

Changing NMEA2000 instances

1. Introduction

1.1 Purpose of NMEA 2000 instances

Instances are used in an NMEA 2000 network to identify multiple similar products connected on the same network.

As an example, take a system with two battery monitors (one for the main battery bank, and another for the hydraulic-thruster bank) and also a Quattro inverter/charger. All three of those devices will send their battery voltage measurements out on the N2K network. For the displays to show these values at the right place, they need to know which voltage belongs to what battery. That is what instances are for.

1.2 Different types of instances

There various types of instances, and for marine systems are two that matter: the Device instance and the Data instance. The data instance goes by various different names, like Fluid instance, Battery instance and DC instance.

Details and differences of each type are explained in detail in the [Cerbo GX manual, NMEA 2000 chapter](#).

1.3 Considerations before changing instances

First of all, note that it is only rarely necessary to change either device or data instances like the Fluid-, Battery- and DC-instance. Other sections in this document explain the specifics per make of MFD brand (Garmin, Raymarine, others).

When it is necessary or required to change instances, then depending on the type of product and system used, changing instances could also affect behaviour of the Victron equipment. In more detail:

GX devices, NMEA2000-out feature

All instances of the transmitted by a GX device can be set to any value, there is no related functionality. Includes device instances, and data instances.

Other products: Device instance

The NMEA 2000 device instance, which is the same as the VE.Can device instance, is by default set to 0 on all Victron products.

Some products, such as Lynx Smart BMS, Solar chargers with a VE.Can port and Battery chargers with a VE.Can port, have functionality that relates to the device instance. Therefore make sure to read the manual of that product before changing the device instance.

Battery chargers: Data instances (Battery- and DC instance)

Whilst it is possible to change the Battery and DC instances on a Skylla-i battery charger, changing those will render it impossible for a GX device to properly read the data. This is because the GX device expects the charger's output one to be on Battery & DC instance 0, output two on Battery & DC instance 1, and output three on Battery & DC instance 2.

1.4 Recommend instancing setup for main MFD brands

The only common use case we encounter in testing and support where it is necessary to change instances is with older Raymarine hardware & software:

- Raymarine i70: max number of tank levels is 5; fluid instance 0-4 and type must be fuel.
- Raymarine i70s: max number of tank levels is 5; fluid instance 0-4 and type must be fuel.
- Axiom MFDs: per Lighthouse version 4.1.75, a maximum of 16 tanks can be connected; fluid instance 0-15.

Further details per brand are in these documents:

- [NMEA 2000 configuration for Raymarine](#)
- [NMEA 2000 configuration for Garmin](#)
- [NMEA 2000 configuration for Furuno](#)
- [NMEA 2000 configuration for Navico \(B&G, Simrad and Lowrance\)](#)

1.5 Different methods for setting up instances

As the NMEA2000 protocol prescribes commands to change an instance by sending commands to a device, there are various ways of changing instances. The purpose of this document is to describe all commonly used methods.

Besides the here described methods there are more, for example some MFDs allow changing instances as well.

1. GX Device: Device- instances only
2. Actisense software + NGT-1 USB: Device- as well as data-instances
3. Maretron software + USB adapter: Unknown
4. Commandline of a [GX device](#): Device- as well as data-instances. Note that this required advanced Linux skills; and is listed here only for benefit of experienced software developers

Chapter 2, 3, 4, 5 and 6 explain these methods in detail.

1.6 Further reading on Victron and NMEA 2000

- [Marine Integration Guide](#)
- [Data communication white paper](#)
- [Cerbo GX manual, NMEA 2000 chapter](#)
- [NMEA2000 related discussions on Victron Community](#)

2. GX Device: changing device instances

The Settings → Services → VE.Can → Devices menu shows a list of all devices on the N2K / VE.Can network:

VE.CAN devices		22:24
BlueSolar Charger MPPT 150/70 [15965]	Device#	0
ALTREG [231308]	Device#	0
Waste water [530]	Device#	8
Aft Fuel Tank [533]	Device#	9
Waste water [531]	Device#	10
Day tank [532]	Device#	11

By clicking the right button, a detailed menu is shown:

