DOI: 10.31548/itees.2025.02.063
ISSN 3083-7502

UDC 629.3.017:004.318

Nikitenko Yevheniy
PhD in Physical and Mathematical Sciences, Associate Professor of the Department of Computer Systems, Networks and Cybersecurity,
National University of Life and Environmental Sciences of Ukraine
ORCID: http://orcid.org/0000-0002-9222-644X
E-mail: ev.nikitenko@nubip.edu.ua

Gladkij Anatolij
PhD in Physical and Mathematical Sciences, Associate Professor of the Department of Computer Systems, Networks and Cybersecurity,
National University of Life and Environmental Sciences of Ukraine
ORCID: https://orcid.org/0000-0001-8852-0884
E-mail: amglad@nubip.edu.ua

SPECIALIZED MODULE BASED ON ARDUINO FOR A CAR SECURITY SYSTEM

Abstract. *The high rate of car theft has led to an increased demand for efficient and reliable security systems capable of providing continuous monitoring of a vehicle's condition and promptly responding to attempts of unauthorized access. A significant portion of existing commercial solutions is characterized by high cost, closed architecture, and insufficient adaptability to the specific requirements of car owners. The use of open hardware platforms such as Arduino makes it possible to develop cost-effective, flexible, and specialized security modules that can be integrated into any vehicle model while maintaining a high level of safety.*

The Arduino platform, due to its open hardware architecture and wide selection of compatible modules, is one of the most common foundations for building automotive security systems. Among the main advantages of Arduino are its low cost, support for standard interfaces (UART, I2C, SPI), compatibility with numerous digital and analog sensors, availability of libraries, and ease of debugging. These factors make Arduino a convenient tool for implementing fully functional security modules.

Keywords: *Automotive Security System, Specialized Module, Arduino Platform, Sensors.*

Introduction. The purpose of this work is to develop a specialized hardware and software module based on the Arduino platform, designed for automated monitoring and signaling of unauthorized interference or attempted vehicle theft.

To achieve this goal, the following tasks have been defined:

1. Analyze similar automotive security systems, identifying their advantages and disadvantages.
2. Develop a structural model of the specialized security system module.
3. Select the optimal hardware configuration of the Arduino platform for solving the stated tasks.
4. Develop software for controlling the module and implementing the main monitoring and signaling functions.

The practical significance of the work lies in the development of an accessible and effective solution for vehicle security systems, which can be used in the real business sector.

The growing level of vehicle automation, the introduction of intelligent functions into onboard electronic systems, and the expansion of wireless communication have significantly increased the vulnerability of cars to external cyber and physical threats. At the same time, the widespread use of keyless access, telematics modules, and CAN buses in standard security systems has revealed numerous shortcomings of existing security architectures, including:

- weak user authentication and the absence of multi-factor access verification
- the technical possibility of conducting relay attacks, breaches via OBD-II, and spoofing identifiers in CAN packets
- insufficient control by the end user over updating or expanding security functions

- lack of support for flexible integration of new sensor, cryptographic, or tracking modules.

In addition, most OEM systems are closed both in software and hardware, which makes it impossible to promptly update or adapt such systems to new requirements or real threat scenarios.

Under such conditions, the development of an autonomous, modular security solution becomes especially relevant—one that not only performs basic functions of intrusion detection or blocking but also provides:

- independence from the manufacturer's central ECUs or CAN bus
- support for adaptive real-time threat response
- the ability for remote event indication and object tracking
- an open structure for integration with additional protective modules.

Using the Arduino platform as the basis for implementing such a module is considered appropriate.

The engineering task includes forming the logical structure of the system, synthesizing the hardware configuration based on available modules, developing program code using Arduino IDE libraries, and conducting simulations and real-world testing. Special attention is given to modularity, structural autonomy of each component, and the compliance of event-processing logic with the requirements for time sensitivity, minimization of false alarms, and safe handling of input signals.

Within the defined purpose and considering the specifics of embedded security systems, particular attention must be paid to the technical requirements of the security module, which must operate under conditions of limited computational and energy resources. These requirements are formed taking into account the principles of energy efficiency, response time, event detection accuracy, resistance to external interference, and the ability to integrate additional modules. These characteristics must be incorporated at the system design stage, including the architectural model, choice of hardware platform, software configuration, and the logic of input signal processing.

Literature Review. Vehicle security systems are an integral part of the overall concept of automotive safety and serve the function of preventing unauthorized access, theft, and damage to the vehicle or its individual components. Due to the increasing information and physical threats in the automotive sector, security systems are continuously evolving in both technical and software aspects. Their functional complexity is constantly growing, which necessitates precise classification to analyze the advantages, limitations, and application areas of each type.

In [1], the authors describe automated systems that activate without user intervention after a certain period of inactivity. They usually block engine start, starter operation, or fuel supply. A typical example of such systems is the immobilizer, which does not require additional actions for activation but has limited functionality.

