**Journal of Electrical Engineering, Electronics, Control and Computer Science – JEEECCS, Volume 1, Issue 1, pages 15-20, 2015**

# The conception and execution of a simulation workbench for an automotive fuel gauge system

Chiper Ana-Maria
University of Piteș ti
Electronics, Communication and Computer Science Faculty, Applied Electronics Specialization
Piteș ti, Romania
Ana.M.Chiper@gmail.com

***Abstract*** **– The present article covers measurements and experiments regarding the display of fuel consumption on Dacia vehicles, as there have been errors in the industry, concerning the level bars update of the real quantity of fuel of various Dacia automobiles.**

***Keywords- fuel consumption; electric resistance; dashboard; Dacia vehicle***

## I. INTRODUCTION

Displaying the correct number of bars (i.e. the level of fuel in the tank) on the dashboard has been an important aspect for the Dacia automobile company, especially for Dacia Duster and Dacia Logan models.[1] However, there have been problems when the tank is being supplied with fuel.

Such a situation can be encountered when traveling a considerable distance after the fuel tank has been supplied, as it can be noticed a sudden change in the number of bars. For example, 2 or 3 bars can suddenly drop at the same time, which is inadmissible, because the person driving the vehicle has to permanently and correctly be informed about the level of fuel in the tank.

Subsequently, in the first phase of the project a complete system modeling and simulation had to be developed. There had to be found variables which determine a change in the dashboard display[2] and, also, a way to use them in such a manner that the defect which appears in practice can be reproduced.

This way, through simulation, the traveling of the real distance has been avoided.

## II. SYSTEM DESCRIPTION

The simulation workbench practically created refers to the simulation of the system used to indicate the level of fuel contained in the tank. This fuel gauge system is a constituent element of the fuel management system.

The main functions of the fuel management system are:

- Fuel storage;
- Fuel supply;
- Engine supply (flow, pressure);
- Indicating the level of fuel.[3]

A general schematic representation of this simulation workbench is illustrated in Figure 1.

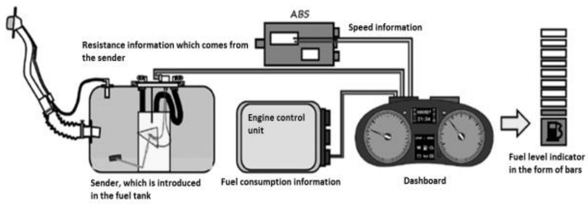


Figure 1 – The general schematic of the fuel gauge system

The level of fuel is indicated using the sender, which can be observed in Figure 2. This represents a device which works as a level sensor, with the main features being:

$$h=f(L) \quad (1)$$

$$R=f(h) \quad (2)$$

where *h* is the height of the sender's float, *L* is the amount of fuel, expressed in liters, and *R* is the information of resistance transferred from the sender to the dashboard.[4]

From relations (1) and (2) results that:

$$R=f(L) \quad (3)$$

The relation (3) indicates a dependency between the resistance provided by the sender and the amount of fuel contained in the vehicle's tank.

The sender is introduced in the tank and its role is to obtain the information which would be of use in the calculus of the level indicator on the dashboard, which is a part of the fuel system.[3]

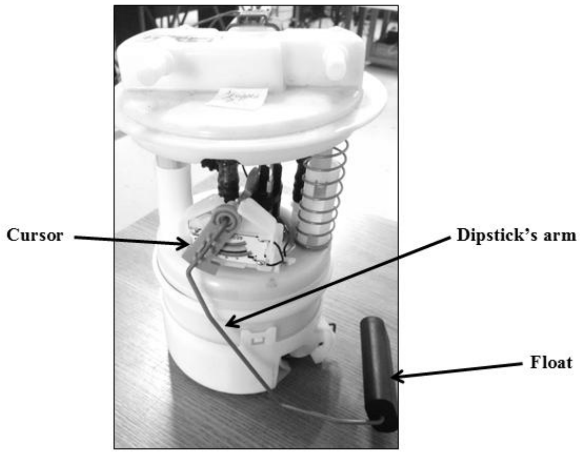


Figure 2 – The sender placed in the tank of a Dacia Logan vehicle

When the level of fuel is shown on the dashboard's display, there are two aspects taken into consideration:
- The level of fuel in the tank – this information (expressed in ohms) is given by the sender existent in the tank, as shown in Figure 2, sender which is supplied with 12V and connected to the dashboard.

