

# Injection self learning Challenger5

Release: V1.20 – 12/08/2021

<http://www.skynam.com>



Machine management

## Injection self learning Challenger5

Skynam reserves the right to make changes, corrections, modifications, improvements, to this document, to products and to the services which he describes, at any time and without notice preliminary.

No part of the documents may be reproduced or transferred, whatever the reason or the means used, whether mechanical or electronic, without prior authorisation from Skynam.

Skynam's general sales conditions are fully applicable.

WINDOWS is a Microsoft Corporation registered trademark

The WINDOWS logo is a™ Microsoft Corporation trade mark.

## OVERVIEWS

### **I) WINJALL SOFTWARE FOR ECUs TUNING:**

The Challenger5 ECU tuning pack is supplied with accompanying of the program Winjall. It is subjected to the same rules of use and license.

The Winjall program requires a license of use. In its complete shape, this license is established by a file license and a material dongle key

To make access to Challenger5 ECUs easier, this pack contains a licence made of one single file, without a dongle.

This licence gives you access to the main tuning functions for Challenger5 ECUs.

However, to have access to the complete features, it is necessary to obtain with the Company Skynam a complete license.

Though it is possible to make copies of the Winjall software distributed in this pack, you have to comply with the regulations for use and the licence you have accepted when installing it on your computer.

### **II) TUNING PACK DOCUMENTATION:**

This documentation version has been specifically upgraded for the Tuneware Challenger5 versions distributed from August 10<sup>th</sup>, 2021, notably:

- throttle / rpm: version 2.40
- pressure / rpm: version 2.40

These Tuneware versions only work with the Winjall V9.70 version or a later one, which also requires a XP SP3, 7, 8, 10 or any later version of the Windows operating system of the Microsoft Corporation.

### **III) WARNING:**

The CHALLENGER5 ECUs are meant for racing cars, boats or bikes and not for any other purpose. Consequently, the softwares to be found in this pack are for racing use only and are not meant to match the emissions standards required for standard or road cars.

Remember it mustn't be used on open roads, as it does not comply with road regulations.

## PRINCIPLES

The self learning of the injection time base map allows a speed up of the engine tuning.

When the ECU is provided with a Lambda sensor, it is able to read the richness of the engine. By this means, it can

- dynamically make corrections of the injected fuel quantity to maintain the richness to a given target (to see file 'Use of Lambda sensor').
- make a self learning of the injection time base map according to the given map of richness target.

It is strongly advised to use **Wideband** Lambda sensors with a module interface of specific electronic conversion of the sensor. These modules are able of measuring a very important range of richness quickly and precisely.

Skynam recommends the **modules DigiLSU or LSU49-5V** which were tested for a long time and have given the proof of their high precision and quality of manufacture. They must be used with Lambda Bosch LSU 4.9 sensors.

However, it is possible to use also 0-1 volt 4 wire Lambda sensors, if it is in perfect state, if possible new.

### **I) AUTO NOTATION OF THE LEARNING QUALITY:**

To make its learning of injection time, the ECU owns a map added to the cartography of injection time, with the same number of lines and columns, in which it notes the quality of every point of the injection time map with a value going from 0 % to 100 %.

This map is called quality map of self learning.

During the self learning, every time the ECU is going to write a value in the injection time map, it also writes the estimated quality of this point.

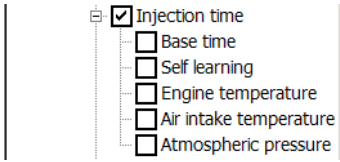
**We can consider that the self learning is correct when the majority of the points of the quality map indicate a value of at least 96 %.**

We can then stop the self learning, and verify if need that the injection times given by the injection time map are correct.

