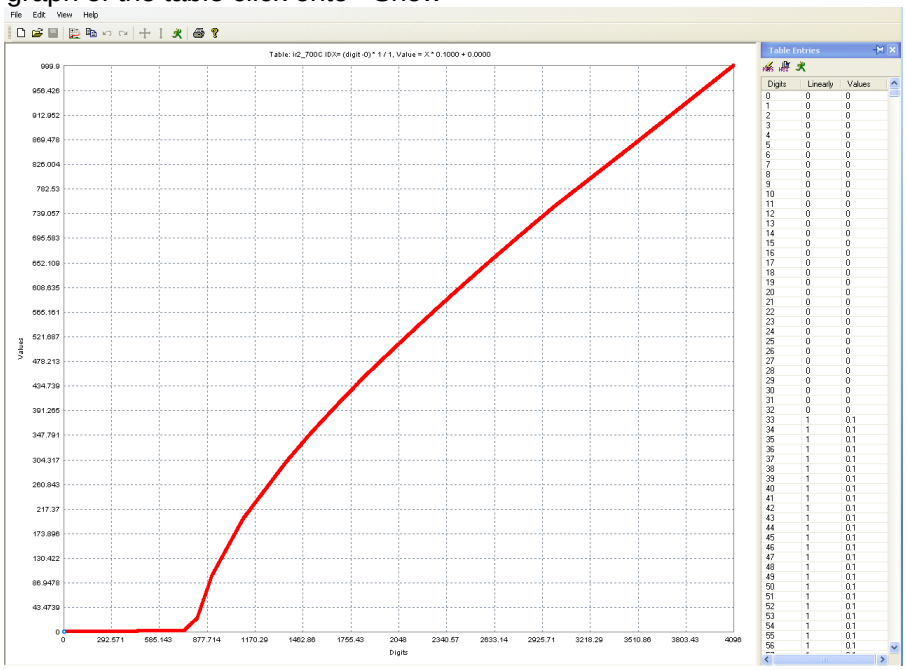
6 fixed tables can be saved into the permanent memory of the dashboard and can be assigned to any input channel.

2D delivers a set of predefined tables in the folder **Race_xx.y/System/Tables**. For examples of creating user defined tables refer to Section 12 Special Dashboard functions.

To save a table into the display click onto **<Load>** and choose the table in the explorer window.



To show the graph of the table click onto >Show>



To erase the table of the display memory click **<Unload>**

9.2.2 Recorded

9.2.2.1 Gap

The Gap-table contains the Section times for every 32 meters in the fastest lap.

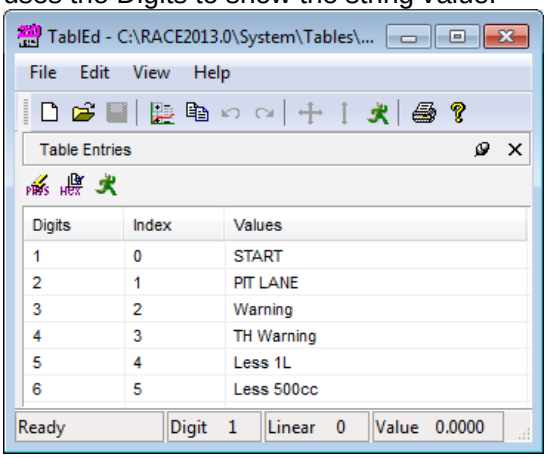
9.2.2.2 Laptime

The Laptime-table contains all the laptimes received by the display.

9.2.3 String STR1,STR2

The String-tables contain predefined messages which can be put onto the display as warnings or driver information.

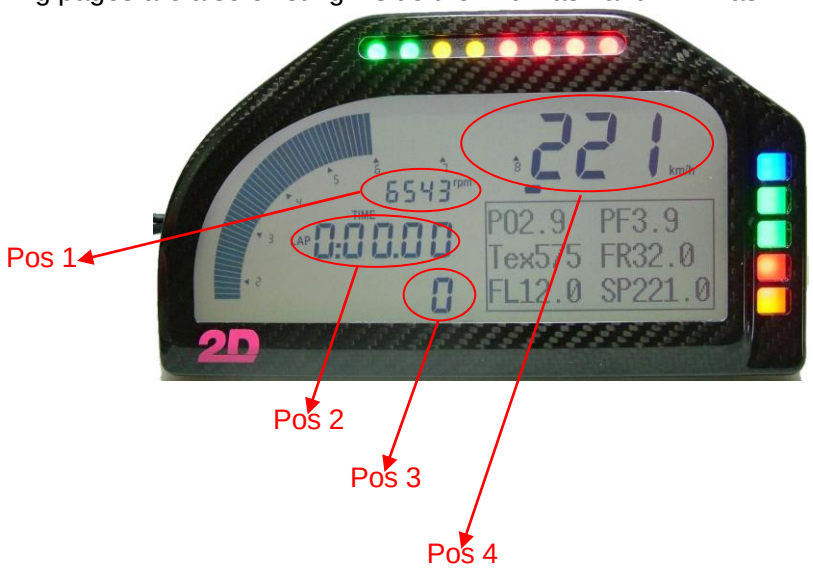
Be aware that there are differences when using string tables with a CAN channel or with the DIAG function. E.g when the channel value is 1 it will show “Pitlane” as the channel uses the Index of the table, but when the DIAG function is used and the channel value is 1 it will show “Start” as this uses the Digits to show the string value.



10 Special format functions

10.1 Format Functions - BigDash

Many areas of the 2D Dash can be formatted according to the units or scaling of the associated data channel. The output areas of the 2D BigDash are summarised below. Many of the formats listed in the following pages are also existing inside the MidiDash and MiniDash!



To activate fixed formats, go to ***<Parameter>*** in the input mask, tick the box **<Use format function>***and choose the de*sired ***<format function>*** in the drop down box (as shown below).

Pos 1

Dimension	Description
none	Shows channel dimensions like set on tab parameter with no unit; can be used for any channel type
rpm	Shows channel dimensions like set on tab parameter + unit “rpm” ; can be used for any RPM channel
k	Shows channel dimensions like set on tab parameter + unit “k” ; can be used for any distance channel
m	Shows channel dimensions like set on tab parameter + unit “m” ; can be used for any distance channel
km	Shows channel dimensions like set on tab parameter + unit “km” ; can be used for any distance channel

Pos 2

Dimension	Description
none	Shows channel dimensions like set on tab parameter with no unit, maximum 4digits are displayed ; can be used for any channel type
LAP	Shows channel dimensions like set on tab parameter + unit “Lap” , maximum 4digits are displayed ;should be used for channel Lapcnt
PO	Shows channel dimensions like set on tab parameter + unit “PO” , maximum 4digits are displayed ;can be used for any oil pressure channel
LAPTIM	Shows channel dimensions mm:ss.hh + unit “LAP TIME” ; can be used for any timing channel (GAP, Laptime, Sectiontime, Laptime_run)
FUEL	Shows channel dimensions like set on tab parameter + unit “FUEL” , maximum 4digits are displayed ;can be used for any fuel level channel
TIME	Shows current time in dimensions hh:mm + unit “TIME” ; display output is independent of channel
V	Shows channel dimensions like set on tab parameter + unit “V” , maximum 4digits are displayed ;can be used for any speed channel
MIN	Shows channel dimensions like set on tab parameter + unit “MIN” , maximum 4digits are displayed ;can be used for any channel
VMIN	Shows channel dimensions like set on tab parameter + unit “VMIN” , maximum 4digits are displayed ;can be used for any speed channel
F	Shows channel dimensions like set on tab parameter + unit “F” , maximum 4digits are displayed ;can be used for any channel
R	Shows channel dimensions like set on tab parameter + unit “R” , maximum 4digits are displayed ;can be used for any channel

