

Examples of the use of the embedded systems for the long-term collection of slowly-changing parameters in the traction of a car

ARTICLE INFO

Received: 15 July 2021
Revised: 6 September 2021
Accepted: 8 September 2021
Available online: 15 September 2021

The article presents how embedded systems can be used to collect data in the long-term traction of a car. It is assumed that the long period is the time of a travelled distance, e.g. a few thousands of kilometres, or a time, e.g. a month. Such data can be used to optimize the control systems and to diagnose unusual faults in mechatronic systems. The research paper presents how, with the use of very cheap devices, it is possible to collect data that quite often could not be collected even with the use of very expensive measuring devices. The possibility of simple analysis of signals in real time was also pointed out.

Key words: *embedded system, collecting and archiving data, software, signal matching systems, real-time analysis*

This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>)

1. Introduction

A large number of vehicles with internal combustion engines requires constant efforts from manufacturers to develop as eco-friendly a car as it is possible to protect the human environment. The legislator dictates the standards for the toxicity of exhaust fumes from car engines. Scientists and car manufacturers try to meet the requirements set out in the standards. It is becoming more and more difficult. The concept of "eco-driving" has been in use for some time. It can be understood as applying a driving style by drivers that reduces fuel consumption and thus the emission of toxic substances, especially CO₂ [1–3]. There have been many scientific publications on this issue [4]. Many of them present the parameters of the traction of the car and the behaviour of the driver. These are data collected from on-board diagnostic systems or directly from sensors and actuators thanks to the use of a variety of measuring devices. On-board diagnostic systems provide a number of parameters that determine their operation. They can be read with a diagnostic tester in the real-time data reading mode [4]. However, many testers do not provide an option to save read data. There are specialized devices available on the market that can record selected real-time data of mechatronic systems through a diagnostic connector. Unfortunately, the cost and limitations of such recorders are an obstacle to obtain the necessary data. The second method of collecting information in traction is the registration of specific signals directly from sensors and actuators (e.g., fuel injectors). Various mobile and stationary recorders are used for this purpose. Often applied and very universal, they are compatible with computers (e.g., a laptop). Their goal is to sample selected signals with an appropriate frequency and save the values in disk sets. This method has both advantages and disadvantages. The biggest advantage is the possibility to configure the measuring device in a very simple way and a number of options available for the user in this respect. There are also downsides, the first of which is the cost of the device, ranging from several to even tens of thousands of zloty. The second disadvantage is the inability to analyse the measured signals in real time, and the

ability to only save their values. Due to that, the data obtained by this method are formed in files of a very large volume, whereas the useful information contained is often-times relatively small. There are also restrictions as far as the recording time is concerned due to the amount of memory samples installed in the measuring device.

The analyses of the behaviour of drivers and the work of mechatronic systems carried out as a part of the "eco-driving" often require collecting data over long periods of time (e.g., several days or several months). These are slowly-changing data (e.g., vehicle speed, accelerator pedal position, temperature, etc.) Thus, their measurement and recording can take place at a low frequency (e.g., every 0.5 seconds). High accuracy is not required for many of these signals (e.g., it is enough to measure the vehicle speed with an accuracy of 1 km/h).

In analyses related to eco-driving, accurate and fast-changing data may also be needed (e.g., duration of all fuel injections with accuracy of 1 μs). Collecting them in the form of an injection signal recording at a frequency that provides the required accuracy over long distances will result in the creation of large data sets. For this reason, it will be impossible to implement, even for many professional measurement systems.

As a result, there is a problem of collecting various data from mechatronic systems in a long-distance traction.

The aim of this article is to present embedded systems as recorders of slowly-changing parameters (e.g., vehicle speed) that occur in mechatronic systems of cars over long periods of time.

Such measurements are especially useful in various analyses of mechatronic systems of cars related to the "eco-driving" and wherever the observation in a long period of time is necessary (e.g., a month).

The proposal to use embedded systems to collect slowly-changing data in the car traction is justified primarily by the low price of the devices and the ability to analyse signals in real time. In some cases, the real-time data analysis allows you to significantly reduce the amount of data collected without losing useful information. The method of the

analysis is precisely defined by the researchers in the form of a program implemented by a microcontroller. The idea of using embedded systems to collect data in traction arose from the need to collect data and, at the same time, the lack of financial resources to purchase a professional measurement system. Therefore, it is directed to young researchers of automotive issues as well as students of automotive faculties. However, it should be remembered that it has limitations related to the computing power of the microcontrollers used as well as the quality of the A/D converter.

Being able to register and then analyse the results can certainly support the work related to increasing the reliability and optimization of the car's mechatronic systems.

