



Introduction

DashCommand is an app that acts as an advanced in-car computer. It is currently available for Windows, iOS (iPhone/iPod Touch/iPad), and Android. The following troubleshooting topics are covered in this guide:

- [Connection Troubleshooting](#)
- [Connection Errors](#)
- [Data, Gauge, and PID Troubleshooting](#)
- [Inclinometer / Skid Pad Troubleshooting](#)
- [Performance Issues & Troubleshooting](#)

Having a problem not listed in this Troubleshooting Guide? Please contact support@palmerperformance.com for more help. The most recent version of DashCommand while creating this document is v3.4.0.

Connection Troubleshooting

The connection process is the most common place to have troubles. First off, you need to make sure that your interface is supported by the platform you are using. If you are using an iPhone, you need to have WiFi interface or a GoPoint GL1/BT1/BT1A . The only supported Bluetooth hardware for the iPhone is the GoPoint BT1/BT1A. If you have an Android phone, just the opposite is true. Android supports Bluetooth interfaces, but does not support WiFi (more specifically, Android doesn't support Ad-Hoc networks).

You also need to make sure you have the correct interface type selected in Settings. You should select GoPoint if you are using a GoPoint GL1 or BT1 interface, and ELM if you are using any ELM interface (OBLink, Kiwi, OBDKey, etc).

Now you know that you've got the correct interface selected, you need to make sure your device is connected to your interface. For the GL1, just plug it in to the dock connect on your iPhone. For a WiFi interface, you need to join the WiFi network created by your interface (you can find more information on connecting to WiFi from the DashCommand WiFi Setup Guide on our downloads page). If you are using Bluetooth, you need to go to the system settings and make sure you have paired with your interface.

You have the right interface, it's selected in Settings, and your phone is connected to it. Now it's time to get DashCommand connected. If you haven't created a vehicle in the Vehicle Manager yet, do that now. You should be ready to just hit the Connection button in the main menu.

***Note, if you have a Ford 7.3L Powerstroke Diesel, you will need to check the box to disable Mode \$09 in the Vehicle Manager OBD-II Settings. Not doing so could result in your vehicle shutting off when you make a connection. This is a bug in the 7.3L Powerstroke PCM causing it to shut off the vehicle if a mode \$09 request is made while the vehicle is running. This only affects the 7.3L Powerstroke.

Connection Errors

OBD-II interface not found - This means that DashCommand could not find your OBD-II hardware. First step, make sure your key is on and the OBD-II hardware is powered up (vehicle can be running also) . Go to Settings and make sure you have selected the correct Hardware Type. If those conditions are correct, then close DashCommand and make sure the device is connected correctly to your device. If it is Bluetooth, check to make sure it is paired. If it is WiFi make sure you are connected to the WiFi network and your IP is correct if you are not using an OBDLink WiFi (only the OBDLink WiFi does not require an IP address). The ONLY Bluetooth that will connect with the iPhone/iPod/iPad is the GoPoint BT1/BT1A, if you are on iOS and have an ELM327 Bluetooth, you will need a different hardware interface. If you are on Android you cannot use a WiFi OBD-II hardware, if you have a WiFi OBD-II hardware on Android, you will need a different hardware interface.

Error detecting OBD-II Protocol – This error means that we communicated with your OBD-II hardware, but your vehicle did not respond to any OBD-II message requests. This error could be for a variety of reasons. First thing to check is ensuring your vehicle is OBD-II compliant, just because you have an OBD-II shaped plug does not mean it is OBD-II compliant. Check your emissions label under the hood for text saying it is OBD-II/JOBD/EOBD compliant. If you are OBD-II compliant, you could try some settings changes in the Vehicle Manager. Enter the Vehicle Manager, click on your vehicle, then click OBD-II Settings.