Pos 3

Dimension	Description
none	Shows channel dimensions like set on tab parameter with no unit, maximum 4digits are displayed ; can be used for any channel type
U	Shows channel dimensions like set on tab parameter + unit “U” , maximum 4digits are displayed ;can be used for any voltage channel
UB	Shows channel dimensions like set on tab parameter + unit “UB” , maximum 4digits are displayed ;can be used for any battery voltage channel
V	Shows channel dimensions like set on tab parameter + unit “V” , maximum 4digits are displayed ;can be used for any speed channel
MAX	Shows channel dimensions like set on tab parameter + unit “MAX” , maximum 4digits are displayed ;can be used for any channel
VMAX	Shows channel dimensions like set on tab parameter + unit “VMAX” , maximum 4digits are displayed ;can be used for any speed channel
BEST_T	Shows channel dimensions mm:ss.hh + unit “BEST” ; should be used for channel Laptime to show the best laptime since power up
GAP_T	Shows channel dimensions ss.h + unit “GAP” ; should be used for channel GAP to show the time difference to the best laptime since power up
F	Shows channel dimensions like set on tab parameter + unit “F” , maximum 4digits are displayed ;can be used for any channel

R	Shows channel dimensions like set on tab parameter + unit “R” , maximum 4digits are displayed ;can be used for any channel
PF	Shows channel dimensions like set on tab parameter + unit “PF” , maximum 4digits are displayed ;can be used for any fuel pressure channel
TW	Shows channel dimensions like set on tab parameter + unit “TW” , maximum 4digits are displayed ;can be used for any water temperature channel
TOTAL	Shows channel dimensions like set on tab parameter + unit “TOTAL” , maximum 4digits are displayed ;can be used for any channel
BEST	Shows channel dimensions like set on tab parameter + unit “BEST” , maximum 4digits are displayed ;can be used for any channel

Pos 4

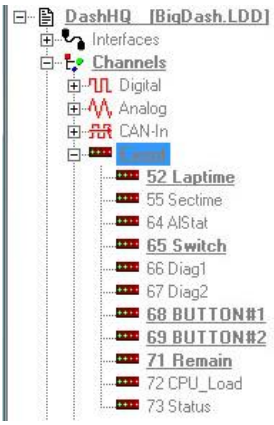
Dimension	Description
none	Shows only pre dot channel values with no unit, maximum 3digits are displayed ; can be used for any channel type
min	Shows only pre dot channel values + unit “min”, maximum 3digits are displayed ; can be used for any channel type
mph	Shows only pre dot channel values + unit “mph”, maximum 3digits are displayed ; can be used for any speed channel
km/h	Shows only pre dot channel values + unit “km/h”, maximum 3digits are displayed ; can be used for any speed channel
LAP	Shows only pre dot channel values + unit “LAP”, maximum 3digits are displayed ; should be used for channel Lapcnt
Gear	Shows only pre dot channel values + unit “Gear”, maximum 1digit is displayed ; can be used for any gear channel
SPEED	Shows only pre dot channel values + unit “SPEED”, maximum 3digits are displayed ; can be used for any speed channel

11 Setting properties of the 2D Dash with 2 buttons

11.1 Setting BigDash with 2 buttons

Various settings of the BigDash can be made without a connected computer, but by using 2 connected buttons on the dash.

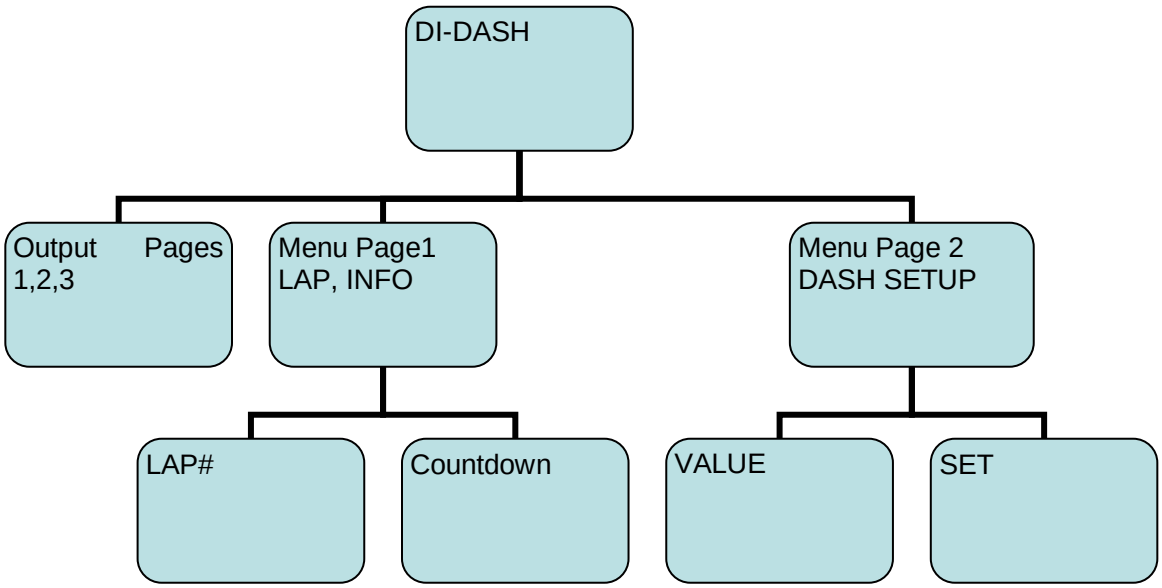
In order to use the 2 input buttons they have to be activated via Winit as shown below.



Nr	Recor	On	Name	Sampli	Multiplicator	Digits	Offset
52		☐	Laptime	200	0,005	8820	0,000
55		☐	Sectime	200	0,00	39483	0,00
64			AlStat	200	1,00	0	0,00
65		☐	Switch	25	1,00	0	0,00
66		☐	Diag1	25	1,00	0	0,00
67		☐	Diag2	25	1,000	0	0,000
68		☒	BUTTON#1	25	1,000	0	0,000
69		☒	BUTTON#2	25	1,00000	0	0,00000
71		☐	Remain	1,	1,000	0	0,000
72			CPU_Load	25	1,00	0	0,00
73			Status	1,	1,000	0	0,000

11.1.1 BigDash Menu structure

The following picture shows the menu structure of the 2D Dash.
Button 1 - is used to **browse vertically**, to **change values** and **confirm actions**
Button 2 - is used to **browse vertically in the submenus**



11.1.1.1 LAP#

Shows the number of driven laps with elapsed time and fastest laptime.

11.1.1.2 Countdown

Start and stop the countdown by pressing button 1.

11.1.1.3 Value

The main properties of the 2D DASH are set in the submenu “Value”. Choose the value to be set and jump to submenu “SET” to change the value. To change time and date, set to zero and reset the dash keep button 1 pressed for 5 seconds.

