UDC 004.45:004.67:004.4
DOI: 10.31673/2409-7292.2025.017899

Nikitin D.M., Rybitskyi O.M.

AUTOMATED MODELS OF SOFTWARE SYSTEMS FOR VEHICLE DIAGNOSTICS: APPROACH TO DATA INTEGRATION

The introduction of the OBD-2 interface provided standardized access to vehicle diagnostic data, enabling real-time monitoring of the system's status. Effective utilization of this data requires the development of intelligent algorithms capable of analyzing the obtained information while considering signal variability and the probability of faults. One of the approaches to improving analysis accuracy is the use of finite state machines (FSMs), which allow structuring the decision-making process based on a set of defined states and transitions between them. This study explores the method of integrating FSM into OBD-2 data analysis processes to create an automated diagnostic system that enhances fault detection accuracy and reduces the number of false-positive results. The proposed diagnostic model employs FSMs to build a flexible and scalable logic for analyzing a vehicle's condition. As part of the research, a mathematical FSM model was developed, considering the temporal variation of OBD-2 parameters and identifying critical deviations based on signal timing characteristics. A software package for system modeling and testing was created, allowing the verification of its effectiveness based on both synthetic data obtained in the MATLAB/Simulink environment and simulated scenarios. A comparative analysis of fault detection accuracy using the proposed FSM model versus traditional threshold methods demonstrated an increase in diagnostic reliability. The testing results showed that fault detection accuracy increased to 92.2%, while the false-positive rate decreased to 4.1% compared to classical OBD-2 data analysis methods. The proposed approach reduced processing delay to 250 milliseconds per diagnostic cycle, making it applicable for real-time fault detection.

Keywords: finite state machine, automotive diagnostics, OBD-2, software architecture, data integration, simulation

Introduction and problem statement

Automated vehicle diagnostic systems are a crucial element of modern automotive maintenance, aimed at ensuring timely fault detection, enhancing road safety, and reducing repair costs. One of the key standards that allows real-time access to actual operational parameters of automotive systems is OBD-2 (On-Board Diagnostics). This system provides a unified interface for accessing data from electronic control units (ECUs), enabling diagnostics without the need for vehicle disassembly. However, processing OBD-2 data presents several challenges, including significant signal variability, the possibility of erroneous faults, and the need for rapid decision-making.

Traditional methods of OBD-2 data analysis, based on simple comparisons of obtained values with predefined thresholds, have significant limitations. They do not account for dynamic parameter changes, which may lead to missed faults or the generation of irrelevant alerts. For example, a brief engine temperature spike does not always indicate a critical failure, yet the system might register it as a fault.

At the same time, combined faults, where multiple parameters change simultaneously (e.g., increased temperature and excessive engine load), might remain undetected by traditional algorithms. This highlights the need for more adaptive analytical methods capable of considering the sequence of parameter changes, their interdependencies, and temporal characteristics.

One approach to addressing this issue is the use of finite state machines (FSMs), widely applied in control systems and data processing. FSMs allow the diagnostic process to be represented as a set of states with transitions between them depending on the current OBD-2 parameter values. This approach formalizes decision-making by considering previous system states, significantly improving analysis accuracy. FSMs also provide the flexibility to adapt to different vehicle conditions by adjusting the transition parameters between states.

Literature review

Researchers have long recognized the limitations of threshold-based strategies for automotive fault detection. Yadav and Swetapadma illustrated these shortcomings in their paper on transmission line diagnostics [1], highlighting how finite-state machines (FSMs) can better classify faults by examining the sequence and duration of anomalies rather than just instantaneous values.

In the area of Moore FSM design, Solov'ev provides structural models that reveal how to detect subtle failures [2] by carefully observing transitions within a state machine. His approach transfers readily to automotive diagnostics, where real-time states of system health can shift rapidly due to variations in OBD-2 signals.

When it comes to managing timing issues—for instance, delays and clock drift—Köhl and Hermanns proposed a robust method for diagnosing real-time systems [3]. Their work is key for OBD-2 scenarios, which often involve streaming data arriving at different intervals, making it challenging to interpret sensor values.

On a more application-oriented note, Górski and Stecz (2024) examined FSM-based models for near-real-time testing [4], showing how to break down system operation into clearly defined states and transitions.

