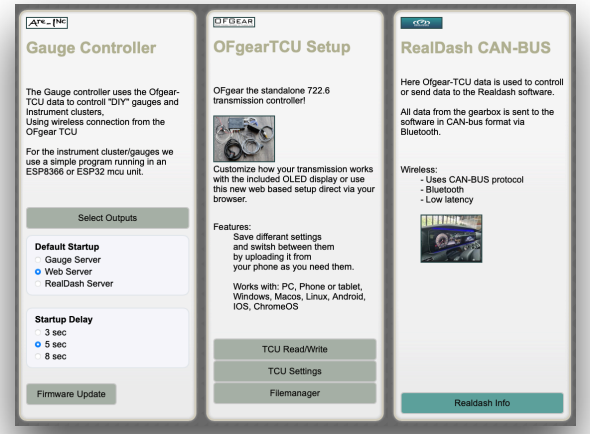
Are_Inc

Ofgear TCU Controller

INTRODUCTION:

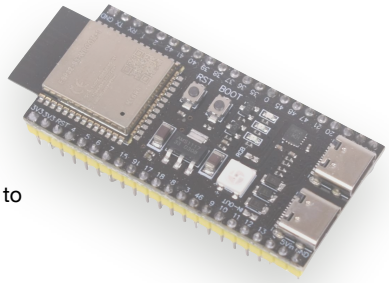


After some years using the OFgearTCU in our car fleet together with a verity of engines and setups, we found that we wanted a smoother way of setting up the TCU controller for different cars & use.
The PC software offered by OFgear is working well and fill the functions needed to setup the TCU, but it require a PC with Windows to run it…

The tool we use when adjusting the engines is a Tablet or a Phone, and we wanted the same for the OFgear system.
The goal set for the project where as follows:
- No software to install.
- Platform independent.
- Easy GUI that is relatable for OFgear PC software users.
- Possible to store several setups with load setup function.
- Easy installed hardware.
- Option of remote control the unit from ex. dashboard.
- Support for exporting live data to dashboards.
- Support for RealDash CAN software.
- Possible to create own gauges/dashboards with free software and easy Arduino code examples

During the development we went trough several versions for solving the connection between the OFgearTCU and the Are-Inc_TCU controller, but we ended up using the existing USB connection on the OFgear unit to connect the our system.

THE HARDWARE:



We ended up using a MCU with some powerful data:
240 MHz, 32-bit LX7 microprocessor
8 MB RAM
16 MB Quad speed Flash drive.

We are running true USB-OTG to connect to OFgearTCU. We run a web server onboard, Bluetooth for RealDash connection, and a wireless system to connect to DIY dashboards and mechanical gauges.

Easy installation with just an USB cable and 9-16volt power.

USER GUI:

There is only a button and a LED-light on the unit to guide you as a user. Both the button and the LED has external connection as well, to offer a remotely located GUI if needed. (If you like to add a button in your dashboard to for easy access to the setup.

For the first start you just hookup the power and wait for the led to turn “Yellow” then you can login to the wifi A.P and start setup your unit.

When you are done with your settings, the LED-light will always give you “status” information for the unit.

There is no need for connected laptops/Tablet/Phones when you are up and running. And if you need to change any settings all you need to do is, reboot the unit holding down the button, wait for the “Yellow” light and then logon via the wifi A.P and start altering the settings.

OFGEAR

OFgear has no affiliation to Are-Inc and has no reliability towards this product and software.!

FIRST STARTUP:

Hold the button on the unit down and connect the power to the unit (9-16 volt).
The LED-light on the unit will start glowing in a repeated pattern,

