

rusEFI Manual

Contents

Quick Start	2
Minimal Setup	2
Waiting for your ECU to arrive?	2
Once your ECU has arrived	2
Getting Your Car Running	3
Tuning	3
Logging	3
Analyzing Logs	3
Share Your Tunes and Logs	3
Get Support	3
FAQ	3
How To Create a TunerStudio Project	5
Download and Install	5
Creating a Project	5
Get Running With a Wire-in ECU	8
Summary	8
Hardware requirements	8
Get Running With a Plug & Play ECU	14

Quick Start

Minimal Setup

All official units are shipped with firmware installed. Modern Windows comes with USB CDC drivers.

1. Download and Install TunerStudio (See Download and Install on page 5).
2. Click the 'Detect' button and let TunerStudio automatically find and download the correct definition file from the internet.

You are ready to go control boost!

Continue reading for more detailed setup steps.

Waiting for your ECU to arrive?

1. Download the rusEFI bundle (<https://wiki.rusefi.com/Download>) for your ECU.
2. Extract the bundle and launch the rusEFI console. Hit the "Install Drivers" button to install the Virtual Com Port and DFU drivers. If necessary the bundle will also help you install the Java runtime.

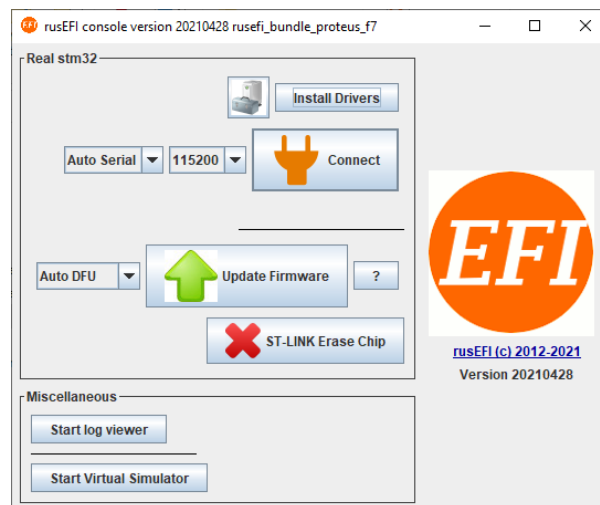


Figure 1: console

3. Get friendly on the rusEFI forums (<https://rusefi.com/forum>) Introduce yourself, start a build thread, and ask questions.
4. Download and Install TunerStudio (See Download and Install on page 5)
5. Create a project in TunerStudio (See Creating a Project on page 5)

Once your ECU has arrived

Your board comes with rusEFI firmware installed on it, but it should be updated.

How to update your firmware (<https://wiki.rusefi.com/HOWTO-Update-Firmware>)

Getting Your Car Running

Wire-in ECUs

Follow the Get Running With a Wire-in ECU (See Get Running With a Wire-in ECU on page 8) guide.

Plug & Play ECUs

Follow the Get Running With a Plug & Play ECU (See Get Running With a Plug & Play ECU on page 14) guide.

Tuning

Get Tuning (<https://wiki.rusefi.com/Get-tuning-with-TunerStudio-and-your-rusEFI>)

Logging

Any troubleshooting begins with logs. One of rusEFI powers is the amount of data points available for logging!

Shaky videos will not do. Even high-resolution screen shots will not do—*it has to be a proper log file*.

See the Logging Guide (<https://wiki.rusefi.com/Logging-Guide>).

Analyzing Logs

EFI Analytics MegaLogViewer (<https://www.efianalytics.com/MegaLogViewer/>) is an industry leading log charting application.

Share Your Tunes and Logs

You will definitely need your tune and log online for any community discussion.

rusEFI Online (<https://wiki.rusefi.com/Online>) is the platform to share and compare your tune with other tunes!

Get Support

See Support & Community (<https://wiki.rusefi.com/Support>)

FAQ

Q: *How do I change settings?*

A: You can change settings and your engine tune using TunerStudio (See How To Create a TunerStudio Project on page 5).