The following values can be changed:
Dash Modes - Race, Endurance, Road, Delta Sections
Time - current time in the 2D DASH
Date - current date in the 2D DASH
Flash Brightness - sets the luminosity of the LED’s
Countdown - start time of the Countdown
Set to zero - sets all “Autozero” channels to 0
Reset Dash - reset the 2D Dash to factory default settings

If the 2D DASH is set back to factory defaults, all previous settings will be lost ! The display cannot be set up by the buttons as button 2 is deactivated by default.

12 Special Dashboard functions

12.1 Gear-dependent shift light settings

The gearshift lights (LED's) of the 2D Dash can be switched on at different RPM values according to the gear that is selected.

Required Channels
To make this setting, the dashboard must have access to valid data channels for:

- Engine RPM
- Selected Gear (either calculated or measured directly from a gearbox sensor)

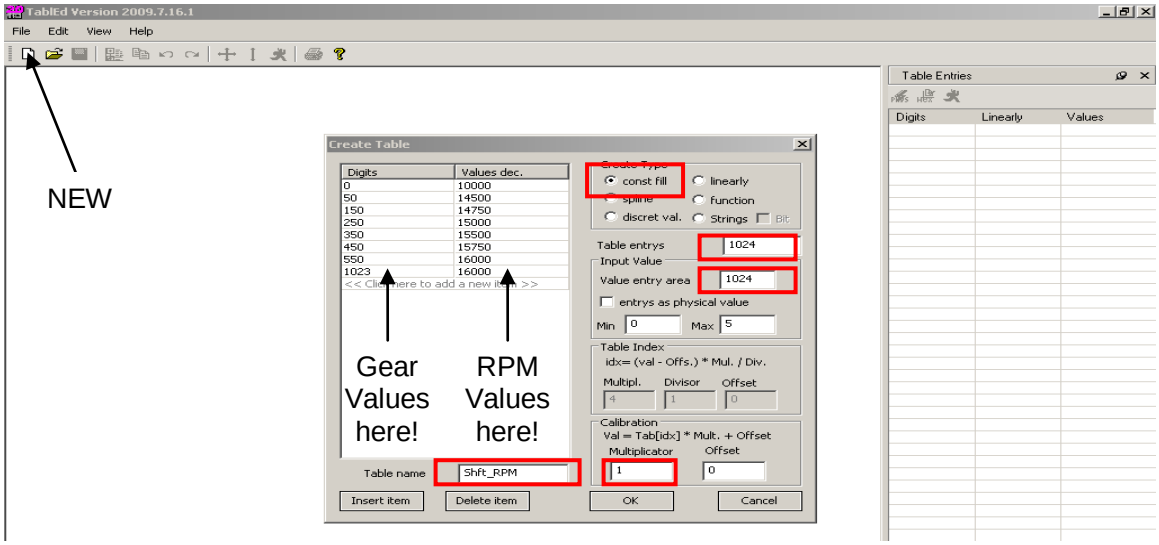
If there is no available channel for selected gear, this can be calculated from wheel speed and RPM. See Chapter 12.2.

It is important that whatever the method of determining its value, the selected gear data channel must be represented by whole numbers (integers) where 0 = neutral, 6 = 6th ,etc.

Step 1: Create a Lookup Table
A lookup table must be produced to define the RPM value at which the final (highest) shift light will switch on at the Dash for each gear.

The required actions are as follows:

- Open the 2D program "**TablEd.exe**", located inside the Race_xx.y installation folder
- Click "**New**" to open the "Create Table" Window (as shown below)
- The **Gear** and **RPM** values are entered in the areas shown below



- Ensure the "**Create Type**" field is set to "**const fill**"
- Set the "**Table entrys**" field to "**1024**"
- Set the "**Value entry**" area to "**1024**"
- Set the "**Multiplicator**" field to "**1**"

- Name the table in the "**Table Name**" field, e.g. **"Shft_RPM"**, then click "**OK**"
- Next go to "**File**" and click "**Save**" to permanently store this table inside your computer
- To correctly save the table, save the file with the same name as defined above, e.g.**'Shft_RPM'**, AND BE SURE OF THE DIRECTORY INTO WHICH IT IS SAVED.

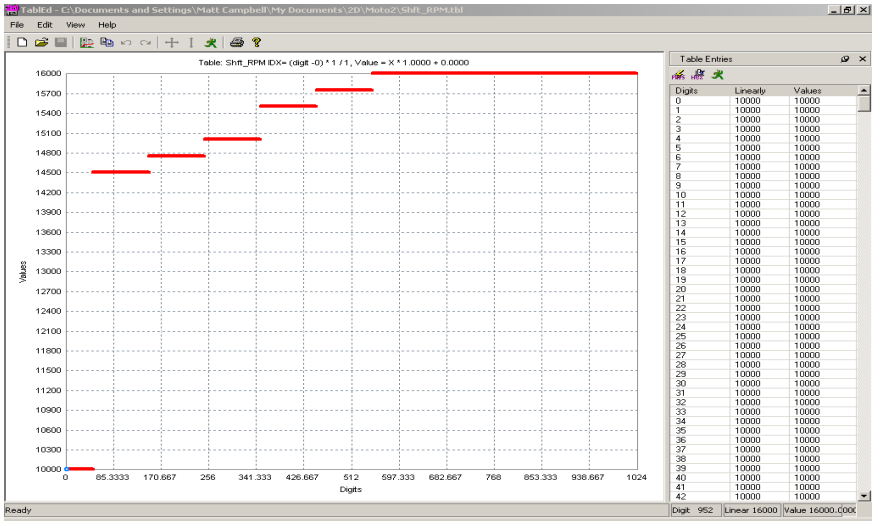
The defined table is used to assign the relationship between the selected gear and the RPM value that the dash must display on the shift lights for each gear.

However to make this function operate correctly it is necessary to modify gear input values, as shown by the example below. The table on the left side shows the RPM values that gear shift should occur for each gear. The table on the right shows the **Table Inputs** that are required to achieve the **Desired Settings** shown on the left.

Desired Settings	
Selected Gear	**SHIFT RPM**
0	10000
1	14500
2	14750
3	15000
4	15500
5	15750
6	16000

Table Inputs	
Digits	**Value Dec**
0	10000
50	14500
150	14750
250	15000
350	15500
450	15750
550	16000
1023	16000

Inputting the gear values (under column **digits**) improves the effectiveness of this dashboard configuration. Once the table inputs are defined and the table is saved, it can be viewed as shown below.

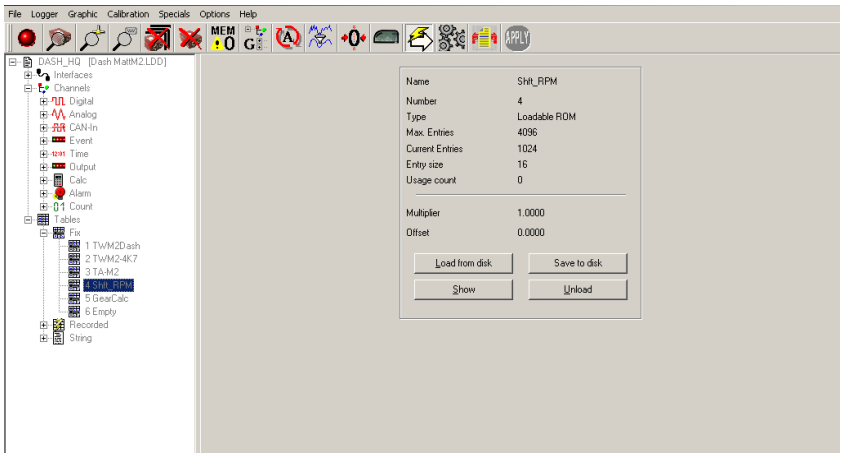


This is the data table from which the Dash will determine what RPM values to switch on the shift lights when a gear is selected.