Active systems require explicit user action to turn the system on or off (pressing a button, entering a code, using a key fob). They provide extended configuration options, allowing the activation of various sensors (impact, volume, tilt) and alarm modes [2].

The systems proposed in [3], integrated into the vehicle's electronic architecture (OEM), allow synchronization with control units via the CAN bus, ensuring deep interaction with other electronic systems of the vehicle. They may include telematic functions and interact with mobile applications or cloud platforms.

Physical vehicle protection means include mechanical blockers of the steering wheel, gearbox, hood, and pedals. These devices provide a direct physical barrier but do not offer remote control or notification capabilities [4].

Smart key and GSM alarm systems incorporate authentication algorithms, encryption, dynamic signal coding (rolling code), and intrusion detection [4].

In [5], a comparison is presented between the traditional electronic device communication architecture in vehicles and a CAN-bus-based architecture. In the non-CAN variant, each device has a separate connection to the electronic control unit (ECU), creating a complex network of wired channels. In the CAN-based variant, all devices are connected to a common two-wire bus, through which data exchange occurs, providing a more efficient and scalable network topology.

DOI: 10.31548/itees.2025.02.063
ISSN 3083-7502

Security systems also actively use sensor modules, among which the most common are ultrasonic, infrared, gyroscopic, and magnetoresistive sensors. They enable the detection of movements inside the cabin, vehicle body tilt, impacts, attempts to break locks, or hood opening. Due to low power consumption and compactness, these sensors are widely used in embedded solutions based on microcontrollers such as Arduino, STM32, or ESP32 [6].

Analyzing the results of recent studies, it is advisable to focus on the development of flexible, autonomous, and adaptive security modules with built-in self-diagnostic capabilities, cryptographic protection, and integration with mobile services, which constitutes the goal of further system design based on the Arduino platform.

Purpose. The purpose of this article is to develop and study a specialized hardware-software module based on the Arduino platform for an automotive security system, designed to automatically monitor the vehicle's condition, detect unauthorized access attempts, and promptly generate alarms and lock critical vehicle components.

Results and Discussion.

Architecture of the security module

The design of an effective automotive security module is based on the principles of modularity, autonomy, and adaptability to external threats, which is especially relevant given the continuous increase in both cyber and physical attacks on vehicles. The architecture of the system must ensure not only the detection of and response to unauthorized access attempts, but also support the scalability of functionality, integration of new components, and flexible configuration of operating logic according to specific usage conditions.

The central element of the developed security module is the Arduino UNO microcontroller, based on the ATmega328P, which performs the functions of controlling all subsystems, processing input data from sensors, and generating control signals for actuators. The choice of this platform is determined by its open architecture, low power consumption, compatibility with numerous peripheral modules, and extensive library support for implementing monitoring and signaling algorithms [7].

The structural diagram of the security module architecture is shown in Figure 1. This UML diagram reflects the main hardware components of the system, their functional interconnections, and the directions of data transmission between the modules.