Q: *Do I need to compile source code myself?*

A: Nope, not at all. Prebuilt bundles are available on the Download Page (<https://wiki.rusefi.com/Download>).

Q: *I've made some changes in TunerStudio & hit Burn. 'Need Burn' has appeared but the values are not there after rusEFI reset?*

A: rusEFI does not write to internal flash memory while your engine is running. In order to really write the values into the permanent memory you would need to stop the engine (or disconnect

your stimulator). Online tuning is not affected by this issue; you can tune and the new values will be applied right away, but they will not be saved until you stop the engine.

Q: *I've remapped some analog inputs and outputs but nothing has changed, looks like the old pins are still used for output?*

A: For most of the pinout changes to be applied you need to reset rusEFI. Do not forget to stop the engine so that the new config is saved into internal memory!

Q: *What does this error code mean?*

A: See Error Codes (<https://wiki.rusefi.com/Error-Codes>)

How To Create a TunerStudio Project

EFI Analytics TunerStudio (<https://www.tunerstudio.com/index.php/tuner-studio>) is a third party commercial tuning application used to configure and tune your rusEFI ECU. TunerStudio is available on Windows, Mac and Linux.

Most rusEFI ECUs can be powered through the USB port, so you can set it up in the comfort of your living room and have it communicate with Tunerstudio before it is installed in the car. This approach is highly recommended.

Download and Install

TunerStudio (<https://www.efianalytics.com/TunerStudio/>)

MegaLog Viewer (<https://www.efianalytics.com/MegaLogViewer/download/>)

The paid versions of both programs are recommended.

Creating a Project

Connect TunerStudio to your ECU. You will need a USB cable to establish communication between your computer and the rusEFI unit. Most rusEFI units feature a USB port. microRusEFI requires an externally-wired USB port.

Your laptop should give you an audible notification when you plug in the ECU. This indicates that it sees a new device connected to a USB port.

Open TunerStudio and click *“Create New Project”*.



Figure 2: First project

TunerStudio needs an ECU Definition file (usually with .ini extension) in order to start communicating with rusEFI.

Auto-detect Mode

All official rusEFI boards support automatic ECU definition file download if connected via USB, serial port or Bluetooth. Just hit 'Detect' and wait for the name of your ECU to appear.

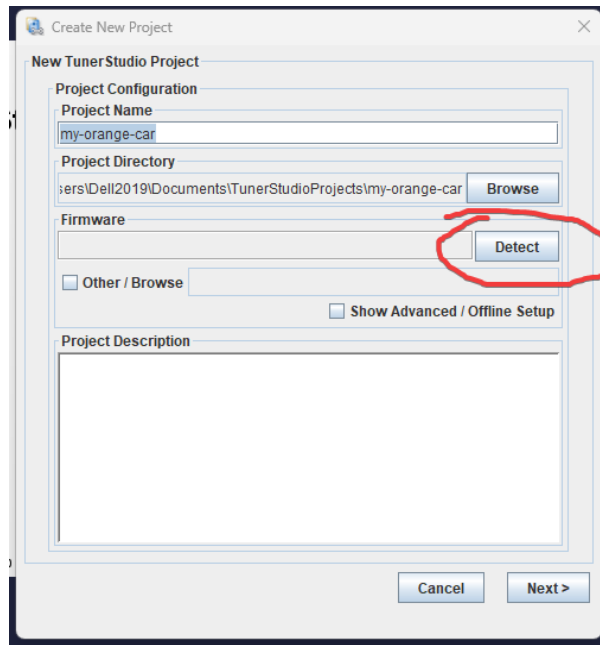


Figure 3: TunerStudio Detect Button

Manual Definition Selection

If the definition for your unit wasn't detected automatically, select the file from your rusEFI bundle (<https://wiki.rusefi.com/Download>).

Your rusEFI board should also present itself as a USB storage device to your computer, containing the .ini file you need. This is the most foolproof method of making sure you have the correct .ini file.