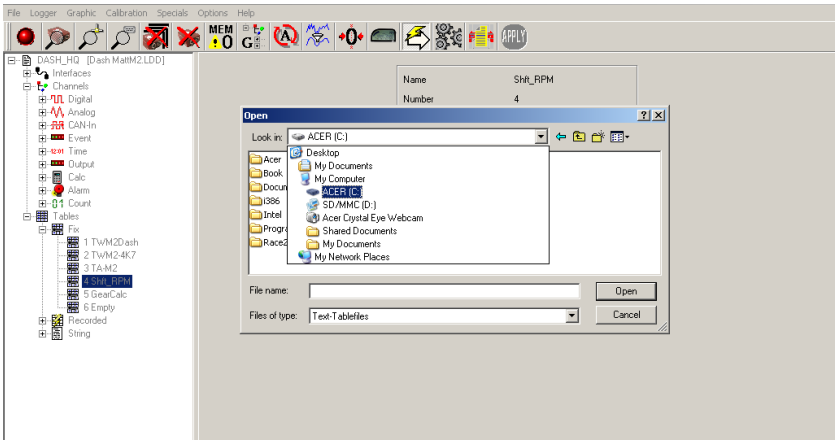
Step 2: Load Table to the Dash

Next the defined table must be loaded to the 2D Dash. This is performed in the 2D program WinIt by the following actions:

- Select the node "**Tables**" from the system tree
- Expand the "**Fix**" section within the "**Tables**" node
- Select any of the 6 spaces available for loading tables, but be sure not to select a space that is already being used for another dash function! Remember which space number is chosen as this is important later!
- Click **<Load from disk>** on the main window



- Locate the directory into which the "**Shft_RPM**" table was saved
- Select the table "**Shft_RPM**" and click **<Open>**



Now the table is correctly defined, and loaded to the Dash. It can now be used within a calculation channel.

Step 3: Configuration of Calculation Channel

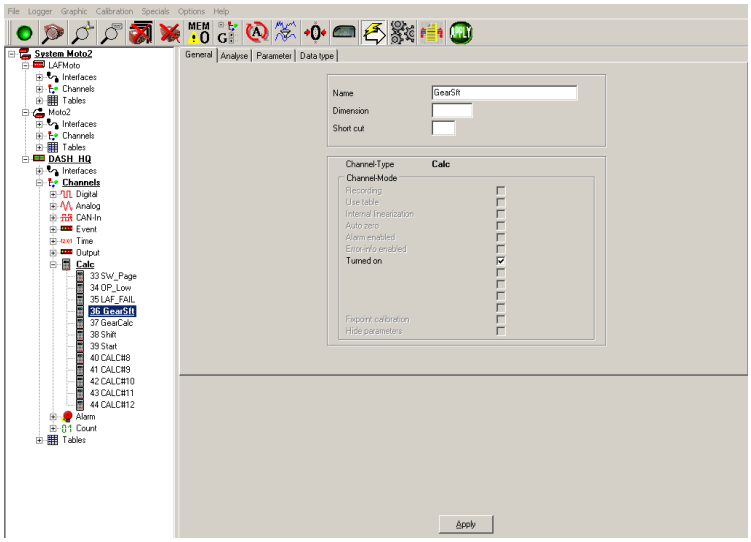
Next a Calculation channel is defined, this is the channel that will be used as the data input for the 2D Dash shift lights. This must be made to correctly operate with the Selected Gear Table that was loaded to the Dash.

The function of the CALC Channel is complex as it will:

- Sample the currently selected gear of the vehicle
- Determine the required RPM shift value from the loaded table for the selected gear
- Compare the RPM shift value to the current engine RPM
- Generate a fixed point number which can be used to control the shift lights

The following actions are required for making the CALC Channel setting:

- Expand the node “**Calc**” from the system tree of the 2D Dash by clicking “**+**”
- Select one of the 12 available Calc channels; make sure it is not already being used!
- Under the tab “**General**”, assign a name to the channel, e.g. “**GearSft**”
- Also check the box “**Turned on**” to activate the channel



- Under the tab <**Analyse**>, set the field “**multiplier**” to "**1**" and “**offset**” to "**0**"

In the tab <**Parameter**>, the formula for the Calc Channel is defined as follows:

- In the field “**Calculation formula**” the channel formula with the following format is entered “(<RPM> /(tab(<shift table position>,(< gear>)*100)))*10000”

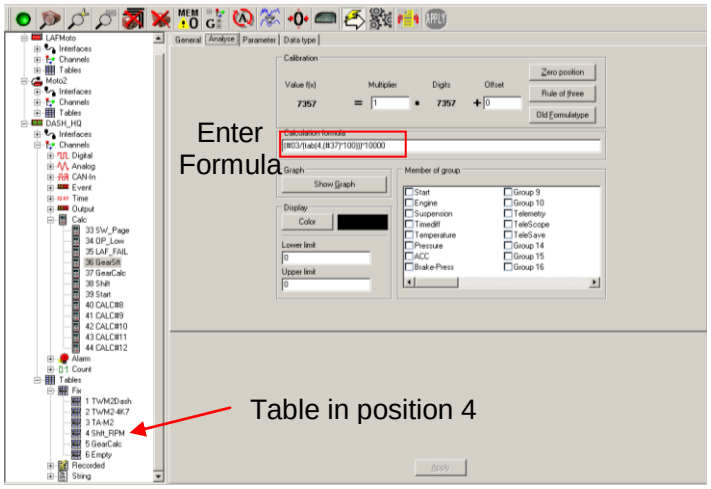
Notes:
<**RPM**>, insert the channel containing the engine RPM Data, e.g. **#03**
<**shift table position**>, input the position number of the saved table, e.g. **4**
<**gear**>, insert the channel containing the selected gear data, e.g. **#37**

In this example the input formula was **“(#03/(tab(4,(#37)*100)))*10000**”.

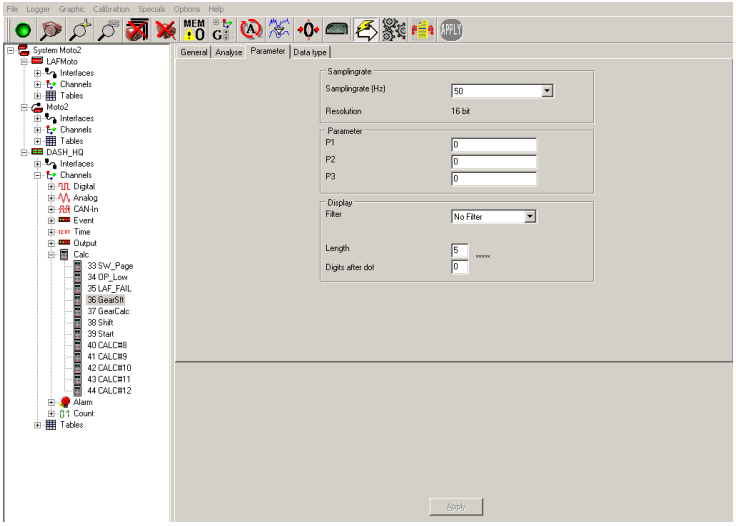
The formula is initially a simple quotient between the measured RPM and the preset shift RPM for the selected gear (from the loaded table). If the measured RPM is half of the shift RPM value, the formula equals “0.5”. If the measured RPM is equal to the shift RPM, the formula equals “1”.
However the final stage of the calc channel involves the calculated number being multiplied by 10000. Therefore if the above values were true, 5000 and 10000 would be respectively generated by the calc channel.