Also, the sender's most important part is represented by a variable resistor – the cursor –, which controls the flowing current through the circuit, based on the dipstick's arm position;
- The fuel consumption estimated by the engine control unit.[5]

In the software which manages the fuel consumption in the tank, there is a variable that the manufacturer refers to as "consumed fuel", noted in the present article as *ConsumedFuel*, because it represents a variable in which the fuel flow is stored.

This variable is a part of the calculus of the final parameter, noted in the software with $L_2$ (t), which determines the number of bars on the screen.

Figure 4 represents the flowchart based on which the level of fuel in the tank of a Dacia Logan car is displayed on the dashboard.[6] A first version of a practical assembly has been created – it can be seen in Figure 3 –, its constituent parts being the sender, the dashboard and a power supply, in order to observe the gradual changes in the number of bars displayed on the dashboard.

The dashboard is a part of the Dacia X90 project, while the sender was destined to the monitoring of the level of gasoline-type fuel.

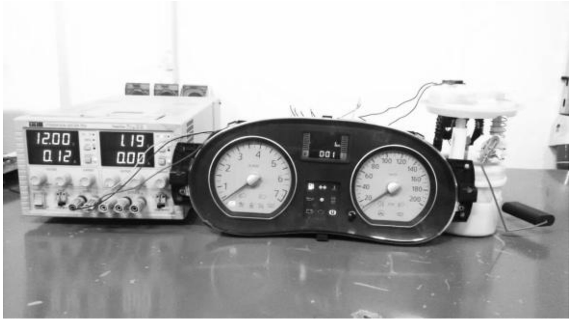

Figure 3 – The first version of the practical assembly which was created

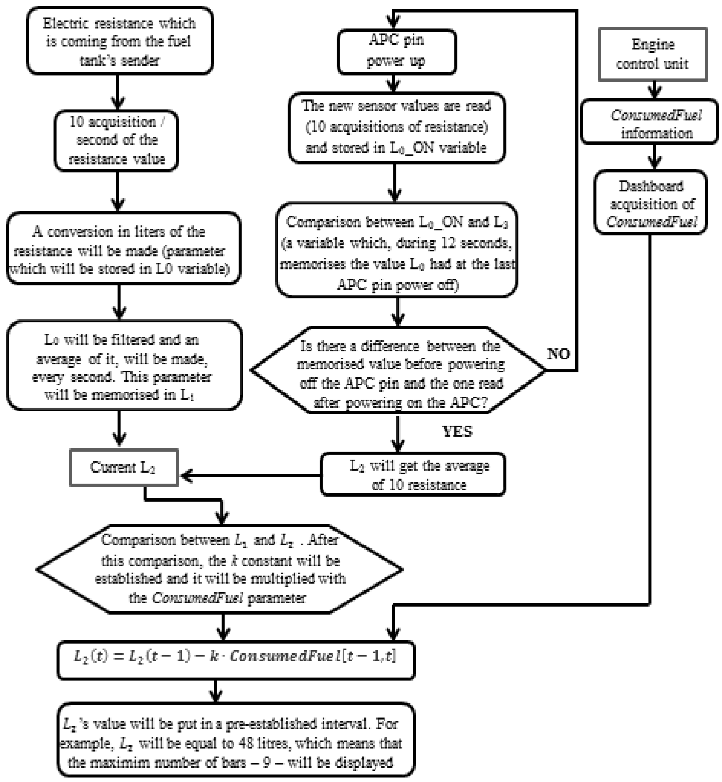


Figure 4 – Flowchart for displaying the fuel level from a Dacia Logan's fuel tank[7]

The main purpose is to verify the interval at which the change in the number of bars displayed on the dashboard occurs. This can be done by checking the indicated area in Figure 5.