Purple -> Yellow -> **Blue**

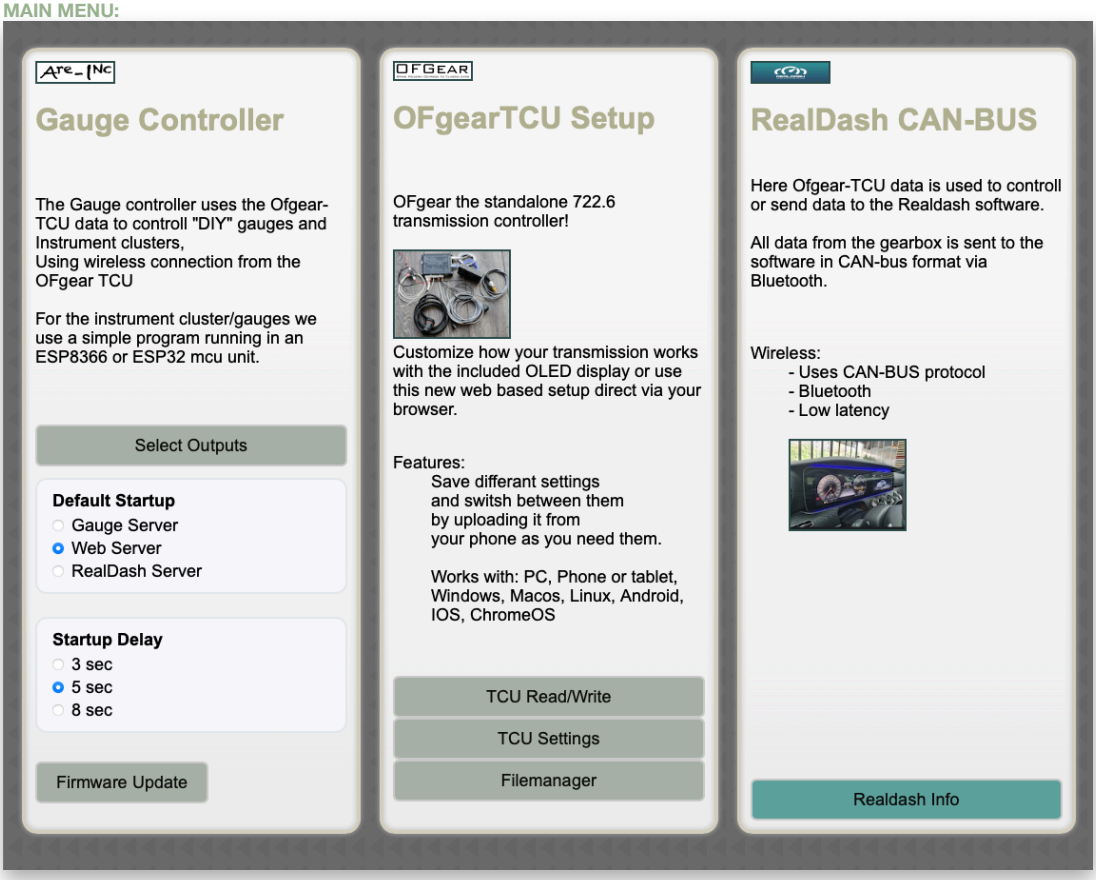
Purple = ESP-NOW master
Yellow = WEB server mode (Setup mode)
Blue = Bluetooth CAN-BUS connection for the RealDash software.

Release the button when the LED is glowing Yellow.
This will start the WEB server on the unit and there will be a WIFI access point presented to you.
Look in your WIFI connection list for the SSID "**Are-Inc_TCU-2025**" and select that.
You will be asked for the password and here you enter "**12345678**"

SSID: **Are-Inc_TCU-2025**
Password: **123456789**
Address: **192.168.4.1**

With that done, you start your favourite browser on any unit (Phone/Tablet/PC) and enter the address 192.168.4.1 and you will be presented with the main menu of the Are-Inc TCU controller.

MAIN MENU:

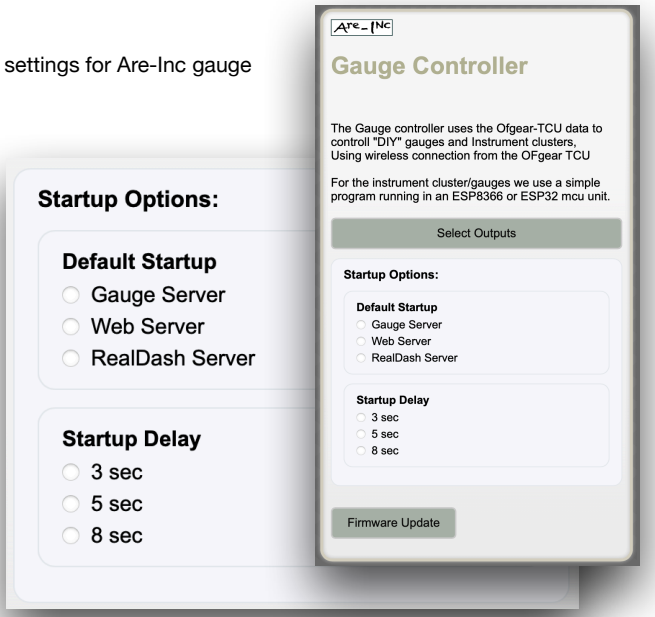


MENU SECTIONS:

The main menu is divided into 3 sections, Gauge Controller, OFgearTCU Setup & RealDash CAN-BUS.