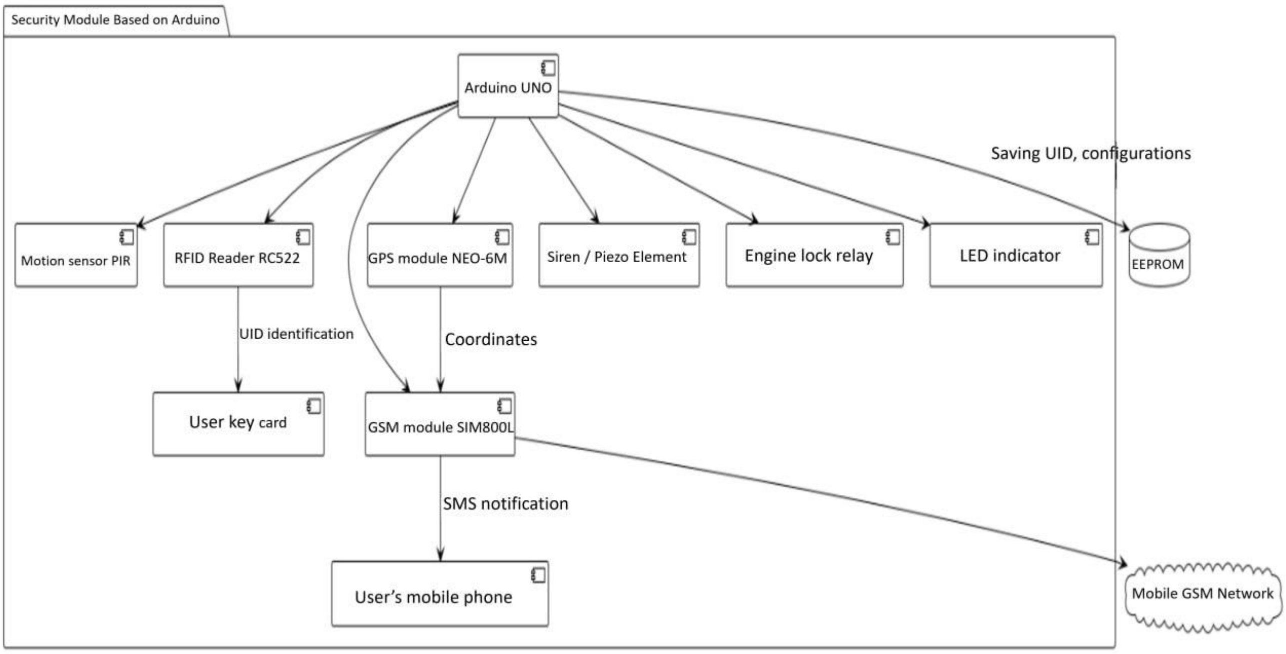


Figure 1 – Architecture of the Arduino-based security module

As shown in Figure 1, the Arduino UNO serves as the central node that coordinates the operation of all peripheral devices. Sensors and actuators are directly connected to the microcontroller, ensuring a rapid response to events. The transmission of alarm signals and coordinates is carried out via the GSM module, which interacts with the mobile network to notify the user.

An important feature of the architecture is its modularity, which enables easy integration of additional functional blocks, such as Wi-Fi modules, tilt sensors, or biometric authentication units. The open Arduino IDE software environment provides flexibility in configuring event-processing logic and adapting the system to specific usage scenarios [8].

Design of the structural and schematic diagrams

The development of a vehicle security module involves constructing both a structural and a schematic electrical diagram. The structural diagram allows the main functional blocks of the system and their interactions to be represented at a conceptual level, while the schematic diagram details the electrical connections between components, defining the connection logic and the nature of the signals.

The structural diagram of the security module is built taking into account the functional distribution of subsystems into threat-detection blocks, indication blocks, and active countermeasure blocks. The central link of the system is the Arduino UNO microcontroller, which processes signals from sensors, generates control signals for actuators, and provides interaction with the user. Inputs to the Arduino UNO include signals from a PIR sensor for motion detection, an ultrasonic distance sensor for proximity monitoring, and an SW-200D tilt sensor for registering changes in the vehicle's position. Control signals are generated for a relay module that blocks the vehicle's critical systems, for LED indication, and for an audible alarm. Interaction with a mode-selection button and a potentiometer for adjusting parameters is also provided (Figure 2).

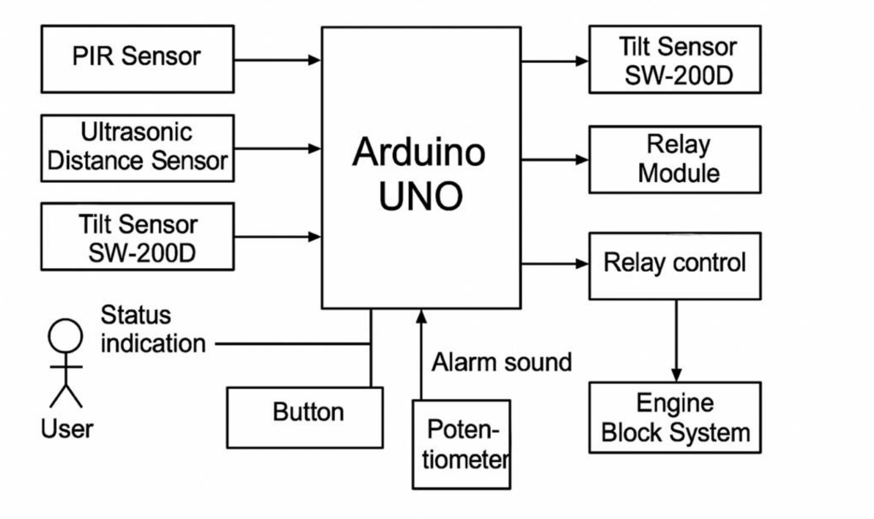


Figure 2 – Structural diagram of the security module

The schematic diagram details the connections between the system components, taking into account power supply, signal connections, and control circuits. Power is provided by a 9V battery, which, through a voltage regulator, supplies the Arduino and peripheral devices. Sensors (ultrasonic, PIR, and tilt) are connected to the microcontroller's digital inputs. LEDs for status indication are connected through 220 Ω current-limiting resistors. The audible alarm is implemented using a piezo element, controlled by a separate digital output. Additionally, a button and a potentiometer are connected, serving for manual control and parameter adjustment, respectively (Figure 3).