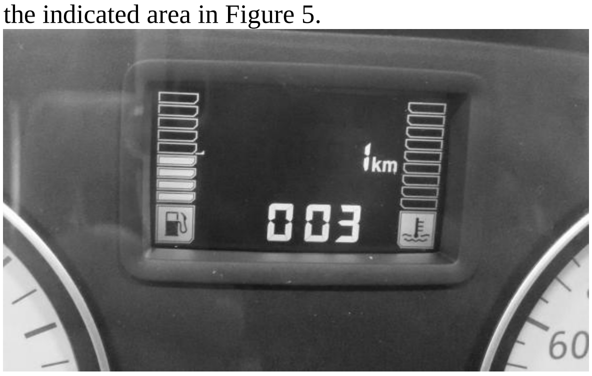

Figure 5 – The area which shows the level of fuel in the vehicle's tank on the dashboard's display.

## III. RESULTS AND INTERPRETATION

According to the flowchart which was presented in figure 3, each bar corresponds to a certain number of liters in the tank.

The $L_2(t)$ parameter represents the level of fuel still in the tank at the *t* moment, and, depending on the value assigned to this variable and, also, on another parameter, whose value is stored in the variable *ConsumedFuel*, the number of bars will appear on the display.

The exact formula to be taken into account for the calculation is the relation (4).

$$L_2(t) = L_2(t-1) - k \cdot ConsumedFuel\,[t-1,t] \quad (4)$$

where *k* represents a variable dependent on $L_1(t)$ and $L_2(t)$.

Also, $L_1(t)$ represents a variable which memorizes the filtered and averaged value of the fuel gauge's resistance value, at the *t* moment, converted in liters.

From relation (4) results that a variation of the electric resistance produced by the height to which the sender's float is positioned will determine a change in formula, therefore a change of the data which contribute to the displaying of the number of bars.

There were brought up for discussion several cases: the interval at which the number of bars updates, with the wire corresponding to the powered or unpowered APC pin (*après contact* – "after contact").

It has been hypothesized that h=f(L), R=f(h) functions and, respectively, R=f(L) are correct and their characteristics provide correct calculated values, according to which, on the dashboard, x bars will be displayed.

Conclusions after doing the present experiment can be visualized in Table I.

As it can be observed in relation (1), it has been concluded that the *ConsumedFuel* parameter has to be taken into consideration, henceforth continuing the measurements.

The stored information in this variable, which is transferred to the dashboard, comes from the engine control unit and it represents the amount of fuel transmitted by the injectors per unit time.

As the waveform is unknown, it is necessary to visualize it with the use of an oscilloscope.

The signal comes from the engine control unit, therefore it has been assumed that the signal has to vary depending on the acceleration and, consequently, on the speed of the engine. Following the experiment, this proved to be true.

It has been chosen that the oscilloscope probe be attached directly to the pin which provides the signal from the engine control unit to the dashboard, this being a more practical solution than disassembling the dashboard.

Several measurements were made at various rotations of the engine, and it has been determined that it is a signal of rectangular form, unipolar, with a slight variation of the duty cycle, with a value of 12Vpp (peak-to-peak value) and a frequency that varies depending on these rotations. The working stand for measuring the *ConsumedFuel* parameter can be seen in Figure 6.