In terms of educational resources, MIT OpenCourseWare provides a well-known collection of lecture notes on building and verifying state machines, giving an entry point for engineers who want to implement FSMs for tasks like OBD-2 data analysis [5]. Similarly, Fuicu implemented FSM algorithms for real-time monitoring in IoT settings [6], demonstrating fast response loops, where timely intervention can prevent damage or accidents.

Machine learning also plays a growing role in predictive automotive diagnostics. Gong explored various AI-driven strategies to automate fault detection, underscoring the idea that combining state-machine logic with data-driven insights could further reduce downtime [7].

Bringing these insights together, the study applies FSM principles to OBD-2 data streams, showing how well-structured state machines can reduce false positives, enhance detection speed, and even anticipate vehicle faults.

Aim and objectives

Despite progress in system diagnostic, several challenges remain regarding the integration of FSMs and OBD-II. These include the need for real-time processing of large data volumes, ensuring compatibility with different vehicle models, and adapting to specific operating conditions.

The aim of this study is to develop an automated vehicle diagnostic model based on the integration of finite state machines (FSMs) with real OBD-2 data to improve fault detection accuracy. Traditional methods of diagnostic parameter analysis often fail to account for the sequence of changes in vehicle system behavior, leading to false positive or false negative results.

The proposed approach is to formalize the diagnostic process as a set of states with defined transition rules, enabling a more structured and precise fault analysis. Implementing this system will enhance the efficiency of vehicle maintenance by reducing the risks of unexpected failures and lowering operational costs through early problem detection.

Furthermore, automating diagnostics with FSMs will allow for the creation of adaptive algorithms capable of functioning across various vehicle makes and models, ensuring flexibility and scalability of the system.

To achieve this aim, the study defines the following key objectives:

1. Analyze existing vehicle diagnostic methods and identify their shortcomings in the context of OBD-2 usage.

2. Develop a mathematical FSM-based diagnostic model that will determine the sequence of parameter changes and generate diagnostic conclusions based on structured rules.

3. Create a software prototype of the FSM system to facilitate real-time acquisition, processing, and analysis of OBD-2 data.

4. Conduct modeling and simulation-based testing of the proposed model using both synthetic and virtual data obtained from vehicles simulated in different operational scenarios.

5. Perform a comparative analysis of the developed FSM model's efficiency against traditional methods, evaluate fault detection accuracy, and optimize the algorithm to enhance performance.

© Nikitin D.M., Rybitskyi O.M. (2025). Automated models of software systems for vehicle diagnostics: approach to data integration. Сучасний захист інформації, 1(61), 105–112.
https://doi.org/10.31673/2409-7292.2025.017899.

6. Formulate recommendations for further development and integration of FSM-based methods into diagnostic systems for vehicle servicing and predictive technical condition analysis.

Presentation of the main material

The automated vehicle diagnostic system proposed in this study is based on the integration of OBD-2 (On-Board Diagnostics) data with finite state machines (FSMs) for accurate analysis of a vehicle's technical condition. The primary goal of the system architecture is to develop a flexible and adaptive software model that not only identifies current faults but also detects trends in changes to the vehicle's systems. To achieve this, a multi-level structural architecture is designed, consisting of a data acquisition layer, a processing and analysis layer, and a visualization and decision-making layer. The architectural diagram is shown in Figure 1.