The constructed circuit implements basic security logic: when any of the sensors is triggered, the system activates both visual and audible alarms while simultaneously opening the relay to block the engine start system. This approach provides comprehensive, multi-level vehicle protection, combining threat detection, signaling, and physical countermeasures against theft attempts.

DOI: 10.31548/itees.2025.02.063
ISSN 3083-7502

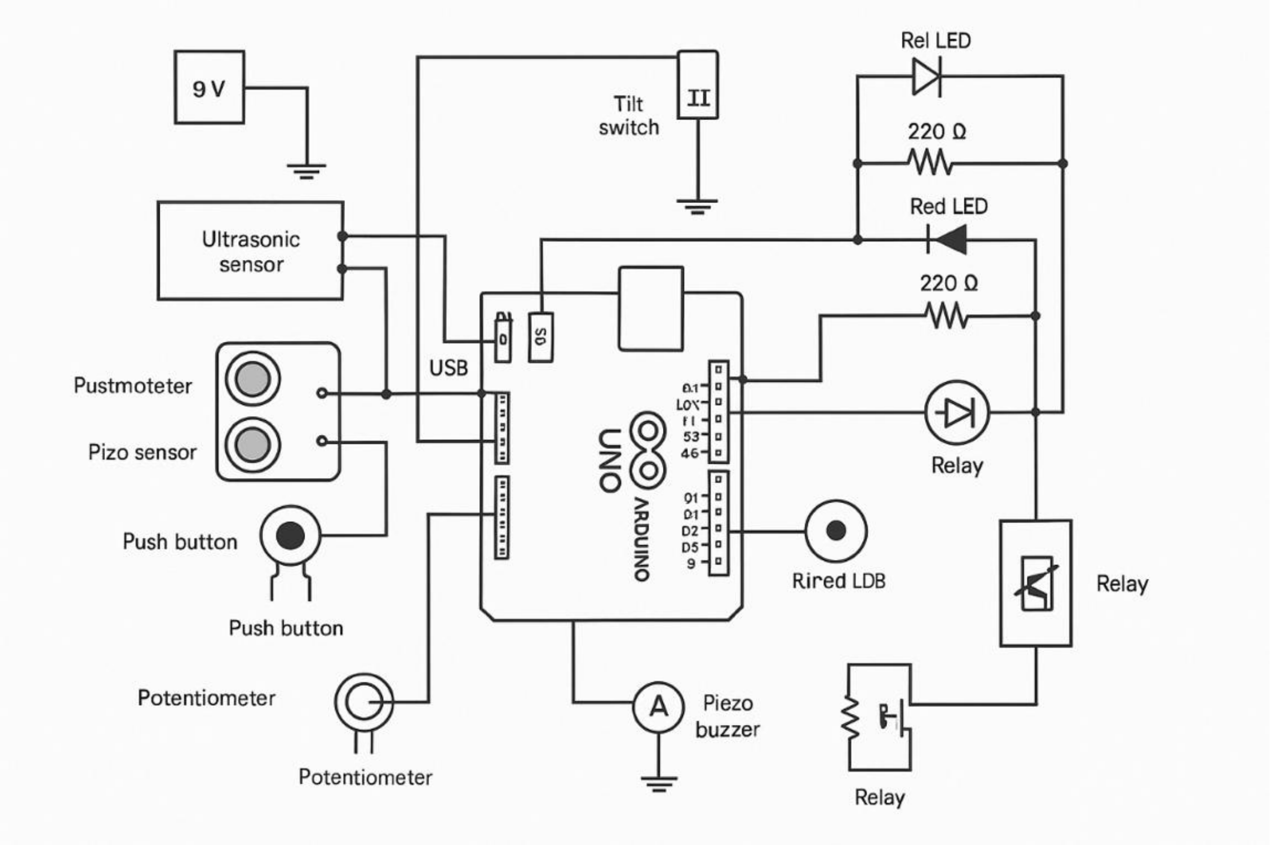


Figure 3 – Schematic diagram of the security module

Development of the system operation algorithm

To ensure the correct functioning of the Arduino-based security module, a system operation algorithm was developed that implements the logic for responding to threats detected by various sensors and activates the corresponding actuators. The main requirements for the algorithm are: continuous monitoring of sensor status, minimization of false alarms, the ability for the user to reset the alarm, and restoration of the system's initial state.

The operation of the module begins with the initialization of all components. The system then checks for the presence of power. If power is absent, a standby mode is activated. Upon detecting a stable power source, the system switches to security mode.

During the monitoring phase, the system reads data from the following sensors:

- **PIR sensor**, which reacts to motion within the monitored area.
- **SW-200D tilt sensor**, which detects changes in the vehicle's position (e.g., during a towing attempt).
- Ultrasonic distance sensor, which allows detection of an object approaching the vehicle.

If any sensor is triggered, the system enters an alarm mode. Specifically, the audible alarm (piezo element) is activated, the visual indicator (red LED) is turned on, and critically, the relay blocks the engine start system, preventing unauthorized vehicle movement.