Finally the settings are made under the tab <**Parameter**>.

- Ensure that the Sampling rate selected is 50 Hz.
- No filter setting is necessary.



Also under the tab <**Data Type**> ensure the “Digits Signed” box is NOT checked.



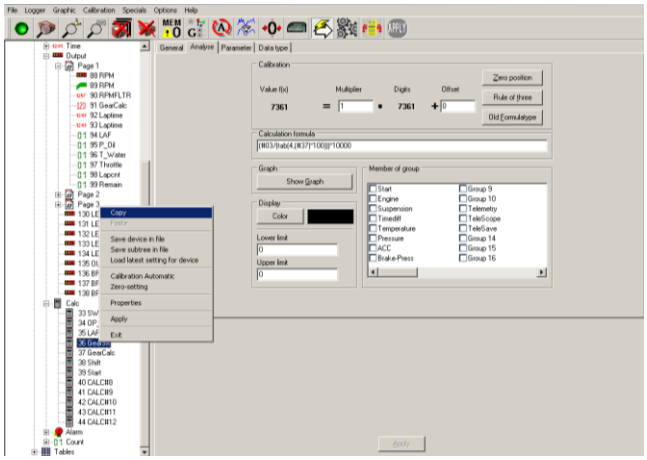
This completes the configuration for the Calculation channel.

Step 4: Configuration of Dash Output

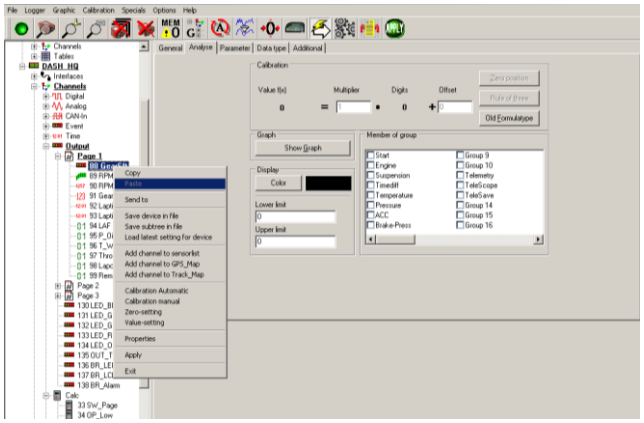
Next the 2D Dash settings for the shift lights are defined.

The following actions are required:

- Select the calculation channel "**GearSft**" from the system tree
- Make a right mouse-click and select **<Copy>**



- Select "**Output**" from the Dash system tree, open the node "**Page 1**"
- Right click on top channel of Page1, and select **<Paste>**. This assigns the calculation channel data to the shift lights of page 1 of the Dash



Next the settings of the shift lights are defined by selecting the tab <**Additional**>.

Index	Value
1	6000
2	7000
3	8000
4	9000
5	9500
6	9750
7	9900
8	10000

The definition of rows 1-8 control the switch on values of shift lights 1-8. To illuminate shift light 8 when the predefined shift RPM has been reached for the selected gear, set the value of row 8 to be "**10000**". This will make the final (right side) shift light illuminate when the correct shift RPM has been reached for the current gear selected. The final 4 shift lights (rows 5-8) can be set to "**10000**" to make all of these lights switch on to give a more clear gear-shift indication!

Alternatively, a good setting value for row 7 might be 9900 (99% of the maximum value 10000). The remaining values of the lights can be set in a similar way, e.g. row 1 = 6000, row 2 = 7000, row 3 = 8000, row 4 = 9000, row 5 = 9500, row 6 = 9750. The entered values can be adjusted to give the correct RPM spacing from the optimal shift RPM.

Finally click **<Apply>** to complete the gear-specific shift light settings.

Below a table shows the RPM values that occur when each shift light switches on for example settings just explained. The table shows just how customisable each shift light can be when the gear dependent shift light settings are used.

Selected Gear	RPM Values for Shift Light Switch							
	1	2	3	4	5	6	7	8
neutral	6000	7000	8000	9000	9500	9750	9900	10000
1	8700	10150	11600	13050	13775	14138	14355	14500
2	8850	10325	11800	13275	14013	14381	14603	14750
3	9000	10500	12000	13500	14250	14625	14850	15000
4	9300	10850	12400	13950	14725	15113	15345	15500
5	9450	11025	12600	14175	14963	15356	15593	15750
6	9600	11200	12800	14400	15200	15600	15840	16000

12.2 Online Gear Calculation

The 2D Dash can be used to display the current gear selected by the transmission. In some cases the gearbox of your vehicle is not fitted with a sensor to directly measure the selected gear position.

The 2D Big Dash is capable of accurately calculating the gear selected by your vehicle at any moment, and can display this information to the driver or rider while they are racing.

To determine the selected gear, the following actions must be taken:

- Create a Calculation channel to determine the overall gearing ratio of the vehicle as it is driving
- Create a lookup table to establish the relationship between the calculated overall gearing ratio and the selected gear number (1-6, 1-5, etc) of the vehicle

Step 1 Creating the Calculation channel

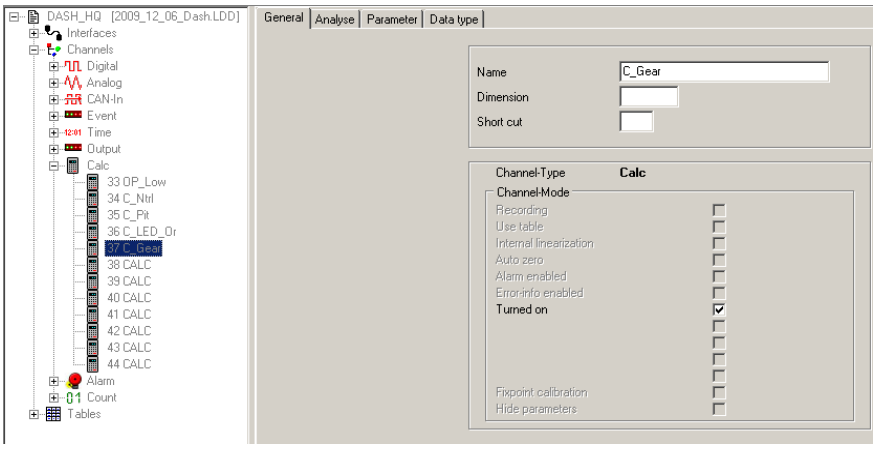
To start, a measured data channel must exist for engine RPM. Also a channel that varies proportionally to the vehicle speed is required. Examples of such a channel include a direct measurement of wheel speed (V_Rear), or a GPS speed channel (V_GPS).

$$\text{Vehicle Gearing Ratio} = \frac{\text{Vehicle Speed Reference}}{\text{Engine RPM}}$$

Using a GPS speed channel provides a more smooth-changing data trace with undesired effects removed (locking, spinning, etc).