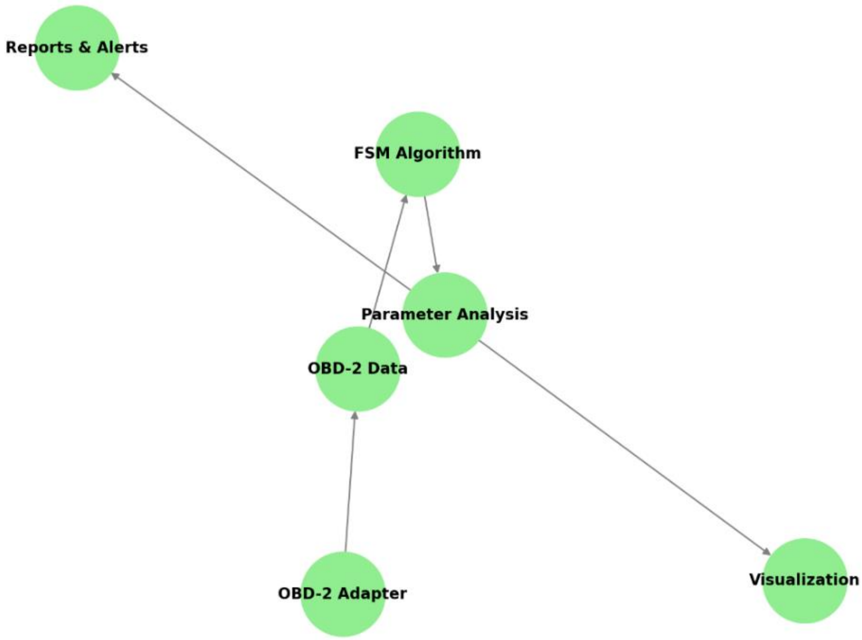

Figure 1. Architecture of the FSM diagnostic model with OBD-2

At the data acquisition level, the standard OBD-2 interface is used to retrieve diagnostic information from the vehicle's electronic control unit (ECU). In simulation-based settings, these OBD-2 parameters are emulated through software tools (e.g., MATLAB/Simulink). Key indicators such as engine RPM, coolant temperature, vehicle speed, engine load, oxygen levels in exhaust gases, and diagnostic trouble codes (DTCs) are generated in real-time or accelerated-time simulations. This setup enables immediate analysis of the vehicle's condition without the variability of physical testing.

The next level is the data processing and analysis system, where finite state machine (FSM) algorithms are applied. The FSM model represents the operation of a vehicle as a set of discrete states, with transitions between them based on the analysis of OBD-2 parameters. For example, to detect engine overheating, the FSM considers not only the instantaneous temperature value but also the historical changes of this parameter over time, which helps eliminate false alerts.

The FSM is represented as a set of states S, a transition function between them δ and a set of input parameters I. Within the FSM model, each state defines normal or deviated system behavior. Examples of states include:

• S_0 (normal engine operation) – all parameters are within the normal range.

• S_1 (minor deviation) – engine temperature exceeds the threshold but only for a short period.

• S_2 (critical state) – engine temperature exceeds 110°C for more than 30 seconds, and RPM surpasses 4000.

• S_f (emergency situation) – immediate engine shutdown is required.

© Nikitin D.M., Rybitskyi O.M. (2025). Automated models of software systems for vehicle diagnostics: approach to data integration. Сучасний захист інформації, 1(61), 105–112.
https://doi.org/10.31673/2409-7292.2025.017899.

The equation of the FSM is presented in formula 1:

$$S_t + 1 = \delta(S_t, I_t), \tag{1}$$

where S_t is the current state of the system, and I_t represents the OBD-2 input parameters at time t. This allows the system to identify faults not only based on individual parameter values but also on their combinations.

To ensure the flexibility and scalability of the proposed FSM model, it has been implemented as a software module for working with OBD-2 devices (or corresponding simulation data). The program code allows for modifying threshold values for each state, enabling adaptation of the algorithm to various simulated vehicle conditions.

The visualization and decision-making layer include the development of a graphical interface that displays the vehicle's status in real time. The visual component consists of:

• FSM state diagram, showing the current operating mode.

• Real-time updated list of (simulated) OBD-2 parameters.

• Notification system, alerting users to critical deviations.

The main advantage of the FSM model is its ability to identify faults not only based on instantaneous parameter values but also through the analysis of their sequence. The diagram of the FSM model for engine fault diagnostic is shown in Figure 2.

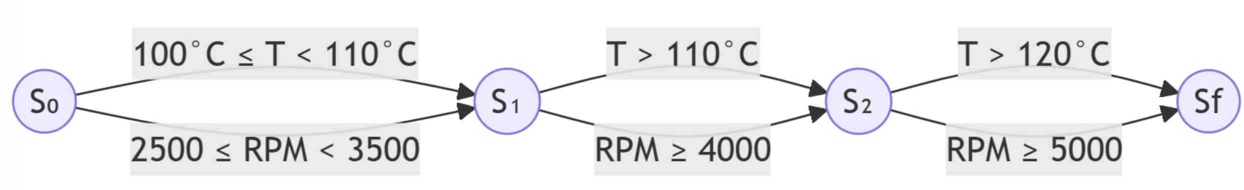

Figure 2. FSM diagram for engine fault diagnostic

The FSM model for diagnostics consists of the following main states:

• S_0 – Normal mode (all parameters are within the normal range).

• S_1 – Warning state (minor deviations that are not critical).

• S_2 – Suspected malfunction (the deviation persists for a certain period).