TABLE I – First set of measurements

| **APC pin state** | **Float position modification** | **Observation during measurements** |
|---|---|---|
| After APC pin is powered to 12V, change the float's position, un-power the APC pin and power it again | Modify from a higher position to a lower one. | The update interval of the bars on the dashboard was measured and it was established that it takes 12 seconds for the update. In order to maintain the 12 seconds interval, it is necessary that once the APC pin is powered and, at the same time, the float's position is changed, the APC pin should be disconnected and after disconnecting it, the timing will begin. |
| | Modify from a lower position to a higher one. | |
| After APC pin is powered at 12V, change the float's position and, at the same time, keep the APC pin in the same state. | Modify from a higher position to a lower one. | If the APC pin is not disconnected from the power supply and the float's position is considerably modified, the number of bars from the dashboard's display will have to vary more. |
| | Modify from a lower position to a higher one. | It has been observed that the bars update is not showing on the display. For that conclusion to be made, 40 minutes were measured since changing the float's position. |

From Table II and Figure 7, it can be deduced that the frequency of the signal *ConsumedFuel* rises together with the speed of the engine.

As a result, the signal obtained from the engine control unit can be replaced with the one produced with the use of a rectangular signal generator.

Moreover, in the next stage, the sender will be replaced, adding a resistance box which comprises the whole range of resistances produced by the variation of the sender's float height.

Figure 6 – *ConsumedFuel* parameter measurement from a Dacia's Engine Control Unit, using the oscilloscope and varying the engine's rotations / minute

TABLE II – Measurements which demonstrate that the *ConsumedFuel* signal rises together with the engine speed

| **Engine speed [rot/min]** | ***Consumed Fuel* period [ms]** | ***Consumed Fuel* frequency [Hz]** | **"*t*" period of *Consumed Fuel* parameter [ms]** | **Duty cycle (τ) [%]** |
|---|---|---|---|---|
| 800 | 350 | 2,8 | 160 | 45,7 |
| 1000 | 287 | 3,5 | 156 | 54 |
| 1500 | 224 | 3,9 | 127 | 56 |
| 1750 | 200 | 5,2 | 120 | 60 |
| 2000 | 166 | 6 | 96 | 57 |
| 2500 | 160 | 6,2 | 96 | 60 |
| 3000 | 138 | 7,2 | 83 | 60 |

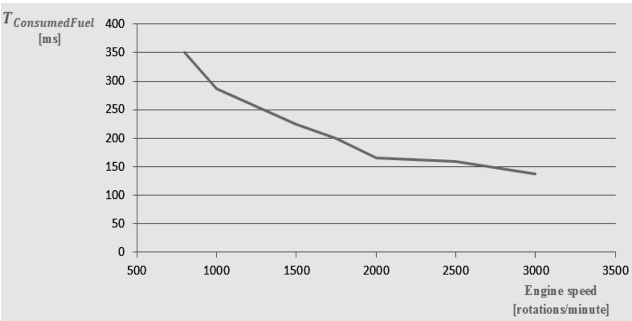


Figure 7 – Function which shows the dependency between *ConsumedFuel* signal period and engine speed

For the final assembly, the sender was replaced with a resistance box, capable of offering precise resistance values, with a low tolerance, in the range which is of interest in this study.

In addition, it was necessary to verify the correspondence between the values of resistances and a certain number of bars, thus a function Number of bars = f(R), in order to perform exact measurements - see Figure 8.

More precisely, it had to be seen at which value of the resistance there is an update of the bars. For this, following the experimental measurements, it was obtained the feature in Figure 8.

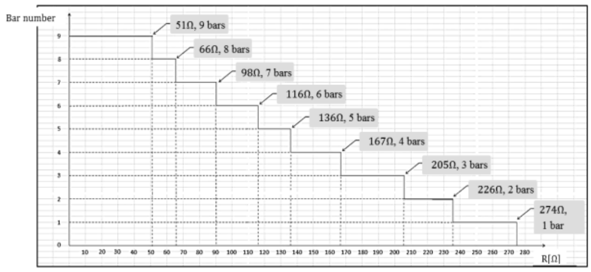


Figure 8 – Characteristic of number of bars displayed on the dashboard and resistance

Remarks about the measurements pointed out in Table III:

The initial resistance was altered with 1Ω / 5sec, until it got to the resistance that had to produce the change in the display of bars. For this information, Table III has to be consulted.