BlueSolar Charger MPPT 150/70 [15965]		22:25
Model Name	BlueSolar Charger MPPT 150/70	
Custom Name		
Careful, for ESS systems, as well as systems with a managed battery, the CAN-bus device instance must remain configured to 0. See GX manual for more information.		
Device Instance	0	
Manufacturer	358	
Network Address	161	
Firmware Version	2.05	
Serial Number	0015965 HQ1250944QQ	
Unique Identity Number	15965	

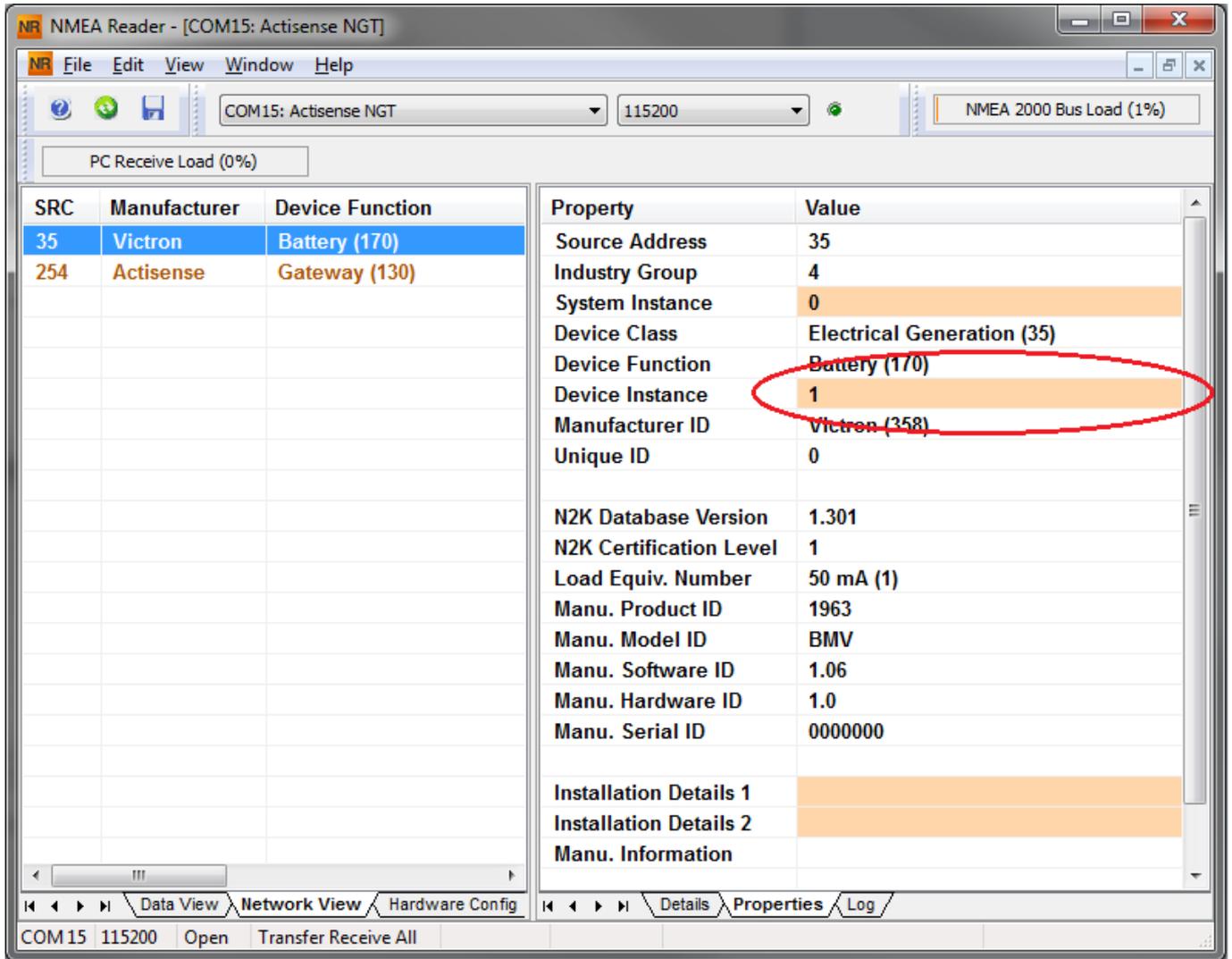
3. Actisense: changing device instances

Note: make sure to use a recent Actisense driver. Otherwise the instance might not 'stick'.