The use of an embedded system as a simple and at the same time specialized computer gives the opportunity to not only collect, but also to analyse data in real time and then save the results. However, in order for a simple and very cheap computer to simultaneously perform a specialized measurement task, it must be properly programmed. Such a program has to be written by a user who knows what information he or she wants to collect and how to do it [5]. Therefore, programming skills are required, which in the case of an engineer (e.g., of mechatronics) is an easy ability to obtain. Many engineers know the basics of microcontroller programming due to their faculty, most often in the form of C++ language [6, 7].

The article presents some examples of measuring signals and parameters in a car with the use of an embedded system.

2. Embedded systems

Embedded systems as simple computers equipped with software have been developed in recent years for various microprocessors. The most famous are as follows: Arduino, Raspberry Pi, Nucleo, etc. Due to the price, hardware availability, and dissemination in the academic and student environment, Arduino was chosen to present as an exemplary measuring device. The microcontroller used in Arduino is the least effective among the above-mentioned platforms. Nevertheless, it is able to carry out some of the measuring tasks with sufficient accuracy and quality. For more demanding measuring tasks, other embedded systems can be used that are more advanced and more expensive than Arduino.

The literature presents a lot of different applications of specialized computers with embedded software. Embedded systems are primarily used to control various processes and devices. The use of embedded systems for recording slowly-changing parameters in running cars was not found. Hence the proposal for such an application of a simple and cheap computer in automotive research.

2.1. ARDUINO boards as simple computers

There are several types of embedded systems for the ARDUINO development platform. The most common ones are the following: Arduino Uno, Arduino Nano, Arduino Mega (Fig. 1).

All of the above-mentioned boards are made of a microcontroller, a quartz resonator, a USB-RS232 converter, a supply voltage stabilizer, a RESET button, signalling diodes, and sockets: USB and power (the latter is not pre-

sent in the nano board). They are in the form of a printed circuit board with led out digital and analogue ports.

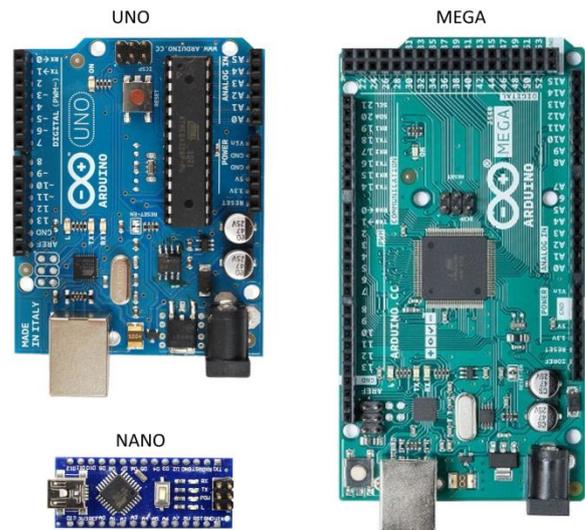


Fig. 1. Examples of Arduino computers (boards)

Arduino Nano and Arduino Uno boards have the ATmega328P microcontroller. It is equipped with three types of memory, such as:

- Flash (32kB) – dedicated to the program code,
- SRAM (2kB) – as operational memory,
- EEPROM (1 kB) – as a non-volatile memory for storing results.

The first 2 kB of flash memory is occupied by the factory-installed software used to start the computer and load the user's software, the boot loader. It acts as such a micro-operating system. The ATmega328P microcontroller is also equipped with one 10-bit analogue-to-digital converter. It enables the measurement of voltage values on analogue inputs in the range from 0 to a given reference value (5 V maximum). Digital ports operate in the so-called 5 V logic. Thus, signals with other voltages require adjustment [6].

2.2. Simple signal matching systems

For the correct identification of binary signals, it is required that their voltage is within the range from 0 to 5 V. Many signals in the mechatronic systems of cars change in this range. There are also binary signals, the amplitude of which reaches approximately 14V (Fig. 2).