Injection time learning quality		Injection time self learning quality from 0 to 100%										
LINE SCALE		Engine rpm (rpm)					0 Standard interpolation					
COLUMN SCALE		Intake pressure (bars)					1.007 Standard interpolation					
	0.295	0.390	0.480	0.565	0.645	0.720	0.790	0.855	0.915	0.970	1.020	
600	92.655	98.229	92.655	82.623	73.529	66.012	59.770	64.394	68.805	72.207	73.529	
900	88.554	92.655	88.554	80.332	72.207	65.182	62.928	68.805	74.973	80.332	82.623	
1300	92.655	98.229	92.655	82.623	73.529	66.012	65.182	72.207	80.332	88.554	92.655	
1700	92.655	98.229	92.655	82.623	73.529	66.012	66.012	73.529	82.623	92.655	98.229	
2200	92.655	98.229	92.655	82.623	73.529	66.012	65.800	72.207	80.332	88.554	92.655	
2700	88.554	92.655	88.554	80.332	72.207	73.529	73.188	71.894	74.973	80.332	82.623	
3200	92.655	98.229	92.655	82.623	80.332	82.623	82.018	79.808	74.599	72.207	73.529	
3800	88.554	92.655	92.655	88.554	88.554	92.655	91.524	87.678	79.808	71.894	66.012	
4400	92.655	98.229	98.229	92.655	92.655	98.229	96.638	91.524	82.018	73.188	73.529	
5000	88.554	92.655	92.655	88.554	88.554	92.655	91.524	87.678	79.808	80.332	82.623	
5600	88.554	85.309	82.623	82.018	88.554	92.655	88.554	85.309	85.309	88.554	92.655	
6300	92.655	88.554	80.332	84.593	92.655	98.229	92.655	82.623	88.554	92.655	98.229	
7000	88.554	85.309	82.623	82.018	88.554	92.655	88.554	85.309	85.309	88.554	92.655	

Here is an example of quality cartography while the learning is in progress.

You access to the learning quality map in the Injection time pages once the Lambda sensor is activated (see lower).



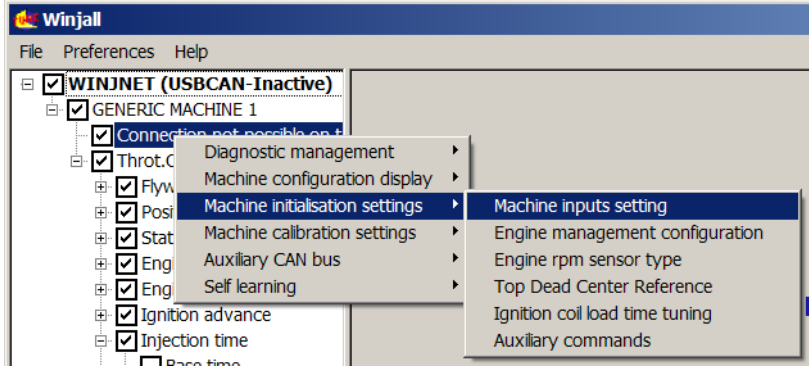
**II) RUN LEARNING:**

The self learning is made only when the PC is in communication with the ECU. If you stop Winjall, the filling of the injection time map by the learning self is stopped too.

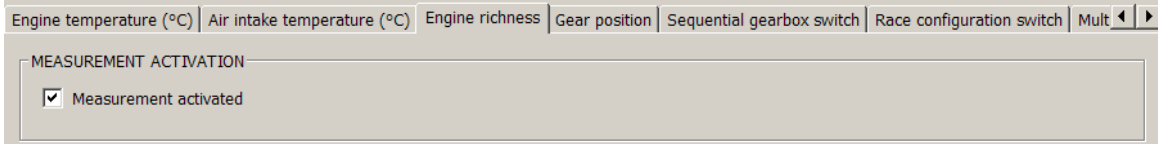
**III) ACTIVATION OF THE MEASUREMENT OF RICHNESS:**

As this self learning uses Lambda sensors, it is necessary having installed the sensors on the engine, to activate the sensor measurement.

This activation is made by means of the function of configuration of inputs, described in detail in the file 'Inputs configuration'.



Activate then the measure of richness (see file 'Use of Lambda sensor').



Once activated the Lambda sensor present on the engine, the functions of self learning become accessible.

## PRELIMINARY TUNINGS IN THE SELF LEARNING

The self learning of the injection time requires a preliminary preparation of the tunings of the ECU. Generally speaking, all the functions of the ECU have to be pre adjusted so that the engine can work correctly during the self learning.

