Speeduino v0.4.3

Compatible with M50 PnP ecu

[1. Getting started 2](#_Toc27469815)

[1. Installing to the car 2](#_Toc27469816)

[2. Connecting to PC 2](#_Toc27469817)

[1. USB-connection 2](#_Toc27469818)

[2. Bluetooth connection 3](#_Toc27469819)

[3. USB-Cable 4](#_Toc27469820)

[4. Connecting to TunerStudio 5](#_Toc27469821)

[2. Configuring Speeduino before the first start 8](#_Toc27469822)

[1. Required Fuel 8](#_Toc27469823)

[2. Injector parameters 9](#_Toc27469824)

[3. TPS Calibration: 10](#_Toc27469825)

[4. Wideband calibration 11](#_Toc27469826)

[5. Starting the engine 12](#_Toc27469827)

[3. Tuning the engine 13](#_Toc27469828)

[1. Idle control 13](#_Toc27469829)

[2. Vanos 13](#_Toc27469830)

[4. Connections 15](#_Toc27469831)

[3. Boost solenoid and relay control 15](#_Toc27469832)

[4. Launch control 15](#_Toc27469833)

[5. Firmware updates 16](#_Toc27469834)

# Getting started

## Installing to the car

When installing the ecu to the car, it is enough to plug it in to engine's wiring harness and run the MAP hose from intake to the MAP sensor connector at the back of the ecu. In addition, for tuning the ecu, run the 0-5v signal from wideband lambda controller to the external rear connector. See Section 4

## Connecting to PC

You can connect Speeduino to your computer / tablet / cellphone using USB or Bluetooth. Unfortunately, these cannot be used at the same time, and if you want to use USB instead of Bluetooth, then the Bluetooth pairing needs to be removed. Of course, the Bluetooth connection works same way as the USB connection. The only difference is that if the firmware wants to upgrade it must be done with a USB cable.

### USB-connection

When connecting a USB cable to your computer, you need to install the appropriate drivers for your computer. If your Arduino has ATMega16U2 windows should install drivers automatically. But if it doesn’t, you need to install Arduino IDE and the drivers included in it:

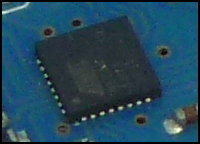
<https://www.arduino.cc/en/main/software>

In other hand if the Arduino Mega has the CH340 chip for USB, the windows drivers for that may cause connection failure, so it is recommended that you install the manufacturer's own drivers before connecting the USB hub. These can be found at: <http://www.wch.cn/downloads/CH341SER_ZIP.html>

The page and the installer are partly in Chinese, but the installation should still be easy. Use a web page, for example, with a Chrome browser and put google translate to translate it into English.

If you don’t know which one your Arduino Mega has, here’s how you can tell which one you have by looking your Arduino:

ATMega16U2 or 8U2:



CH340:



After connecting the Speeduino to the computer with the correct drivers installed, it should show up in Device manager under Ports:



With CH340 it will be named as “USB Serial CH340”. With 16u2 the name will be “Arduino Mega 2560” or “Speeduino Mega 2560” If so, all ok and keep the COM port number in mind.

### Bluetooth connection

When connecting with Bluetooth, Speeduino must be in the car and the currents on. Then, with a computer / tablet / mobile phone, Bluetooth search for new devices. After a moment of search, it should find a Bluetooth device called SpeeduinoM50 (or what name you did gave for Bluetooth module if you did configure it by yourself). Connect to it.



After that, Bluetooth should ask for the PIN code and if not ask for the PIN code for the use of the Bluetooth device. The PIN code can be found on the top of the box (if it has a Bluetooth module inside it) or the PIN number you did configure Bluetooth with, the Speeduino should be connected without problems. On the computer, then, the Device Port under COM ports should have a Bluetooth Serial COM port. Remember this COM port number.

### USB-Cable

USB2.0 compatible USB-A to USB-B cable is enough as a USB cable that can be used with Speeduino. Of course, it’s a good idea to buy relatively high quality in order to avoid possible interference caused by the USB cable. In addition, the maximum cable length is 5 meters, but it’s recommended use shorter cables like 3 meters or less. Also, depending on which car model the box is to be installed on, it may be necessary to purchase a cable with a USB-B connector with a 90-degree angle for space reasons. This helps installation if the USB connector is in tricky place to fit cable with straight connector.