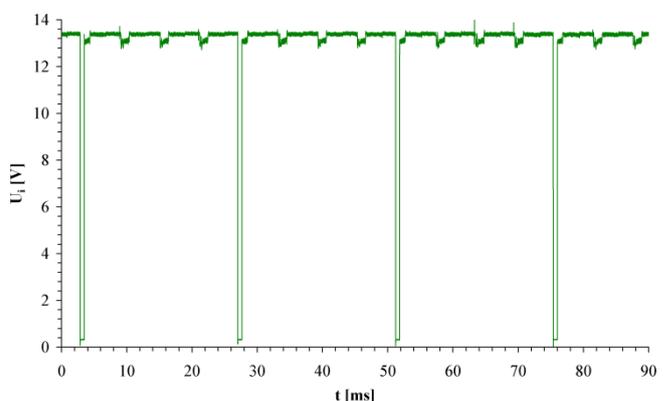


Fig. 2. An example of a binary signal with an amplitude exceeding 5 V

There are also signals similar to square waveforms, but they are not square (Fig. 3). They carry information about the moment of their occurrence and duration, while their amplitude carries no information. Before introducing the microcontroller to the digital port, it is necessary to shape and adjust their amplitude.

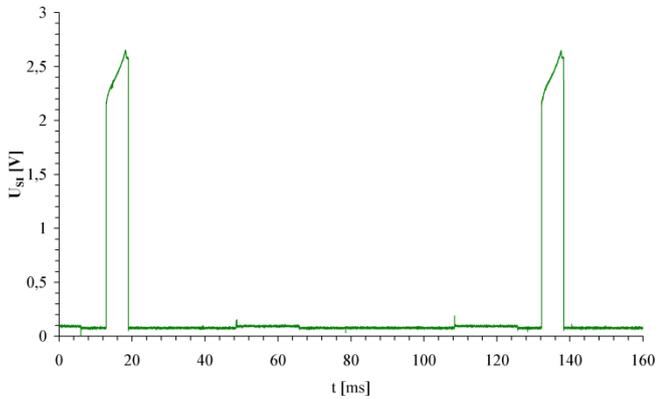


Fig. 3. An example of a signal similar to a square one

Some sensors in mechatronic systems generate analogue signals in which the information is contained in the frequency. An example may be the signal from the inductive speed sensor of the car wheel in the ABS system (Fig. 4).

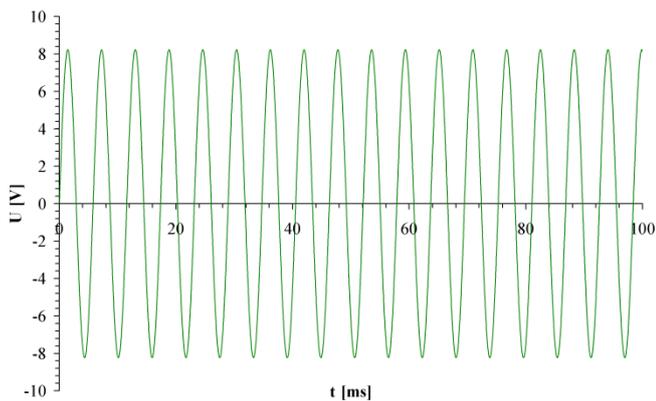


Fig. 4. An example of an analogue signal that carries information only through frequency

Such a signal also changes amplitude with speed. However, its amplitude is not dependent on speed alone. Therefore, transforming it into a square signal will not cause the loss of the carried information, but it will enable the introduction to the digital input of the microcontroller.

The signals shown in Figs 2, 3, and 4 can be converted into square (binary) signals using very simple electronic circuits shown in Fig. 5.

The circuits shown in Fig. 5 are powered by 5 V, which makes their signal output with a voltage of up to 5 V. Each of them requires the selection of resistive elements to the input voltage range. To convert the signals shown in Figs 2 and 3, it is enough to use the circuit shown in Fig. 5a to obtain a square wave signal with an amplitude of 5 V (Fig. 6). However, when transformed, it will be inverted. In order to restore it to its original form in phase, a double transformation should be applied (Fig. 7).

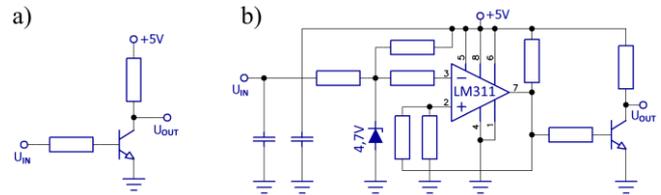


Fig. 5. Examples of two systems of matching the signals in the mechatronic systems of the car to the binary form: a) single-stage inverting voltage amplifier, b) voltage comparator with a hysteresis loop

