VPEB Van Perlo Elektronica & Besturingstechniek

DTGv1

Manual

Author: Leon J.A. van Perlo
Version: 1.0
Date: March 2nd, 2025

Smart electronics for your miniature world

Release management

This manual applies to:

- Module DTG version 1.0 / 1.1
- Module DTG/R version 1.0 / 1.1
- Firmware DTG 1.0 Build 3
- Firmware DTC 6.2 Build 3
- Firmware DTR 2.1 Build 1
- Software DTCnewConfig version 1.2 Build 3

Compatibility

To use the functionality described in this manual, you need at least the firmware and software listed above.

In particular, we mention that to use DTR relay modules in combination with addressable DTC modules, you need at least firmware DTR 2.1 build 1. You cannot program this firmware into the DTR yourself. If you use older versions of the DTR, they will work, but only with a DTC at address 1.

Proper functioning of the DTM manual control cannot currently be guaranteed when used with DTC modules using an address other than address 1.

©2025 This document, or any information contained herein, may not be copied or distributed, in whole or in parts, in whatever form, without the explicit written approval of the original author. The making of copies and prints by users of the DTG module for their own use is allowed.

Contents

1 Preliminaries

1.1 Introduction

The DTC Turntable Controller (Dinasys Turntable Controller) controls various turntables for scale H0, H0m, TT, N and Z. The DTC is controlled via a USB interface. Via this interface, the DTC can be controlled digitally by train control software on a PC.

In addition, the DTC has a CAN interface. This is used to communicate with various accessories, such as relay modules or a hand control.

The DTG (DinaSys Turntable Gateway) is a complementary module intended for use in combination with one or more DTC controllers. The DTG allows to control one or more DTC modules with protocols not directly supported by the DTC.

The DTG features:

- A CAN interface
- An RS485 Interface
- 4 electrically isolated digital outputs
  - An electrically isolated USB interface
  - An electrically isolated DCC inputEen CAN interface

In addition to the full DTG, there is a limited and more economical version DTG/R on which the latter two interfaces are not present..

The DTG(/R) is connected to one or more DTC controllers via the CAN interface. It is possible to control up to 7 DTC controllers via the DTG. The actual number you can control partly depends on the protocol and software used.

Some of the possibilities of the DTG::

1. Control a DTC via the USB interface on the DTG. The possibilities are then identical to the situation where you connect a DTC directly to the controlling PC. This situation seems only cost-increasing, but can be a solution, namely when the distance between your PC and the DTC is large. Long USB cables (>3m) can lead to interference in practice, but the CAN connection between DTG and DTC can be tens of metres long. So you can connect the DTG to the PC with a short cable and connect to the DTC via a long CAN cable.
2. Control multiple DTCs via the USB interface on the DTG. The prerequisite is that your train control software supports DTC addressing.
3. Control one or more DTC controllers via RS485. The DTG behaves on the RS485 interface as an OC32. The prerequisite is that your control software supports DTC over OC32. You can include the DTG in an RS485 bus with OC32 controllers. So that can also be a Dinamo system. The advantage of this setup is that you don't need any additional links from your control PC.
4. Control one or more DTC controllers via the OC32 protocol over USB. The prerequisite is that your control software supports DTC over OC32. The DTG behaves on the USB interface as OC32. The RS485 interface then becomes a so-called 'pass-through' interface. On that RS485 you can then connect further OC32 modules.
5. Control the DTC with protocols other than DTC. The DTG is then interfaced as OC32 to the control PC, either directly or as part of a Dinamo system. The control software commands the DTC as if it controlled OC32 outputs.
6. Control the DTC by means of DCC commands, for example from your digital central unit.

1.2 Scope of delivery

The DTG(/R) is normally supplied as a module including mating connectors, mounting frame and screws. No cables are supplied. The main reason is that the length of the cables is very dependent on the situation in which you want to use the DTG, so a standard cable is rarely appropriate.

The CAN cable between DTG and DTC is a standard RJ45 network cable (cat5 is more than sufficient). These cables can be bought at almost any length, through VPEB sales partners or in (almost) any shop supplying computers, network components or related parts.

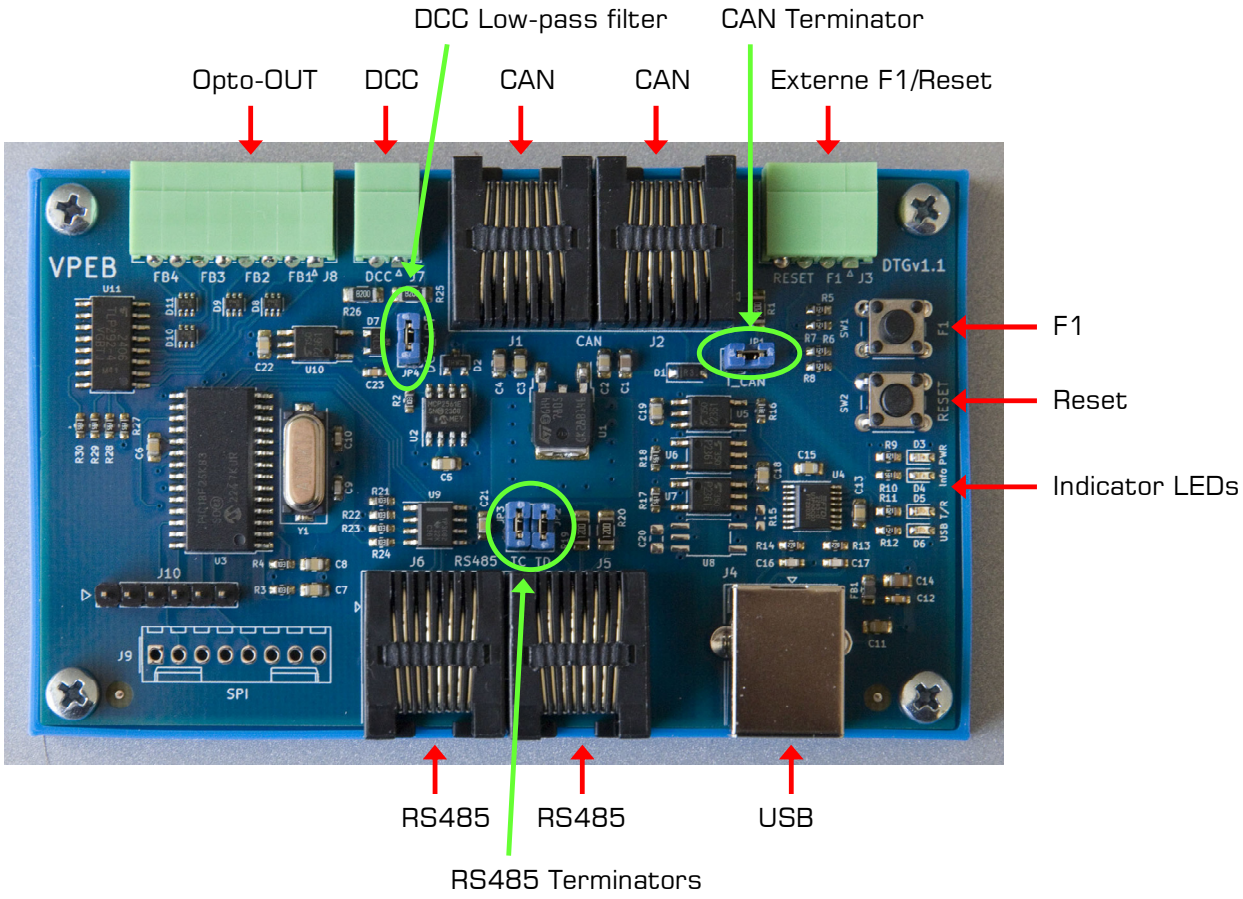
An RS485 cable, like a CAN cable, is also a standard RJ45 network cable.