### Connecting to TunerStudio

Tuning Speeduino happens using TunerStudio. If you don’t already have TS, it can be downloaded here: <http://www.tunerstudio.com/index.php/tuner-studio>

After opening TS, click: Create New Project



Give the project a name and click Detect. The Tuner Studio should automatically detect Speeduino and download configuration from the server. (Only newer TS versions than 3.0.28)



This automatic downloading of the .ini -file works only if your PC has internet connection. If the Internet connection is not available or the detection otherwise fails, the configuration must be added manually. The required files can be found in the Speeduinon firmware packages. The zip files for each Firmware release can be found from here: <https://github.com/noisymime/speeduino/releases>

Choose the zip package that corresponds your Firmware version and unpack it.

After that select: Other / Browse in TS where it says Firmware.



The downloaded and unpacked Firmware package contains a speeduino.ini -file under reference folder. Select it and click next. 

From the menu that opens, choose Temperature Displays as Celsius unless you want to see the temps in Fahrenheit. In this menu, you can also choose Fueling Algorithm as Alpha-N, but Speed ​​Density is usually used. From Enablehardware\_test, you can unblock the injector / ignition output testing, but you probably wont not need it. Of course, if that is needed, you can also enable it later.



Click next.

For Com Port, select the port you saw in Device manager:



Note! The “Test Port” button works only on TS versions newer than 3.0.28

Click next and then to Finish to make TunerStudio connect to speeduino:



Check that the MAP sensor displays about 100kpa and the temperature sensors are sensible values ​​(with a cold engine both about the same ambient temperature). If you have built the ecu by yourself, you need to calibrate the temperature sensors first.

# Configuring Speeduino before the first start

## Required Fuel

Before the engine can be started for first time, the Speeduino must be configured for used setup in TunerStudio. Click on “Settings” and “Engine Constants”.



Everything here should be ok ready from Base Tune, but then click Required Fuel:



Engine Displacement = Set correct engine displacement. Remember to select correct unit.

Injector Flow = M50 B20 / B25 greens flow about 190cc/min. If some else injectors are used, you will need to use the provided injector data for setting flow rate. Note that the basic flow rate is often reported at 3bar base pressure, but m52 uses a 3.5bar base pressure.

Air-Fuel Ratio. = Stoichiometric mixture ratio. 14.7 for gas and 9.8 for e85.

Once set, click ok and TunerStudio will calculate ReqFuel value.

## Injector parameters

To set injector parameters go: Settings -> Injector Characteristics



Injector Open Time is the same as dead time, latency, etc. This means you need to enter the dead time of the injectors and voltage correction. The values ​​for B20 / B25 greens are:

Open Time 0,6 ms

Voltage correction:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Voltage | 6 | 8 | 10 | 12 | 14 | 16 |
| % | 255 | 223 | 138 | 90 | 58 | 38 |

For other injectors, refer the injector datasheet. Note that you want to use the “Open Time Only” Voltage correction mode. Whole PW mode is just for legacy tunes.

## TPS Calibration:

To calibrate the throttle position sensor, go to Tools -> Calibrate TPS:



When you do not touch the gas pedal at all, press Get Current under Closed throttle. Then press the throttle down and Get Current at Full Throttle. Then, the Accept and TPS values ​​should now reasonably move between 0 and 100%.

## Wideband calibration

For wideband lambda calibration, you need to go to Tools -> Calibrate AFR sensor.



From the EGO sensor list, select the broadband controller you are using and click Write to Controller. If your wideband controller is not on the list, you should use either of the custom selections and manually enter the values ​​according to the manufacturer's specification.

Once the values ​​have been written, the AFR gauge can be displayed by clicking the right mouse over a unused gauge and searching for the “Air: Fuel Ratio” in sensor inputs menu.



The gauge should now look like the display of the broadband controller itself (if any). If not, check the connections. The engine then ready to be started.

## Starting the engine

Once all the settings in the previous sections have been made, the engine should run. It should start at first crank, but if not, first try adjusting the bottom of the VE map up or down:



Once the engine is started, you can start tuning it. NOTE! Basic maps are just for the start. Before the car can be driver, Speeduino maps must be tuned to fit the engine.