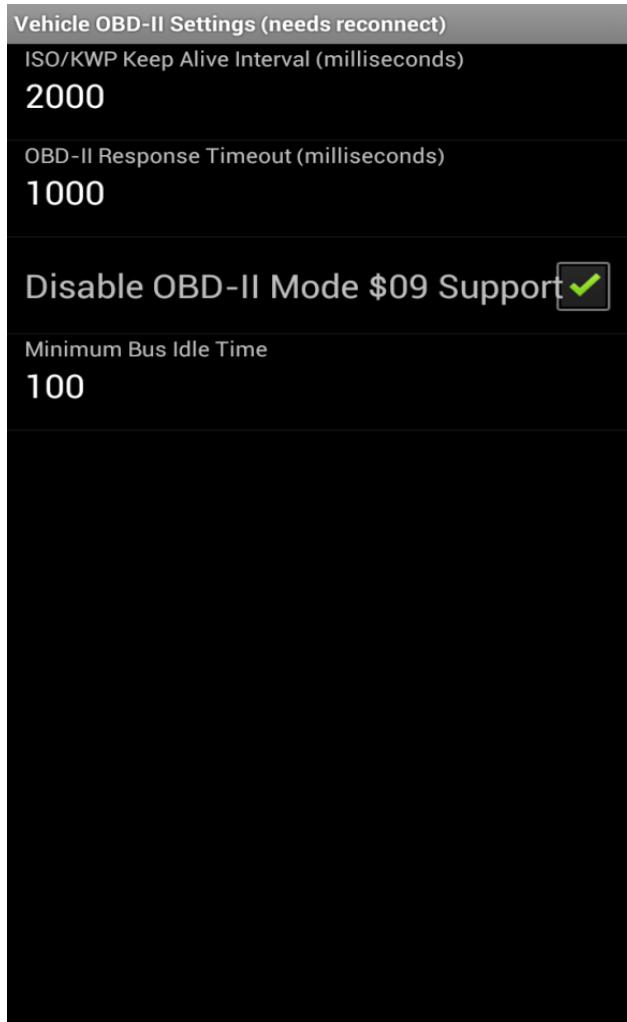
Make the changes as follows:

ISO/KWP Keep Alive Interval – 2000ms

OBD-II Responses Timeout – 1000ms

Disable OBD-II Mode \$09 Support – Checked or On

Minimum Bus Idle Time – 100ms



*Note for iOS the background will be white in your settings view

If you get a connection from these changes, keep reducing the Minimum Bus Idle Time down as low as you can get it while still holding a connection. The higher that number is, the slower it can make your refresh rates, none of the other values have any effect on refresh rates. Most vehicles will be fine at zero, but some earlier OBD-II vehicles will need a value greater than zero entered there.

Failed to open data port – This error means that you have changed the settings inside of DashCommand to select a specific port for your hardware. If this is not the correct port, it will fail. This can be left as 'Auto'. If you are using this to enter in a WiFi IP address, you are using it incorrectly and need to refer to the WiFi Setup Guide for the correct way to enter an IP address.

You must select both an aux input port and type – In most cases if you get this error, you have accidentally checked the box ‘Aux Input Only’, and that means just that. If you click that box you will not get any OBD-II data from your vehicle, you will only get Aux data. If you do not have an iMFD module, then you will not get any data at all even though it shows the green dot at the top like you have a connection. If you do have an iMFD or other supported aux module, you will need to select the Aux input type to make a connection to your aux module.

Invalid Message Filter or Interface Communication Failure – This error occurs most often from the interface getting some interference during the connection. If you see this you should simply try again. If this problem persists, you will need to contact support@palmerperformance.com to diagnose the issue further and find a solution.

If you are still having trouble connecting after following all these steps, please contact support@palmerperformance.com about it. Please include information on what platform you are using (iPhone, Android, Windows), what device you are running (iPhone, iPad, Samsung Galaxy SIII, etc), and what vehicle you are connecting to (Make, Model, Year, and engine).

Data, Gauge, and PID Troubleshooting

Having trouble getting data, or the gauge values seem incorrect? The first step is to check your vehicle settings. The vehicle settings influence how a lot of data is calculated.

Horsepower and torque calculations rely on the Brake Specific Fuel Consumption vehicle setting. Entering a value that is too high will cause power numbers to be reduced. Also keep in mind this is power **at the engine**. This calculation is from the current level of performance of your engine, it will show reduction in power from altitude or poor air conditions, it does not contain any correction factors for less than ideal conditions.