You can use your DTC's USB cable to connect your DTG to your PC if you wish.

2 DTGv1

2.1 Overview

Below is a picture of the DTCv1.1 with an overview of connections and functions. These are described in more detail below.



2.2 CAN bus (J1, J2)

The CAN bus is to connect your DTC(s) and any other DinaSys accessories. There are 2 RJ45 connections. These are identical. The reason there are 2 connections is that a CAN bus must be looped through if you connect more than 2 stations.

NOTE: The RJ45 for the CAN bus is **not an Ethernet connection and not an RS485 bus**. So do not connect a network cable from a computer network or a cable from a Dinamo system to this[1].

The CAN bus is also the power supply for your DTG. Normally, the CAN bus is powered from your DTCv3 with approx. 9V supply voltage. With jumper JP1 on the DTCv3, the output of the stabiliser is routed to the CAN bus.

A CAN bus is formed by connecting modules together in a string topology. The stations at the end of the bus use only one connection, all stations in between use both connections. Only the first and last modules in the RS485 bus should have a **Terminator**.

[1] It doesn't immediately break down if you interchange CAN and RS485. Both systems use more or less the same voltages for communication. However, the protocols are completely different and an essential difference is that the DinaSys CAN bus carries a power supply in it and an RS485 bus does not. With RS485, the contacts for the power supply are not connected, so nothing breaks. That may be different if you connect a network cable on which POE (Power Over Ethernet) is provided.

2.3 RS485 bus (J5, J6)

The RS485 bus is used in a Dinamo system and is used to connect OC32 modules to a PC for controlling accessories. There are 2 RJ45 connections. These are identical. The reason there are 2 connections is that an RS485 bus must be looped through if you connect more than 2 stations.

CAUTION: The RJ45 is **not an Ethernet connection** and **not a CAN bus**. So do not connect a network cable from a computer network to this, nor the CAN cable from your DTC [1].

An RS485 bus is formed by connecting modules together in a string topology. The stations at the end of the bus use only one connection, all stations in between use both connections. Only the first and last modules in the RS485 bus should have a **Terminator**.

2.4 USB (J4)

Via the USB connection, the DTG is connected to a PC. From the PC, you can configure, control and update the DTG with new firmware. You can also use this interface to configure and control the DTCs connected to the CAN bus.

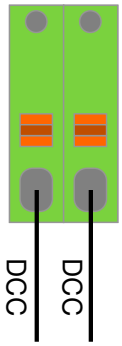
The USB connection is galvanically isolated from the rest of the DTG. This means that there is no electrical connection between the USB connection and the other connections on the DTG. This greatly reduces the risk of interference and increases the reliability of the USB connection.

The USB section on the DTG is powered from the PC's USB connection. This means that the USB connection becomes active when you plug in the USB cable (and the PC is on), so even if the DTG itself is not (yet) powered. So a USB port is then visible on the PC, only the communication makes no sense if the DTG itself is not powered. This is because there is then nothing to receive or send messages.

The USB chip on the DTG is automatically recognised by the currently common operating systems (Windows, MacOS and Linux) **PROVIDED** your PC is connected to the Internet the first time you connect to the DTG. Your PC will then download and install the necessary drivers itself. If this does not work, for example because your PC has no Internet connection or because your operating system is very outdated, you can download the drivers manually from http://www.ftdichip.com

2.5 DCC Input (J7)

The DCC connection is a 2-wire connection to which you can connect a DCC signal from your digital central unit. The DTG does not require any power from the DCC connection. It is therefore best to tap a DCC signal from your central unit before the booster connection. This signal is considerably less polluted by interference signals than the signal on the rails of your layout. Another advantage is that the DCC continues to work even if the booster switches itself off, e.g. due to a short circuit on the track.



DCC is polarity insensitive, so it doesn't matter if you swap the two wires. The DCC connection on the DTG is galvanically isolated. That means there is no electrical connection between the DCC connection and the rest of the DTG.

2.6 Opto-OUT (J8)

This connection consists of 4 digital outputs with 2 contacts each, isolated from the rest of the DTG by optocouplers. These outputs can be used to report the status of the DTC back to your control system, e.g. via a feedback system.

There is a difference here between the DTGv1.0 and the DTGv1.1.

DTGv1.1:

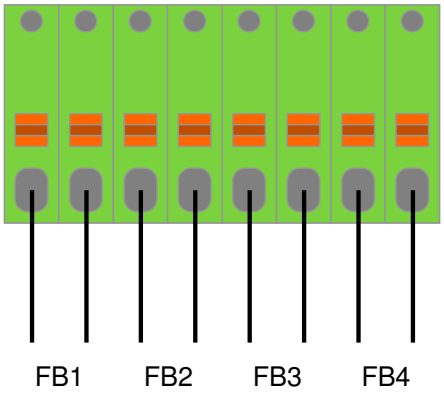
The outputs on the DTGv1.1 are AC voltage capable. That means it does not matter what the polarity of the signal you connect.

The maximum peak voltage each output can switch is 30V.

The maximum current each output can switch is 50mA.

Note: You should limit the current that will flow when the output is active externally, for example by using a resistor in series with the output.