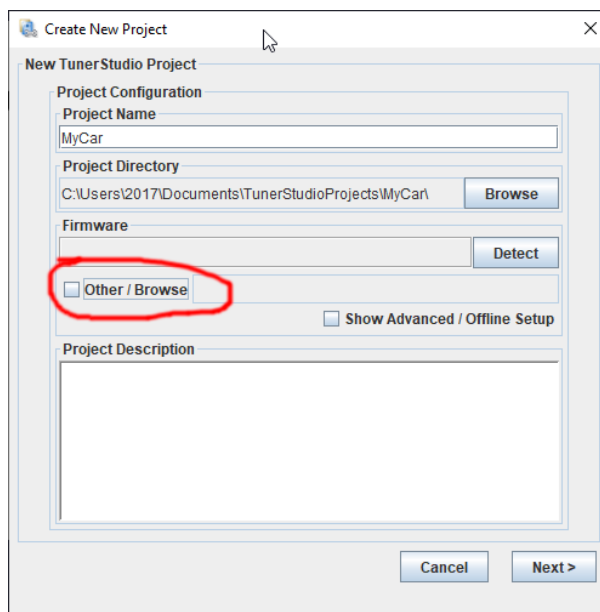


Figure 4: Menu

After clicking on the link you'll see this:

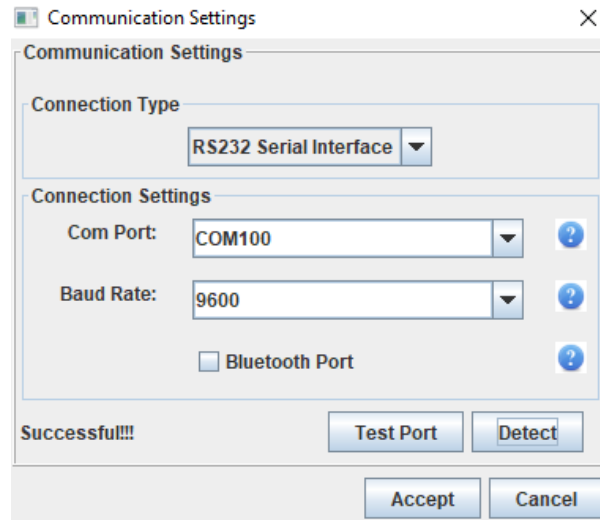


Figure 5: Menu

See Tunerstudio Connectivity (<https://wiki.rusefi.com/Tunerstudio-Connectivity>) if you need detailed instructions on these settings.

With this your TunerStudio screen should come to life! You should see sensor inputs and some output values like ignition timing and dwell. If your ECU isn't connected to a car, the values displayed might not make any sense, but at least you can see that TunerStudio and your computer are communicating.

Get Running With a Wire-in ECU

Summary

So you either have hardware under way, or are considering getting ruSEFI hardware. This manual is intended for those that are using purchased hardware. If you are not, don't worry, we also encourage DIY and like hearing back from those that have. We also encourage people to use the purchased boards as a starting point for a DIY effort, feel free to modify the board we'll likely make suggestions on how to make your effort better.

Hardware requirements

Plan the engine

The engine will need wires that connect to various sensors and devices. You will need crimp tools, soldering tools, and certain electrical and mechanical skills.

We suggest that before you purchase hardware, that you create a plan with a schematic. Even if the schematic is on a napkin, I suggest you take picture of it with your phone or scan it in and get feedback from members in the forum. We can help steer you to a successful build.

Start ordering components Once you have a plan, you can start making a bill of materials. You'll likely be purchasing harnesses, wires, connectors, and all sorts of things. You may have to do junk yard runs to get certain hard to obtain items, ect. If you have your rough schematic at arms length, it will really help you know what you need and when you'll need it.

Preparing your engine

Physically locating the controller Place some place that is away from hot items like the exhaust. Mount on a piece of steal that can function as a heat sink. Preferably in a dry well ventilated location, were the wires can be easily routed to and from.