Gauge Controller:

Here you will find the main settings for the unit and the settings for Are-Inc gauge controller function.



Startup Options:

Default Startup
Sets the boot mode for the unit if the button is not used. The selected option will be the mode the unit starts in of you don't use the button for select.

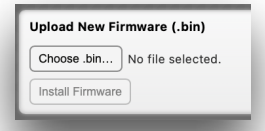
Startup Delay
Sets the interval between the options when the button is pushed during startup.
Ex. If the button is held during statup, the different boot modes will alter in the pattern:

Purple (ESP-NOW) -> Yellow (Web-Server) -> Blue (Realdash)

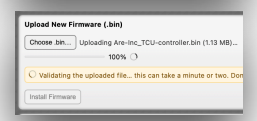
And continues until the button is released.
The mode shown by the LED when the button is released will be the Boot mode.
This Startup Delay sets the time between the altering modes.

FIRMWARE UPDATE:

When a new release of the controller software is released, you can download the new file and update your unit via the "Update GUI".

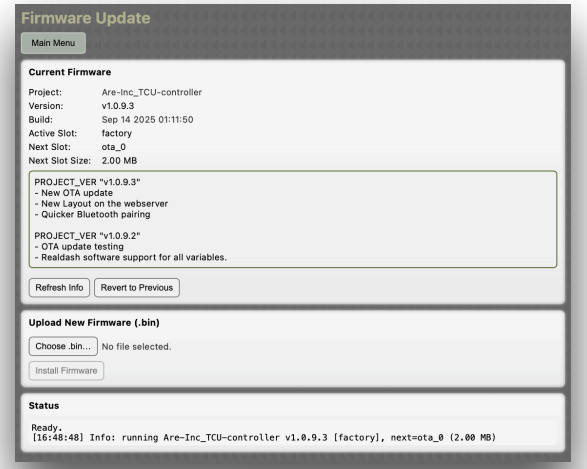


Click on "Choose .bin" and point to your downloaded firmware .bin file.

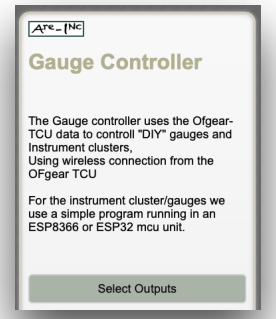


Wait for the file to upload and become verified.

Then click "Install firmware"



This will take around 2-4 minutes to Finnish and you will always se the status during the steps.
The update is done when the unit has restarted and you see the "Install finnished. Now running …… on OTA_ …"

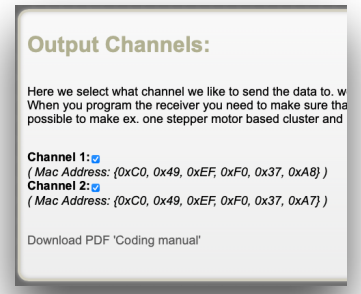


SELECT OUTPUTS:

This will open the setup for the “ESP-NOW” function. This enables the unit to send information from the OFgear TCU to your D.I.Y. gauge cluster or LCD screen.

We use the sturdy ESP-NOW protocol to wireless send data from the TCU to your self produced gauges or LCD screens.

We provide templates in ArduinoIDE for the ESP32 and ESP8266 MCU’s.
Among the templates you will find solutions / examples for LCD-screens, Gauge steppers, RC-servo gauges and more…
We have used this to create mechanical gauges using “gauge stepper motors” for Speed, RPM, Boost, Temperature, and more. You are free to combine that with an LCD screen of choice, or why not create a touch controlled screen with all the available data from the OFgear TCU…



OUTPUT CHANNELS:

The ESP-NOW has 2 output channels. The data sent out on respective channel is selectable via a column of “check boxes” for each variable available from the OFgearTCU.