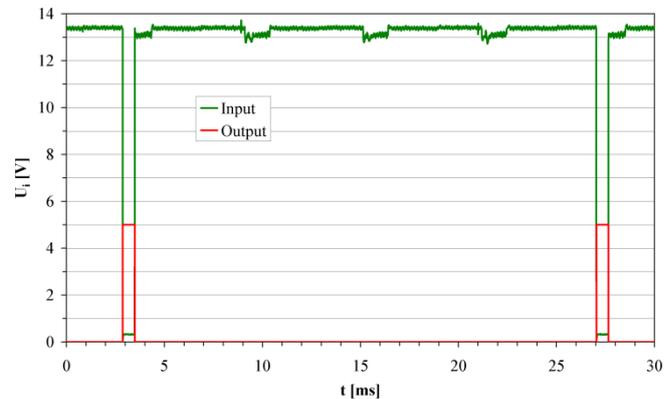


Fig. 6. Signal conversion to a rectangular form with 5 V amplitude limitation and phase reversal

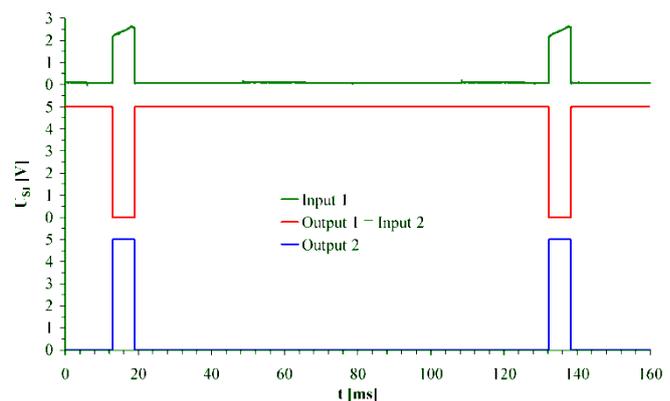


Fig. 7. Double conversion of the signal to a rectangular form with amplitude adjustment up to 5 V

In order to convert an analogue signal (Fig. 4) into a binary one, it is necessary to compare it to the set level. As a result of a disturbance when passing the set threshold, an alternating high and low state can be generated at the output. Therefore, the circuit (Fig. 5b) uses a hysteresis loop in the comparator input that sets the reference level. This causes a certain switching of the comparator output when the signal passes a given threshold. The result of the transformation of the signal from Fig. 4 is shown in Fig. 8.

The presented signal matching systems are characterized by high simplicity and low cost of implementation. At the same time, they enable signal measurements with the use of the embedded system. These are just examples for the presented signals. In a similar manner, other equally simple systems can be designed and implemented for specific needs.

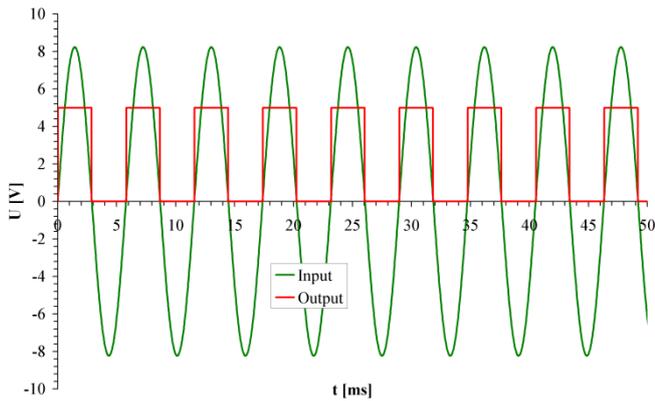


Fig. 8. Conversion of an analogue signal carrying information through frequency into a binary signal with an amplitude of 5 V

3. Measurement system based on an embedded system for the Arduino platform

Embedded systems for the Arduino platform are popular "Arduino boards". These are tiny computers that perform simple tasks. These simple tasks, dictated by the program code, can be a measurement action. The microcontroller is responsible for performing at a high speed many simple actions. Noticing a high or low state on a digital port takes place in less than 1 microsecond. Reading the time of the internal clock also takes a very short period. Such simple actions allow us to measure the duration of binary pulses and count their number. The microcontroller also performs simple arithmetic operations. Unfortunately, these take a longer time. It can also measure analogue voltage signals. It has 8 analogue inputs on which the voltages are measured with the use of one ten-bit analogue-to-digital converter with a maximum sampling rate of 10 kS/s. Though this frequency is not high, the recording of slowly-changing signals is definitely possible and accurate. The measurement resolution at the maximum reference voltage of 5 V is approximately 1.25 mV.



Fig. 9. Measuring systems of slowly-changing parameters were installed in a Mitsubishi Carisma GDI car

In addition to performing measurement actions, the measuring device has to collect measurement results as well. This embedded system has internal memory, yet it is

very small. Therefore, it must use an external non-volatile memory, to which it will transfer the measurement results from time to time. Such external non-volatile memory can be an SD card. The Arduino development platform has libraries to support the SD card and the SPI interface used for the communication process with the SD card. The use of memory in the form of an SD card for measurement data is a very cheap solution, ensuring a very large capacity and easy transfer of measurement data.