Note: The maximum voltage is a peak voltage. The peak voltage of a sinusoidal alternating voltage is considerably higher than the effective voltage. If you provide a pure alternating voltage of 21V effective, its peak voltage is about 30V.



DTGv1.0:

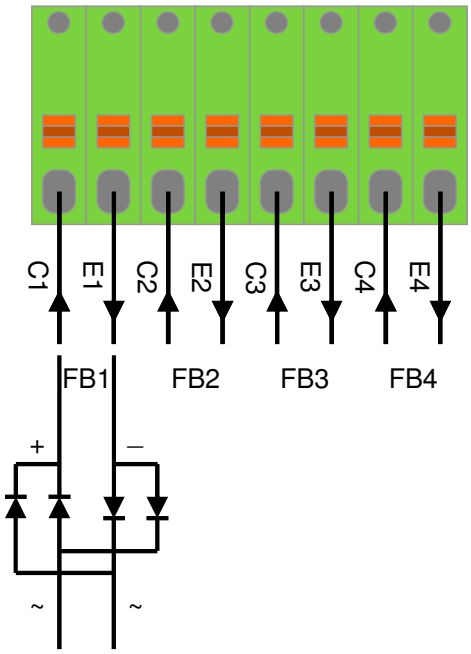
The outputs on the DTGv1.1 are **NOT** suitable for AC voltage. This means that the polarity of the signal you connect is important. Connector C of each pair should be positive with respect to connector E.

The maximum peak voltage each output can switch is 80V.

The maximum current each output can switch is 50mA.

Note: You should limit the current that will flow when the output is active externally, for example by using a resistor in series with the output.

If you need to switch an AC current with the outputs of the DTGv1.0, you can do so by adding a bridge rectifier externally in the connection of FB1..FB4.



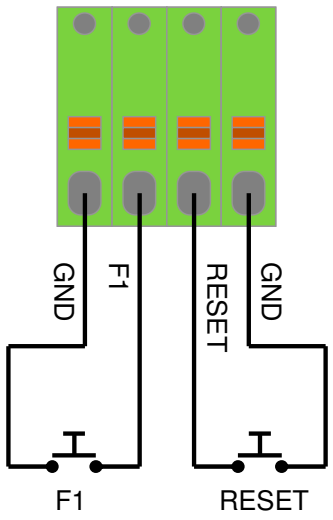
2.7 Reset/F1 (SW1, SW2, J3)

There are two push buttons on the DTG: RESET and F1

Reset is for restarting the DTG. Normally this will never be necessary, but a manual reset is necessary to start the bootloader in the DTG, which allows you to load new firmware (software) into the DTC yourself when it is made available by VPEB.

F1 can have various functions. In the current firmware, this button activates configuration mode. You don't need it in principle, but it can be a godsend if you have forgotten what address and baud rate you set the DTG to.

The Reset and F1 buttons have a different height, so you can feel the difference even when the module is mounted under the track. Since you almost never need Reset, that is the lowest button.



You can also control the F1 and Reset functions externally. You can connect one or two external push buttons to the 4-pin connector J3. **NOTE:** The connectors on J3 are intended for push buttons and are NOT electrically isolated. They can tolerate a voltage of up to +5V with respect to GND (the minus connection of your DTG).

2.8 Indicator-LEDs

There are four indicator LEDs on the side of the module:

- Blue: USB connection to the PC active. I.e. the DTG is connected to the PC and the drivers on the PC are loaded.
- Green: USB data sending/receiving.
- Red: This LED can have various functions. With the current firmware:
  - On = Normal operation
  - Blinking = Configuration mode
- Yellow: Info. This LED can have various functions. With the current firmware:
  - On = Bootloader active
  - On (100ms): Sending RS485

3 Mounting and connecting

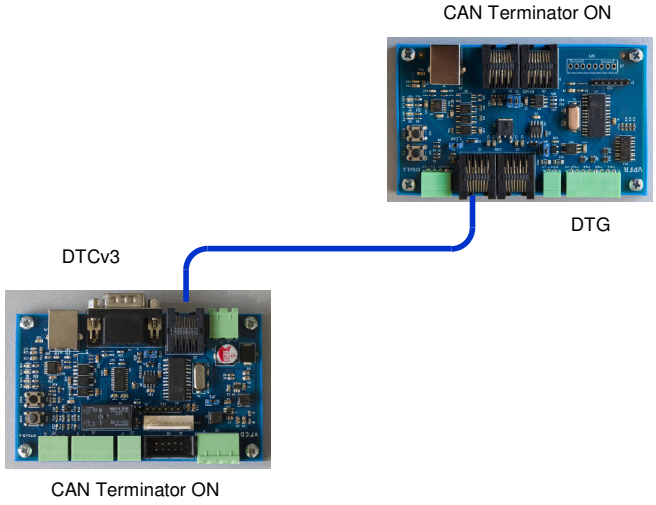
3.1 General

The DTG comes with a mounting frame and screws. Use these to screw the DTG onto a flat surface (usually and preferably wood). Make sure you leave enough space around to plug in the various connectors and it is also useful if the push buttons are still accessible.

If you control the DTG via USB, place the DTG near the PC that controls it so that the USB cable can be short. Long USB cables can lead to communication errors. The CAN cable can easily be tens of metres long.

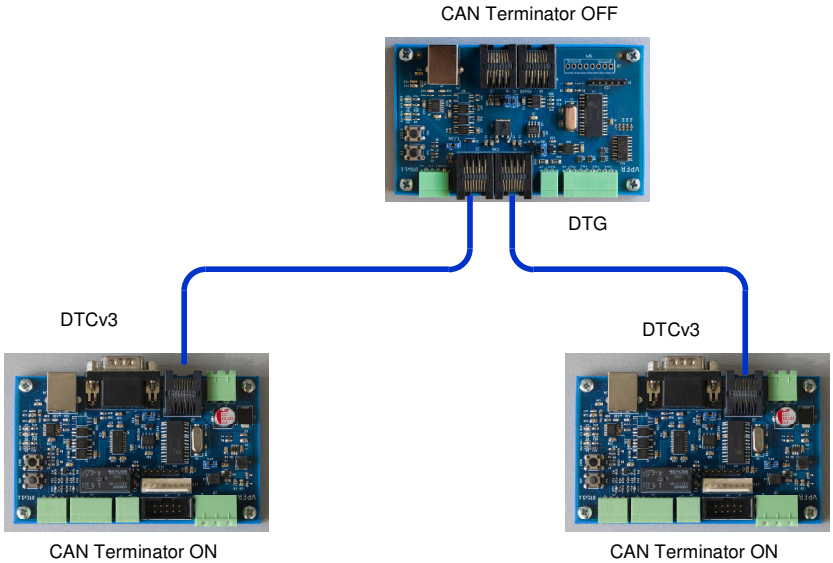
3.2 DTG en DTCv3, communication and power

To connect the DTG to a DTCv3, take a (standard) RJ45 cable and connect the two modules as below. Make sure you plug the RJ45 cable into a CAN socket on the DTG. Since there are only 2 modules in this example, each module forms one end of the CAN bus. So on both modules, the CAN terminator is ON.