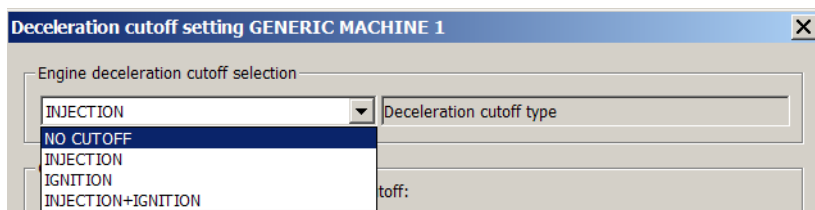
Think that the self learning which the ECU is going to make needs a correct engine working as if you made the tuning of the injection map yourself.

Here is what it is advisable to look particularly:

### I) DECELERATION CUTOFF:

During the self learning, it is better to deactivate the deceleration cutoff. It allows the ECU to learn the injection times in the smallest loads even to average and high rpm, what will allow you more effective reactions of the engine during re accelerations.

However, if you execute the self learning on private track by driving the vehicle, keep in mind that the engine brake is considerably decreased.



### II) RPM LIMITER:

Set the rpm limiter slightly higher than that it will have to be normally, so that the ECU can measure and calculate the good injection times up to the highest rpm.

Deactivate the launch limiter.

### III) LIMITS OF RICHNESS CORRECTION:

The self learning does not use the software switch of richness correction permission, because the self learning is not a correction of richness but a calculation of the injection times to be put in the basic map of the injection times.

Richness correction permission	Value	State
Richness injection correction permission	NO	Allows injection time correction by Lambda sensor

It does not use no more the richness correction limits

Max richness correction	Value	State
Richness max lean correction	0.950012	Sets the richness injection time correction band and limits problem
Richness max rich correction	1.150024	Sets the richness injection time correction band and limits problem

On the other hand, it takes into account limits imposed by the motorist on the richness looping (see file 'Use of the Lambda sensor'):

- the minimum load,
- the minimum rpm,
- the minimum engine temperature

below which the richness looping must not be made.

Min throttle for richness correction	0	Throttle position below which no injection time correction by Lambda sensor is allowed
Min rpm richness correction	0	RPM below which no injection time correction by Lambda sensor is allowed
Min engine temperature richness correction (°C)	+80	Engine temperature below which no injection time correction by Lambda sensor is allowed

#### **IV) PREPARATION OF THE RICHNESS TARGET MAP:**

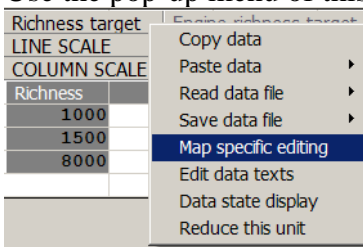
It is this map which pilots the learning of the I.T.

Fill this map carefully according to the richness that you want to get for your engine. For example (in richness display):

Richness target	Engine richness target (richness in thousandths or Lambda in units)			
LINE SCALE	Engine rpm (rpm)	0 Standard interpolation		
COLUMN SCALE	Intake pressure (bars)	1.007 Standard interpolation		
Richness	0.500	1.000	3.000	
1000	1000	1100	1150	
1500	1100	1200	1200	
8000	1150	1200	1250	

This map does not need to be big, but it is essential to prepare it with attention.

Use the pop-up menu of this map to modify the number of lines and columns.



#### **V) PREPARATION OF THE INJECTION TIME MAP:**

You have to give at first the shape wished to the injection time map, that is create the number of lines and columns wished with the points of rpm and load desired in scales.

If you create new lines or columns, use the interpolate function to fill these lines or columns not to leave with null I.T. in the map (see user's manual of Winjall).

For turbo engines, correctly set the column of maxi pressure up to the one the engine is going to work. In throttle/rpm the column of maxi throttle opening is set as usual towards 950. For the line of maxi rpm, set the rpm some ten rounds below the rpm limiter.