The exemplary measuring system presented in this article was made of an "Arduino nano board" and a microSD memory card reader (Fig. 9). The reader is fitted with a 32 GB card formatted for the FAT32 file system with a 16 kB allocation unit.

3.1. Examples of tasks, measurement programs, and their results

The tests of the measuring system were carried out on a Mitsubishi Carisma GDI car. All measured signals were binary in 5 V logic. Some of them were transformed using the matching circuit shown in Fig. 5a. Figure 9 shows two measurement systems. This allows different measurements to be made at the same time. The first system, with a horizontally mounted memory card reader, performs the measurement of:

- the state of three binary signals in 5 V logic that control a three-colour LED diode (the indication of the type of the fuel-air mixture supplying the GDI engine) every 0.5 s.
- the road and speed of the car over time by registering the number that is the sum of pulses from the vehicle speed sensor,
- the engine rotational speed by registering the number that is the sum of pulses from the crankshaft position sensor,
- the average injection time in half-second periods by recording the number that is the sum of the duration of the fuel injection pulses and the number of injections that makes up the sum of the duration times.

The results of the above-mentioned parameters were recorded every 0.5 s. The code of the program that measures the above-mentioned parameters to present its simplicity is presented below. It is related to the hardware capabilities of the microcontroller. Only commands to read ports, compare, sum, and save data are used. No multiplication or division commands were used. Operations on integers were used only. Program code:

```
#include <SPI.h>
#include <SD.h>
File plik;
uint16_t czas[20];
uint32_t droga[20];
uint32_t SCWK[20];
byte pomiar[20];
uint32_t tw[20];
uint8_t lw[20];
uint32_t tw1=0;
uint32_t tp;
uint8_t lw1;
uint32_t S=0;
uint32_t CAS=0;
byte i=0;
byte j1=0;
```

```

byte j2;
byte j3=0;
byte j4;
byte j5=0;
byte j6;
void setup() {
  SD.begin(10);
  pinMode(7, INPUT);
  pinMode(6, INPUT_PULLUP);
  pinMode(5, INPUT_PULLUP);
  pinMode(4, INPUT);
  pinMode(3, INPUT);
  pinMode(2, INPUT);
  plik = SD.open("pomiar.txt", FILE_WRITE);
}
void loop() {
  j2=digitalRead(5);
  if(j2<j1){S++;}
  j1=j2;
  j4=digitalRead(6);
  if(j4<j3){CAS++;}
  j3=j4;
  j6=digitalRead(7);
  if(j6>j5){tp=micros();}
  if(j5>j6){
    tw1=tw1+micros()-tp;
    lw1++;}
  j5=j6;
  if(millis() % 500 == 0){
    czas[i]=millis()/500;
    pomiar[i]=digitalRead(2);
    pomiar[i]=pomiar[i]+digitalRead(3)*2;
    pomiar[i]=pomiar[i]+digitalRead(4)*4;
    droga[i]=S;
    SCWK[i]=CAS;
    tw[i]=tw1;
    lw[i]=lw1;
    tw1=0;
    lw1=0;
    i++;
    delay(1);
  }
  if(i == 20){
    for(BYTE j=0;j<20;j++){
      plik.print(czas[j]*0.5,1);
      for(byte k=0;k<2;k++){
        plik.print(char(9));
        plik.print(bitRead(pomiar[j],k));
      }
      plik.print(char(9));
      plik.print(bitRead(pomiar[j],2));
      plik.print(char(9));
      plik.print(droga[j]);
      plik.print(char(9));
      plik.print(SCWK[j]);
      plik.print(char(9));
      plik.print(tw[j]);
      plik.print(char(9));
      plik.println(lw[j]);
    }
    plik.flush();
    i=0;
  }
}

```

The operation of the measuring system was verified over a distance of 456 km, i.e. On the route from Radom to Gdańsk. The registration results are shown with the use of images in Figs 10, 11, 12. Each waveform shows 42,700 measured values.

High state of a signal in a given colour causes the LED to light in the same colour (green colour indicates that the engine is supplied with a stratified mixture, red colour indicates a homogeneous mixture, and blue colour – no fuel injection, i.e., engine braking).