Requires the [Actisense NGT-1](#).

Changing a device instance:

1. Open Actisense NMEA Reader
2. Select the network view (tab selection is at the bottom left)
3. Select the product whose device instance you want to change
4. Select the properties tab at the bottom right and change the device instance



4. Actisense: changing data instances

Requires the [Actisense NGT-1](#).

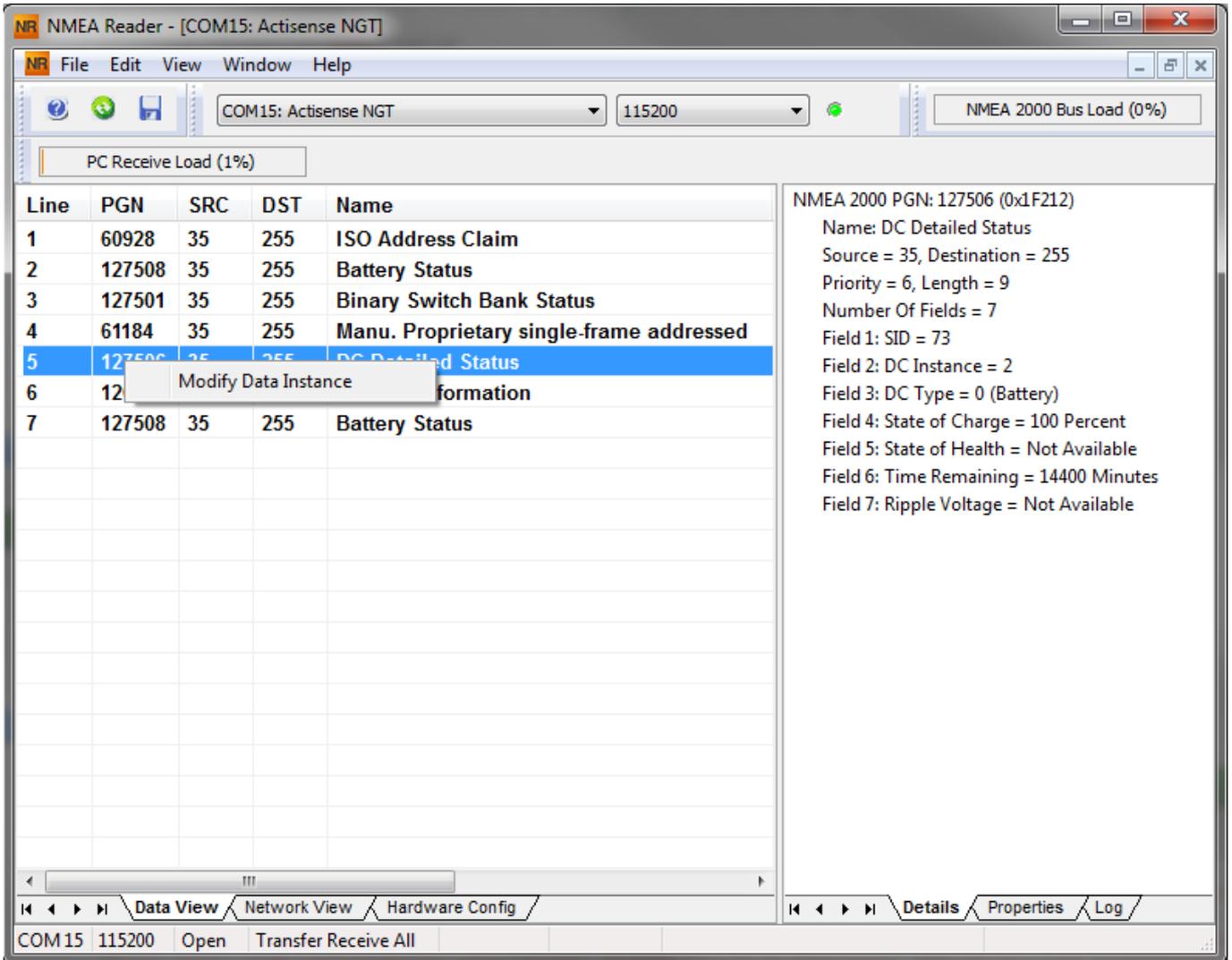
Step by step instructions:

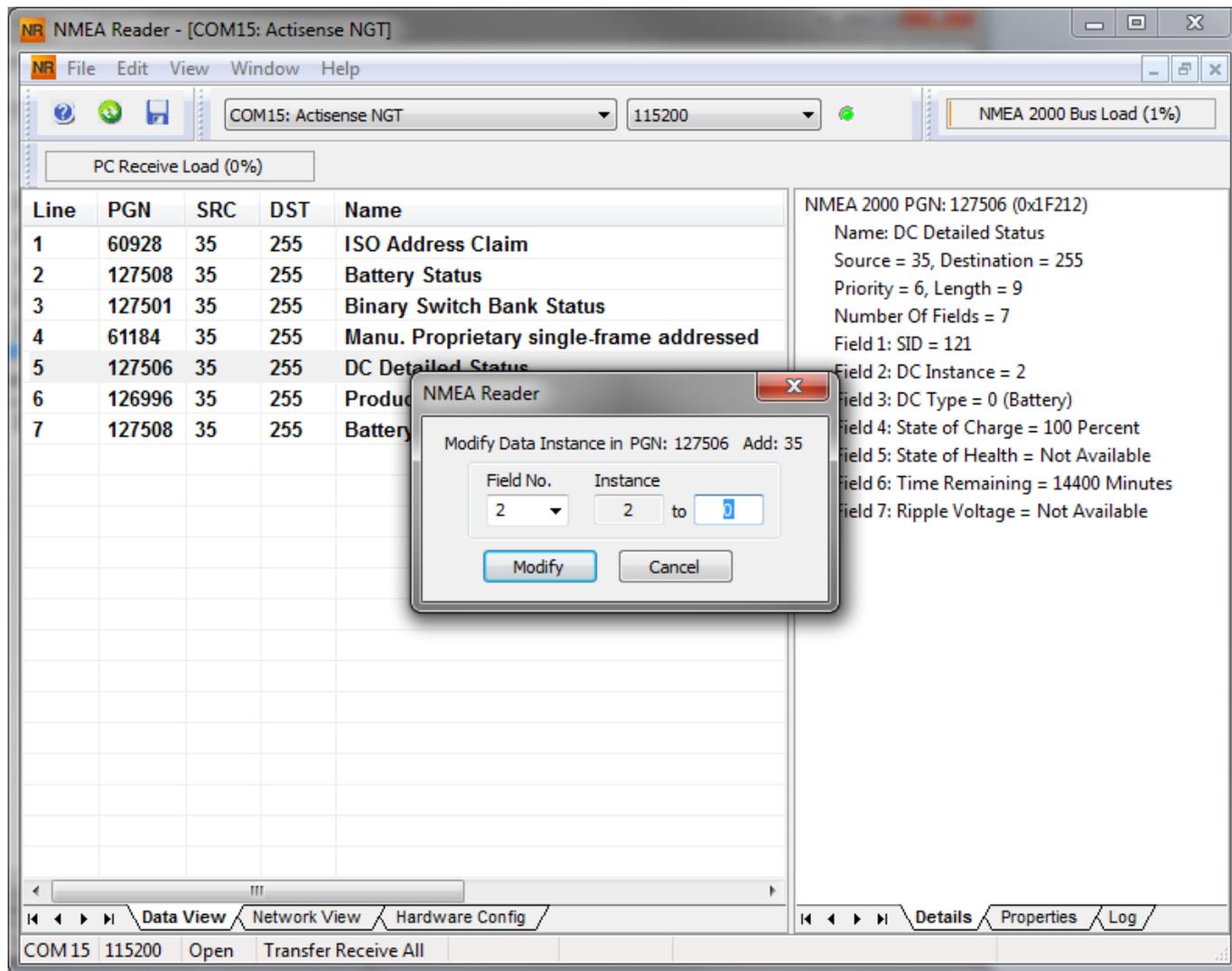
1. Open Actisense NMEA Reader
2. Select data view (tab selection is at the bottom left)
3. Right click on the PGN number. Note that this will only work on PGNs that allow changing their data instance (first screenshot below)
4. And change the value (second screenshot below)

Notes:

- The Battery Instance and the DC instance are the same value within Victron products. Changing one of them, will also change the other one.