To keep the “bandwidth” load as low as possible we recommend to only send the data you need for each channel.
The data is updated several times / second and the less data sent the higher update rte is archived.

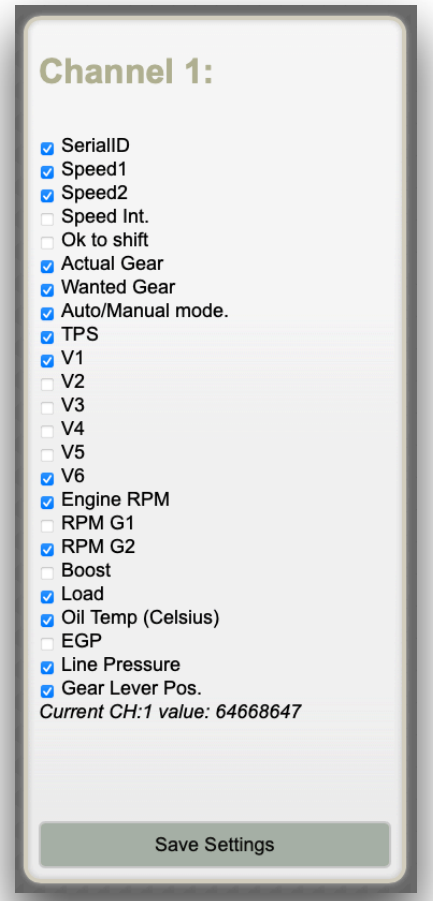
You can select what channel you like to have active, ch1, ch2 or both.
Use the “Check box” to select the channel(s) you like to have active.

SELECTABLE VARIABLES:

Each channel has a column with selectable variables from the OFgearTCU. You select what data you need for your project to be sent out / channel.

Use the “Check boxes” to select the data you are interested of for the active channel(s).

Explanation for each variable is found in the “TCU Variable” section of this document, and is also explained in the “programming manual” for the ArduinoIDE examples.



To use the function you start the unit in “ESP-NOW” mode (Purple LED) and then prepare your ESP32 or ESP8266 with your code that uses the available TCU information.

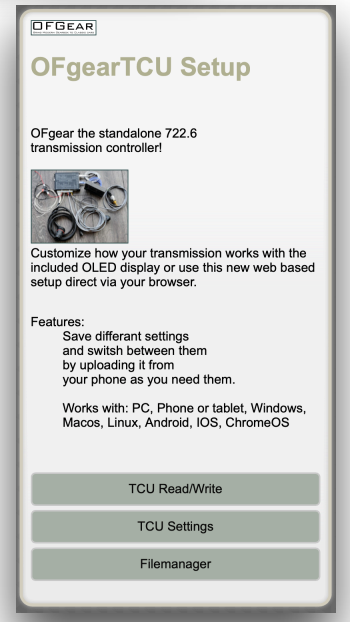
Now you can easily create a “receiver” with the “Stepper Gauge” example that shows the Speed, and just mount the stepper motor in your original speedometer.

Or use the “General Receive” example and make a program that converts the speed output from OFgearTCU to an output that fits your original digital/ electronic speedometer in your car.

TEST AND DEBUG:

This section is prepared for the ArduinoIDE examples but will not be activated until we have verified that our examples are strong and universal enough to actually help you as a user. This will come in a future release.

OFgearTCU Setup



This is the section where you get direct access to the OFgearTCU for altering settings and save/load different setups via the GUI.
You can alter the settings via your Phone/Tablet/PC without cable connection to the OFgearTCU. There is no software to download, you just need to start your favourite web browser and you are ready to start setting up your OFgearTCU.

The benefits are many with this solution:

- No cabled connection.
- No software to install/download.
- “On the fly” changes.
- Possible to store several different setups to files that you can store and then load into the OFgearTCU when you need them.
  - One for Everyday driving
  - One for the Street
  - One for the track
  - summer v.s. Winter
  - Towing trailer
- Full backup of your files.
- a.s.o….

For more information regarding the settings for the OFgearTCU please refer to the homepage of **OF gear**. https://www.ofgear.dk/

The GUI (User Interface) has a lot of similarities to the OFgear PC software and we have tried to keep as much of that software’s functionality & added some of our own to make the settings easier for you as a user.