Route the wires and harnesses

- Start by drawing out your engine layout. This can be as simple as a sketch on a napkin or pizza box. This is handy as it tracks how many injectors you need, what kind of IAC, TPS, IAT, ect options you need.
- It is recommended you route your wires in two different groups, analog and digital/high power. This helps ensure good signals make it to and from the ECU.
- Wire size and fusing go together. The fuse is used to protect the wire against electrical fire. So the wire you can use depends largely on the upstream fuse you select. High-Z injectors are going to draw about 1A, so you can technically get by with tiny stuff like 24AWG or 22AWG. However that is generally not very strong and is prone to breaking and failure. So you would be well advised to follow the NFPA79's suggestion to use a min of 18AWG, as smaller is easily mechanically damaged. In terms of electrical sizing this is very much over sized, which is only suggested to get more rugged mechanical properties.
- You should always reference the MFG's specifications for the wire you are using, as different insulation's have different properties. Some wire is rated for 30A in 18AWG wire, but that is some really special wire which is rated for such conditions. See this general suggestion for general fusing.
 - 18AWG, no larger than a 15A fuse
 - 20AWG, no larger than a 10A fuse

- 22AWG, no larger than a 7A fuse
- 24AWG, no larger than a 3A fuse Those fuse sizes are based on this table (http://www.powerstream.com/Wire_Size.htm) for chassis wiring.
- When selecting a fuse, you can use the below guide to help. Generally you choose the fuse and wire based on what the load needs. If your load is 1A capable, you need wire and fusing that can provide at least 1A. However you don't want too large of a wire as it adds costs, or the wire size might be too large for certain connectors, etc.

Littelfuse Fuseology (http://www.littelfuse.com/~media/automotive/catalogs/littelfuse_fuseology.pdf)

- Take note there is a fatigue issue known as I²t (That's amps squared time) which is a common reason for failures after several cycles.
- If you want to properly select a fuse, you really need a scope with a current measurement probe, then choose the I²t with the suggested method in the PDF. If you do not have such a scope, make sure to have extra fuses on hand just in case it blows a fuse some time down the road.

Littelfuse Mini Data sheet (http://www.littelfuse.com/~media/automotive/datasheets/fuses/passenger-car-and-commercial-vehicle/blade-fuses/littelfuse_mini_datasheet.pdf)

- Check in the forums for additional information. Often members can identify bumps in the road before you hit them. Forum members can also make suggestions that could make things go smoothly.
- If you do something cool, or new, feel free to share in the forums, Discord, Github, e-mail or where ever. It's common that someone doing something new and cool will get more help than someone doing the same old thing.

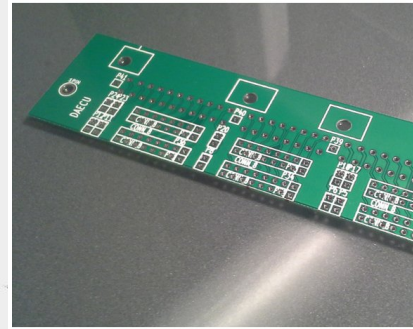
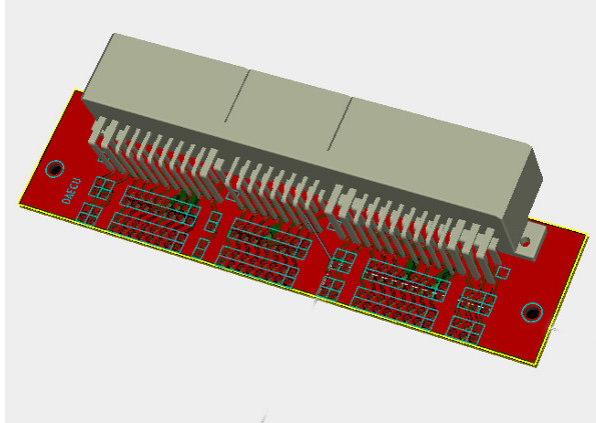
connecting ECU if it's not plug-and-play

- Start by getting junk yard ECU
- Delicately break apart the junk yard ECU salvaging the ECU connector and perhaps the



enclosure.