- Since the BMV sends out two voltages, the main voltage and the aux- or starter-voltage, it comes preconfigured with two battery instances: 0 and 1. When you want to change that to 1 and 2, change the 1 into 2 first, and then the 0 into 1, as they cannot be the same.
- Changing the fluid level instance using Actisense has a bug. Probably due Actisense seeing it as 8 bit number; while in the definition it is a 4 bit number. Work around: using the GX, set the fluid type to Fuel (0), then using Actisense change the fluid instance to the desired value, and then using your GX, set the type back to the desired type.





5. Maretron N2KAnalyzer

Maretron uses a term called “Unique Instance” where the N2KAnalyzer software tool automatically determines if a particular device uses device or data instances.

WARNING: At Victron we do not understand what and how the Maretron software works with regards to this. We advise to use another tool, not Maretron, so that you know what you are doing, ie know what instance you are changing. So far, we have not been able to use Maretron software to change a data instance. And changing the other instance, the device instance can also be done straight from the Victron GX device its user interface. To change a data instance, for example to fix instance conflicts as reported by the Maretron software, we recommend to use Actisense. Not Maretron.

6. Changing the instances from the GX command line

6.1 Introduction

Instead of using Actisense or Maretron software, it is also possible to change the VE.Can aka N2K Device instance from the GX Device shell. To get root access, follow these instructions: [Venus OS](#):

Root Access.

Once logged into the shell, follow below instructions. More back ground information of the used commands such as dbus and dbus-spy is found by reading about root access document.

6.2 Warning: better use an Actisense!

Note that the methods explained in this Chapter 6 are usually not recommended. Use an Actisense instead, see chapters 3 and 4 instead.

6.3 New method - changing a Device instance

All devices available on the canbus are enumerated under the com.victronenergy.vecan service. And for all devices that support the necessary can-bus commands, the Device instance can be changed. All Victron products support changing their Device instance; and most or all non-Victron products as well.

```
# dbus -y com.victronenergy.vecan.can0 / GetValue
value = {
  'Devices/00002CC001F4/DeviceInstance': 0,
  'Devices/00002CC001F4/FirmwareVersion': 'v2.73',
  'Devices/00002CC001F4/Manufacturer': 358,
  'Devices/00002CC001F4/ModelName': 'Cerbo GX',
  'Devices/00002CC001F4/N2kUniqueNumber': 500,
  'Devices/00002CC001F4/Nad': 149,
  'Devices/00002CC001F4/Serial': '0000500',
  'Devices/00002CC005EA/CustomName': 'Hub-1',
  'Devices/00002CC005EA/DeviceInstance': 0,
  'Devices/00002CC005EA/FirmwareVersion': 'v2.60-beta-29',
  'Devices/00002CC005EA/Manufacturer': 358,
  'Devices/00002CC005EA/ModelName': 'Color Control GX',
  'Devices/00002CC005EA/N2kUniqueNumber': 1514,
  'Devices/00002CC005EA/Nad': 11,
  'Devices/00002CC005EA/Serial': '0001514',
  'Devices/00002CC005EB/CustomName': 'SmartBMV',
  [and so forth]
```

To change them, do a SetValue call to the DeviceInstance path like below. Or, perhaps easier, use the dbus-spy tool.

These lines read it, then changes it to 1, then reads it again:

```
root@ccgx:~# dbus -y com.victronenergy.vecan.can0
/Devices/00002CC005EB/DeviceInstance GetValue
value = 0
root@ccgx:~# dbus -y com.victronenergy.vecan.can0
/Devices/00002CC005EB/DeviceInstance SetValue %1
retval = 0
```

```
root@ccgx:~# dbus -y com.victronenergy.vecan.can0  
/Devices/00002CC005EB/DeviceInstance GetValue  
value = 1
```

[note that numbers, like can0, and 00002CC005EB can ofcourse be different on your system].