To prevent false triggers of the ultrasonic sensor, a mechanism for confirming object proximity is implemented, taking into account the signal duration exceeding a predefined threshold. This avoids alarm activation due to short-term movements or minor obstacles.

The alarm state can be reset by pressing a user button. In this case, the siren is turned off, the relay is unlocked, and the visual indicator returns to its initial state. Thus, the algorithm maintains a closed loop — after the alarm condition ends, the system returns to the monitoring phase.

The logic of the algorithm is visualized in a UML activity diagram shown in Figure 4. The diagram illustrates the sequence of events, logical branches, user actions, and transitions between different system states.

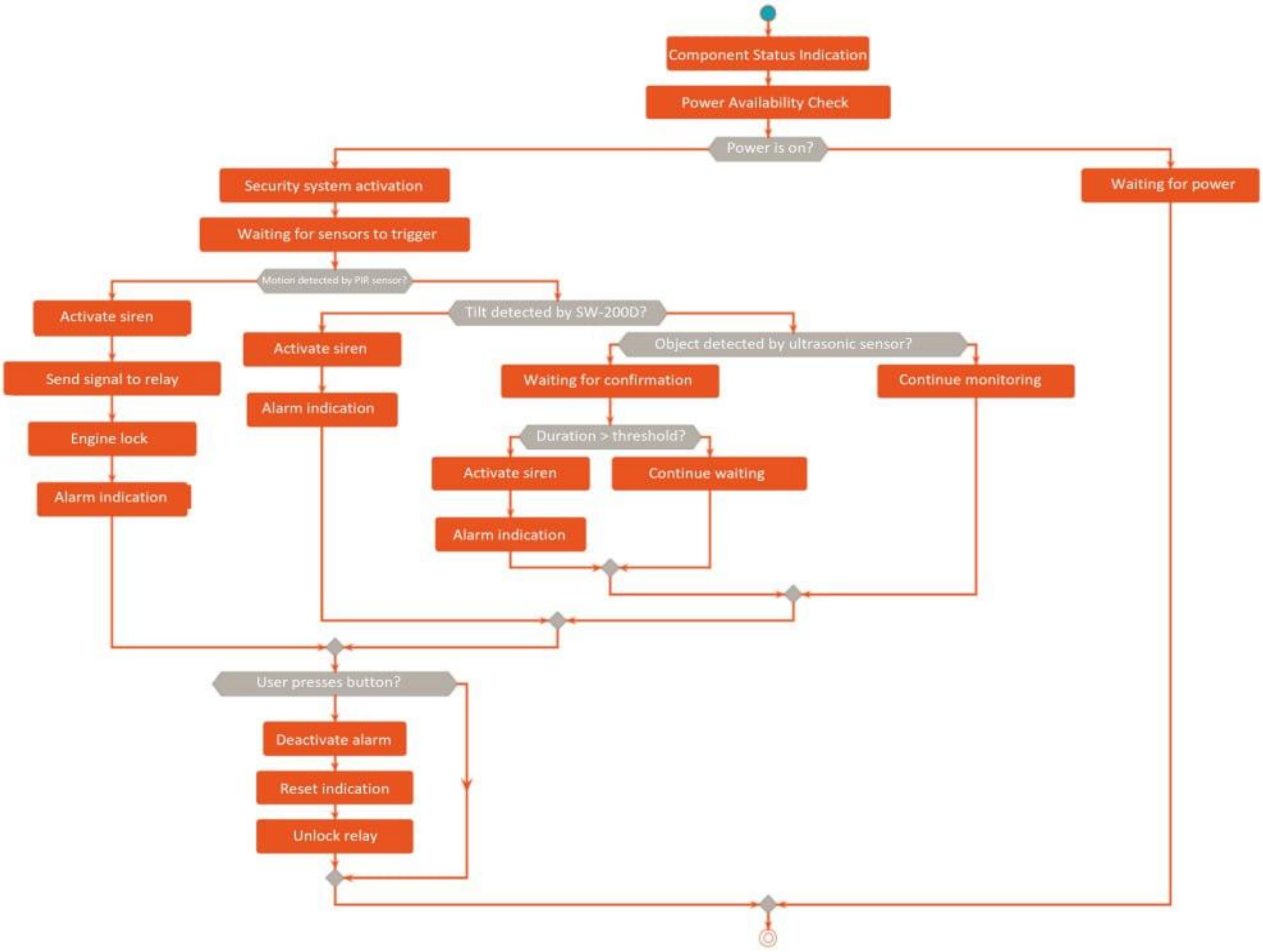


Figure 4 – Operation algorithm of the security module (UML activity diagram)

The implementation of such an algorithm enables a comprehensive system response to potential threats and enhances the vehicle's security level. The presence of multi-level control logic, verification mechanisms, and user intervention makes the system reliable, adaptive, and convenient to use.