- If you have a connector that has a break out PCB board for it, you probably want to obtain the break out PCB for that harness connector. If not you can simply solder wires direct to the connector. Similar to this - Forum (<http://rusefi.com/forum/>)



viewtopic.php?f=4&t=507)

- It will likely be helpful to get a partial harness from a junk yard, and switch to rusEFI circuit-by-circuit.
- Use junk yard harness to make extension harness. Use the junk yard ECU connector to break out the wire harness to the breakout board. Then from the far side of the breakout board, install the junk yard harness connector. Then connect the original ECU. This should allow you to start and run then engine as normal.

Prepare engine wiring

- Connect TPS, MAP, IAT, and other such analog signals to the IO board.
- Connect 12V system / battery to the IO board connector.
- Calibrate the sensors using TS and Java console as required.
- Crank engine and see RPM's are registering correctly on the Java Console.
- Connect LED to pins TODO, which will blink at TDC. Verify that TDC is correct by shining on a crank wheel like a timing light. PS: actually right now we do not have this - see <https://github.com/rusefi/rusefi/issues/297> and <https://github.com/rusefi/rusefi/issues/2732> and <https://github.com/rusefi/rusefi/issues/3120> and <https://github.com/rusefi/rusefi/issues/3120>
- Connect injectors and ignition as required and see if it will start.

Testing the wiring Once connected you should test the wires. Especially the power wires like coil wires and injector wires. A poor connection with a slight resistance like .1 ohms can cause an electrical fire, which I'm sure you do not want. Once everything is connected measure both the voltage drop and current from the ECU connector, or where ever is applicably appropriate. Using your voltage and current readings, calculate the ohms, if it's above about .1 ohms fix the issue. Take note that .1 ohms at 1A is about .1watt that that connection will have to dissipate. If you have a 12 cyl, and 12 .1 ohm connections, the connector will have to dissipate 1.2 watts.

HOWTO start you engine with rusEFI for the first time

test outputs Both rusEFI console and TS allows you to test if rusEFI properly controls things like injectors (you would hear the clicks), cooling fan (you would hear it - if needed), fuel pump (you would usually hear it - if needed), ignition coil (that's challenging if you have a distributor)

get tachometer showing correct cranking rpm

Your tuning software should show correct cranking RPM, usually between 150 and 300 with a fully-charged battery.

See also Trigger (<https://wiki.rusefi.com/Trigger>)

See also Trigger Hardware (<https://wiki.rusefi.com/Trigger-Hardware>)

Confirm TDC position

Assuming you have the hardware ready to spark we now need to find your TDC position - we know trigger shape but we do not know the trigger wheel position in relation to TDC#1 (Top Dead Center, cylinder #1).

Set cranking advance angle to zero for now. Use a timing gun while cranking. We now need to try different values of Engine->Trigger->global trigger angle offset until we get spark at zero advance - that's because we might know the relation between TDC#1 and trigger signal.

On Engine Sniffer tab of rusEFI console TDC#1 is shown with the green vertical line.

