UDC 621.313.33
DOI: 10.25140/2411-5363-2021-2(24)-188-195

Patrik Šarga, Andrej Krempaský

MONITORING OF THE ELECTRIC MOTOR PARAMETERS

The presented paper describes the creation of a monitoring system based on the IoT platform, designed to monitor the parameters of an electric motor. Based on the obtained data, the operator is informed about the current state of the electric motor and can intervene in time so that it is not suddenly damaged or destroyed. In the next phase of the research, a model of predictive maintenance will be processed based on the obtained data. As a result, the operator will not have to estimate the condition of the electric motor, but the system itself will suggest appropriate measures. Such a system will find application in practice, but also in the teaching process, as the preparation of graduates for the modern monitoring systems, which are increasingly used in practice, will be improved. The presented paper is a scientific and methodological publication.

Keywords: *monitoring; Internet of Things; Cloud, Arduino; Predictive maintenance; Electric motor.*
Fig.: 13. References: 17.

Urgency of the research. Monitoring is an activity in which the monitored values are necessary for the process of evaluating the state of a certain system. Monitored values are determined based on the environment or system, which is monitored. Monitoring is used in many sectors, such as the environment (pollution of air or watercourses), industry (chemical, engineering) and many others. The obtained and subsequently evaluated data can signal the occurrence of potential or existing problems, or predict the future state of the monitored system. Such a solution could save considerable financial costs, which are caused by unexpected failures of the system and subsequent downtime, or elimination of the consequences of its failure.

Target setting. The aim of the research was to create a low-cost monitoring IoT device that would be able to monitor the electric motor for a long time. Based on the obtained data, it would be possible to build a model of predictive maintenance of the electric motor and thus prevent its unexpected failure.

Actual scientific researches and issues analysis. Monitoring systems are used mainly in large industrial companies and help to save a lot of money by being able to detect equipment failure in time, thus avoiding unplanned downtime or complete destruction of the equipment [1]. Professional monitoring systems include, for example, WEG Motor Scan [2] or Smart Condition Monitoring [3]. These monitoring devices are based on IoT and their principle of operation is very similar. The devices send real-time recorded data on the status of the monitored object to the cloud, where the user can monitor the current and previous state of the monitored object. Professional monitoring devices also include a predictive maintenance algorithm, i.e. they can analyze and recognize abnormal operating conditions and, based on this, predict the life of the monitored system. It is possible to define critical values and set alarms that trigger an alert and signal the need to check the device. The collected data are subject to detailed diagnostics and analysis. The result is recommendations for performing maintenance and taking specific measures. The system will provide staff with a detailed description of errors, the cause of their occurrence and a proposal on how to proceed in eliminating the fault. This issue is discussed in publications [4], [5], [6] and [7].

Uninvestigated parts of general matters defining. Existing professional solutions are cost-effective, and this can be an obstacle to setting them into practice. The present article aims to contribute to the development of affordable IoT monitoring systems.

The research objective. Based on the results of this research, the creation of a model of predictive maintenance of industrial equipment will be addressed in its next phase. Another goal is to contribute to the teaching of modern monitoring systems and thus prepare students for the current requirements of industries.

The statement of basic materials. The solution was focused on monitoring the parameters of the electric motor, specifically on measuring rotor vibrations and stator temperature.

Fig. 1 shows the construction of the electric motor and its individual parts. The basic elements are the stator and the rotor. Part of the stator is the winding, the temperature of which was monitored. There is a shaft on the rotor, which is mounted in bearings at both ends and its vibrations are monitored during operation. These could indicate a rotor or bearing failure [8].