NOTE that power for the DTG module is supplied by the DTCv3 module and that the DTCv3 is set to supply power to the CAN bus. If you connect a USB cable from your PC to the DTG module, the blue LED on the USB interface will light up. However, the DTG only works when the red LED on the DTG module also lights up!

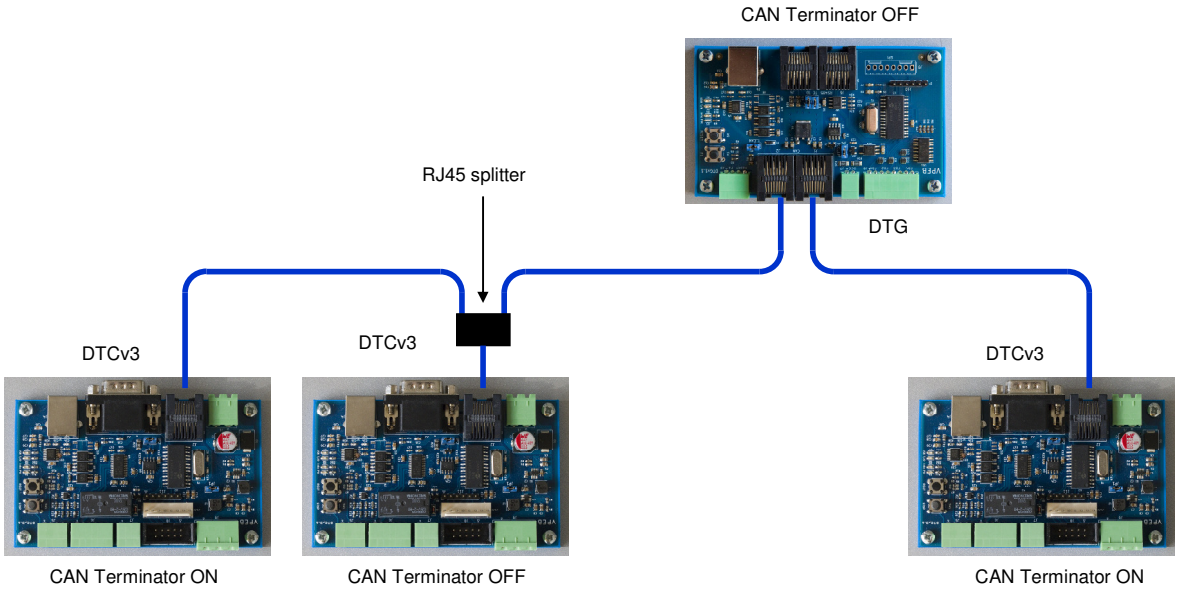
3.3 Multiple DTCv3 modules



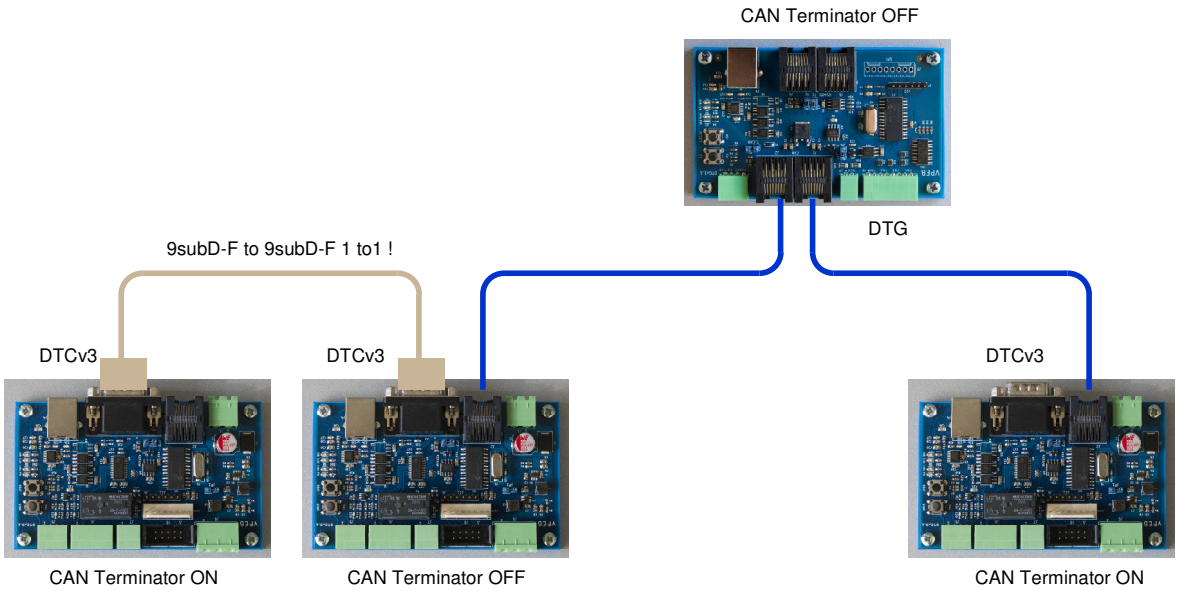
To connect two DTC modules to a DTG, repeat the game from section 3.2.The difference here is: Since the DTG is connected with both RJ45-CAN connectors, it is NOT on the end of the CAN bus and so the CAN terminator is OFF.

For this configuration to work, the two DTCv3 modules will each have to have a different DTC address (see section 4.2).

If you want to connect three or more DTCv3 modules, the challenge is that a DTCv3 module has only one RJ45 CAN socket. You can solve this by applying an RJ45 splitter. Place this as close as possible to the DTCv3 module (<0.25m).