Description of the module software

The security module software is implemented in C++ within the Arduino IDE environment, using a procedural model for event and signal management. The main functionality of the module includes monitoring input signals from sensors, processing breach logic, generating the corresponding alarm response, and controlling actuators.

The software architecture employs a modular approach to handling data from heterogeneous sensors: passive infrared (PIR), ultrasonic (HC-SR04), tilt sensor (Tilt), analog sound sensor, potentiometer, and button. Each sensor is connected to a separate digital or analog input of the Arduino UNO R3 microcontroller. Signal processing is performed in the main program loop (loop()), where each sensor is checked according to its threshold values or activation conditions.

During initialization (setup()), ports are configured, pin modes (input/output) are set, the serial monitor is activated, and libraries for external components are loaded. For digital inputs such as PIR, Tilt, and Button, the signal level is read directly using the digitalRead() function. Analog sensors (Potentiometer, Sound Sensor) are processed using analogRead(), with subsequent comparison of values against predefined sensitivity thresholds.

Figure 5 presents the UML component diagram of the developed system, illustrating the logical connections between the central controller (Arduino UNO R3), the sensor subsystem, actuators, and

DOI: 10.31548/itees.2025.02.063
ISSN 3083-7502

auxiliary infrastructure elements. The diagram indicates connection types (digital input/output, analog input, I2C interface), corresponding to the program code structure and port configuration.

The implementation logic also takes into account optimization of event response time, achieved by avoiding blocking functions such as delay() in critical sections of the code. Instead, time control is implemented using the millis() function, allowing for pseudo-parallel operation.

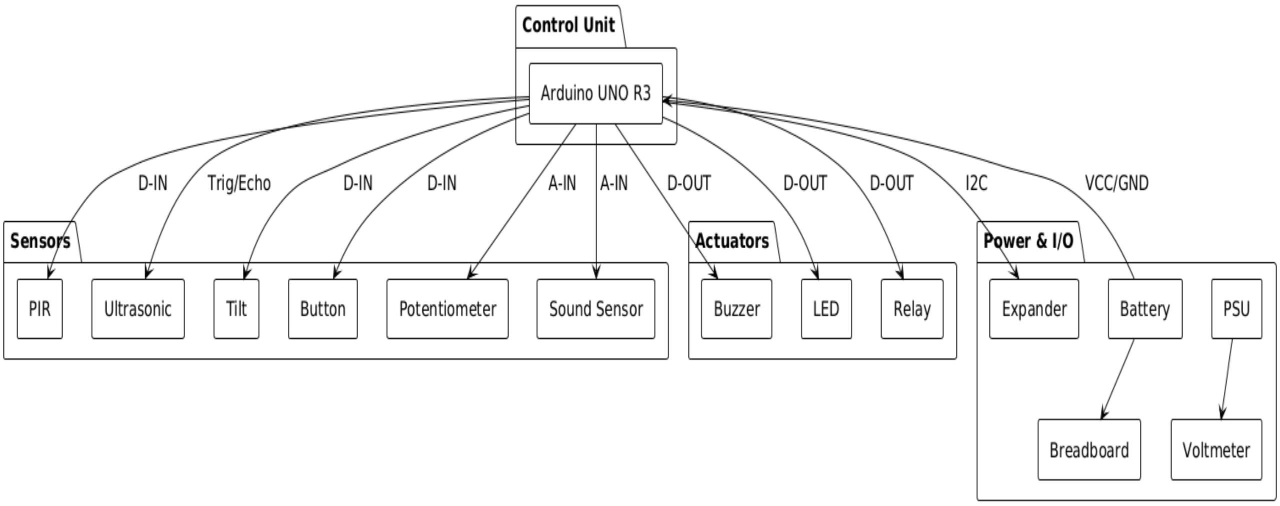


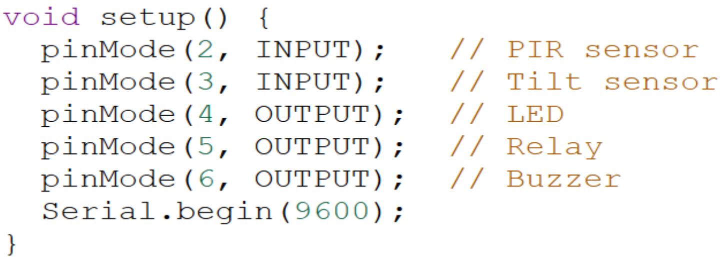
Figure 5 – UML component diagram of the Arduino UNO-based security module

The developed software ensures full integration of all physical components of the security system, guarantees real-time event response, and provides a foundation for further functionality expansion within secure embedded solutions.