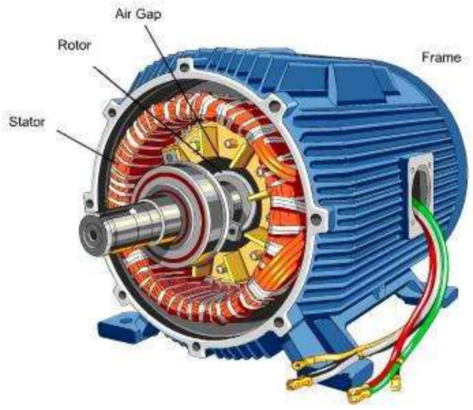

Fig. 1. Electric motor [8]

By monitoring the vibrations and winding temperature, it is possible to prevent the most common failures of the electric motor in time, which include:

- Bearing failures - are among the most common, accounting for almost 40% of all failures of induction machines. These defects immediately create a series of shock vibrations as the moving part of the bearing passes over the defect surface. These frequencies are unique for each bearing size and rotational speed. Bearing failures are most often caused by incorrect lubrication, improper mechanical loading, incorrect bearing assembly and the like. These faults can affect all the parts of which the bearings are composed [8].
- Stator winding faults - most often due to overheating or design errors that cause a short circuit. These faults cause an electrical imbalance on the individual stator coils [8].
- Rotor failures - are caused mainly by shaft deflection. Failure of the rotor can cause mechanical damage to the entire electric motor [8].

As part of the research, a monitoring system was designed that is not costly. The monitoring system built on the IoT platform was connected to the cloud, where the measured data were collected and subsequently evaluated [9], [10] and [11].

An ESP32 microprocessor (Fig. 2) was used to monitor the parameters of the electric motor in conjunction with sensors suitable for this solution.

Fig. 2. ESP32 DevKit [12]

The vibration sensor module based on the vibration sensor SW-420 and comparator LM393 is used to detect vibrations. The threshold can adjust using an on-board potentiometer. The sensor module contains two LEDs - Power and status LEDs. The Power LED indicates the power supply to the sensor. The status LED lights up when the sensor detects vibrations. During no vibration, the sensor provides logic LOW and when the vibration is detected, the sensor provides logic HIGH [13].

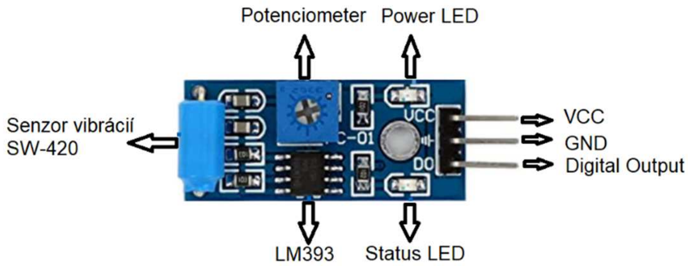

Fig. 3. Vibration sensor SW-420 [13]

A DS18B20 digital sensor was used to measure the temperature. The sensor is manufactured in several versions. In this case, a waterproof variant was chosen (Fig. 4), in which the sensor is sealed in a stainless-steel rod. Such a solution was used to ensure the reliable operation of the sensor even in adverse working conditions. [14]

Fig. 4. Temperature sensor DS18B20 [14]

Fig. 5 shows a proposed circuit diagram of the individual components needed to monitor the condition of the electric motor. The SW-420 vibration sensor contains 3 wires: power supply (VCC), ground (GND) and data (DO). The sensor is connected to the power supply by the red wire, to the ground by the black wire and by the brown wire to the pin 22 on the ESP32 board.

The DS18B20 temperature sensor is also connected using three wires, the first is the power supply, the second the ground and the third the data. The red wire is connected to the power pin, the black wire to the ground, and the yellow wire is connected to data pin 15 on the ESP32 microcontroller. The power supply and data pin of the sensor must be interconnected by a resistor.

The sensors are powered via ESP32 via the VCC and GND pins, which are connected to the power board.

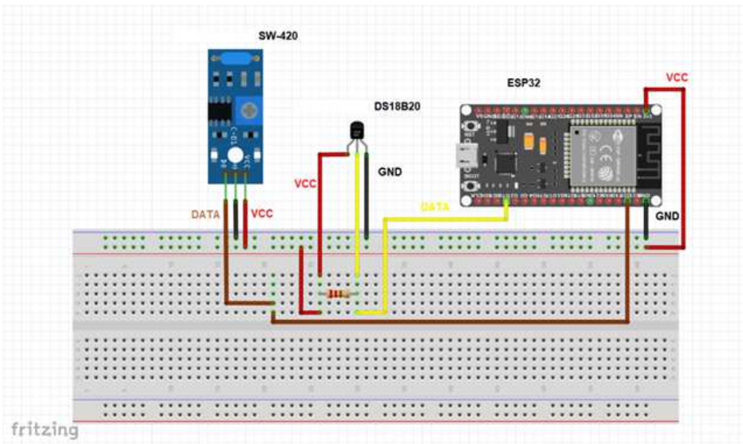

Fig. 5. Equipment connection design

Fig. 6 shows the actual connection of the sensors and ESP32 according to the diagram of Fig. 5.