• S_f – Critical state (a malfunction has been diagnosed, requiring corrective actions).

The transition function between states is determined based on a combined analysis of parameters read via OBD-2. Threshold values for transitions between states are defined for each parameter, for example:

1. Coolant temperature $T_{coolant}$:
   - If $T_{coolant} < 100°C \Rightarrow S_0$
   - If $100°C \le T_{coolant} < 110°C$ (up to 20 seconds) $\Rightarrow S_1$
   - If $T_{coolant} > 110°C$ for more than 30 seconds $\Rightarrow S_2$
   - If $T_{coolant} > 120°C \Rightarrow Sf$
2. Engine RPM:
   - If $RPM < 2500 \Rightarrow S_0$
   - If $2500 \le RPM < 3500 \Rightarrow S_1$
   - If $RPM \ge 4000$ more than 10 seconds $\Rightarrow S_2$
   - If $RPM \ge 5000 \Rightarrow Sf$

The combination of these parameters determines the conditions for transitions between FSM states. In addition to tracking direct OBD-2 signals, the system introduces a “state confidence factor”, which quantifies the certainty of the model’s current state. This factor is dynamically updated based on how consistently parameter values match the thresholds for S_0, S_1, S_2, or S_f. By using this extra

© Nikitin D.M., Rybitskyi O.M. (2025). Automated models of software systems for vehicle diagnostics: approach to data integration. Сучасний захист інформації, 1(61), 105–112.
https://doi.org/10.31673/2409-7292.2025.017899.

measure, the FSM can provide more nuanced feedback, such as indicating the likelihood of progressing from S_1 to S_2 if certain warning signs remain persistent.

To assess critical malfunctions, the model utilizes a risk function (formula 2) that evaluates the overall level of parameter deviations from the norm.

$$R = w_1 \cdot f_1(I) + w_2 \cdot f_2(I) + \cdots + w_n \cdot f_n(I), \tag{2}$$

where w_n are the weight coefficients of the parameter's significance, and $f_n(I)$ is the normalization function of the input parameter. If R exceeds a certain threshold $R_{threshold}$**,** the transition to state S_f (critical state) occurs.

An additional enhancement involves a reinforcement learning policy that can fine-tune the weights w_n over time. By feeding historical diagnostic outcomes into a learning module [8], the system can gradually emphasize parameters that have proven most reliable to failures, further boosting detection precision.

To verify the effectiveness of the developed FSM-based vehicle diagnostics model, two main testing stages were conducted: modeling processes in MATLAB/Simulink and software-based testing of vehicles under simulated conditions to replicate varied automotive behaviors. The objective of this study was to assess accuracy, response speed, and the rate of false positives compared to traditional vehicle parameter analysis methods.

At the first stage, a digital twin of the diagnostic system was created in MATLAB/Simulink, where changes in key OBD-2 parameters were simulated, including coolant temperature, engine RPM, oxygen levels in exhaust gases, and diagnostic trouble codes (DTCs). Signal generators were used to simulate real vehicle behavior under normal and abnormal operating conditions.

During modeling, special attention was paid to analyzing the dynamic changes in parameters. For instance, during the simulation of engine overheating, a scenario was created where the coolant temperature gradually increased from 90°C to 120°C, while engine speed exceeded 4000 RPM. Traditional threshold-based analysis methods immediately registered overheating when exceeding 110°C, whereas the FSM model considered the duration of the parameter staying in the critical zone, reducing the number of false positives.

The next step involved software-based testing intended to replicate real-world conditions as closely as possible. For this, four simulated vehicle profiles of different makes and model years were created, reflecting typical engine characteristics. Data was collected within each simulation using a virtual OBD-2 adapter interface. The data was recorded under various driving conditions:

- Idling for 5 minutes;
- City driving at speeds up to 60 km/h;
- Highway driving at speeds over 100 km/h;
- Fault simulation (e.g., disabling the MAF sensor).

For each state predicted by the FSM model, a comparative analysis was conducted between the obtained simulated values and the theoretical calculations. For example, during the simulation of engine overheating, the FSM identified the state as critical under the condition expressed in formula 3.

$$T_{coolant} > 110°C \; for \; 30 \; seconds \wedge RPM > 3500. \tag{3}$$

However, certain simulation runs revealed that temperature exceeded 110°C for only 15–20 seconds before returning to normal. To address this issue, the state transition logic was modified by adding a hysteresis function, which required the critical state to persist for at least 25 seconds before registering a fault. The modified transition function is shown in formula 4.