To make the Calculation channel setting:

- Expand the node “**Calc**” from the system tree of the 2D Dash by clicking “**+**”
- Select one of the 12 available Calc channels; make sure it is not already being used!
- Under the tab <**General**>, Assign a name to the channel, e.g. “C_Gear”
- Also check the box “**Turned on**” to activate the channel



Next under the tab <**Analyse**>, set "**multiplier**" to 1 and "**offset**" to 0
The Calculation formula must be defined with the form:
"tab(<lookup table position>,(<speed channel>/< RPM>)*10000)"

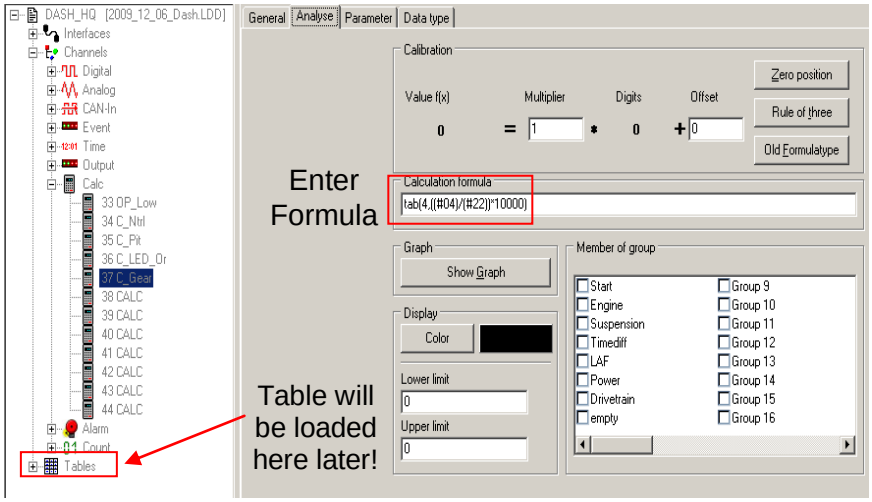
NOTES:
The <lookup table position> must be defined in the calculation so that the dash understands where to find the correct selected gear values. At this stage the lookup table has not been defined or loaded to the dashboard. These steps will be explained after this calculation channel, so for now consider the lookup table position as equal to "1".

The <speed channel> that is entered is the CAN In channel number of the speed reference you want to use in the calculation (e.g. RPMSprkt, V_Rear, V_GPS). In this example "#04" is entered to represent "RPMSprkt", as used for the setting of the Moto2 gear calculation.

Into the field <RPM> you must input the CAN In channel number for your engines measured RPM value. In this example "(#22)" is entered to the calculation formula.

The result of the calculation is most likely to be a number between 0 and 1, e.g. 0.3854. To represent such a number in the datalogger, and also to generate the lookup table, it easier to use full numbers (integers). Therefore the result of the calculation is multiplied by **10000** to instead make a rounded number, e.g. 3854.

For this example, the exact input formula for the calculation is: **tab(1,((#04)/(#22))*10000)**, as shown below.



Inside the tab <**Parameter**>, no filter is necessary.
Also under the tab <**Data Type**>, ensure the "Digits Signed" box is NOT checked.

This completes the configuration for the Calculation channel.

Step 2 Defining the gearing ratios of the vehicle

The lookup table must be produced to define the precise relationship between the calculated vehicle gearing ratio and the gear that is selected when that gearing ratio is made. Therefore you must know the gearing ratios of your vehicle before you can make the table!

If you are using the gearbox output shaft as the speed reference for the gear ratio calculation (as used for Moto2), and already know the gear ratios used by your vehicle, the generation of the lookup table is relatively easy. If you are using the vehicle speed as the speed reference, the total gearing ratio of the vehicle must be determined (including all speed reduction within the engine, gearbox, and final drive sprockets or differential).

The total gearing ratio of the **engine** is calculated for each gear by multiplying the primary gear ratio by each selectable gear ratio. The results of these calculations are shown below in column 3.

The engines overall gear ratios (for each gear selected) have values between 0 and 1. By multiplying by 10000 the gearing ratio can be expressed as whole numbers. This will make it easier to construct the lookup table!

Gear Step Name	Gear Step Ratio	Overall Gear Ratio (Primary x Gear)	Overall Gear Ratio x 10000
Primary	36/76	NA	NA
Gear 1	15/39	0.1822	**1822**
Gear 2	16/32	0.2368	**2368**
Gear 3	18/30	0.2842	**2842**
Gear 4	18/26	0.3279	**3279**
Gear 5	23/30	0.3632	**3632**
Gear 6	24/29	0.3920	**3920**

If you do not know the gear ratios used, the 2D program Anna-Liza can be used to attain a good approximation of the ratio between engine RPM and the speed reference you have used in your formula for the calculation channel “C_Gear”. It is required that you make the first run of your event with no gear displayed on the dash, during which data is recorded for engine RPM and the speed reference channel you want to use.

After downloading the data from your first measurement you can use Anna-Liza to make an **XY plot** of engine RPM and the speed reference used for the gearing calculation, e.g. wheel speed. See example below.

The above plot shows 6 distinctive straight lines, each demonstrating the 6 different gearing ratios that occur when each gear of the vehicle is selected. If you move the cursor about the screen, the X and Y values of the cursor position are viewable at the bottom of the screen (see above).
By placing the cursor on each of the gear ratio “trend lines”, exact values of engine RPM and vehicle speed are shown at the bottom of the screen for each gear.

The ratios determined from the XY plot are shown below.

Gear	X Value	Y Value	Overall Gear Ratio	Overall Gear Ratio x 100000
Gear 1	131.5	14362	0.00916	916
Gear 2	170.3	14733	0.01156	1156
Gear 3	203.3	14659	0.01387	1387
Gear 4	241.6	15122	0.01598	1598
Gear 5	276.0	15586	0.01771	1771
Gear 6	299.0	15640	0.01912	1912

By taking the vehicle speed (X Value) and dividing by the engine RPM (Y Value), the Overall Gear Ratio of the vehicle is determined. As was previously explained, it is better to multiply the resulting gear ratio by a large number, so it can be expressed as whole numbers in the lookup table. In this example, using the vehicle speed (measured at driven wheel or by GPS), the gearing ratio is multiplied by 100000. This is required because the overall gearing ratio you calculate using road speed as the speed reference is much smaller in value.

It is important that the number you multiply by in determining the table input values (e.g. 100000) is the same multiplication number used in the calculation channel!