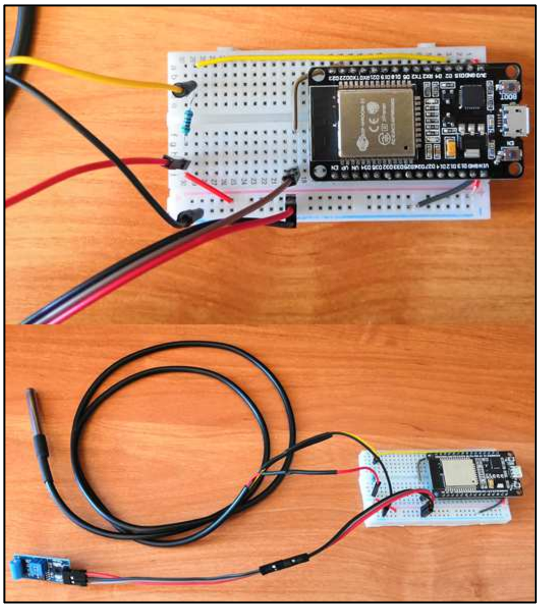

Fig. 6. Real connection

The Arduino IDE was chosen as the software environment for creating the driver, as this software is compatible with almost all Arduino modules, including the ESP32. This software is used to write, compile, and upload code to devices. The software is freely available for common operating systems such as Linux, Windows and MAC. The Arduino IDE supports C and C ++. [16]

An example of the created code is shown in Fig.7.

```
int EP =22;
void setup(void)
{
  Serial.begin(115200); //Nastavenie rýchlosti serial monitora
 WiFi.mode(WIFI_STA);
 ThingSpeak.begin(client); // Inicializácia ThingSpeak

  //Senzor vibrácií
  pinMode(EP, INPUT);
  sensor.begin();
}
void loop(void)
{
 if (WiFi.status() != WL_CONNECTED)
 {
   Serial.print("Pripájanie k SSID: ");
   Serial.println(SECRET_SSID);
   while (WiFi.status() != WL_CONNECTED)
   {
     WiFi.begin(ssid, pass);
     Serial.print(".");
     delay(5000);
   }
   Serial.println("\nConnected.");
 }
  // senzor vibrácií
  long measurement =TP_init();
  delay(0);
  Serial.print("Vibrácie: ");
  Serial.println(measurement);


  //senzor teploty
  sensor.requestTemperatures();
  while (!sensor.isConversionComplete());  // čakanie kým je senzor pripravený
  Serial.print("Teplota v °C: ");
  Serial.println(sensor.getTempC());

  delay(0);
```

Fig. 7. Sample of code in Arduino IDE

The ThingSpeak product from MathWorks, the creator of the Matlab and Simulink environment, was used as the cloud platform, thus ensuring mutual support of ThingSpeak and Matlab [15]. Using Matlab code, a graphical representation of the measured data was created in the ThingSpeak environment. The data is stored in channels, each of which can contain up to 8 fields. An example of a created channel with realized temperature measurement can be seen in Fig. 8. In the next phase of the research, the connection of Matlab and ThingSpeak will be advantageously used in the creation of a predictive model in the Matlab environment.