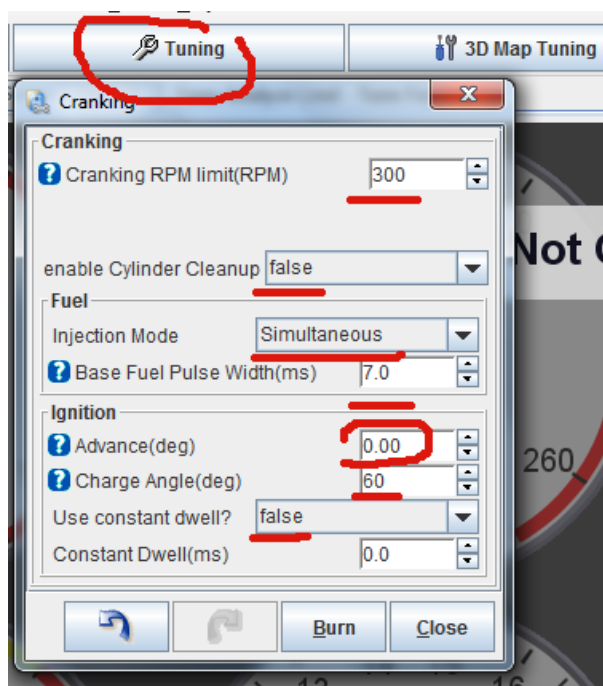


Figure 6: Initial Cranking Parameters

cranking parameters

rusEFI has separate cranking control strategy for your first couple of engine revolutions - usually you want more fuel, different timing and simultaneous injection to start an engine.

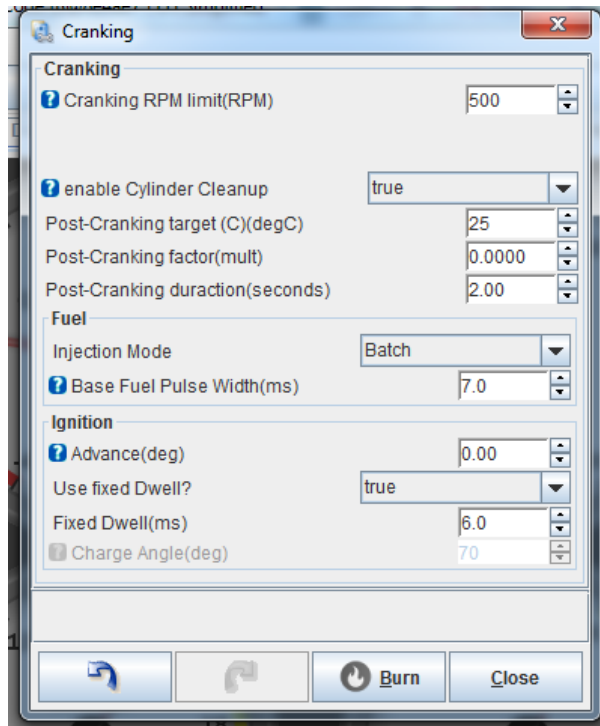


Figure 7: Cranking Dialog

Engine would start rich, as long as it's not too rich, as long as you have close-enough cranking timing angle. By default, cranking mode is active if RPM is below 500 RPM.

Please note that trigger synchronization point often does not match TDC, so just try different values between 0 and 720. For example, try 0, then 20, then 40 etc. Use `showconfig` to see current setting.

running parameters

For first run I suggest running based on MAF sensor - even if you do not have MAF sensor, and flat maps.

Once plain MAF works next step is running with proper MAP sensor calibration & flow rate setting.

next steps & troubleshooting

There are three ways to produce similar logs - the intention is for these three to have same exact data.

1. SD card logging
2. ruseFI console logging
3. TunerStudio logging

See also https://github.com/rusefi/rusefi/blob/master/firmware/console/binary/output_channels.txt
(https://github.com/rusefi/rusefi/blob/master/firmware/console/binary/output_channels.txt)

See also Error Codes (<https://wiki.rusefi.com/Error-Codes>)

External links

Fuel injectors at first start - Video (https://www.youtube.com/watch?v=lgvt0mh_UB8)

Diagnostics and trouble shooting of your engine

Basic tests List basic tests here, like is LED on, are jumpers installed correctly if applicable, find hot components and do basic visual checks for burn things and such.

Test equipment tests List tests that can be done with O-Scopes, multimeters, scan tools, and other such options for diagnosing a problem.

Get help from a local We provide much more info than most OEM options. If you are stuck, you may be able to get help from a local mechanic or someone local. Try asking for help in the forums there may be a member or a club meeting that's near by. It's common you can find local people who are willing to help.