#### **INJECTION TIME PREPARATION**

**Do not forget that a wideband Lambda sensor cannot measure more than 1550 thousandth of richness** (0.645 Lambda), that is if the injection time map with which you start has I.T. more than 1.5 times than what requires the engine (in touch with the injectors flow), the sensor is going to saturate and certainly to indicate an error.

**A 0-1 volt Lambda sensor cannot meanwhile measure more than 1300 thousandth of richness** (0.77 Lambda), that is if the injection time map with which you start has I.T. more than 1.3 times than what requires the engine (in touch with the injectors flow), the sensor is going to saturate and specify a value that can be aberrant.

The self-learning on these extremely rich points will not be possible. You will track down easily these points because the engine is going to smoke "like a truck".

#### **SPECIFIC PREPARATION FOR TURBO ENGINES WITH BANG-BANG**

The ECU does not make the bang-bang during all the phase of the self-learning: in the maps for which you inserted a column 0 which gives the working in bang-bang, this column 0 is simply not used (see the file 'Atmo Pressure-rpm turbo and bang-bang' for details).

You do not thus need to modify your bang-bang tunings, except for the injection time map (and for it only), on which you have to remove the column of load 0 to make the self learning.

#### **VI) PREPARATION OF THE IGNITION ADVANCE MAP:**

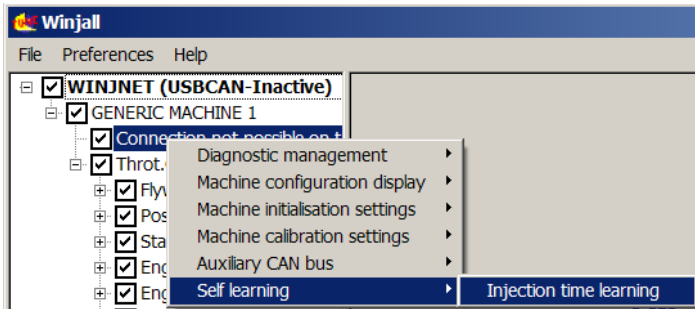
Remind you that for the tuning of the ignition advance map, it is necessary to leave an advance under estimated and to raise back then the advance if the engine allows it, to obtain the desired torque.

Before launching the learning of I.T., do not hesitate has to decrease the advance of several degrees, you will put them back later when you will have a correct richness.

## LAUNCHING THE SELF LEARNING

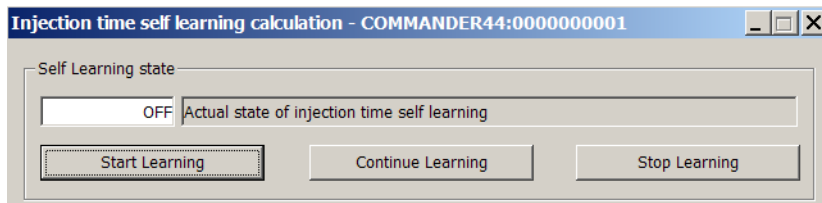
Once activated the Lambda sensor equipping the engine, the functions of self learning become accessible.

Functions of start up of the self learning is generated from popup menu obtained by a right click on the system state of the machine (System menu) – the 1st line below the name of the machine.



### **I) START LEARNING:**

A dialogue box appears which allows to choose the way of launching the learning:



#### **START LEARNING**

This launches the learning from the beginning. Any previous learning is cleared and the learning quality map is set back to 0 %.

#### **CONTINUE LEARNING**

This continues the learning at the stage where it was previously stopped.

If the quality map has no same shape as the injection time map (the same number of line and columns), this choice is not accessible.

It happens for example when we added or remove lines and columns to the injection time or quality map since the last learning.

#### **STOP LEARNING**

This stops the learning if it is in progress and clears the dialog box.

To click "X" at the top right produces the same effect as the button of Stop learning: the learning is stopped and the dialog box disappears.

#### **EXECUTION OF THE LEARNING**

Once launched the learning, it is possible to minimize this dialog box to icon (in the taskbar of Windows) by clicking on the "-" at the right top. It allows to free the screen without stopping the learning.