Another important parameter in the FSM model was the determination of increased engine load under simulated conditions. Theoretically, the FSM identified an overload state when engine load was greater than 80% for 10 seconds. However, data showed that short-term peak loads on the engine

© Nikitin D.M., Rybitskyi O.M. (2025). Automated models of software systems for vehicle diagnostics: approach to data integration. Сучасний захист інформації, 1(61), 105–112.
https://doi.org/10.31673/2409-7292.2025.017899.

were not always indicative of a malfunction. For example, a sudden acceleration could temporarily exceed 80%, but this did not necessarily indicate serious technical issues.

$$S_{t+1} = \begin{cases} S_2, if\ T_{coolant} > 110°\text{C for } 25s \\ S_f, if\ T_{coolant} > 120°\text{C } \wedge \text{RPM} > 4000 \\ S_0, otherwise \end{cases}. \quad (4)$$

To reduce false diagnoses, a filtering mechanism based on the dynamics of parameter changes was added. This adjustment considered not only the absolute load value but also the average rate of change over time:

$$\frac{\Delta L}{\Delta t} < 5\%.$$

This allowed the system to register an overload only in cases of sustained load increase over a certain period, rather than reacting to short-term peaks.

The experimental results demonstrated that the FSM-based approach reduces false positive alerts compared to other methods [9]. For instance, in engine overheating diagnostics the FSM model reduced the false alarm rate to 4.1%. Similar improvements were observed in oxygen sensor fault analysis and fuel system failure detection. The flexibility of FSM allowed handling different simulated vehicle profiles with minimal adjustments [10]. The details of the improvements can be seen in table 1.

Table 1

Comparison of FSM model with Traditional Diagnostic Methods

Method	Fault Detection Accuracy	Signal Processing Time	False Diagnosis Rate (%)
Threshold	85%	300-600ms	7.5%
FSM Model	92.2%	250ms	4.1%

One of the key aspects of testing was verifying FSM reliability in complex scenarios, such as temporary sensor failures [11]. In many simulated environments, sensors could transmit short-term incorrect values, which might interpret as actual faults. The FSM model filtered out such transient anomalies and responded only in cases of stable exceedance of critical parameters.

Another area of testing was the analysis of combined faults, where multiple parameters simultaneously exceeded normal limits. For example, a mass airflow sensor (MAF) malfunction combined with increased engine RPM could indicate a fuel system issue. In this case, the FSM model was able to determine the interdependencies between parameters, comparing to analyzing each parameter separately.

The test results also confirmed the stability of the FSM model across different simulated profiles, mirroring popular OBD-2 data transmission protocols (such as CAN, ISO 9141-2, SAE J1850) [12]. During testing, several runs showed gradual increases in catalyst temperature and instability in oxygen sensor readings, indicating possible fuel system clogging. In the testing simulation these changes were detected before diagnostic trouble codes (DTCs) would typically appear.

Conclusions

The study demonstrated that the use of Finite State Machines (FSM) for OBD-2 data analysis significantly enhances the accuracy and speed of vehicle fault diagnostics. The implementation of a hysteresis mechanism and cross-parameter analysis enabled the model to consider not only instantaneous diagnostic parameter values but also their dynamics, making the system more resilient to temporary deviations and sensor instability.

The filters for short-term parameter variations embedded in the model reduced the risk of erroneous diagnostic decisions, which was previously a common issue in classical OBD-2 data processing methods. The alignment of the FSM model with simulated datasets confirmed its high efficiency in complex scenarios, such as temporary sensor deviations, peak engine loads, and combined faults.

Moreover, this research suggests a strong potential for integrating multi-agent architectures into the FSM framework. In such systems, each critical vehicle subsystem (e.g., powertrain, emission control) could operate as a semi-autonomous agent, collaborating with others in real time to vote on diagnostic outcomes. This cooperative decision-making can further minimize false alarms and generate timely warnings for developing malfunctions.

The findings of this study confirmed that the integration of an FSM model into OBD-2 based tools for automated vehicle diagnostics, offers high accuracy, fast response times, and adaptability to different types of vehicles. The developed system architecture ensures flexibility and efficiency, making it suitable for use in systems and platforms for vehicle condition monitoring.