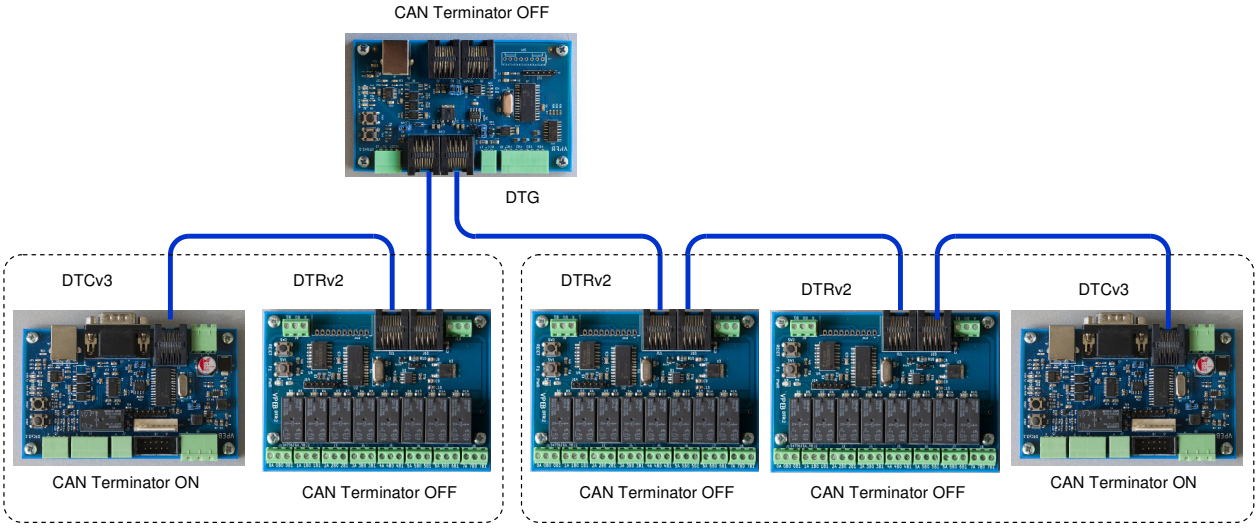
Instead of using a splitter, you can also resort to using the subD-9 CAN connection to interconnect DTCv3 modules:



3.4 DTCv3 and DTRv2 modules

Are you using DTRv2 modules in addition to DTCv3 modules? In all cases, you simply loop through all the CAN modules you are using. The order in which you connect the modules is not important. The only important thing is, that the whole chain of all modules has only 2 ends.

In the example below, we choose to put the DTCv3 modules each on the end of the bus, This has only a practical reason: The DTCv3 module has only one RJ45 CAN socket and is therefore less easy to loop through to a subsequent module.



The diagram above shows a grouping, which DTR modules belong to which DTC. **NOTE:** This grouping is determined solely by configuration and is not influenced by the order in which you connect the modules.

You can find how to configure the DTR in the DTRv2 manual. If you want to use the DTR in combination with several DTCs and the DTG, it is important that the DTR firmware is at least **DTR 2.1 Build 1**. This version recognises the address of the DTC through which it is controlled.

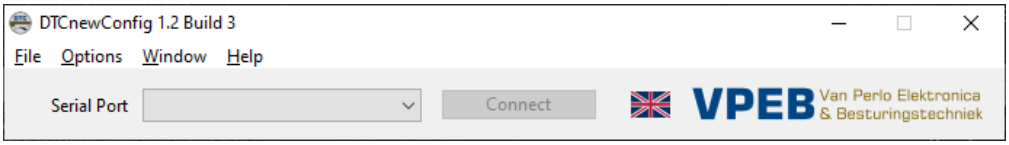
4 Setup and testing

4.1 DTCnewConfig

If you haven't already done so, download the programme DTCnewConfig. For use with the DTG, you need at least DTCnewConfig version 1.2. The version at the time of publishing this manual is 1.2 Build 3.

DTCnewConfig is available for Windows, MacOS (Intel and Apple M processors) and Linux. The download is a compressed .zip (or .tar.gz) file. Unpack this file somewhere on your PC.

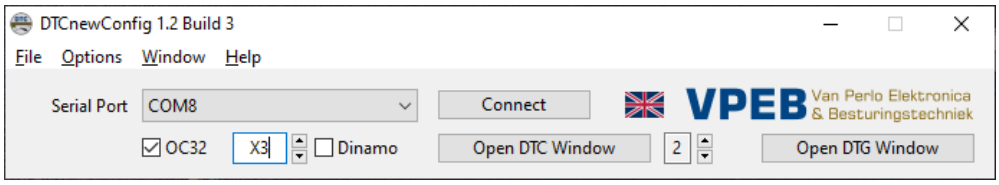
Start DTCnewConfig. You will then see the window as below: [2]



Users used to DTCnewConfig 1.1 will notice that the opening screen of version 1.2 is considerably sparser than that of version 1.1. The reason is that the communication part is now separated from the configuration and operation of the DTC.

As in version 1.1, you can choose the interface language in the opening screen. Also, you can choose the serial (USB) port to which your DTC (or DTG) is connected. Then click on 'connect' to open this port and connect. DTCnewConfig 1.2 then opens the corresponding DTC configuration screen itself. If you use DTCnewConfig in this way, version 1.2 works largely identically to version 1.1, with the difference that it is 2 separate windows that you can move separately if you wish.

However, there are more options. In the top menu of the main screen (the screen that starts DTCnewConfig), under 'Options' you'll find the option 'Advanced'. Switching that on gives you some additional options. These are generally only useful if you use the DTG.



Open DTC window:

Allows you to manually open a window to configure and control a DTC. A DTC now has an address 1..7. So you can also open 7 windows for 7 DTCs.

Open DTG window:

Allows you to manually open a window for configuring and controlling the DTG.

Checkbox OC32:

This allows you to choose to communicate with the DTG via the OC32 protocol. You need to select this if you communicate via the RS485 connection or if you have set the USB connection on the DTG to the OC32 protocol. You also need to choose the OC32 address your DTG is set to. This can be a base address or an eXtended address. With the up/down arrows you can adjust the address from 1..16. If you go higher, an 'X' appears to indicate that you are using eXtended Addressing. The extended address can be X1..X96. You can also type the desired address yourself in the corresponding field, whether or not preceded by 'X'.

DTCnewConfig currently only supports OC32 extended addressing according to the 'enforced extended addressing' method. This means that when extended addressing is used,

[2] The images are from Windows. On Mac and Linux the layout may differ.

the base address is no longer relevant. 'Underwater' in this case, DTCnewConfig sends base address 0.

Checkbox Dinamo:

If you have chosen OC32, you can then also check 'Dinamo'. In this case, DTCnewConfig controls a DTG connected as an OC32 in a Dinamo system. If you choose Dinamo, the OC32 base address can be up to 32 (instead of 16), to support the addressing of Dinamo systems with a Dual Bus.