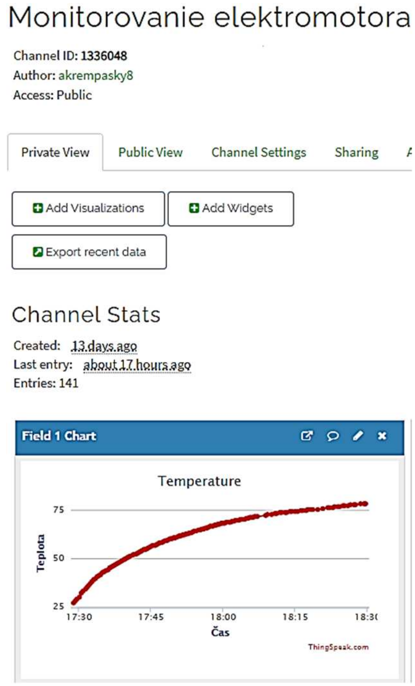

Fig. 8. Channel of electric motor monitoring in ThingSpeak

After creating and testing the functionality of the monitoring system, its practical testing was realised. A single-phase electric motor with a power of 15W and a maximum speed of 2780 rpm was used. It was powered from a 230V network with a frequency of 50Hz. The electric motor was firmly attached to a wooden base with a sheet metal strap, backed with a rubber pad to minimize vibration. Subsequently, the sensors were mounted to record the most accurate values. The temperature sensor was installed directly on the stator winding and the vibration sensor using a screw on the outer cover in which the rotor housing is located. The mounting of the electric motor and the location of the sensors are shown in Fig. 9.

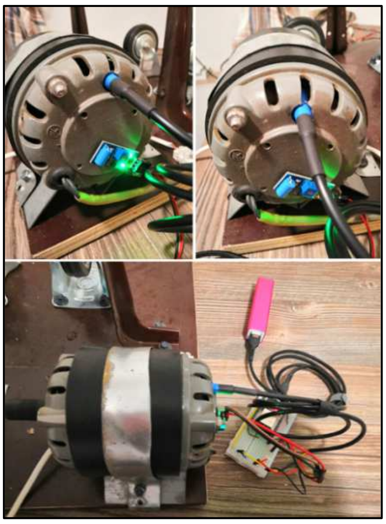

Fig. 9. Mounting of electric motor and sensors

In Fig. 10 is an example of measured vibrations.

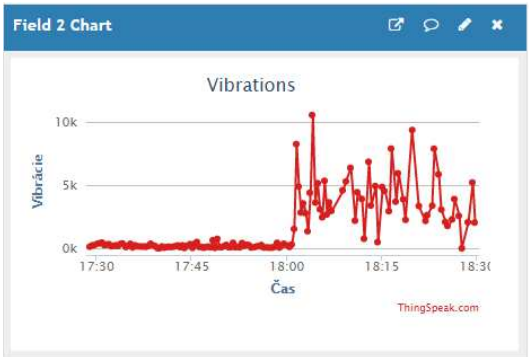

Fig. 10 Measured vibrations in ThingSpeak

After confirming the functionality of the system and finding out the limit values of temperature and vibration, which were still considered standard, the notification part was created in the ThingSpeak environment. This section serves as a warning when limit values are measured. An example of the code is shown in Fig. 11. When the critical value is reached, the operator will be notified via email, but it is possible to adapt the system to notification using other platforms.

ThingSpeak™ Channels Apps Support

MATLAB Code

```
% ID kanála z ktorého sú čítané údaje o teplote
channelID = 1336048;

%API key pre odosielanie upozornení
alertApiKey = 'TAKJQEGCVRHQ1J5WJ4HB4';

%URL Adresa pre odoslanie upozornenia
alertUrl="https://api.thingspeak.com/alerts/send";
options = weboptions("HeaderFields", ["ThingSpeak-Alerts-API-Key", alertApiKey ]);

% Názov predmetu v emailovom upozornení
alertSubject = sprintf("Teplota elektromotora");

% Čítanie posledných dát
TemperatureData = thingSpeakRead(channelID,'NumDays',30,'Fields',1);

% Kontrola a zadefinovanie posledných prečítaných dát
lastValue = TemperatureData(end);
if (lastValue > 100)
        alertBody = ' Vysoká teplota elektromotora ';
elseif (lastValue < 100)
        alertBody = ' Optimálna teplota elektromotora ';
end
% Požiadavka na odoslanie emailu
try
    webwrite(alertUrl , "body", alertBody, "subject", alertSubject, options);
catch someException
    fprintf("Failed to send alert: %s\n", someException.message);
end
```

Fig. 11. Created Matlab code in ThingSpeak

The resulting monitoring environment in ThingSpeak is shown in Fig. 12.