Implementation of control and signaling functions

The control and signaling system of the security module is built around the Arduino UNO R3 microcontroller, which interacts with motion, tilt, and sound sensors and controls actuators — an LED, a relay, and a buzzer. The response logic architecture ensures the detection of unauthorized actions and the immediate activation of the alarm system.

Figure 6 shows a fragment of the system's initial setup, where digital input and output ports are configured for operation with sensors and signaling devices. Specifically, ports D2 and D3 are configured to receive data from the motion (PIR) and tilt sensors, while ports D4–D6 are used to activate the LED, relay, and buzzer, respectively.

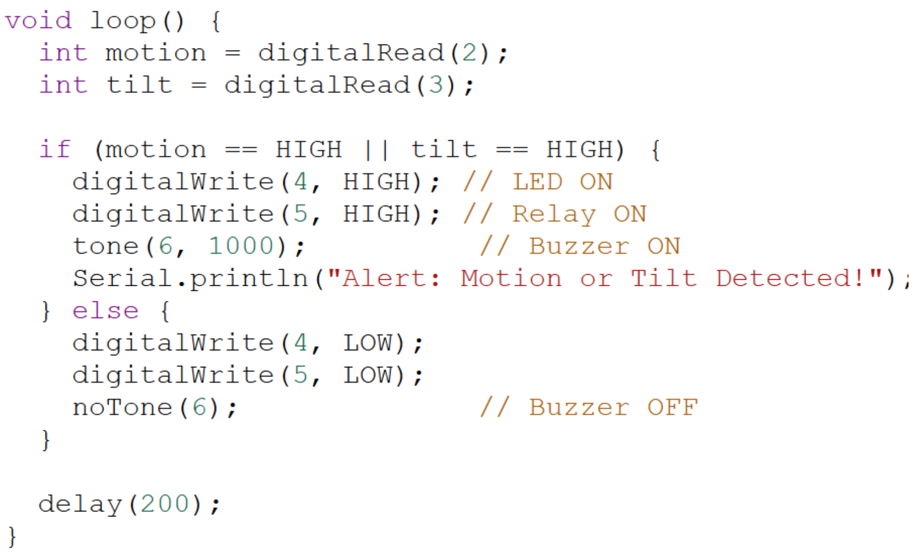


```
void setup() {
  pinMode(2, INPUT);    // PIR sensor
  pinMode(3, INPUT);    // Tilt sensor
  pinMode(4, OUTPUT);   // LED
  pinMode(5, OUTPUT);   // Relay
  pinMode(6, OUTPUT);   // Buzzer
  Serial.begin(9600);
}
```

Figure 6 – Fragment of digital port initialization in the security module

The module's operation logic involves continuous monitoring of sensor states. When an active signal is detected from any sensor (motion or tilt), the microcontroller immediately sends control signals to the actuators, triggering the alarm. Figure 7 shows the implementation of the main loop() cycle, which is responsible for event analysis and the corresponding response.

The module also implements a function for monitoring acoustic changes in the environment using an analog microphone sensor. Noise level readings are taken through port A0, and alarm activation is triggered when the predefined threshold is exceeded. A fragment of the corresponding code listing is shown in Figure 8.

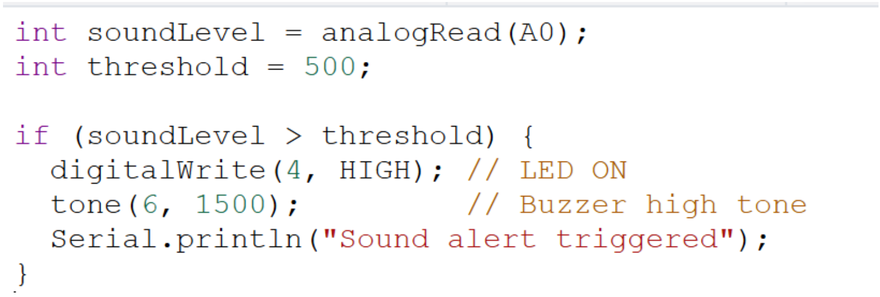


```
void loop() {
  int motion = digitalRead(2);
  int tilt = digitalRead(3);

  if (motion == HIGH || tilt == HIGH) {
    digitalWrite(4, HIGH); // LED ON
    digitalWrite(5, HIGH); // Relay ON
    tone(6, 1000);          // Buzzer ON
    Serial.println("Alert: Motion or Tilt Detected!");
  } else {
    digitalWrite(4, LOW);
    digitalWrite(5, LOW);
    noTone(6);              // Buzzer OFF
  }

  delay(200);
}
```

Figure 7 – Implementation of the main loop() cycle



```
int soundLevel = analogRead(A0);
int threshold = 500;

if (soundLevel > threshold) {
  digitalWrite(4, HIGH); // LED ON
  tone(6, 1500);          // Buzzer high tone
  Serial.println("Sound alert triggered");
}
```

Figure 8 – Code fragment for acoustic change monitoring

This implementation allows for multiple threat detection channels, thereby increasing the overall reliability of the module.