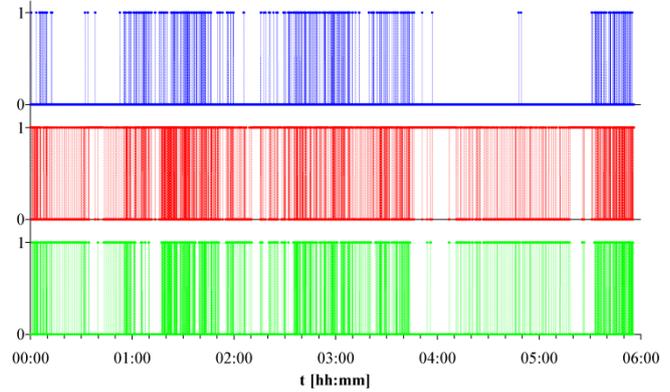


Fig. 10. Three-colour led colour led control signals, (the indication of the type of the fuel-air mixture)

Only one of the three signals is high at a given time. The registration of the results allows us to assess the state of the system at a given moment and allows us to conduct the analysis of its operation. They can also be used in the statistical analysis, e.g., to determine the percentage share of the supply of the engine with a stratified mixture.

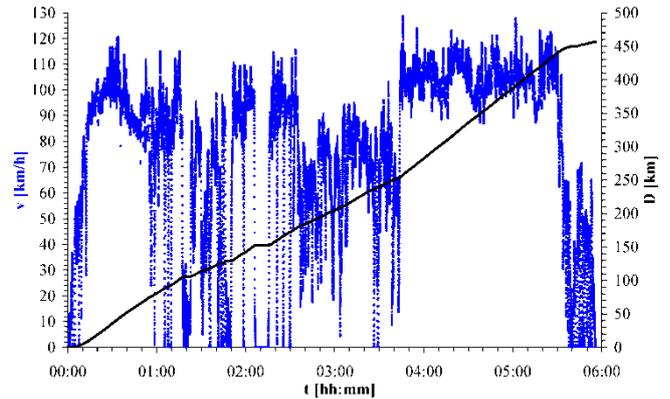


Fig. 11. The waveform of the route and vehicle speed while driving

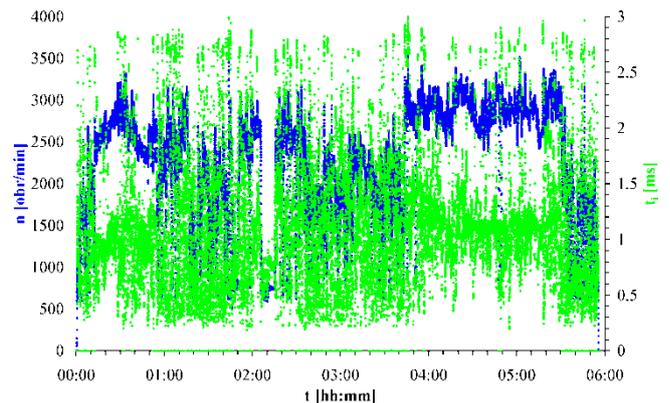


Fig. 12. The waveform of the changes in the rotational speed of the engine and the average injection time in half-second periods

The graph in Fig. 12 shows, inter alia, average injection times in half-second measurement periods. Due to the occurrence of many injections in one measuring period, it is impossible to record the duration of each of them. In addition, there is a different number of injections in each measurement period depending on the rotational speed of the

engine. An attempt was made to record the duration of each fuel injection. The second measuring system was applied (with a vertically installed memory card reader in Fig. 9). This measuring system is designed to measure and record the duration of each injection. During the appearance of the rising slope, the current time is recorded. Immediately after the appearance of the falling slope, the duration of the injection pulse is determined and stored. The code of the program which measures the times of all fuel injections is very simple and looks as follows:

```
#include <SPI.h>
#include <SD.h>
File plik;
byte i=0;
uint32_t t[32];
uint32_t tw[32];
uint32_t tp;
byte j1=0;
byte j2;
void setup() {
  SD.begin(10);
  pinMode(7, INPUT);
  plik = SD.open("tw.txt", FILE_WRITE);
}
void loop() {
  j2=digitalRead(7);
  if (j2>j1) {tp=micros();}
  if (j1>j2) {
    tw[i]=micros()-tp;
    t[i]=millis();
    i++;
  }
  j1=j2;
  if (i==32) {
    for (byte j=0; j<32; j++) {
      plik.print(t[j]);
      plik.print(char(9));
      plik.println(tw[j]);
    }
    plik.flush();
    i=0;
  }
}
```

The results of measurements of selected injections on the route of 456 km are presented with the use of a pictorial diagram (Fig. 13). The graph shows 376 thousand fuel injections. The injection time measurement results can be converted into injected fuel doses. These can be analysed in a number of ways.