On board hardware diagnostics Don't have a scope, no problem, the IO board has basic scope built inside. You can connect pin blah to nearly any point on the board and you can measure a variety of points synchronized with the logging software.

Get Running With a Plug & Play ECU

Step 1 (If Possible)

Run your car on your stock ECU to warm it up. This may make it easier to start, especially when it's cold outside.

Step 2

Unplug your stock ECU and plug in your rusEFI ECU. It's best to leave the ECU's case open for now, so you can observe blinking lights etc.

Step 3

Turn on the ignition, but **do NOT start the engine!**

Step 4

Connect TunerStudio to your ECU. See the TunerStudio guide (See Creating a Project on page 5) if you haven't yet created a project in TunerStudio.

Assuming the prerequisites for sensors (<https://wiki.rusefi.com/HOWTO-Start-An-Engine#sensor-requirements>) are met, your TunerStudio screen should come to life!

You should see sensor inputs and some output values like ignition timing and dwell.

If things are set up properly, you should see reasonable values for:

- Coolant temperature (CLT)
- Intake Air temperature (IAT)
- Battery Voltage (BV)
- Manifold Air Pressure (MAP/MAF) (should be near 100kpa if you're using a MAP sensor)
- Throttle Position Sensor (TPS)

The values may show some jitters. RPMs should be 0. Push your throttle pedal and see if you get a response. Congratulations—your rusEFI ECU can “see” your car.

Step 5

Calibrate your throttle pedal.

1. In TunerStudio, go to “Tools” -> “Calibrate TPS”.
2. Without touching the throttle, click on “Get Current” next to “Closed Throttle ADC count”.
3. With the throttle wide open, click on “Get Current” next to “Full Throttle ADC count”.

Step 6

Save (Burn) calibration to ECU.

Step 7

Test your outputs. Under “Controller” -> “Bench Test & Commands” you can see a lot of options for outputs you can test. Let's focus on the basic ones for now. Spark and Fuel. First, if your car's fuel pump is controlled by the ECU, test “Fuel Pump”. You should hear the fuel pump running.

Then move on to spark. Here it depends on how your car is set up. If you have a 4-cylinder with wasted spark (which is likely the majority of users), you will likely be using “Spark #1” and “Spark #3”. When you click the test button, you should hear the corresponding ignition coil firing. If you can't hear the spark, you may want to pull a spark plug (or use a spare) and hook it up to the spark plug connector and ground it. Test again and you should see a spark arcing. Don't shock yourself.

Now it's time to test your injectors. Again, this depends on your vehicle. Most cars will at least have two banks of injectors. So you should definitely hear injectors firing when you're testing "Injector #1" and "Injector #2". If you have full-sequential fuel, you should try as many injectors as your engine has cylinders. **Fuel is flammable. Have a fire extinguisher near by just in case.**

That's the basic outputs tested. With this you should be able to get the engine to run. You can go ahead and test the other outputs as well. Like Check Engine Light. Engine-Fan, AC, AC-Fan etc.

Close the test window. We're getting close to starting the car.

Step 8

Turn the ignition off and unplug the USB cable to remove all power sources from the unit, just to make sure we're in a defined state. Then reconnect the USB cable, turn on the ignition, wait for TunerStudio to come to life, and then crank the engine. You should see the RPMs jump to a positive value (typically anywhere between 200 and 600 RPM) and your engine should start. It may take a few cranks for RPMs to register. If you see RPMs in Tunerstudio, but the engine isn't starting, keep at it. It may take quite a bit longer than the stock ECU. rusEFI takes about two seconds of cranking before it can identify the position of the crankshaft, then it will fire the injectors and ignition coils. It may take more than ten seconds for the engine to catch, which seems like an eternity when cranking, but don't give up. There may be coughs and sputters. Play with the throttle a little, and it will start eventually. Post your findings to the forum (<https://rusefi.com/forum/>) and we can help!

We're done with the initial startup. Congratulations—you have your car running on your standalone aftermarket ECU! Now the fun really begins! 😊