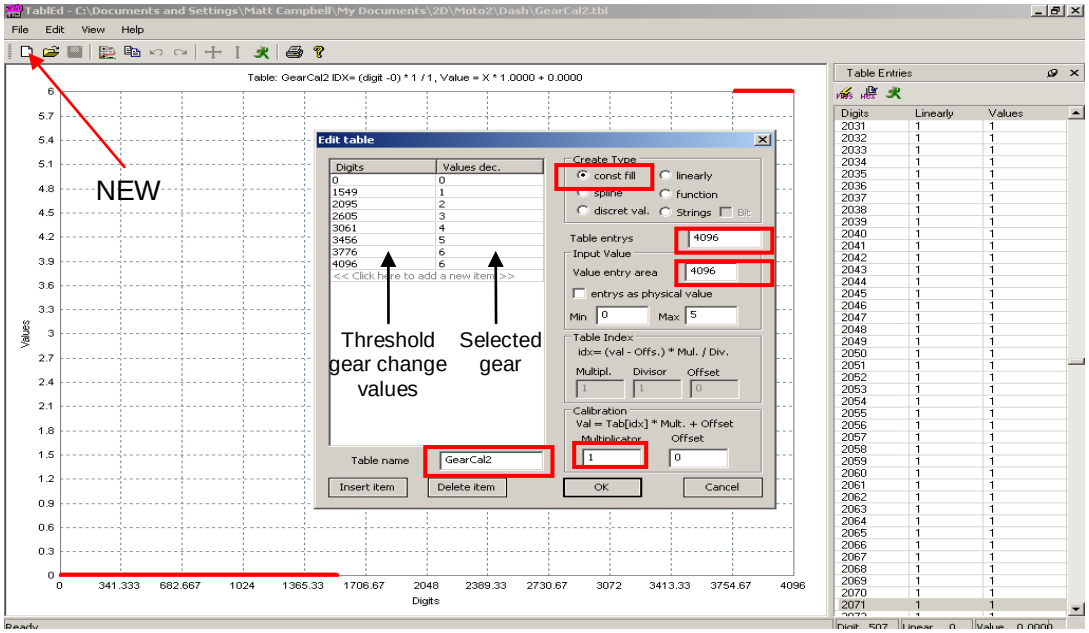
Step 3 Defining the lookup table

Before you make the lookup table you must be certain that:

- You have generated a calculation channel which will calculate the actual gearing ratio used by the vehicle, and expresses the number as an integer (whole number)
- You have attained understanding of the overall gearing ratios used by the vehicle and can predict accurately the value that will be made by the calculation channel when the vehicle is racing

To assemble the lookup table:

- Open the 2D program "**TablEd.exe**", found inside the Race_xx.y installation folder
- Click **<New>** to open the "Create Table" Window (as shown below)



- In the column "**Digits**" you enter the threshold values for defining what selected gear will be determined by the 2D Dash.
- In the column "**Values dec**" you enter the gear number that is selected when the corresponding Digits are input to the lookup table

The values that you enter to the table are not the same as the values generated by the calculation channel!

Instead the values are modified to equal the 'middle point' between each gear ratio (as generated by the calculation channel). This gives increased reliability of the gear calculation function. The modifications are necessary to ensure that the correct gear is displayed on the dash, even if the calculated value of gear ratio varies slightly as the vehicle is racing.

Examples are given below to further explain the process of making the table entry values. The values you must enter to the table are highlighted in green.

Determining the table entry values for the Moto2 bike (speed reference = RPMSprkt).

Selected Gear	Value generated by Calc Channel	Threshold values calculation	Required Table Entries	Gear Value Entries for Table
Neutral	0	0	0	0
Gear 1	1822	0.5*(1822-0) + 0	911	1
Gear 2	2368	0.5*(2368-1822) + 1822	2095	2
Gear 3	2842	0.5*(2842-2368) + 2368	2605	3
Gear 4	3279	0.5*(3279-2842) + 2842	3061	4
Gear 5	3632	0.5*(3632-3279) + 3279	3456	5
Gear 6	3920	0.5*(3920-3632) + 3632	3776	6
		Max table value = 4096	4096	6

Determining the table entry values for another application (speed reference = V_Rear or V_GPS).

Selected Gear	Value generated by Calc Channel	Threshold values calculation	Required Table Entries	Gear Value Entries for Table
Neutral	0	0	0	0
Gear 1	916	0.5*(916-0) + 0	458	1
Gear 2	1156	0.5*(1156-916) + 916	1036	2
Gear 3	1387	0.5*(1387-1156) + 1156	1272	3
Gear 4	1598	0.5*(1598-1387) + 1387	1493	4
Gear 5	1771	0.5*(1771-1598) + 1598	1685	5
Gear 6	1912	0.5*(1912-1771) + 1771	1842	6
		Max table value = 4096	4096	6

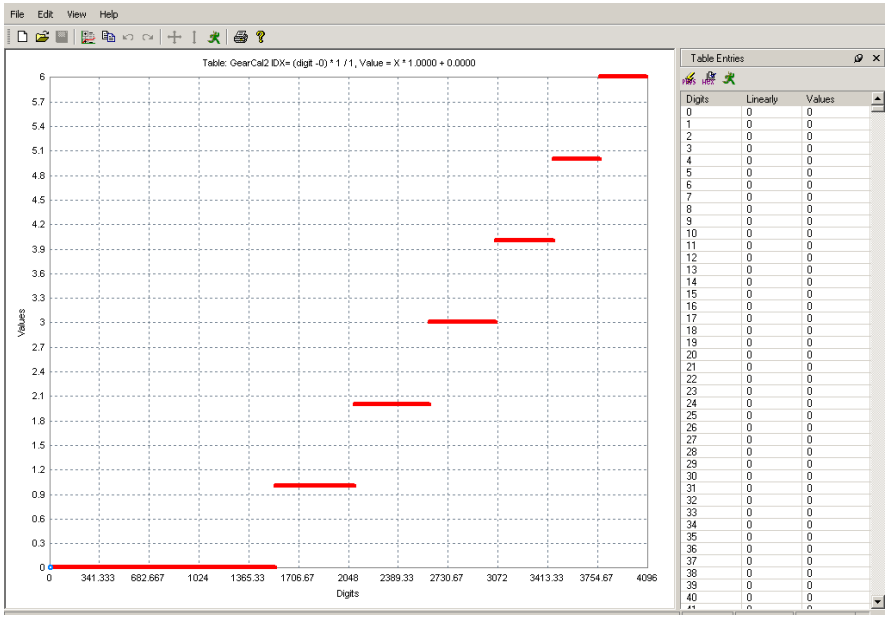
In either case, the following settings are made to complete the lookup table:

- "**Create Type**" is set to “const fill”
- Set "**Table entrys**" to 4096
- Set "**value entry area**" to 4096
- Set “**Multiplicator**” to 1 and make “**offset**” equal to 0
- Name the table in the "**Table Name**" field, e.g.’GearCal’, then click “**OK**”

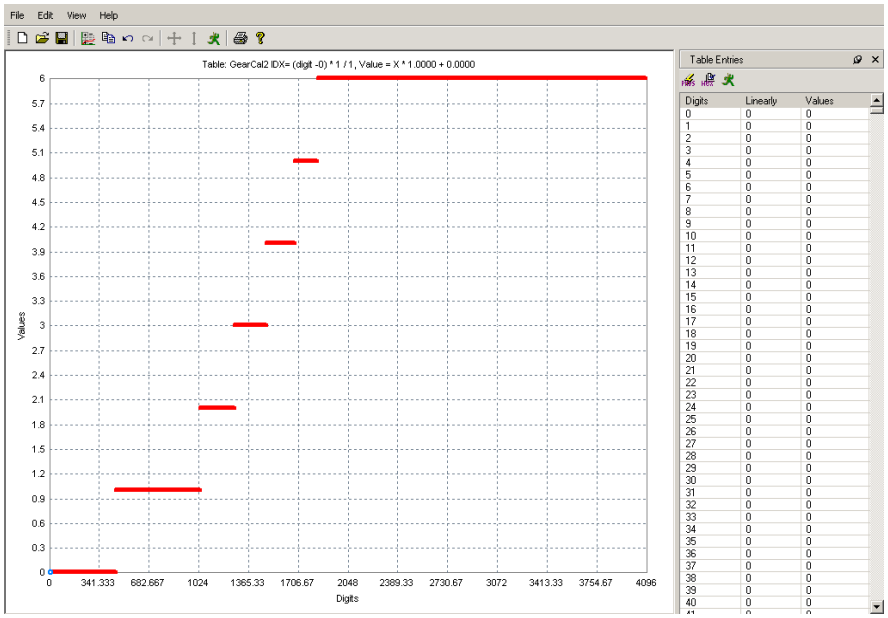
The table you have defined will then be viewable on the screen, as shown on the next page. Before you can close the table editor you must follow the remaining instructions:

- Next go to “**File**” and click “**Save**” to permanently store this table in your computer
- Save the .tbl file with the same name e.g.’GearCal’, AND BE SURE TO REMEMBER THE DIRECTORY TO WHICH IT IS SAVED.