6.4 New method - changing Data instance

This applies only the NMEA2000-out feature. See links on top of page for what the NMEA2000 out feature is.

The data instances used for the NMEA2000 out feature are stored in local settings. Here is a snippet of the lines, taken by using the dbus-spy tool that also allows changing entries:

The Data instances are the "Battery-", "DCDetailed-", and so forth instances.

```
Settings/Vecan/can0/Forward/battery/256/BatteryInstance0      0      <-  
Data instance for main voltage measurement  
Settings/Vecan/can0/Forward/battery/256/BatteryInstance1      1      <-  
Data instance for starter or mid-voltage measurement  
Settings/Vecan/can0/Forward/battery/256/Description2  
Settings/Vecan/can0/Forward/battery/256/IdentityNumber        15  
Settings/Vecan/can0/Forward/battery/256/Instance              1  
Settings/Vecan/can0/Forward/battery/256/Nad                   233    <-  
Source address - no need, also not good, to change this  
Settings/Vecan/can0/Forward/battery/256/SwitchInstance1       0      <-  
Data instance for switchbank  
Settings/Vecan/can0/Forward/battery/256/SystemInstance        0  
Settings/Vecan/can0/Forward/solarcharger/0/DcDataInstance0    0  
Settings/Vecan/can0/Forward/solarcharger/0/DcDataInstance1    1  
Settings/Vecan/can0/Forward/solarcharger/0/Description2  
Settings/Vecan/can0/Forward/solarcharger/0/IdentityNumber     25  
Settings/Vecan/can0/Forward/solarcharger/0/Instance           0  
Settings/Vecan/can0/Forward/solarcharger/0/Nad                 36  
Settings/Vecan/can0/Forward/solarcharger/0/SystemInsta        0  
Settings/Vecan/can0/Forward/solarcharger/1/DcDataInstance0    0      <-  
Battery voltage & current  
Settings/Vecan/can0/Forward/solarcharger/1/DcDataInstance1    1      <- PV  
voltage & current  
Settings/Vecan/can0/Forward/solarcharger/1/Description2  
Settings/Vecan/can0/Forward/solarcharger/1/IdentityNumber     24  
Settings/Vecan/can0/Forward/solarcharger/1/Instance           0  
Settings/Vecan/can0/Forward/solarcharger/1/Nad                 36  
Settings/Vecan/can0/Forward/solarcharger/1/SystemInstance     0  
Settings/Vecan/can0/Forward/solarcharger/258/DcDataInstance0  0  
Settings/Vecan/can0/Forward/solarcharger/258/DcDataInstance1  1  
Settings/Vecan/can0/Forward/solarcharger/258/Description2  
Settings/Vecan/can0/Forward/solarcharger/258/IdentityNumber   23
```

Settings/Vecan/can0/Forward/solarcharger/258/Instance	0
Settings/Vecan/can0/Forward/solarcharger/258/Nad	36
Settings/Vecan/can0/Forward/solarcharger/258/SystemInstance	0

6.5 Old method

(Only allows changing Device instances - not data instances as used in the NMEA2000-out function)

Step 1. List the devices:

```
root@ccgx:~# dbus -y  
com.victronenergy.bms.socketcan_can0_di0_uc10  
com.victronenergy.charger.socketcan_can0_di1_uc12983
```

It shows a Skylla-i (the charger). di1 in the name means that it is currently on DeviceInstance 1.

Step 2. Change it, for example, to 4:

```
root@ccgx:~# dbus -y com.victronenergy.charger.socketcan_can0_di0_uc12983  
/DeviceInstance SetValue %4  
retval = 0
```

Step 3. Wait a few seconds, and double check:

```
root@ccgx:~# dbus -y  
com.victronenergy.bms.socketcan_can0_di0_uc10  
com.victronenergy.charger.socketcan_can0_di4_uc12983
```

Device instance changed successful!

From:
<https://www.victronenergy.com/live/> - **Victron Energy**

Permanent link:
https://www.victronenergy.com/live/ve.can:changing_nmea2000_instances

Last update: **2025-02-06 22:04**