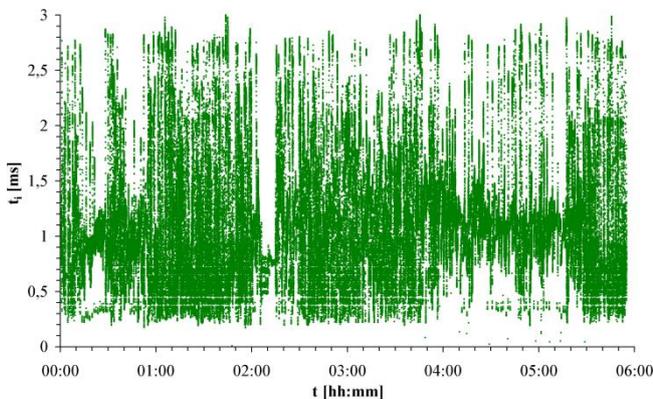


Fig. 13. Times of fuel injections measured on the distance of 456 km

3.2. Verification of a measuring device constructed on the basis of an embedded system

The presented proposals of measuring devices for collecting slowly-changing data were verified. The verification consisted in the comparison of the measurement results obtained with the presented device and a professional measurement system. The first of the two presented measuring systems counts pulses from two sensors and records their numbers every 0.5 s. The second of the presented systems measures the duration of injection pulses and records each of them with an accuracy of 1 μ s. It is connected with the necessity to detect signal slopes, as well as calculate and save the injection pulse duration to a file on the SD card. There is; therefore, a risk that slopes will be noticed with different delay, which will result in incorrect time measurements.

In order to verify the accuracy of the proposed measuring device, a comparative test was conducted. The same signal (of fuel injection for cylinder 1) was introduced on the input of the tested system and on the input of the professional measuring system. The professional measuring system consisted of a PC with an A/D converter card of GAGE manufacturer, Octopus ComuScope 8380 series [9]. The computer measuring system was configured so it records the signal every 1 μ s. The recording time was 50 s, which result from the sampling frequency and the amount of memory of the measurement card. The measurement results were synchronized in such a way that both measurement lines were initialized first, and then the engine was started. Therefore, the analysis of the measurement results was started with the first generated and recorded injection pulse. During the test, the position of the accelerator pedal was changed multiple times, so as to generate pulses with duration times from the entire work range. During the test, the computer measuring system recorded 575 injection pulses. Their times were determined based on the recorded injection signal course and adopted as real values. Then the errors were determined: absolute and relative of the duration measurement for all 575 injections:

$$\Delta t_w = t_{wA} - t_{wp} \quad (1)$$

where: Δt_w – the absolute error of the injection time measurement, t_{wA} – the injection time measured with a device based on the Arduino platform, t_{wp} – the injection time measured with a professional measuring system,

$$\delta t_w = \frac{|\Delta t_w|}{t_{wp}} \cdot 100\% \quad (2)$$

where: δt_w – the relative error of the injection time measurement.

The values of the errors were presented in the form of diagrams.

During the verification of the measuring device, the injection times were changed in the range from approximately 200 μ s to 3350 μ s.

The analysis of the injection time measurement errors presented in Fig. 14 and 15 provides us with the chance to positively assess the measuring device constructed on the basis of a simple computer for the Arduino platform. The measurement of the duration of the vast majority of injec-

tions differs by $\pm 2 \mu\text{s}$ from the real value. What is more, the absolute error of the duration of the vast majority of injections is lower than 1%.

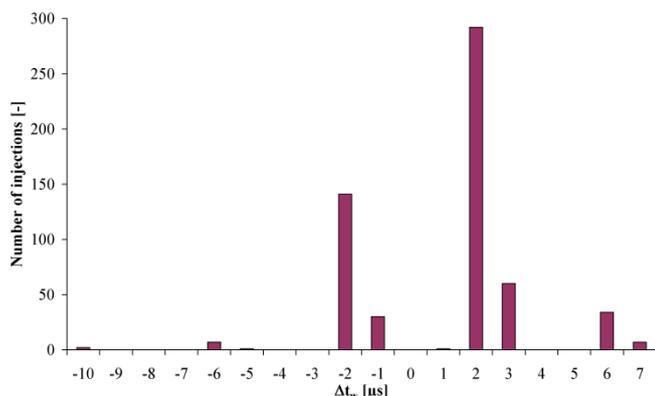


Fig. 14. Diagram of discrepancy (absolute error) of injection time obtained during verification