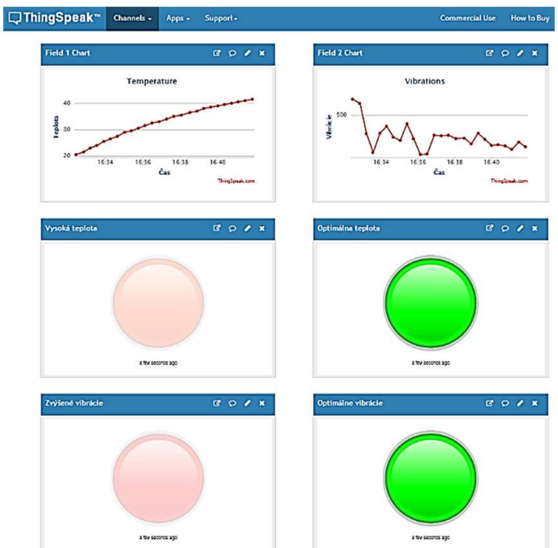

Fig. 12. The resulting monitoring environment in ThingSpeak

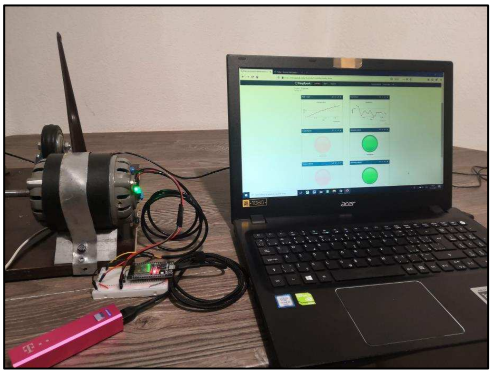

Fig. 13. Testing of the electric motor monitoring system

Conclusions. In the last phase, the overall testing of the resulting IoT monitoring system was realised (Fig. 13). Long-term trouble-free operation of the electric motor was monitored and subsequently, the fault states of the electric motor were simulated and monitored. The tests confirmed the full functionality of the monitoring system, including the notification part. In the next phase of the research, long-term data collection will be implemented, with several modes of operation of the electric motor, including various fault conditions. The obtained data will be used for the creation of a model of predictive maintenance in the Matlab environment. The result will be a fully functional and autonomous low-cost monitoring system that can be used in practice [17].

Acknowledgement. This work has been supported by the Slovak Grant VEGA 1/0330/19.

References

1. How A Machine Monitoring System Benefits Your Industry? (2021). https://www.biz4intellia.com/blog/how-a-machine-monitoring-system-benefits-your-industry.
2. WEG Motor Scan (2021). https://www.weg.net/wegmotorscan/en.
3. Smart Condition Monitoring (2021). https://simap.sk/prediktivna-udrzba-zariadeni/.
4. Ayvaz, S., Alpay, K. (2021) Predictive maintenance system for production lines in manufacturing: A machine learning approach using IoT data in real-time, Expert Systems with Applications, Elsevier, 173, https://doi.org/10.1016/j.eswa.2021.114598.
5. Sun, M., Wu, F. (2021) Effects of imperfect IoT-enabled diagnostics on maintenance services: A system design perspective, Elsevier, 153, https://doi.org/10.1016/j.cie.2020.107096.
6. Qolomany, B., Mohammed, I. (2020). Trust-Based Cloud Machine Learning Model Selection for Industrial IoT and Smart City Services, 99 IEEE Internet of Things Journal, DOI:10.1109/JIOT.2020.3022323.
7. Rákay, R., Galajdová, A. (2020) Testing properties of smart condition monitoring system In: Technical Sciences and Technologies. Chernihiv National University of Technology 21, 3, 266-273.
8. 13-common-causes-of-motor-failure (2021), Retrieved from https://www.fluke.com/en-gb/learn/blog/motors-drives-pumps-compressors/13-common-causes-of-motor-failure.
9. Condition Based monitoring system using IoT (2021). https://www.ripublication.com/ijaer18/ijaerv13n12_08.pdf [in English].
10. Identifying Mechanical Faults with Motor Current Signature Analysis (2021). https://www.reliableplant.com/Read/28633/motor-current-signature-analysis [in English].
11. Predictive Maintenance by Electrical Signature Analysis to Induction Motors. (2021). https://www.intechopen.com/books/induction-motors-modelling-and-control/predictive-maintenance-by-electrical-signature-analysis-to-induction-motors.
12. ESP32 DevKit ESP32-WROOM GPIO Pinout (2021). https://circuits4you.com/2018/12/31/esp32-devkit-esp32-wroom-gpio-pinout.
13. SW-420 Vibration Sensor Module (2021). https://components101.com/sensors/sw-420-vibration-sensor-modul.
14. DS18B20 Temperature Sensor (2021). https://components101.com/sensors/ds18b20-temperature-sensor.