The Engine Displacement setting determines how boost, MAP, and MAF are calculated. If your vehicle has a MAP sensor only (supports SAE.MAP), MAF values are calculated using the displacement. MAF values are required in the fuel economy calculations. If your vehicle uses a MAP sensor, you need to fill in this setting to get fuel economy data. On the flip side, vehicles with a MAF sensor only use this value to calculate MAP – which in turn is used to determine boost.

Distance to empty (or fuel range), is determined by using the Fuel Tank Capacity vehicle setting, and the SAE.FLI PID if its available from your vehicle. Many older vehicles don't support SAE.FLI at all, meaning you can't get DTE from these vehicles.

Incorrectly setting the Vehicle Speed Correction Factor will give you vehicle speed numbers that are wrong. This value is multiplied by the speed we get from the OBD-II port, so that users with oversized tires, or a not-stock axle ratio, can adjust to get the correct speed in DashCommand. Unless you need to adjust your speed for tires changes or similar, you should leave this setting blank (or set it to 1 – the default).

What if you're missing a value that isn't calculated, like RPM or vehicle speed? This is usually not an issue unless you are using a GoPoint GL1 (iPhone only) with an older vehicle. The GL1 is constantly requesting engine RPM from the vehicle, and in some cases it delays the vehicle's response to DashCommand long enough for DashCommand to think that a PID isn't supported. To fix this, (before you connect) go to the vehicle settings for your car. Decrease the ISO/KWP Keep Alive Interval slightly to 2000ms. And increase the OBD-II Timeout to 600ms. Then go connect to the vehicle. After you've connected, press the Validate PIDs button near the bottom of the vehicle settings.

Inclinometer / Skid Pad Troubleshooting

The Inclinometer and Skid Pad seem like simple tools, but the math behind them is very complicated and can introduce a lot of issues. Firstly, these screens both “zero” themselves when DashCommand is started. If you are holding the phone when you start DashCommand, you will need to manually re-zero them from the menu after you get your phone situated in a mount or cup holder. If your phone can move or wiggle, you’ll get bad readings from these accelerometer-based features.

If you see that the Inclinometer or Skid Pad show readings while the vehicle is not moving, on level ground, you need to select zero from the menu. For the Skid Pad, it should be perfectly centered and you’re done. This is where the complicated math comes in. The accelerometer in these devices doesn’t measure pitch and roll directly. Rather, it measures acceleration (hence the name “accelerometer”). DashCommand calculates pitch and roll by determining the difference between the gravity vector, and any other vector. We’ve chosen to use the vector that is from the center of the phone, directly out the screen. We are assuming that the screen is always facing the back of the vehicle, and it also means that the phone needs to be mounted more or less vertically for the Inclinometer feature to work correctly. If your phone isn’t vertical, you will always read some pitch even when on level ground.

Some newer phones also include a gyroscope. While this does allow for an accurate pitch to be calculated, we currently don’t have support for this sensor in DashCommand.

Performance Issues & Troubleshooting

Slow dashboards are the worst, and unfortunately there is more than 1 factor. There are 2 big influences in how fast the dashboards update, and a large number of smaller ones. One of the biggest bottlenecks is usually the vehicle itself. Older ISO or J1850 vehicles don't communicate very quickly, and can make dashboards seem choppy or slow. The other big slowdown is a combination of the device you are running on, and the DashXL skin set you are using. If you are using an ISO vehicle, it will be slow no matter what you do (J1850 is better, but still slow). There is **nothing** that can be done to speed up a vehicle itself that uses an old protocol. However, some things can be done to help DashCommand speed things along slightly.

The default Tuxedo skin set is a very large, detailed skin that shows just about every PID your vehicle could support. The only downside to this is that it's just too much for some older devices to display smoothly. Try using a different skin set from DashXL.net.

You can also try disabling the Persistent PIDs in Settings. Persistent PIDs are PIDs that are monitored all the time, no matter what screen or dashboard you are on. They can add as many as 10 extra SAE PIDs that the vehicle has to supply us with, and will greatly slow down vehicles using older protocols like ISO.