(We hope to get some of you to help us put a document/guide for all the settings for the OFgearTCU. If we manage to do gather that information, our next add-on will be an A.I assisted setup to help speed up the settings…)

The OFgearTCU Setup has 3 main sections:

#1. TCU Read/Write

Here you can download the settings from the OFgearTCU to the GUI and look at the settings that you have in the TCU right now. You can save the settings to a config file for backup or later use. You can directly start changing settings and save those direct “live” to the TCU or just save them to a config file.
OR…
You can load a setup file that you have saved in the drive and write the whole setup direct to the TCU.
You have close to 6MB of storage on the unit for your own setup files. (Over 850 setups can be stored…)

#2. TCU Settings

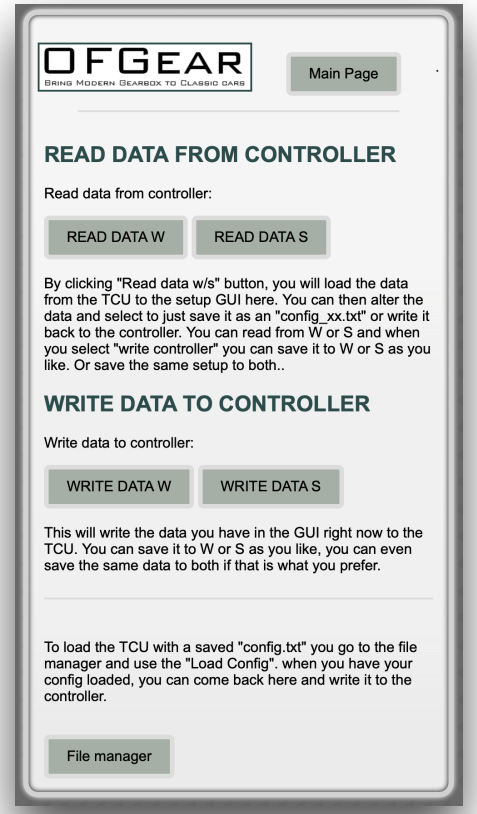
This is the Setup GUI where you have access to every editable data in the OFgearTCU.
Everything from basic settings, to shift times, Boost control, Shift firmness and more…

#3. File manager

This is your storage where you can save/download/upload your setup files. There are some simple functions to rename/ delete and load files as well as looking at the file size…

We added a simple “Load Config” button to give you a direct list of all your “Configuration files” saved in the storage, and easily load them into the settings GUI or just send it direct to the TCU.

TCU READ/WRITE:



"Read Data W" and "Read Data S" buttons do just that, It reads the TCU EEPROM for the settings that is stored in the TCU and present the data in the WEB-GUI for you to work on.
When you work with the data you can use buttons in the GUI to send the changes direct, or you can save the new settings to a config file and store it in the storage.

"Write Data W" and "Write Data S" will flush the current settings from the GUI to the TCU EEPROM and also save it as the "default" config.txt in storage.

Usage:

The process of reading the EEPROM or writing to it takes a lot of time, and you should never do this with the car in use. Make sure to have the car parked and the gearbox I "P" to be safe.
Always let the process finish before you cut the power to the gearbox and this unit, if not you might risk to "***BRICK***" the OFgearTCU and then you need to flash the firmware back on in again.

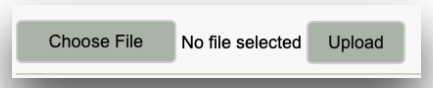
Always make sure you saved your setup before attempting to transfer it to the TCU incase you run into problems when flashing the EEPOM.

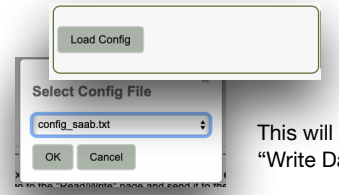
FILEMANAGER:

The file manager is used for updating the HTML files on the server and to save and manage the config files in your storage.
We have added some small functions to help you with the most common routines.

Upload files:

Click "Choose File" and select the file you like to upload.
The server only support small files (storage is about 6mb in total).
Then click "Upload". If the file already exists you will be prompt to overwrite the file or cancel the upload.