15. ThingSpeak for IoT (2021). https://thingspeak.com/pages/commercial_learn_more.

16. Arduino Software (IDE) (2021). https://www.arduino.cc/en/guide/environment.

17. Krempaský, A. (2021) Monitorovanie parametrov elektromotora na báze Arduina s prepojením na cloud, Košice 2021, 78.

УДК 621.313.33

Патрік Шарга, Андрій Кремпаський

МОНІТОРИНГ ПАРАМЕТРІВ ЕЛЕКТРОДВИГУНА

Моніторинг - це діяльність, при якій відстежувані значення необхідні для процесу оцінки стану певної системи. Отримані та згодом опрацьовані дані можуть сигналізувати про виникнення потенційних або існуючих проблем або передбачати майбутній стан системи, що контролюється. Таке рішення може заощадити значні фінансові витрати, які спричинені несподіваними збоями.

Системи моніторингу застосовуються переважно у великих промислових компаніях. Професійні системи моніторингу засновані на IoT, і їх принцип роботи дуже схожий. Професійні пристрої моніторингу також включають алгоритм прогнозного обслуговування. Існуючі професійні рішення є економічно невигідними, і це може стати перешкодою для їх застосування на практиці. Ця стаття має на меті внести вклад у розвиток доступних систем моніторингу IoT.

Метою дослідження було створення недорогого пристрою IoT для моніторингу, який зможе тривалий час контролювати електродвигун. На основі отриманих даних передбачено побудувати модель профілактичного обслуговування електродвигулювача, тим самим запобігти його неочікуваному виходу з ладу. Інша ціль - навчання студентів сучасним системам моніторингу, готуючи їх до існуючих вимог галузі.

В роботі запропонована повністю функціональна недорога система моніторингу IoT, яка попереджає оператора про досягнення критичних значень для електродвигуна. Здійснено повне тестування системи моніторингу IoT. Тести підтвердили повну функціональність системи моніторингу.

На основі результатів цього дослідження на наступному етапі буде впроваджено довгостроковий збір даних. Отримані дані будуть використані для створення моделі прогнозного обслуговування промислового обладнання в середовищі Matlab. В результаті пердбачається отримати повністю функціональну і автономну недорогу систему моніторингу, яку можна буде використовувати на практиці.

Представлена в статті інформація має науково-методичний характер.

Ключові слова: *моніторинг; інтернет речей; хмара, Ардуїно; інтелектуальне обслуговування; електричний мотор.*

Fig.: 13. References: 17.

Patrik Šarga – Doctor of Technical Sciences, Associate Professor, Department of Automation and Human Machine Interactions, Faculty of Mechanical Engineering, Technical University of Košice, Letná 9 Košice, Slovakia
E-mail: patrik.sarga@tuke.sk
ORCID: https://orcid.org/0000-0002-5566-8871
Scopus Author ID: 15128102900

Andrej Krempaský – Student of mechanical engineering, Faculty of Mechanical Engineering, Technical University of Košice, Letná 9 Košice, Slovakia
E-mail: andrej.krempasky@student.tuke.sk

Šarga, P., Krempaský, A. (2021). Monitoring of the electric motor parameters. *Technical sciences and technologies,* (2(24)), 188-195.