TABLE III - Measurements in which were considered two cases: the absence, and, respectively, the presence of *ConsumedFuel* signal, at the dashboard input

| | | |
|---|---|---|
| **Initial resistance [Ω]** | 50 | |
| **The resistance value which should produce the change on the display [Ω]** | 53 | |
| ***ConsumedFuel* signal** | Active | Inactive |
| **Interval of time that has passed from the moment of the modification of the resistance that should produce a changing in the bars display.** | ≅6 minutes | 11 minutes, 8 seconds |
| **Number of bars which are displayed after update** | 8 | 8 |
| **Initial number of bars / Number of bars which should be displayed after update** | 9 / 8 | |
| **Observations** | The bars update time is reduced considerably when the *ConsumedFuel* is activated. This demonstrates the validity of the formula presented in Figure 3. | |

Another important observation would be that on "startup", after APC, the engine will be idling for a little period of time, in order to simulate the real situation of the engine's state (after activating the APC, there has to be a waiting interval, just a few seconds - for measurements, it was chosen a 20 seconds duration). This implies 20 seconds maintenance of the 2,8 Hz frequency of the *ConsumedFuel* signal (which corresponds to the 800 rotations of the engine in an idle state).

After the 20 seconds pass, there will be a simulation consisting of raising the speed of the engine, an increase which will also translate into the actual increase in the frequency of the *ConsumedFuel* signal, according to Figure 10.

The second variant of assembly is exemplified in Figure 9.

Thus, an almost real simulation of the car's state of drive was obtained. Almost real, because it is a little bit unlikely to simulate fuel consumption without having some information about the speed at a given moment. In order to also simulate this parameter, it is necessary to have a generator capable of simulating the information concerning the speed transmitted to the dashboard, i.e. a generator which produces a signal similar to the one that is in fact transmitted.

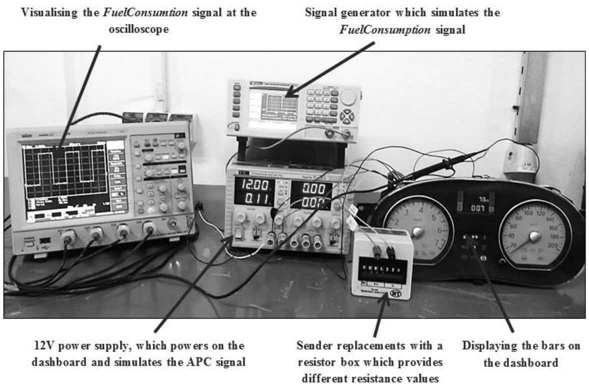


Figure 9 - Assembly which simulates the display of fuel level on the dashboard

It has been concluded that the speed signal is a rectangular signal received from the speed sensor, unipolar, with the amplitude of 5Vpp and a variable frequency. Depending on this frequency information, the dashboard's indicator which displays the speed will be actuated.

By simulating the signal, a linear characteristic will be obtained.

Figure 10 indicates this characteristic of the speed that is indicated on the dashboard and the signal which comes from the speed sensor.

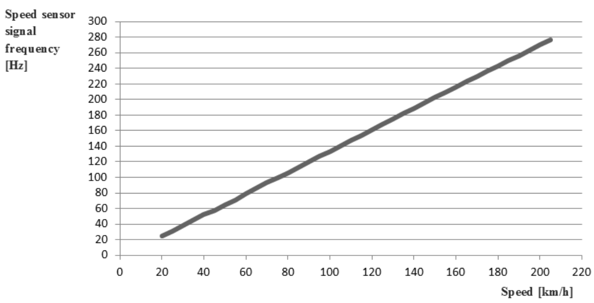


Figure 10 – A characteristic regarding the speed displayed on the dashboard and the speed sensor signal's frequency

Once the characteristics of the waveform have been discovered, measuring it on the Engine Control Unit by using the oscilloscope (Figure 6), the final simulation workbench has been achieved. It can be visualized in Figure 11.