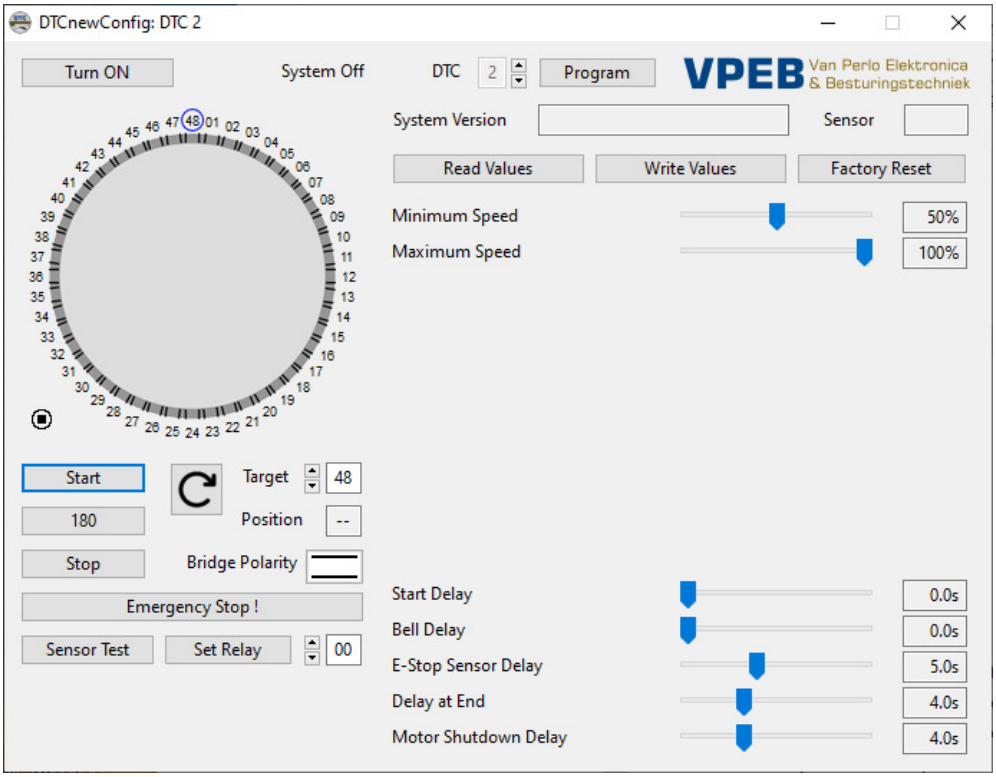
4.2 DTC configuration

If you are in 'advanced' mode, DTCnewConfig no longer automatically opens a DTC window when you connect to the communication port. After all, DTCnewConfig does not know which DTC you want to communicate with. So now you have to make that choice explicitly and open the desired window yourself.

If you work in 'advanced' mode, you will also find an extra button 'Turn ON' in the DTC window. With this you explicitly ask the DTC window to connect to the DTC in question and read in the parameters. Of course, this can only be done after you have connected to the DTG at port level, otherwise you will get an error message. Once the connection is made, you get the option 'Turn OFF'. With this, you break the logical connection to the relevant DTC.

The terminology turn ON and turn OFF was once coined by PiCommIT. You don't actually turn the DTC 'on' or 'off', but we maintain this terminology to avoid confusion with 'connect' and 'disconnect' from the communication window.

If you work in 'advanced' mode, the DTC configuration window has an additional option: The ability to set the DTC address. If you connect multiple DTCs to the DTG, each DTC should have a unique address.



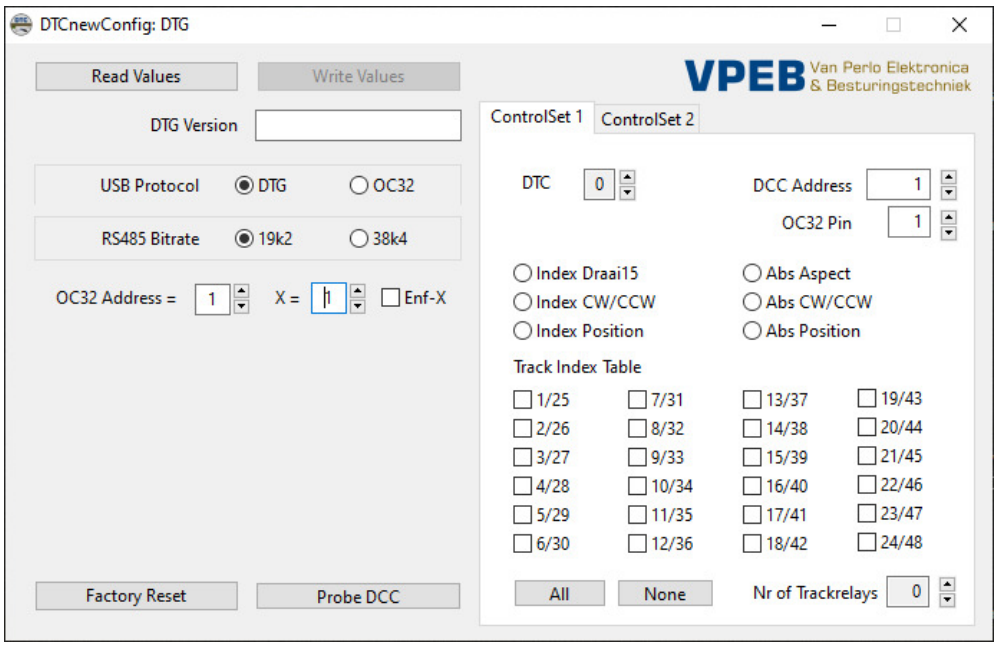
Note: If you change the address of a DTC, it changes only after you reset (or turn off and on) the relevant DTC module. The reason is that otherwise the address could change while you are configuring and that would lead to strange behaviour. Also, DTCnewConfig continues

to communicate at the 'old' address until you open a new DTC window with the new address.

Otherwise, the DTC configuration window offers the same options as with DTCnewConfig 1.1. So we refer to the DTC manual here for a detailed description.

4.3 DTG configuration

To set up the DTG itself, open the DTG window from the main screen. You will then see the following screen:



You cannot write away changes to the DTG if you have not first read in the current values. The reason we do it this way: The DTG already has a very large number of configuration settings, and writing out parameters without knowing the current settings can lead to unwanted effects. So the first thing to do is 'Read Values'. In fact, this is equivalent to 'connect and read everything'. Of course, this can only be done if you are connected to the DTG at port level, otherwise you will get an error message.