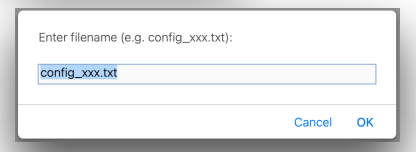
Load Config:

Click this button to see a list of the different config files you have stored in memory.
Select one from the list in the popup window that appear and click "OK"
This will load the selected config into the GUI and you can start working on the settings, or go straight to "Write Data W or S" (See above information about Read/Write)



Save As:

When you are satisfied with the settings in the GUI and you want to save it to a "Config file" in storage, you click the "Save As…" button and enter a name in the popup that appear.
The name should always begin with "config_" then your unique name, then end with the extension ".txt"..

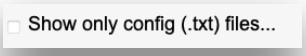


EX: *config_myname.txt*

File handeling:

You find a list of all the files you have in the storage.
There will be a lot of HTML files and files that is used by the system here as well as your config files.

We have let the users of our system to have access to the files of the system so that if you have HTML / JAVA knowledge you can change the layout or add functions of your own.



To make the file list less cluttered you can use the check box
“Show only config (.txt) files…”

This will clear the file list from all files except the config files.



Filename	Size (bytes)	Actions
favicon.ico	15406	Download | Delete | Rename
realdash.html	8449	Download | Delete | Rename
logo_areinc.png	33343	Download | Delete | Rename
tcu.html	54130	Download | Delete | Rename
can_ofgear.xml	1761	Download | Delete | Rename
are-inc.css	10012	Download | Delete | Rename
boost.html	18600	Download | Delete | Rename
auxpwm.html	17192	Download | Delete | Rename
readwrite.html	10439	Download | Delete | Rename
realdash.css	10012	Download | Delete | Rename
lookup.csv	5810	Download | Delete | Rename
are-inc_user_manual.pdf	117610	Download | Delete | Rename
config_saab.txt	5552	Download | Delete | Rename
ota.html	21135	Download | Delete | Rename
update.txt	190	Download | Delete | Rename
ofgear.css	13200	Download | Delete | Rename
config.txt	5549	Download | Delete | Rename
gauge.html	19314	Download | Delete | Rename
app.bin	1183184	Download | Delete | Rename

Download:

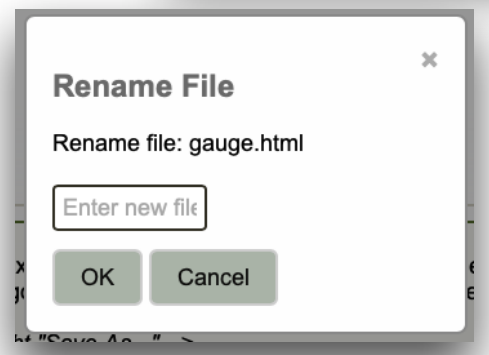
Click on the Download link if you like to download a file.
This will go fairly quick unless the file is large.
This is just an MCU so the computing power is a bit limited.

Delete:

Click to delete a file, you will be prompted with a Warning that you will delete the file if you continue.
Answer OK or Cancel.

Rename:

This function can be used to rename a config file if you feel you need a better name for the file.



Be careful using the “Rename” / “Delete” functions because they also work on any of the system files in the storage.
If you delete the “filemanager.html” there is no way of getting access to the system storage, and you will have to contact us to get a reset of your unit.

(There is some commands that you can use to save your system incase you delete the file manager by misstake, but you need to contact our support to get help with that.)

TCU Settings

This is a bundle of pages with settings for the OFgearTCU. We have divided the settings into several different pages for an easier overview. There is a lot of settings in here and information on what they do and how to use them, you will find on the OFgear home page “https://ofgear.dk/“ or in forums online.

We will show some of the settings we have made a different approach to compared to the OFgear PC software, the rest is mostly close to the settings via the OFgear display or PC software.