## **ID RUN LEARNING:**

The self learning is made only when the PC is in communication with the ECU. If you stop Winjall, the filling of the injection time map by the self learning is stopped too.

## SELF LEARNING USE METHOD

### I) RUN OF THE SELF LEARNING:

When you launch the self learning, the ECU waits that the conditions of engine working are favorable enough to fill its map of I.T.:

- 1) If the measures of throttle position or of Lambda sensor are in error, the learning is not made.
- 2) For pressure/rpm engines, the injection time self-learning takes into account the intake pressure movement and in areas where intake instabilities are significant, the learning cannot be performed. So intake pressure instabilities prevent the self-learning. Check the setting of the pressure averaging map (see full explanation in document 'Pressure/rpm Atmo Turbo and Bang-bang').
- 3) As long as the procedure of engine start enrichment is in progress, the learning is not made.
- 4) As long as the lambda sensor is not warm, the learning is not made.
- 5) In the phases of unlooping of the Lambda sensor, the learning is not made. These phases are:
  - looping forbidden by the motorist
    - engine rpm lower than rpm limit of richness correction
    - position pedals lower than position limit of richness correction
    - engine temperature lower than temperature limit of richness correction
  - injection stopped on one or several cylinders (deceleration cutoff, rpm limiter)
  - ignition stopped on one or several cylinders (deceleration cutoff, rpm limiter)
  - accelerating pump in action
  - gearshift

### II) STATE OF THE SELF LEARNING:

The ECU displays on the page of the quality map of learning the state actual of the learning.

Self learning state	Value	State
State of injection self learning	<b>WAITING</b>	OFF self learning disabled, WAITING waiting to learn, ACTIVE measuring data, RECOR

The possible states are

- OFF: the function of self learning is not launched
- WAITING: the function of self learning is launched but the conditions of calculation are not gathered (for example, Lambda is unlooped)
- ACTIVE: the function of self learning is launched and the calculation is in progress
- RECORD: the function of self learning is launched, a calculation was made and its result is being recorded in the injection time and the learning quality maps.

In normal operation of learning, the state passes constantly from ACTIVE (currently measuring) to RECORD (currently recording).

If the state of the learning stays on WAITING, it is because a condition to make the measure is not present. See for example on the same page the state richness looping:

Richness correction fast help	Value	State
Lambda looping state	HEAT	HEATING-LOOPWAIT-LOOPING-ERROR. LOOPWAIT : when the sensor works but the
Richness target	1100	Engine richness target (richness in thousandths or Lambda in units)
Engine bank 1 richness	1414	Engine richness value (richness in thousandths or Lambda in units)
Bank 1 injection time richness correction	1.000000	Richness calculation coefficient on injection time

### **III) DRIVING OF THE SELF LEARNING:**

#### **LIMIT OF LAMBDA OPERATION**

Do not forget that a wideband Lambda sensor cannot measure more than 1550 thousandth of richness (0.645 Lambda) and that a 0-1 volt sensor cannot measure more than 1300 thousandth. The self learning on these extremely rich points will not be possible.

Report to lower chapter 'Self learning problems' to solve these problems.

#### **ADVICES FOR A GOOD USE**

To obtain a fast self learning, here are some advices of use of the engine.

- Display the page of display of the learning quality map.
- If you run the self learning by driving in the vehicle, on private track, use the highest possible gear position to reach the zone of the map which you want to make learn.
- It is often impossible to run the engine in certain zones of the map, as the strong loads with very low rpm. The self learning owns capacities of extrapolation (to fill the points where it cannot go) and can fill all the same the map in these zones with a good quality note.
- Remain stable with the throttle because abrupt movements prevent the learning.
- You can return several times on points already learnt to improve their quality note, but it is better to make learn several different zones and return there then than staying for a very long time on the same zone to try to obtain to it a very high quality at once.