Conclusions. During the course of this work, the stated goal was fully achieved – a specialized automotive security system module based on the Arduino platform was developed, providing a basic level of protection against unauthorized access and mechanical tampering.

It was determined that modules based on Arduino microcontrollers with open architecture demonstrate high adaptability to specific user requirements. The main advantages of such systems were identified: low cost, ease of implementation, and customization possibilities. Among the disadvantages are the need for configuration, lack of protection against scanners, and limited energy efficiency in the basic implementation.

Functional and structural models of the specialized security module were developed. The model provides multi-channel monitoring: door status, nearby motion detection, and vehicle body tilt. All events are processed centrally by the microcontroller, after which the response is initiated – activation of visual and audible alarms, engine start system blocking, and signal transmission to an external module.

The developed software ensures continuous scanning of sensor states, event processing, and execution of responses according to predefined scenarios.

Results of modular and system testing confirmed the functionality of all components – sensors, logic, and actuators. The system responded to threats within an average of 140–200 ms, which is acceptable for this class of devices. Stability of event handling was verified under simultaneous signals from multiple sensors.

Thus, the developed security module demonstrated effective operation, high stability in monitoring and signaling modes, and flexibility for future expansion. The obtained results allow the recommended solution to be used for educational, research, and practical purposes, as well as serving as a foundation for building full-featured automotive or stationary security systems with additional functionalities.

References

1. Tverdokhlebov, V. I. (2018). Vehicle security systems. National Aviation University.
2. Mikhieiev, A. V., & Demchenko, S. O. (2020). Microcontrollers for embedded systems: Arduino and its applications. Kharkiv National University of Radio Electronics.
3. Struk, A. O., & Yavorskyi, I. P. (2021). Fundamentals of digital electronics and microprocessor technology. Lviv National University Publishing House.
4. Sharma, K., & Patel, D. (2021). Vehicle security system using GSM and GPS: A review. International Journal of Engineering Research and Applications, 11(4), 11–15.
5. Martin, J. (2019). Arduino for automotive projects. Packt Publishing.
6. Volodin, S. V. (2017). Automatic control and vehicle security systems [Methodical guidelines]. Kyiv National University of Construction and Architecture.
7. Maksimović, D., & Erickson, R. (2020). Fundamentals of power electronics. Springer. https://doi.org/10.1007/978-3-030-43881-4.
8. Zozulia, S. M. (2022). Working with sensors in Arduino IDE. Vinnytsia National Technical University.

Нікітенко Євгеній Васильович
кандидат фізико-математичних наук, доцент кафедри комп'ютерних систем, мереж та кібербезпеки,
Національний університет біоресурсів і природокористування України
ORCID: http://orcid.org/0000-0002-9222-644X
E-mail: ev.nikitenko@nubip.edu.ua

Гладкий Анатолій Михайлович
кандидат фізико-математичних наук, доцент кафедри комп'ютерних систем, мереж та кібербезпеки,
Національний університет біоресурсів і природокористування України
ORCID: https://orcid.org/0000-0001-8852-0884
E-mail: amglad@nubip.edu.ua

СПЕЦІАЛІЗОВАНИЙ МОДУЛЬ НА БАЗІ ARDUINO ДЛЯ АВТОМОБІЛЬНОЇ СИСТЕМИ БЕЗПЕКИ

***Анотація**. Високий рівень викрадень автомобілів призвів до зростання попиту на ефективні та надійні системи безпеки, здатні забезпечувати постійний моніторинг стану транспортного засобу та оперативно реагувати на спроби несанкціонованого доступу. Значна частина існуючих комерційних рішень характеризується високою вартістю, закритою архітектурою та недостатньою адаптованістю до конкретних вимог власників автомобілів. Використання відкритих апаратних платформ, таких як Arduino, дозволяє розробляти економічно ефективні, гнучкі та спеціалізовані модулі безпеки, які можна інтегрувати в будь-яку модель автомобіля, зберігаючи при цьому високий рівень безпеки.*

Платформа Arduino, завдяки своїй відкритій апаратній архітектурі та широкому вибору сумісних модулів, є однією з найпоширеніших основ для побудови автомобільних систем безпеки. Серед основних переваг Arduino — низька вартість, підтримка стандартних інтерфейсів (UART, I2C, SPI), сумісність з численними цифровими та аналоговими датчиками, наявність бібліотек та простота налагодження. Ці фактори роблять Arduino зручним інструментом для реалізації повнофункціональних модулів безпеки.

***Ключові слова**: система безпеки для автомобілів, спеціалізований модуль, платформа Arduino, датчики*