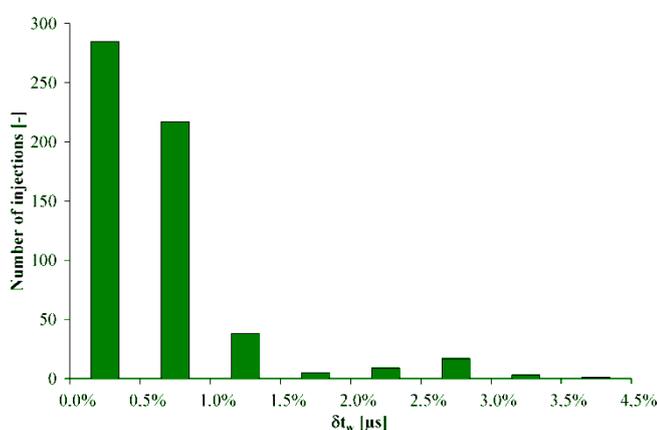


Fig. 15. Diagram of the relative error of injection time obtained during verification

4. Summary and conclusion

The use of embedded systems, i.e., simple computers based on microcontrollers allows for the measurement of various parameters of the signals present in the mechatronic systems of vehicles. Such systems have certain limitations related to the amount of operating memory and the speed of operation. Nevertheless, they provide us with the opportunity to apply quite complex measurements over a very long period of time, e.g., a six-hour measurement of fuel injection duration takes up 5.1 MB of memory on a 32 GB memory card. The only difficulty related to the use of embedded systems for long-term measurements of slowly-changing parameters is the need to write a program and possibly build simple signal matching circuits. However, it can be concluded that these are not tasks beyond the capabilities of every engineer. It is evidenced by the following:

- very low complexity of signal matching circuits, as presented in this research paper,
- simple program code as evidenced by the two examples of the presented programs.

The paper mainly presents the possibility of using embedded systems for the long-term data collection in mechatronic systems of cars. Presented registration results should be treated only as an example of the possibilities offered by this type of microprocessor system. In particular, emphasis should be put on the very low price compared to the potential measurement possibilities. The proposal to use embedded systems to collect slowly-changing data is addressed primarily to young researchers who do not have professional measuring equipment and students of automotive faculties. Another advantage of the presented measuring system is the possibility to permanently install it in the car and collect the necessary data in everyday use of the car. This will reduce the cost of the conducted research and result in the data collection during typical operation of the vehicle.

The accuracy that can be obtained in the more difficult task of measuring the timing of successive fuel injections is satisfactory.

Bibliography

- [1] LOIS, D., WANG, Y., BOGGIO-MARZET, A. et al. Multivariate analysis of fuel consumption related to eco-driving: Interaction of driving patterns and external factors. *Transportation Research Part D: Transport and Environment*. 2019, **72**, 232-242. <https://doi.org/10.1016/j.trd.2019.05.001>
- [2] HUERTAS, J.I., DÍAZ, J., GIRALDO, M. et al. Eco-driving by replicating best driving practices. *International Journal of Sustainable Transportation*. 2018, **12**, 107-116. <https://doi.org/10.1080/15568318.2017.1334107>
- [3] PUCHALSKI, A., KOMORSKA, I. Driving style analysis and driver classification using OBD data of a hybrid electric vehicle. *Transport Problems*. 2020, **15**(4), 83-94. <https://doi.org/10.21307/tp-2020-050>
- [4] FAFOUTELLIS, P., MANTOUKA, E.G., VLAHOGIANNI, E.I. Eco-driving and its impacts on fuel efficiency: An overview of technologies and data-driven methods. *Sustainability*. 2021, **13**(1), 226. <https://doi.org/10.3390/su13010226>
- [5] RIBBENS, W.B. Understanding automotive electronics: an engineering perspective. *Butterworth-Heinemann an imprint of Elsevier*. 2017, Paperback ISBN: 9780128104347, eBook ISBN: 9780128104354.
- [6] MCROBERTS, M. *Beginning Arduino*. Apress 2011.
- [7] CICOLANI, J. *Beginning Robotics with Raspberry Pi and Arduino*. Apress 2018.
- [8] ISERMANN, R. Mechatronic systems – Innovative products with embedded control. *Control Engineering Practice*. 2008, **16**(1), 14-29. <https://doi.org/10.1016/j.conengprac.2007.03.010>
- [9] Gage. Egmont Instruments. http://www.egmont.com.pl/gage/techn/dso_PCI.html

Zbigniew Wołczyński, DEng. – Faculty of Mechanical Engineering, Kazimierz Pulaski University of Technology and Humanities in Radom.
e-mail: z.wolczynski@uthrad.pl