The lookup table generated for the Moto2 bike is shown below.



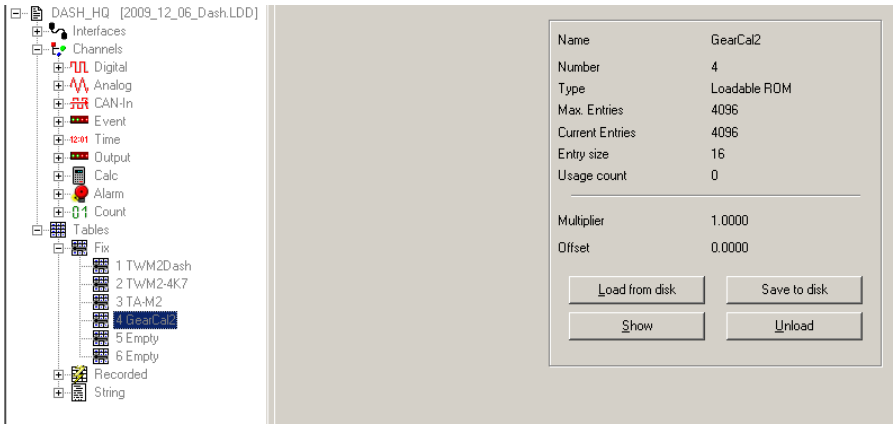
Here the lookup table generated for an alternative application is shown.



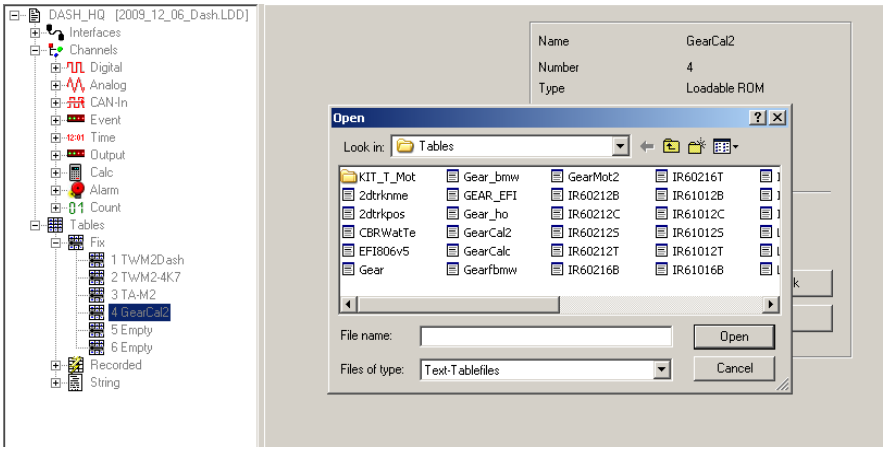
Step 4 Loading the Table to the 2D Dash

Next the defined table must be loaded to the 2D Big Dash using the 2D program WinIt:

- Select the node "**Tables**" from the system tree
- Expand the "**Fix**" section within the "Tables" node
- Select any of the 6 spaces available for loading tables, but be sure not to select a space that is already being used for another dash function! Remember which space number is chosen, e.g. 4, as this is important later!
- Click **<Load from disk>** on the main window



- Locate the directory into which the "GearCal" table was saved
- Select the table "GearCal" and click **<Open>**



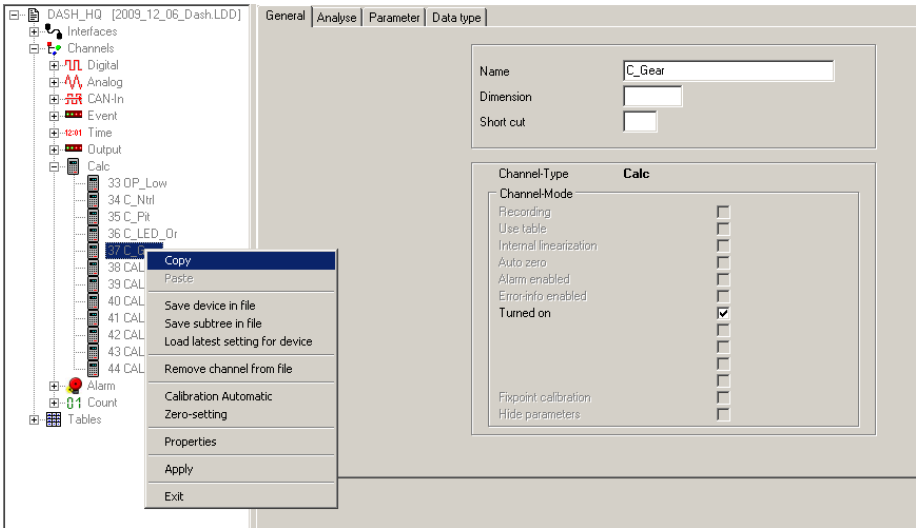
As the table is loaded to the Dash it can now be used by a calculation channel. In this example the table has been loaded to **position 4** of the Dash. Previously when the calculation channel was created the "lookup table position" was entered as "1".

Now that the table is correctly created and loaded to the dash, you must go back to the calculation channel and modify the table position to "4".

Step 5 Setting the Dash Output

Next the calculated channel is displayed as the selected gear on the display of the 2D Dash.

- Select the created calculation channel, e.g. “C_Gear”, from the system tree
- Make a right mouse-click on the channel, select **<Copy>**



- Now select “**Output**” from the Dash system tree, open the node “**Page 1**”
- Right click on the 4th channel from the top of Page1, and select **<Paste>**. This assigns the calculated gear position to the top right on page 1 of the display.
- In the <**General**> tab, select the dimension “**GEAR**” from the drop down box
- In the <**Parameter**> tab, within the field <**Display**>, set the character ‘**length**’ to 3, and ‘**digits after dot**’ to 0.

This completes the online gear calculation settings for the 2D Big Dash.

12.3 Filtering Input Channels

Analogue and CAN input channels can be filtered for better readability on the dashboard. Channels that contain fast-changing values, e.g. air/fuel ratio, can be made more easily read by using the input filter.

The implemented Tau filter uses the specified number of samples to give the desired filter characteristic. Use a filter with a high number e.g. 7 for a very smooth signal but with a greater delay if sudden signal changes occur. Use a small filter e.g. 1 for small signal delays and less filtering.

12.3.1 Applying a filter to a channel - Practical Example

The process for filtering a channel is made according to the equation below:
Displayed value=(previous value*(N-1)+current value)/N

NOTE:

- N represents the number of samples over which the data channel will be filtered.
- N is equal to $2^{\wedge filter\ number}$, e.g. $2^{\wedge 1}$= 2 samples (filter setting 1), $2^{\wedge 7}$=128 samples (filter setting 7)
- That also means that the filtering rate should be adapted to the sampling rate of the input channel.