Despite the high efficiency of the FSM framework, some challenges remain, requiring further research. One of these is the integration of predictive algorithms based on artificial intelligence, which would not only detect faults but also anticipate their potential occurrence before they impact vehicle performance. Incorporating machine learning for predicting future malfunctions might further improve the safety and efficiency of diagnostic systems. Additionally, future work could focus on expanding the range of analyzed parameters, including brake control systems, suspension systems, and auxiliary electronic modules, allowing the FSM model to cover a broader spectrum of vehicle malfunctions.

Furthermore, investigating robust cybersecurity measures for protecting OBD-2 data – such as encrypted vehicle-to-cloud communication protocols or blockchain-based ledgers could strengthen the reliability of remote diagnostics and reduce risks of tampering or data spoofing. By addressing these security issues, the next generation of FSM-driven diagnostic platforms can confidently integrate connectivity services without compromising data integrity.

References

1. Yadav, A., & Swetapadma, A. (2015). A finite-state machine-based approach for fault detection and classification in transmission lines. Electric Power Components and Systems, 44(1), 43–59. https://doi.org/10.1080/15325008.2015.1091862

2. Solov'ev, V. V. (2023). Structural models for failure detection of Moore finite-state machines. Journal of Computer and Systems Sciences International, 62, 977–990. https://doi.org/10.1134/S1064230723060102

3. Köhl, M. A., & Hermanns, H. (2023). Model-based diagnosis of real-time systems: Robustness against varying latency, clock drift, and out-of-order observations. ACM Transactions on Embedded Computing Systems, 22(4), Article 68, 1–48. https://doi.org/10.1145/3597209

4. Górski, T., & Stecz, W. (2024). A method for modeling and testing near-real-time system scenarios. Applied Sciences, 14(5), 2023. https://doi.org/10.3390/app14052023

5. MIT OpenCourseWare. (2011). Lecture notes: State machines in system design. Retrieved February 16, 2025, from https: //ocw.mit.edu /courses/6-01sc-introduction-to-electrical-engineering-and-computer-science-i-spring-2011/pages/unit-1-software-engineering/state-machines/

6. Fuicu, S., Avramescu, A., Lascu, D., Padurariu, R., & Marcu, M. (2015). Real-time monitoring using finite state-machine algorithms. In R. Giaffreda et al. (Eds.), Internet of Things. User-Centric IoT. IoT360 2014. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering (Vol. 150). Springer, Cham. https://doi.org/10.1007/978-3-319-19656-5_27

7. Gong, C.-S. A., Su, C.-H. S., Chen, Y.-H., & Guu, D.-Y. (2022). How to implement automotive fault diagnosis using artificial intelligence scheme. Micromachines, 13(9), 1380. https://doi.org/10.3390/mi13091380

8. Rasheed, R., Qazi, F., Dur e Shawar Agha, Ahmed, A., Asif, A., & Shams, H. (2024). Machine learning approaches for in-vehicle failure prognosis in automobiles: A review. VFAST Transactions on Software Engineering, 12(1), 169–182. https://doi.org/10.21015/vtse.v12i1.1713

© Nikitin D.M., Rybitskyi O.M. (2025). Automated models of software systems for vehicle diagnostics: approach to data integration. Сучасний захист інформації, 1(61), 105–112.
https://doi.org/10.31673/2409-7292.2025.017899.

9. Wagner, F., Schmuki, R., Wagner, T., & Wolstenholme, P. (2006). Modeling software with finite state machines: A practical approach (1st ed.). New York, USA: Auerbach Publications. https://doi.org/10.1201/9781420013641

10. Embedded Staff. (2016). Implementing finite state machines in embedded systems. Retrieved February 19, 2025, from https://www.embedded.com/implementing-finite-state-machines-in-embedded-systems/

11. Chen, J., & S, R. (2021). Model-based validation of diagnostic software with application in automotive systems. IET Cyber-Systems and Robotics, 3(3), 140–149. https://doi.org/10.1049/csy2.12016

12. Kim, K., & Roush, F. W. (2003). Algebra, abstract. In R. A. Meyers (Ed.), Encyclopedia of physical science and technology (3rd ed.). Academic Press. https://doi.org/10.1016/B0-12-227410-5/00019-3

Надійшла 03.02.2025

© Nikitin D.M., Rybitskyi O.M. (2025). Automated models of software systems for vehicle diagnostics: approach to data integration. Сучасний захист інформації, 1(61), 105–112.
https://doi.org/10.31673/2409-7292.2025.017899.