- 1) Start the engine and make it warm to small rpm and at low load (on 1500-2000 rpm)
- 2) When the engine and Lambda are warm, to launch the self learning
- 3) In neutral gear, without load, accelerate the engine to reach the high rpm during some seconds, until the quality of the zone of the engine cursor (cells written in red) rises.  
Decrease a little the rpm by loosening the accelerator and begin again on the a little lower rpm zone.  
Make learn in this way all the small loads by coming down towards the tick over.
- 4) Make learn the tick over zone.
- 5) By loading a little more the engine (loaded in the test bench) or on tracks or on the roll dyno in the vehicle, make learn the medium loads by stages by beginning from the low rpm, until the high rpm.
- 6) By loading the engine, make learn maxi loads by stages by beginning from the low rpm, until the high rpm.
- 7) The learning can be considered as finished when the majority of the quality map reaches 96 %.

#### **NEUTRALIZATION OF LEARNING**

If you think that some areas of the injection time map are correct and should not be changed by learning, manually rise the quality of these areas to 100%.

For example the area of engine start:

Injection time learning quality		Injection time self learning quality from 0 to 100%							
LINE SCALE		Engine rpm (rpm)						0 Standard interpolation	
COLUMN SCALE		Throttle position angle						0 Standard interpolation	
	0	15	50	100	150	200	300	600	900
600	100.000	100.000	100.000	100.000	100.000	100.000	100.000	100.000	100.000
900	0.000	0.000	0.000	0.000	100.000	100.000	100.000	100.000	100.000
1300	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
1700	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
2200	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
2700	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
3200	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000

#### SETTING POINTS MANUALLY

During the learning, you can modify points of injection time or quality map manually, but if you modify points of I.T. and that the learning estimates that the quality of the point that it can give is better, it is going to rewrite on these points behind you.

If you want to fix values definitively to points of I.T., set to 100% the quality of the corresponding points in the quality map as indicated above in the neutralization of learning.

#### IV) AFTER THE SELF LEARNING:

Once ended the self learning, it will still remain for you

- 1) To make the tuning of the accelerating pumps, either by working with their coefficient, or by modifying directly their maps.
- 2) To verify the quality of the warm start, by possibly manually modifying the first line of low rpm of injection time map (600 rpm and lowers)
- 3) To work your ignition advance map.
- 4) Having let the engine cool down some hours, work the cold start by modifying coefficient of cold enrichment or directly the maps of enrichment of cold start, starter and rise in temperature.

#### V) SELF LEARNING PROBLEMS:

##### TOO RICH ZONES SATURATE THE LAMBDA

If your map of start is far too much rich on some points, you will obtain generally two reactions:

- the wideband Lambda sensor is going to pass in error or the 0-1 volt sensor will give aberrant values.

- the engine is going to have nonstop knocks and can not cross these points of map.

Examine then the zone concerned by the I.T. map, decrease the by hand the I.T., and decrease the corresponding zone of the quality map below 50 %.

##### ZONES WITH BAD RICHENESS

If when the self learning is ended and when you have adjusted the accelerating pumps, there are still zones in which the engine has not a good richness, it is mostly that you miss lines of rpm in the injection map.

It occurs generally on the strongly developed engines which have sudden rise of torque and/or on the engines which have resonances in the intake manifold.

So that these engines can be well tuned, it is needed an important number of lines at the level of these rough modifications of engine working, for example a line of regime every 250 rpm, possibly even more closer, so as to allow a big sharpness of regulation.

Note: it is absolutely useless, and even harmful, to position a big number of lines of rpm and columns of load everywhere in the map. It must be made only on the fast modifications of engine working.

To add lines of rpm (or in an equivalent way columns of load) without that the self learning forces you to redo all the learning:

- 1) Save your Tuneware to be able to return to the point of current tuning in case of error.
- 2) Create your lines and your columns in the injection time map by means of the function of map specific edition (see higher the preparation of the injection map). Make these lines of rpm or columns of load be filled by Winjall by using the function of interpolation of the specific edition (see user manual of Winjall).
- 3) Create the same lines of rpm and columns of load in the learning quality map, but leave the quality to 0 on these new lines and columns.
- 4) On lines and columns which surround the new lines and the columns created in the quality map, decrease widely the quality of learning, at least below 50 %.
- 5) Restart the learning by choosing the option "Continue learning".
- 6) Run on the engine on these zones of low quality.