# Tuning the engine

The tuning of the engine happens same way as with any other Aftermarket ecu. If you can tune Megasquirt with TunerStudio, you can also tune Speeduino. There is still A few noteworthy differences, like: Speeduino will not turn off when car ignition is off if the USB cord is attached to the Speeduino. Also, the Speeduino does not control anything other than the priming pulses or the fuel pump and injectors before the machine runs. Thus, the boost solenoid etc. cannot be tested if the engine is not running.

## Idle control

By default, idle control works in the Open loop mode, which can be changed to the Closed loop when the engine is otherwise tuned. In Open loop mode, idle rpms are adjusted from the “Idle - PWM Duty Cycle” -map. Cranking Duty is the idle valve map that is used during cranking, so if there is starting problems, tuning this map may help.



## Vanos

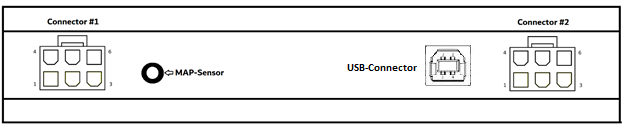
In case you have m50TU, the vanos tuning can be done from “VVT duty” map from Accessories menu:



0% duty means that VANOS is off in the basic position and 100% duty means that it is on. Speeduino also allows PWM control, but the m50 vanos doesn’t work properly with open loop PWM control so use only 0% and 100% duty values.

# Connections

In addition to the original 88-pin connector the ecu has USB connector on the side of the box which is for the PC connectivity. Then at the behind, there is two 6-pin molex minifit connectors and spot for MAP hose. Connector #1 is meant for wideband signal input, flexfuel sensor signal and other engine related stuff. Connector #2 is meant for connecting speeduino additions like SD-logger.



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Connector #1 |  |  | Connector #2 |
|  |  |  |  |  |
| **Pinni** | **Toiminto** |  | **Pinni** | **Toiminto** |
| 1 | Flex fuel sensor signal |  | 1 | GND |
| 2 | Wideband lambda 0-5v signal |  | 2 | Serial3 RX |
| 3 | Launch input |  | 3 | Serial3 TX |
| 4 | Spare relay output D37 |  | 4 | +5 Volts |
| 5 | Fan relay control |  | 5 | Spare relay output D31 |
| 6 | Boost solenoid output |  | 6 | Spare relay output D33 |

## Boost solenoid and relay control

M52 speeduino is capable of driving boost solenoid and relays like for fan control, without modifications. Both boost solenoid output and relay outputs work by grounding. Which means that the other pin of the boost solenoid or relays will be connected to switched 12v. Preferably through small fuse (For example. 3-5 A). The red wires with white stripe are handy place to get switched 12v in the m52 wiring harness. Connecting to constant 12v is not recommended, because it can cause weird behavior. Like the ecu not turning off when switching ignition off.

## Launch control

Launch input is connected directly to the processor and there is no protection in it. So be careful with the launch input. It’s recommended to run the launch input to button (For example button in clutch pedal) which grounds the input when launch is activated. With this kind of connection the pull-up resistor needs to be enabled in TS (Named as “Clutch input”).

Example settings:



# Firmware updates

Speeduino is a continuously developing engine control unit, with new firmware updates released every few months, which include bug fixes and new features. You should keep track of what the new versions bring with you and upgrade your firm to the newer if the new features are available.

The most up-to-date instructions for updating firmware can be found at: <https://speeduino.com/wiki/index.php/Compiling_and_Installing_Firmware#Downloading_the_firmware>

When writing these instructions, best way to load in new firmware is speedyloader, which can be found from the link above.



Using the speedyloader is easy. Choose the FW version you want to use (normally the latest one) and after that “Choose Port” and select the COM port you saw in device manager previously. Then click “upload” and speedyloader will automatically upload the FW version. It also adds corresponding TunerStusio .ini -file to downloads folder. If you are working with new ecu without any previous firmware, there is M50 basetunes available under “Base tunes”

NOTE! Before upgrading, take Speeduino out of the car, because if it is plugged in during the upgrade, the car may be horrible, such as coils may burn in the worst case. And after updating, review all the settings, maps, and calibrations that they are like before the upgrade.