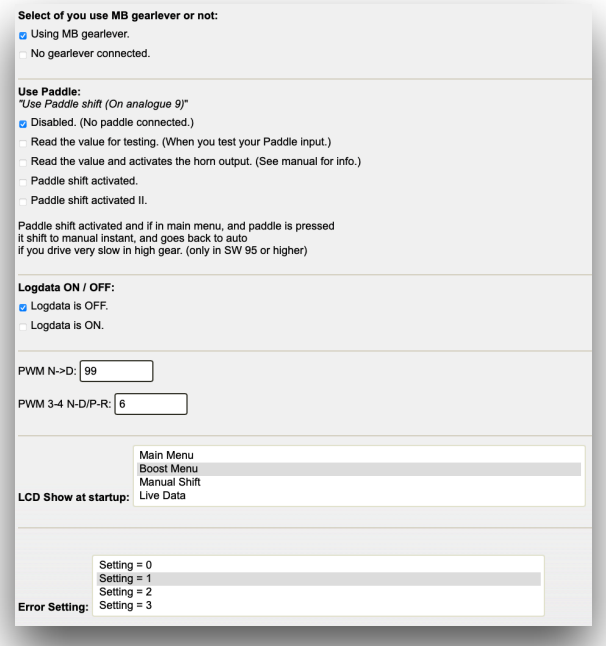
First some general information on the buttons found on the pages.:

Save all to Config	This button will save all settings/Data in the GUI to the file “config.txt” in storage. It will write direct without warning of writing over the existing config.txt. If you like to save the data to a differant file, use the file manager “Save As…” function.
Save To TCU	This will send the data in the section you currently are in, direct to the OFgearTCU EEprom. But it will NOT save it to the config.txt.
Update Chart	If you find this button it will “update” the diagram/chart next to it with the latest changes in data.

The setup pages are devided into the following pages:

Main Menu	The basic setup for the TCU, like gear-lever, Ext/Int speed & RPM, TPS and some other basics.
Shift Speed	The standard shift speeds for Up/Down shift, Kick-down, TPS correction,
Timing	Shift valve times, Lockup spikes, Delay 3-4 gear, time between shifts…
Shift firmness	Shift for each gear, Line High/Low, Line pressure…
Boost Menu	Anything regarding the boost controller…
Aux PWM output	Set the outputs for start relay, Boost valve …

We have tried to make some of the settings easier to get around and we will continue changing the layout as we get feedback from you users.



We have changed the settings for the Gear lever, Paddle, Log-data, LCD show @ startup and error level settings.

We have some more changes in the pipeline for next version, where we try and add some guides for each setting to make it clearer what the setting effect.

Are-INC

RealDash Software

We realised that several of our “Test users” during the development used a software named “RealDash” on a tablet or screen to create an instrument cluster for the car.
When adding the OFgearTCU to the gearbox, there was a wish to be able to easily add data from the gearbox to the instrument cluster.

As we already had the “Live Feed” from the TCU in our unit, we added a bluetooth connection to send out data to the RealDash software in formatted CAN-BUS frames according to the RealDash CAN-bus API.
This gives you the possibility to setup gauges and indicators in you RealDash screen showing any of the TCU data.

We have a Low latency Bluetooth connection and feeds the data as fast as the MCU can push it out.
The limitation is the speed of the OFgearTCU main boards MCU and the serial connection created by it.
We still have a feed that updates several times per second and gives a “jitter” free feed for the gauges and indicators in your dashboard.

The data sent out from the TCU are:

Speed1
Speed2
SpeedInt
OkToShift
ActualGear
WantedGear
AutoManual
TPS
V1
V2
V3
V4
V5
V6
EngineRPM
RPMG1
RPMG2
Boost
Load
OilTemp
EGP
LinePressure
GearLever

Sent as CAN-bus frames:

// Frame 0xC90
frame0 + 0, SerialID
frame0 + 2, Speed1
frame0 + 4, Speed2
frame0 + 6, SpeedInt

// Frame 0xC91
frame1 + 0, OkToShift
frame1 + 2, ActualGear
frame1 + 4, WantedGear
frame1 + 6, AutoManual

// Frame 0xC92
frame2 + 0, TPS
frame2 + 2, V1
frame2 + 4, V2
frame2 + 6, V3

// Frame 0xC93
frame3 + 0, V4
frame3 + 2, V5
frame3 + 4, V6
frame3 + 6, EngineRPM

// Frame 0xC94
frame4 + 0, RPMG1
frame4 + 2, RPMG2
frame4 + 4, boost
frame4 + 6, Load

// Frame 0xC95
frame5 + 0, OilTemp
frame5 + 2, EGP
frame5 + 4, LinePressure
frame5 + 6, GearLever

In the RealDash software you can upload your own "XML" file to give you easy access to the variables that is sent from the OfgearTCU.