The workbench from Figure 11 was made by using the following equipment:

- Oscilloscope and probes;
- Signal generator;
- Power supply;
- Dacia dashboard;
- Resistance box.

The simulated waveforms can be observed in Figure 12. More details about the waveforms' parameters will be found in Table II of the present article.

The signal with a smaller frequency is the *ConsumedFuel* signal and the second signal represents the speed signal, which offers support, so that the instantaneous consumption would be displayed on the dashboard.[8]

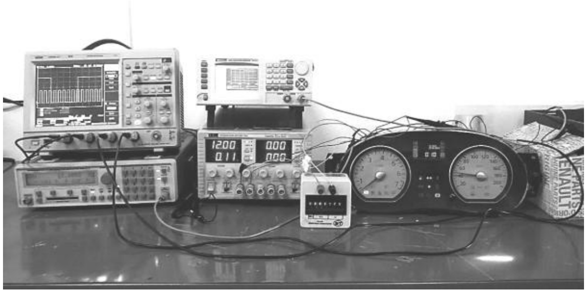

Figure 11 – The final assembly which simulates the fuel consumption, speed and equivalent resistance which occurs when the sender's float position is modified.

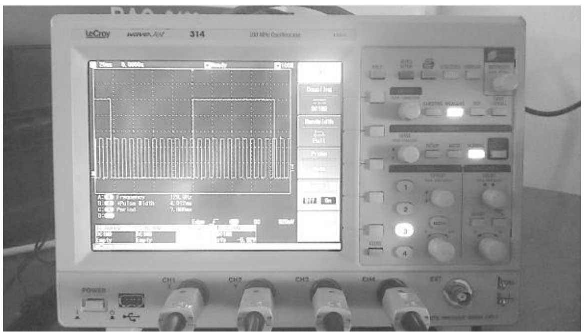

Figure 12 – The two waveforms which had been used in the final simulation worbench, visualized with the oscilloscope

## IV. CONCLUSIONS

By using Dacia automobile parts – dashboard and sender –, other electronic parts which replaced the car battery – power supply – and the signals generated by the Engine Control Unit – simulating them with a signal generator –, it has been demonstrated through measurements that the algorithm which controls the fuel level display is correct.

The realised system, as it was presented in the paper, would be able to simulate most of the fuel gauge systems of Dacia automobile models, for gasolyne-type fuel.

In this way, it will not be necessary to travel a specific distance in order to obtain a specific display – and, in some cases, an erroneous information – of fuel consumption on the dashboard.

The simulation of the consumed fuel system and, also, the results obtained through testing, were made for vehicles on which the communication protocol called CAN (i.e. Controlled Area Network) hasn't been implemented.

For the future projects, special programs will be used. Such programs are CANalyzer and CANoe, comprehensive software tools which allow the simulation and analysis of bus communication.

With CANalyzer analysis tool, it will be possible to complete both simple network analysis and complex debugging problems and it will represent the most appropriate software to use for continuing the present project, for vehicles with inbuilt CAN bus.[9][10]

## References


[1] Renault, "Le tableau de bord", 2009.

[2] Renault, "L'unite de contrôle habitacle", 2009.

[3] Renault, "Formation Circuit à Carburant".

[4] Renault, "Système d'alimentation en carburant", 2009.

[5] Renault, "Specifications for gauging and trip computer".

[6] Renault, "Formation électronique vie serie ", 2009.

[7] Renault, "Specifications for gauging and trip computer".

[8] C. Ana-Maria, "Concepţ ia ş i punerea în funcţ iune a unei mese de simulare pentru cablajul electric auto", UPIT-FECC, Applied Electronics Specialization, 4th year, 2015.

[9] Vector CANtech, "Quick introduction to CANalyzer", 2009.

[10] Vector CANtech, "CAPL Function Reference Manual", 2004.