The only thing you can do before reading the current settings is to reset the DTG to factory settings.

Once you have read in the settings, the DTG Version field shows the current firmware version of the DTG.

4.3.1 General settings

On the left side of the window you will find the 'general settings'.

USB Protocol:

By default, this is set to DTG. You can change this to OC32. The DTG then emulates an OC32 connected to the USB port. You can think of this as the equivalent of a U485(J) + OC32. In this case, the RS485 interface on the DTG becomes a 'pass-through' interface: everything that comes in (according to valid OC32 protocol) on the USB port is transparently passed on to the RS485 port and everything that comes in on the RS485 is passed on to the USB port. The advantage: you can connect OC32s to the RS485 port and the DTG plays DTG + U485 converter.

RS485 Bitrate:

This determines the speed of the RS485 interface. An 'automatic setting', as the OC32 knows, is not implemented (yet). 19,200bps is generally fast enough. But if you connect the DTG in a Dinamo system and its communication speed is set to 38,400bps you should therefore adjust this accordingly.

Note that the USB interface is always set to 19,200bps

Note that the RS485 interface is always set to Protocol OC32. The reason is that the DTC/DTG protocol is a full-duplex protocol while RS485 is a half-duplex medium. So a half-duplex protocol has to be selected for it to work.

OC32 Address:

Here you set the OC32 address that the DTG uses when communicating via the OC32 protocol. The base address can be 1..16, the eXtended part 1..96, just like with an OC32. The Enf-X option activates the 'Enforce eXtended Addressing' option. That means that the OC32 (=DTG) only responds to eXtended Addressing and that the base address is no longer relevant in that process.

4.3.2 Control Sets

On the right side of the window you'll find tabs with 'Control Sets'. You need these (only) if you want to control the DTC(s) via OC32 or DCC commands.

There are 2 Control Sets. With each set you can control a DTC (1..7) according to one of several control methods. Each control method is supported by both DCC and OC32 SetAspect instructions.

If you don't use a Control Set, set the DTC number to 0. The control method selection is then cleared. You cannot select a control method as long as DTC is set to 0.

Any control method can be controlled with OC32 SetAspect instructions.

With DCC, there is a difference between standard DCC Accessory commands and Extended DCC Accessory commands. A standard DCC command knows only the positions 0 (straight) and 1 (thrown) and is basically intended for turnouts. Extended DCC commands know 32 aspects per address, but in the more recent DCC specifications this has been extended. It depends on your central unit what it supports.

A control method can be controlled with standard DCC commands OR with Extended DCC commands (so never both). Methods intended for control with standard DCC commands respond only to standard DCC commands and methods intended for control with Extended DCC commands respond only to Extended DCC commands.

Because it is sometimes unclear which DCC address a particular control unit transmits for a particular setting, the option has been provided to "discover" the DCC address. First select the tab with the ControlSet you want to use. Click on the "Probe DCC" button. This gives a command to the DTG that means something like "do nothing with the DCC command you receive, but tell me what address it is". Then use your digital control unit to give a DCC command to **the first** address you want to use to control the DTC in question. For each method, this is the command for the Emergency Stop or its cancellation. The DCC address that was sent by the control unit appears in the "DCC address" field in the active tab. It also appears whether it is an Extended or a standard DCC address. If a control method was selected that does not fit the address type in question, the choice is changed to an available method that does fit. You can then adjust this yourself if you want.

Note: The address that you have found in this way is only shown and has not yet been saved in the DTG. This only happens when you write the settings.

4.3.3 Control Methods

Each control method also provides an emergency stop for the rotation of the bridge. The emergency stop is always the (first) address selected for the relevant Control Set.
For control by DCC, this is the specified DCC address.

For control via OC32 Pin – Aspect instructions, this is the specified OC32 Pin number. We count from 1 here, so possible Pin addresses are 1..128.
Position/Aspect = 1 (thrown) means Emergency stop.
Position/Aspect = 0 (straight) means Cancel Emergency stop.

The first address1 for the control of the relevant method (see below) is therefore always the specified address + 1.

Index Draai15

The "Draai15" ("Turn15") protocol was once invented by Kees Moerman and implemented in Koploper, among other things. The control uses 7 addresses for controlling "magnetic devices". These can be OC32 addresses or DCC addresses, for example. The 7 addresses have the following function:

1 Thrown = Reset
2 Thrown = Bit 1
3 Thrown = Bit 2
4 Thrown = Bit 4
5 Thrown = Bit 8
6 Thrown = Turn counterclockwise
7 Thrown = Turn clockwise

The intention is that the software first gives a Reset command (address 1) and then sets an Index by setting one or more bits (address 2..5). This allows you to select 15 possible exits. With "Draai15". You can therefore use a maximum of 15 exits + the opposite side. You determine which exits these are by checking which ones you are using in the Exit Index Table. This can therefore be a maximum of 15 pairs. Index 3 (as an example) means that the third exit pair that you have checked in the table is selected.

You can give the above commands as OC32 Pin Aspect commands or via DCC.

The number of reserved addresses for the Index Draai15 method is 8.

CW/CCW

The "CW/CCW" method is derived from the so-called "Fleischmann control". In the DTG, this comes in two variants: Index and Absolute.

Address 1, the first address after the emergency stop, is the address used to be able to turn the bridge over 180 degrees.

Aspect 0 (straight) = turn clockwise, Aspect 1 (thrown) = turn counterclockwise.

In the Index version, addresses 2 and following determine the Index from the Track Index Table. Here, Address 2 corresponds to Index = 1. Index 3 (as an example), so Address 4, means that the third exit pair that you have checked in the table is selected.

In the Absolute version, an address is assigned to each exit pair, whether it is checked or not. However, if a command is given on an address belonging to an exit pair that is not activated, the command will not be executed. This method therefore costs more addresses, but the addresses are absolute and will therefore not change if you expand the turntable again.

The number of addresses used for the Index CW/CCW method is the number of selected exit pairs + 2.

The number of addresses used for the Abs CW/CCW method is 26.

Position

This method also comes in two versions: Index and Absolute.

Aspect 0 (straight) = turn the bridge with the cabin to exit 1..24
Aspect 1 (thrown) = turn the bridge with the cabin to exit 25..48

The DTG itself determines whether the bridge turns clockwise or anticlockwise, depending on what the shortest route is.

With the Index version, the addresses determine the Index from the Track Index Table. Address 1 corresponds to Index 1. Index 3 (as an example) means that the third exit pair that you have checked in the table is selected.