You find the XML file in the file manager or you can download from the support page www.are-inc.se/support

Follow the instructions for the RealDash software to setup a Bluetooth serial connection. The name of the Bluetooth connection will be "OfgearRealDash"...

The values will be streamed from the OfgearTCU "live" for you to show in your dashboard. You will find more information what the values stand for and hints how to use them in the "Programming Manual" In our homepage.

You will find More information about the RealDash software on there home page at: https://realdash.net/index.php





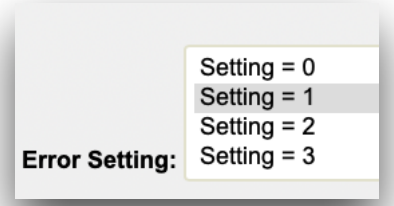
Simple guide to some of the settings:

Error setting:
You find this setting on the “Main Setup” page.

Setting:
0 (Please do not use this setting) No fail messages and no Lock shift.
1 1,2,3 gives a fail message no lock of shift.

If internal speed sensor,
4,5 Locks shift if they happens, and gives a fail messages.
If external speed sensor
4 Does not lock shift
5 Locks shift if they happens, and gives a fail messages.

2 3 only gives a speed fail message no lock of shift.
1,2,4,5 Locks shift if they happens, and gives a fail messages.
3 1,2,3,4,5 Locks shift if they happens, and gives a fail messages.



Starter Lockout:
To use a relay to stop the engine from starting you use AUX-2 as an output to control the relay.
Go to the page “Aux PWM output” and enter the number “2” in the “Use PWM AUX-2” field.

AUX-2 output setting:

If set to:
0 Takes input from the TPS, EGP, RPM and gives out the lowest of the them.
1. Then the signal is used to pull the solonoid on the Gearlever, then it can not be moved in to R and P while driving more then 10 km/h (It gives a ground signla, you have to supply 12V to the other side of solenoid)

2. Then it is used as starter lockout. (See above…)
3. Then the AUX-2 PWM out. Takes input from Gear oil temp and Pulse to ground at higher then 90° and releases at 85°

Lockup 2-4 gear:
You find this setting under the “Main Setup” page.
The field “Lock up 2-4 gear:” gives you the following options:

0 Only lockup in 5th gear
-1 Like 0, but do not unlock if TPS goes below 10%
1 Lockup in 2,3,4 and 5th gear. This is useful for keeping the temperature down when pulling heavy loads, or driving uphill. The 2 – 4 overdrive feature is activated if you put gear lever in position 4.
2 Always lockup in 2-5 gear also if the lever is in “D”
3 Like 2 but also no unlock with no throttle.
4 Like 3 but also shifts gear with locked converter, converter only unlocks if below 30 km/h This is only useful at Racetrack.
5 Special only unlock if speed is below 10 Km/h and lock if in “2” and speed > 10, this is a special setting for an Electrical vehicle (only from ver_140)
6 Lockup over a switch 12V in on (D0 kick-down 12V in), but then you can NOT use kick-down on this input, no soft lockup in this mode as it is for Racing / Drifting (only from ver_143)
7 Lockup if speed bigger then 30 km/hr and locked under shift And it unlocks if TPS > 75% And it unlocks if speed is < 30 km/h Does "NOT" unlock it TPS < 10% (only from ver_170)
8 Lockup if speed bigger then 30 km/hr and locked under shift And it unlocks if TPS > 75% And it unlocks if speed is < 30 km/h Does unlock it TPS < 10% (only from ver_170)
9 Locks if Gear > 2 and Throttle > 10% and speed > 30, locks hard, this setting is for 1⁄4 mile

Index:

Ofgear TCU Controller 1

- Introduction: 1
- The Hardware: 1
- User GUI: 1
- First Startup: 2
- Menu Sections: 3

Gauge Controller: 3

- Firmware Update: 3
- Select Outputs: 4
- Output Channels: 4
- Selectable variables: 4
- Test and Debug: 4

OFgearTCU Setup 5

- TCU Read/Write: 6
- Filemanager: 6

TCU Settings 8

RealDash Software 9

Simple guide to some of the settings: 11