With the Absolute version, an address is assigned to each exit pair, whether it is checked or not. However, if a command is given on an address belonging to an exit pair that has not been activated, the command will not be executed. This method therefore costs more addresses, but the addresses are absolute and will therefore not change if you expand the turntable again.

The number of addresses used for Index Position method is the number of selected exit pairs + 1.

The number of addresses used for Abs Position method is 25.

Abs Aspect

This method is intended for use with OC32 Aspects or Extended DCC addresses. The method uses 2 addresses, or 3 if you also have Track Relays connected (see section 4.3.4).

Address 0 Aspect 1 = Emergency stop.
Address 0 Aspect 0 = Cancel Emergency stop.

Address 1 Aspect N = Turn to position N (N = 1..48). The DTG chooses the shortest route.

Address 2 Aspect 0 = Turn off all Track Relays
Address 2 Aspect R = Turn on Track Relay R (R = 1..48)

Commands for exit pairs that you have unchecked in the Track Index Table will not be executed by the DTG.

4.3.4 Track Relays

A track relay is a relay to activate a siding. It is only used in block-controlled systems, such as Dinamo, HCC!m and G&R. Track relays can be added in the form of DTR modules and are then controlled via the DTC.

By setting the number of track relays, you reserve the necessary addresses to be able to control these relays. The number of required addresses is 1 + the number of relays. The first address is used to switch off all relays.

If you set “Number of track relays” to >0, the first address, used for resetting all relays, is the first available address after all reserved addresses for controlling the bridge..

4.3.5 Feedbacks

Via the optocoupler outputs (J8) you can inform your digital system about the status and position of the bridge.

FB1 and FB2 are linked to ControlSet 1, FB3 and FB4 are linked to ControlSet 2. The respective Feedbacks react to the status of the DTC that you have assigned to the ControlSet.

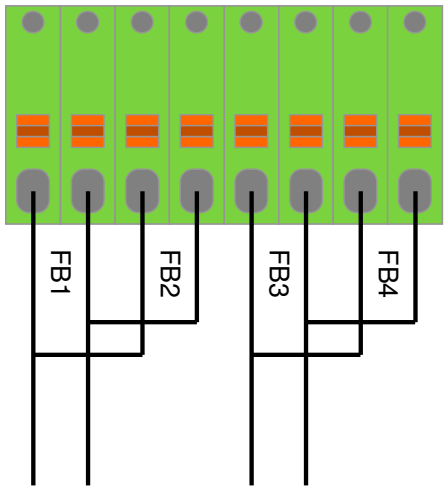
As long as the bridge is stationary, both feedbacks are inactive (off).

As soon as the bridge turns clockwise, FB1 (FB3) becomes active. This feedback remains active until the bridge has reached the end position and the bridge motor has stopped. Every time the bridge approaches the next position, FB2 (FB4) changes status. In this way you can signal to your control system how fast the bridge is turning and your control

system can deduce from this what the position of the bridge is. When the end position has been reached and the motor has stopped, FB1 and FB2 (FB3 and F4) both go off.

As soon as the bridge turns counterclockwise, FB2 (FB4) becomes active. This feedback remains active until the bridge has reached the end position and the bridge motor has stopped. Every time the bridge approaches the next position, FB1 (FB3) changes status. When the end position has been reached and the motor has stopped, FB2 and FB1 (FB4 and FB3) both go out.

If you only want to signal that the bridge is turning or has reached the end position and it is not relevant for your control system to know in which direction the bridge is turning and when the next positions are reached, you can choose to wire FB1 and FB2 (FB3 and F4) in parallel, as shown in the image to the right. This results in one feedback per Control Set.



4.4 F1

With the F1 button you can put the DTG in “Configuration mode”. By pressing the F1 button you will cycle through the following modes:

- Red LED lights up: Normal operation
- Red LED flashes quickly:
  USB = DTG protocol
  RS485 = 38.400bps, OC32 address = 1, OC32 eXtended address = 1
- Red LED flashes slowly:
  USB = DTG protocol
  RS485 = 19.200bps, OC32 address = 1, OC32 eXtended address = 1

This can be useful:

- If you have set the USB port to OC32, but you want to access the USB with the DTG protocol
- If you have a DTG/R (without USB) and you don't remember which OC32 address you have set.
- If you buy a new DTG/R and you want to record it in a Dinamo bus that is set to 38,400bps speed. In that case you have to disconnect the other (OC32) modules, because the DTG gets OC32 address 1, but you can at least access it and set it.

5 Firmware Update

If firmware updates are released for your DTG, you can install these updates yourself.

Firmware update only works (at the moment) on a **Windows** computer.

To perform an update, follow these steps:

- Download (if you haven't already done so) the program "mikroBootloader" from www.dinamousers.net Unpack the .zip file to a location where you can find it.
- Download the firmware you want to install. Make sure that this firmware is intended for the DTG. You need a *.hex file. Normally this is packed in a .zip file. Unpack the .zip file to a location where you can find it.
- Make sure that the DTG is connected to the PC via USB or vis RS485 and that the DTG has power.
- Start mikroBootloader by double-clicking mikroBootloader.exe
- Make sure that “PIC18” is selected at the top right next to “Select MCU”.
- “Change Settings” of the COM port:
    - Port = the COM port to which your DTG is connected
    - When connecting via USB: Baud rate = 115200
    - When connecting via RS485: Baud rate = 38400
    - Data bits = 8
    - Stop bits = 1
    - Parity = none
    - Flow control = none

  Click “OK”.
- Click “Connect”. The text “Waiting MCU response …” will appear on the right. Press the RESET button on your DTG. “Connected” will appear on the right in mikroBootloader.
- Click “Browse for HEX”. Find and select the .hex file you want to install.
- Click “Begin uploading”.
- The “Bootloading progress bar” at the bottom is now filling up. Get some coffee, do the dishes, walk the dog. It may take a while.
- When it is done you will get a pop-up window that it is ready. Click OK.
- Close mikroBootloader.
- Reset the DTG with the RESET button.

Your DTG is now working with the new firmware.

Note: If your DTG is embedded in a Dinamo system you should be able to update the DTG by putting the RM-x in Bootloader Transparent Mode and then going through the update procedure as described above. Unfortunately that does not seem to work. This is probably due to timing. Since Mikrobootloader is an external product that VPEB has no control over we cannot determine the exact cause at this time. For the time being a firmware update only works if your DTG is connected via USB or via RS485 via a U485(J).